



138 107 THS

THESIS



This is to certify that the

dissertation entitled

# MODELS OF CHANGE IN THE WOODLAND SETTLEMENT

OF THE NORTHERN GREAT LAKES REGION

presented by

Susan Rapalje Martin

has been accepted towards fulfillment of the requirements for

Ph.D. degree in Anthropology

Major professor

Charles Cleland

Date May 13, 1985

MSU is an Affirmative Action/Equal Opportunity Institution

O-12771



# RETURNING MATERIALS: Place in book drop to remove this checkout from your record. FINES will be charged if book is returned after the date stamped below.

2000 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	(COX 2 4 7)5 03,0 2,0 5 03,0 1,4 2008	
Sep 2 9	520808	
TEP 25 1993		
MAR 2 9 2006 19 25 07 MAY 22 1 2612		

# MODELS OF CHANGE IN THE WOODLAND SETTLEMENT OF THE NORTHERN GREAT LAKES REGION

Ву

Susan Rapalje Martin

## A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Anthropology
1985

Copyright by SUSAN RAPALJE MARTIN 1985

### ABSTRACT

# MODELS OF CHANGE IN THE WOODLAND SETTLEMENT OF THE NORTHERN GREAT LAKES REGION

By

# Susan Rapalje Martin

This study applies the simulation approach to a problem of prehistoric settlement pattern change in the northern Great Lakes of North America. The simulation model examines the interaction of seasonal resource availability and technological repertoire on the distribution of archaeological sites at the Great Lakes coastlines. Specifically it models the development of the coastline fisheries adaptation, incorporating a set of general assumptions about hunter-gatherer social forms and demographic profiles, and examines the notion that technological innovations produced changes in the locational pattern of prehistoric settlement during the Woodland time period.

The study begins with an examination of ethnographic, historical, and archaeological data to catalog the variation that existed in the aboriginal fisheries adaptation. Then, the study examines the locational evidence of prehistoric settlement distributions, concentrating on temporal changes in location related to hypothetical changes in the technologies of the fisheries adaptation. Finally, the study arranges the assumptions and implications of a proposed technology shift into three simulation models of

coastline site distributions. Each simulation model is evaluated for similarities to the known distributions of archaeological sites and for general consistency with their numbers and aggregation tendencies.

General support is lent to models of change that assume low initial population levels, low population growth rates, and low environmental failure rates. The introduction of the gill net at the beginning of the Late Woodland time period is consistent with observed changes in site distributions in the area under study. The changes are visible as broad-area shifts in the location of populations rather than in the selection of specific kinds of site locations.

### **ACKNOWLEDGMENTS**

I wish to acknowledge the Richard Lounsbery Fund for pre-doctoral research in Anthropology of the American Museum of Natural History for generous financial support. The following persons served well as consultants and advice-givers about the intricacies of SPERRY 1100/80:

George Fox, John Tucker, Cynthia Ware MacDonald and Ted Soldan of the Academic Computing Service at Michigan Technological University, and Kamal Bijlani and Dr. Linda Ottenstein of the Department of Mathematics and Computer Science, Michigan Technological University.

To Dr. Larry D. Lankton and Dr. John H. Winslow, able administrators in the Department of Social Sciences at Michigan Technological University, go sincere thanks for spiritual and financial assistance while this project slouched toward completion.

I would like to thank my graduate committee for unfailing confidence: Dr. Charles E. Cleland, Chairman, Dr. Moreau S. Maxwell, Dr. Lawrence Robbins, and Dr. William A. Lovis.

Finally, to the staff of the Morning Program, KSJN-FM, Minneapolis/St. Paul (an NPR affiliate), thanks for entertaining a lonely long-distance writer.

# TABLE OF CONTENTS

		P	age
LIST OF	TABLES	• • `	Vll
LIST OF	FIGURES	• •	х
Chapter			
One.	BACKGROUND INFORMATION AND STANDING MODELS	;	1
	Introduction		19 21 23 24 26 27 30 31 36
Two.	THE FISHERIES AND THEIR CONDITIONS: THE ARCHAEOLOGICAL, ETHNOGRAPHIC, AND HISTORICAL DATA BASE  The Methods of fish capture Without gear	• • •	53 55 56 58 63 67 73 76 82

Chapter		Page
	Fish preservation	. 94
Three.	A STATISTICAL STUDY OF PREHISTORIC LAKESHORE LOCATIONS	.108
	The Sites and their locational attributes. Case selection procedures	.110 .112 .123 .127 .130 .133 .135 .136 .142 .142 .152
	Clustering of Middle Woodland components	.170
Four.	THE SIMULATION OF ARCHAEOLOGICAL SITE DISTRIBUTIONS	.184
	Introduction	.187 .188 .190 .194 .199 .206 .208 .209 .211 .217 .222 .228
	GILLNET Model results	.233

Chapter					
Five	. CONCLUSIONS AND SUGGESTIONS FOR THE FUTURE246				
	Introduction				
APPEND	APPENDICES				
Appendi	Appendix				
A. (	. Catalog of sites used in the analyses259				
в. О	B. Codebook of variables and values2				
c. r	Data tables288				
D. I	Program listing300				
TTCM OF	P DEPEDENCES 366				

# LIST OF TABLES

Table	Page
1.	Research intensity for archaeological components in the study area
2.	Site density and survey intensity within sectors of the study area 32
3.	Species of fish analyzed
4.	Typology of fish capture methods 55
5.	Artifacts/features from Middle and mixed Woodland components 95
6.	Artifacts/features from Late Woodland components 96
7.	Faunal identifications from Middle Woodland components
8.	Faunal identifications from mixed Woodland components
9.	Faunal identifications from Late Woodland components100
10.	Work group composition by species, method, and season103
11.	Comparison of weir methods with net methods105
12.	Susceptibility of fish species to gears106
13.	Nominal/ordinal scale variable definitions114
14.	Nominal/ordinal scale variables: environmental setting/archaeological setting118
15.	Values on variables examined by the chi- square statistic
16.	Results of 2x2, 2x3, and 2x4 contingency table analyses, CCPER with categorical variables

Table		Page
17.	Univariate statistics over all components (n = 84)	.129
18.	Univariate statistics measured over Middle, Late Woodland components (n = 25, n = 46)	.131
19.	T tests between all Middle and all Late Woodland components (n = 25, n = 46)	.134
20.	T tests between isolated Middle and Late Woodland components $(n = 14, n = 29)$	137
21.	T tests between Middle Woodland and mixed locations (n = 6, n = 17)	139
22.	T tests between Late Woodland and mixed locations (n = 22, n = 17)	140
23.	LDFs and related statistics	146
24.	Measures associated with subregion-based discriminant function analysis of location	159
25.	Centroid values by cluster, Middle Woodland cluster analysis	168
26.	Centroid values by cluster, Late Woodland Cluster analysis	172
27.	Attributes of principal entities	190
28.	Seasonal resource scores by locality	198
29.	Input data for the simulations	199
30.	Input values for ZPG, STRESS, and GILLNET models	211
31.	ZPG model results	218
32.	STRESS model results	229
33.	GILLNET model results	235
34.	Case identification	288
35.	Case members, TEMPDATA file	290
36.	Case members, LOCDATA file	292

Table		Page
37.	FISHDATA data file	. 293
38.	TEMPDATA data file	. 296
39.	LOCDATA data file	. 298

# LIST OF FIGURES

Figure	P	age
1.	The Study area	8
2.	Structure of the Woodland transition (after Lovis and Holman 1976)	11
3.	Structure of the intensification of Woodland subsistence (after Cleland 1982)	15
4.	Hypothetical structure of Woodland settlement system transition	20
5.	Analytical sectors of the study area (after Goodyear et al 1982)	28
6.	Histogram for Middle Woodland components1	48
7.	Histogram for Late Woodland components1	49
8.	Histogram for components of unknown age1	51
9.	Scatterplot for locations, by age19	5 <b>6</b>
10.	Scatterplot for locations, by sub-region1	60
11.	Middle Woodland cluster analysis	5 <b>7</b>
12.	Late Woodland cluster analysis	71
13.	Flow diagram of simulation program19	95
14.	Component distribution, Middle Woodland period21	۱4
15.	Component distribution, Late Woodland period21	15
16.	Component distribution, all periods21	16
17.	Lodge aggregates, ZPG model/RNS #321	9
18.	Lodge distributions, ZPG model/RNS #322	20
19.	Lodge aggregates, ZPG model/RNS #222	23

Figure	·		Page
20.	Lodge	distributions, ZPG model/RNS #2	224
21.	Lodge	aggregates, ZPG model/RNS #1	225
22.	Lodge	distributions, ZPG model/RNS #1	226
23.	Lodge	distributions, STRESS model/RNS #1	230
24.	Lodge	distributions, STRESS model/RNS #2	232
25.	Lodge	distributions, STRESS model/RNS #3	234
26.	Lodge	aggregates, GILLNET model/RNS #1	237
27.	Lodge	distributions, GILLNET model/RNS #1	238
28.	Lodge	aggregates, GILLNET model/RNS #2	240
29.	Lodge	distributions, GILLNET model/RNS #2	241
30.	Lodge	aggregates, GILLNET model/RNS #3	243
31.	Lodge	distributions. GILLNET model/RNS #3	244

### CHAPTER ONE

### BACKGROUND INFORMATION AND STANDING MODELS

# Introduction

A common situation facing the archaeologist has to do with incomplete data, a fact that perpetually confounds those interested in the explanation of culture change.

Lengthy time spans provide the backdrop for understanding changing lifeways and for studying culture change. Yet time passing has the unfortunate property of destroying data essential to the task of explaining events of the past.

There are of course varied routes to compensate for missing data, ranging from the unbridled speculations of past decades to rigorous and frankly unenlightening deductions.

Some have advocated the use of models to experiment with the characteristics of the entities under study, and to predict the qualities of entities that due to the nature of archaeological data may no longer be observable.

We are concerned with real-world entities which have a compound structure and which change with the passage of time...Such entities we call systems. Now suppose that there is a system which we can observe in a variety of ways, and the future course of which we wish to predict. Often we wish to go further and control the system, deflecting its course to our own advantage. One approach to these objectives is to set up a "model" --- that is, a system that we have

ourselves created, which as far as possible mirrors those behavioural characteristics of the target system that we regard as important. We then answer our questions about the target system by studying or experimenting with the model (Doran 1973:428).

A relatively new direction implicit in Doran's statement is the simulation or artificial generation of archaeological data which, when carefully structured and utilized, may fill in the gaps created by the passing of time and the demise of cultures. A simulation is a particular kind of model that acts as a simplified analogue to a real-world process. Hamond defines a simulation as "the modelling of a process by a process" (1978:1-2). is not merely a semantic puzzle, for simulations represent simplified versions of the structural links and interactive routes among parts of a whole. Then, they go beyond static models by generating histories of interactions among those routes. "Thus the properties of a simulation model resemble, in form, some of the processual properties of the real-world situation under examination" (Hamond 1978:2). A simulation may then be considered a dynamic model of a process, whose output and behavior mimics that of the process, as well as predicts behaviors of the process that are otherwise unobservable.

In anthropology and archaeology the application of the computer-assisted simulation approach lagged a bit behind applications in related fields such as biology, economics,

and geography. Demographic anthropologists made relatively greater headway in using this approach than did other interest areas of anthropology (Dyke and MacCluer 1974, Swedlund 1975). Several archaeologists have recently adopted simulation as a technique for appropriate problems, especially those whose methodological positions incorporate a general systems theory point of view (Doran 1970, Renfrew 1981). In the last decade, a number of exemplary and artful uses of simulation modelling in archaeology made it clear that there was a role for this approach in the study of culture process (Hodder 1978). Some innovative studies included those of archaeological formation processes (Thomas 1973, O'Shea 1978, Aldenderfer 1981), hunter-gatherer organizational processes (Wobst 1974, Zubrow 1975), and settlement processes (Zimmerman 1977, Chadwick 1978).

This study applies the simulation approach to a problem of prehistoric settlement pattern change in the northern Great Lakes of North America. The simulation model examines the interaction of seasonal resource availability and technological repertoire on the distribution of archaeological sites at the Great Lakes coastlines. Specifically it models the development of the coastline fisheries adaptation, incorporating a set of general assumptions about hunter-gatherer-fisher social forms and demographic profiles, and examines the notion that technological innovations produced changes in the locational pattern of prehistoric settlement during the Woodland time period. In Clarke's terminology this study concentrates on

the spatial "macro level," both on the relationships among sites and on the links among sites and resource spaces (Clarke 1977:16).

The study begins with an examination of ethnographic, historical and archaeological data to catalog the variation that existed in the aboriginal fisheries adaptation, focussing on the ways in which the locational and labor requirements of the fisheries regulated and were regulated by household organization and settlement placement. the study examines the locational evidence of prehistoric settlement distributions, concentrating on temporal changes in location related to hypothetical changes in the technologies of the fisheries adaptation. Both of these bodies of data serve as sources of information about the operation of the fisheries systems over time, and result in a novel hypothesis about prehistoric subsistence system operation. Finally, the study arranges the assumptions and implications of a proposed technology shift into three simulation models of coastline site distributions. computer versions of the models allow experimental control over the timing and the magnitude of technological innovation and other changes vis a vis patterns of archaeological site distribution over the coastal landscape.

The ultimate goal of the simulation is an experimental situation in which the simulated settlement distributions, the actual product or output of the simulation, respond to alterations in the internal conditions of the model. These conditions and their values can be seen as competing

hypotheses about real-world processes of change; the simular responses give some indication of how such changes may have affected real system operation. The advantages of such an approach to studying culture change include experimental control over the key factors that hypothetically induced real system change, and a rigorous and unforgiving research environment in which to examine the assumptions about the dynamics of real system operation.

It is important to understand that the replication of real-world site distributional patterns is not the sole goal of the simulation approach. There are several reasons why this is so. First, the settlement pattern as it is now known to archaeologists is an artifact of many processes. Some of these processes include differential preservation, differential discovery, and differential excavation. Secondly, it is a fact of simulation modelling that any pattern can, with enough model modification, be reproduced at some stage. On the surface, correspondence between simulated and real site distributional patterns would suggest that the simulated system behaved in the same manner as the real one. It is necessary, however, to account for the relative ability of competing alternatives to produce recognizeable patterns, and most importantly to account for the sensitivity of the output to changes in the conditions under which the model is operating. In this way key factors and threshold values that accompany system change may be isolated. The goal, then, concentrates on the responses of the analog system to provide insights into aspects of real

system transformation and behavior that are otherwise unobtainable. For example, the generation of quantitative population estimates for various periods in prehistory in the northern Great Lakes area has been difficult if not impossible, although the qualitative factor of "population growth" has figured largely in explanations of settlement/subsistence system change (Lovis and Holman 1976, Cleland 1982). Using the simulation approach, the initial size of a population and the rate of population growth may be experimentally varied, until it is possible to specify within what range of size and growth the simular model behaves in accordance with known aspects of the real system. This approach will yield thresholds of population estimates; to exceed these levels is to create simular responses that are inconsistent both with other aspects of the simular system and with known system operation. The simulation exercise will, of course, not give precise numerical answers to the question of population size and growth, nor will it "test" whether one or another estimate is correct. however, examine the implications of such estimates in a formally constructed framework of inquiry. In this sense a simulation is not a testing mechanism, but rather a way of refining hypotheses.

# Background of the problem

The publication of McPherron's (1967) volume on the archaeology of 20MK1, the Juntunen site, represented the first comprehensive treatment of prehistoric occupation in

the area of interest (Figure 1). Located on an island in the Straits of Mackinac, Juntunen enjoyed a lengthy sequence of prehistoric occupation, and, by virtue of careful excavation of its stratified deposits, revealed a long sequence of ceramic complex changes and subsistence system adaptation to the abundant local protein resource, fish (Cleland 1966). Subsequent researchers in the area (Janzen 1968, Brose 1970, Lovis 1973) found a long-lived prehistoric adaptation to the fisheries resources. This adaptation was a supra-regional phenomenon that extended throughout the Lake Forest or Northern Tier area (Mason 1966, Stoltman 1973). Moreover, studies concentrating on the historic period archaeology in the region evidenced that the local dependence on the fisheries intensified during the historic period (Fitting 1976, Noble 1983). Cleland, in a recent treatment of the development of the fisheries adaptation, argued that "the unique prehistoric fishery, which was extant in this region during European contact and survived through most of the historic era, provides the most important single organizing concept for understanding the cultural development of this region" (Cleland 1982:761). There seems little doubt that, given available local resources and the content of the region's archaeological components, the key to understanding local prehistoric adaptations and settlement was the organization of the fisheries.

It is critical to evaluate standing explanations for the causes and sequences of cultural development within the

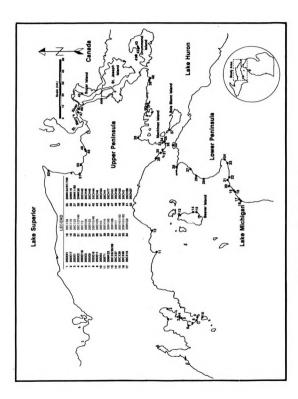


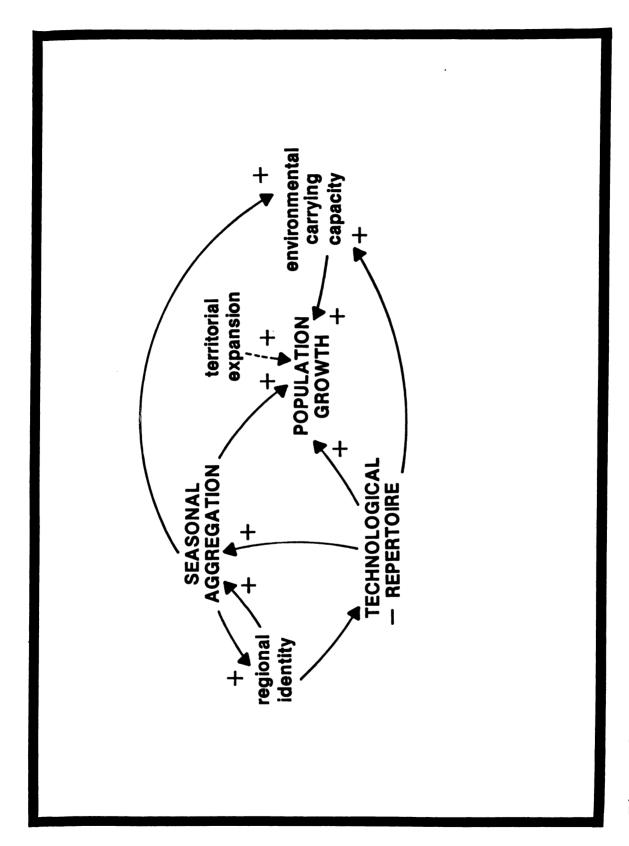
Figure 1. The Study area

region, particularly as they relate to settlement pattern change. In the following discussion, Middle Woodland refers to Laurel and related Lake Forest cultural complexes. This is primarily a temporal category, beginning with the introduction of ceramics into the study area, and ending with the development of the Mackinac Phase as defined by McPherron (1967) and Holman (1978), or roughly A.D. 0 to A.D. 650. Late Woodland begins with the Mackinac Phase and closes with the Juntunen Phase and related manifestations defined as part of the Late Woodland by McPherron (op. cit.), or the years A.D. 650 to A.D. 1450.

The first descriptions of archaeological settlement patterns in the region drew analogies with documented historic period ethnic group adaptations. Some researchers hypothesized that, in the Canadian Biome at least, warm-season coastline settlement focussing on fisheries resources, as practiced by the Ojibwa of the seventeenth and eighteenth centuries, accounted for the distribution of Late Woodland occupations at the major lakes (Fitting and Cleland However, when Late Woodland settlement choices were 1969). compared with those of earlier cultural-chronological periods, some interesting contrasts arose. When comparisons were drawn with Late Archaic settlement choices, it was clear that Woodland people favored the use of the coastlines (Cleland 1974). Cleland attributed this situation, at least in part, to the acquisition of fish nets by the Woodland people (Cleland 1966:94). Within the Woodland time period, moreover, there appeared to be an intensification of

coastline utilization. Whereas the Middle Woodland occupations of the coastlines appeared sporadically across the region as rather small solitary occupations making use of shallow-water fish species, the Late Woodland occupations were frequently stratified with many occupations, were large in area, and suggested occupation by great numbers of people for long periods of time. In addition, upon excavation, some of the Late Woodland occupations included the remains of deep-water fish species, that appeared less frequently within Middle Woodland faunal assemblages.

In 1976, Lovis and Holman presented a stress model for local Late Woodland development that cited population growth and pressure as key variables around which to understand culture change and apparent shifts in settlement patterns (Figure 2). A brief synopsis of Lovis and Holman's argument follows. People that shared the Laurel culture entered the Straits of Mackinac-Sault Sainte Marie area by A.D. 400. They occupied the main lakeshore beaches and the result was a rather intensive utilization of the region, evidenced by an increased number of sites over the number of Late Archaic components in the area. The Laurel population base and stylistic complexes provided the source for local Late Woodland developments following ca. A.D. 750, and population density increased following that time. Population growth referred both to the original expansion of Laurel people into a relatively unoccupied region and to the later development of distinctive sub-regional Late Woodland complexes.



Structure of the Woodland transition (after Lovis and Holman 1976) Figure 2.

Lovis and Holman noted that directional environmental flux was probably unimportant in causing changes in population levels because available evidence suggested no drastic change in the abundance of the key resources. Internal changes related to a larger seasonal social unit were partially responsible for the changed appearance of local Late Woodland manifestations. The instability of population growth rates during the Middle-Late Woodland periods derived from the adoption of collective practices of fishing such as net or weir technologies, which in turn encouraged higher rates of population growth and required larger seasonal groups for their successful implementation, "a major shift in intergroup integration" (1976:274).

The material results of local Late Woodland development, 1) regional differentiation of ceramic styles, and 2) reductions in the frequency of individually-operated items of fishing gear, derived from the tendency to aggregate seasonally into cooperating fishing groups. The nature of actual changes in social form, whether simply increased numbers of people present at the same place and time, or whether a novel basis for their coming together, was not developed in detail. Implicit in this argument, however, were two suggestions. First, net/weir use required more labor than the minimal Laurel social unit had to provide. Secondly, local population densities infringed on the capacity of the local environment at several critical points in time. Lovis and Holman did not specifically discuss the alterations in physical setting that might have

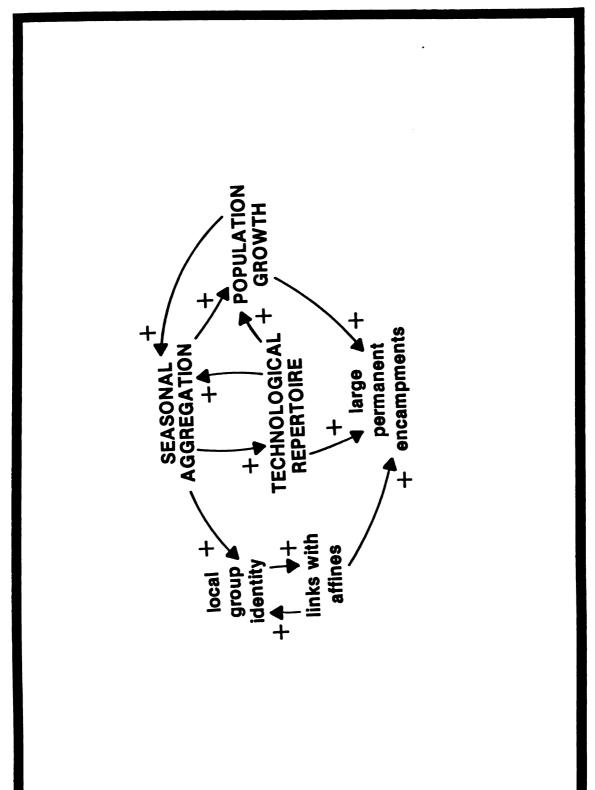
accompanied the process of subsistence intensification other than to suggest that large densely occupied sites were the archaeological result.

In response to this model, Fitting (1979) suggested that Laurel occupations of the Straits of Mackinac were internally diverse both in terms of temporal trends in artifact styles and in terms of population levels. attributed early Laurel stylistic homogeneity to widespread trading contacts between Laurel people and Hopewell cultures to the south, and the development of localized style pools to the eventual breakdown of supra-regional contacts. Population trends included an initial high density that may have declined in the period A.D. 500-700. Population growth, as described by Levis and Helman after A.D. 750, was more likely population movement to areas south of the Straits of Mackinac, but after A.D. 1350 the population of the entire region declined sharply. Fitting attributed shifts in population levels and increasing local stylistic differentiation to changes in the nature of intercultural contacts rather than to the utilization of new subsistence technologies, but he never expanded upon the nature or the causes of apparent locational shifts in settlement other than to attribute them to intra-regional movement.

Cleland (1982) attributed Woodland changes in site characteristics, both locational and depositional, to the development of improved means of fish catching and subsequent population growth, suggesting that the relatively unspecialized nets of the Middle Woodland period were

redesigned for deep-water use by the beginning of the Late Woodland period. Because deep-water fish frequented different habitats and were most readily captured in the fall, the gillnet modification not only encouraged new physical settings for settlements, but, most critically, extended the fishing season into a time of the year in which few other food resources were available. The evidence for such a shift existed primarily in changing frequencies of species of fish at Late Woodland components and secondarily in the changed locational and depositional characteristics of settlements. "The shift in settlement systems from Middle to Late Woodland seems to have accomodated the fall fishery" (Cleland 1982:775). Specifically, people began to select for specialized fishing sites as well as to occupy villages at coastlines throughout the warm seasons if those places corresponded with the preferred fall-spawner habitat. Cleland attributed apparent increases in site sizes and occupational densities during the Late Woodland time period to the discovery of and increased reliance on critical fall-season food resources and subsequent population growth (Figure 3).

Such changes of course did not exist in isolation from other, less archeologically-visible alterations in adaptive styles. Cleland suggested that the successful incorporation of the seasonally-abundant food source into the subsistence regime required a new kind of labor input. Though the gillnet technology itself did not require an increased number of individuals for its operation, the utilization of



Structure of the intensification of Woodland subsistence (after Cleland 1982) Figure 3.

the food required increased labor for preservation. If women contributed this labor, a kin or marital system that maximized seasonal nearness and cooperation among consanguine women might have developed as the adaptation to the fall fisheries intensified. Subtle shifts in kinship systems occurred that probably emphasized and strengthened links with affines, and population growth during the Late Woodland time period probably accompanied increased local group identity. These hypothetical trends were consistent with protohistoric artifactual and social organizational patterns in the region.

Both the Lovis and Holman and the Cleland models included technological repertoire, population growth, and seasonal aggregation as key elements that contributed to systemic change during the Middle-Late Woodland transition. These elements and their interactions drove system changes that were archaeologically visible as locally differentiated styles of artifacts and changed characteristics of settlement location. The basic structures of key elements and their feedback relationships are presented in Figures 2 and 3. The notation follows that of Low (1981); arrows show the direction of change and the signs the increase or decrease of effects. Dashed lines refer to external conditions.

# Interaction among model elements

In order to conceive of a model of any system it is essential to define key elements of the model and to

establish their structures and the nature of their interactions. There are, of course, almost infinite numbers of elements to consider in modelling any process as complex as one of social change. In order to simplify and render comprehensible the process of change, some elements must be emphasized and others must be eliminated or held constant. The process of selection and definition of key model elements is not only essential to problem definition but provides necessary boundaries to the system under study (Bell 1981:54-55). Thus in my view, territorial expansion is de-emphasized as a key element despite the fact that it may have played some role in the appearance of settlement system change. The more critical and, by implication, causal factors, especially in terms of evaluating standing models, appear to be population growth, seasonal aggregation, and technological repertoire. A fourth essential element, environmental condition, will also be examined, because this study focusses on the importance of environmental setting not as an external factor but as an interacting element in system change. Environmental condition refers to the stability or instability of fish abundance as an internalized system component. This view takes into account the fact that alterations in environmental condition may cause (or be caused by) alterations within cultural elements of the system and stresses both culturally-induced and stochastic fluctuations in resource abundance. The model presumes that over the

long run the system includes no directional climatic change.

Low, advancing a point of view common to systems thinking and system dynamicists in the specific sense, suggested that examination of the interaction of pairs of key elements was a basic step in the construction of a feedback model (Low 1981:277). In this manner the hypothetical direction and magnitude of effects that produced system change over time could be specified. The exercise is a basic one regardless of the utilities of the system dynamics approach itself. The model constructed in this paper does not explicitly represent a system dynamics point of view; instead it expresses a widely-held anthropological postulate: that all societal changes are both caused by and cause in turn alterations elsewhere in society, directly or indirectly. Some system responses run counter to intuition, or deduction for that matter. The system under investigation here, never observable as a social entity, is wholly an archaeological phenomenon. Nevertheless it is a patterned system whose archaeological as well as sociological aspects may be enhanced by arranging its elements and interactions as though feedback relationships accounted for all system change. Renfrew's position on the issue of systems thinking in archaeology is clearly reflected here, in which systems thinking represents "an approach to the world within an intuitional framework" and nothing more (Renfrew 1981:287).

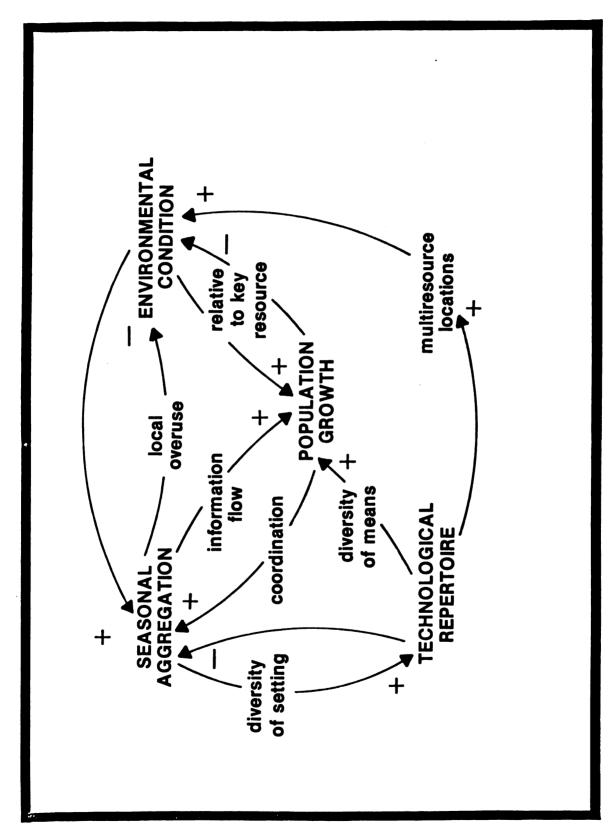
Both of the standing models treat the condition of the physical environment as a factor external to the operation

of the cultural systems involved, although environmental carrying capacity operated to damp runaway population growth. Both models assert that seasonal aggregation and technological repertoire cause increases in one another through the medium of satisfied labor demands. Likewise, both models presume that the incorporation of new technological means solves the fundamental inhibitor to growth, food shortages, and therefore increased technological repertoire boosts population levels. In both models increasing seasonal aggregation causes "increases" in local group identity. Both models tend to omit negative feedback relationships among key elements.

Figure 4 presents an alternative to standing models of system change during the Middle Woodland - Late Woodland transition. It is different because it approaches environmental condition as an internal system factor. Secondly it attempts to trace out forces of stability, that is, negative feedback relationships that might have operated to offset the positive loops of change. Finally, the model extracts common elements associated with system change in a framework that can incorporate multiple hypotheses about the Woodland transition.

Seasonal aggregation and population growth

These two elements appear to exist in a positive loop, i.e. an increase in the value associated with one leads to an increase in the other, through the media of information sharing and the coordination of resource-related efforts.



Hypothetical structure of Woodland settlement system transition Figure 4.

The emphasis here is not on the achievement of minimum labor levels for the provisioning of extra people, but on the aspects of system operation that are less directly related to actual food capture. Both standing models assert that system change involved greater and greater labor investments for some critical resource-related tasks; long-seine operation, large weir construction, preservation of periodic fish abundances are examples. These facts may well have been accurate for some periods of aboriginal life, but beyond the initiation of some novel technology, the system's dynamics probably derived from an additional source, namely, the increasing significance of the hidden costs of the new technology, in terms of managerial tasks such as the coordination of labor and the increasing importance of information about resource distributions in the face of a capture technique of great specificity. The archaeological results of this positive relationship, as noted by both of the standing models, were the large and densely occupied sites of Late Woodland times. Whether growth is responsible for the changed appearance of the sites is arguable, however, since the process of aggregation may present the same archaeological characteristics. The revised model suggests very low population growth levels for the area under study.

Seasonal aggregation and environmental condition

Some environmental conditions are at risk as seasonal aggregation creates larger and more permanent settlements;

this is especially the case when food resources are territorial. In addition a resource with yearly variations in abundance provides a limit to seasonal aggregation imposed by the probability of resource failure. Conversely, as environmental conditions improve relative to a critical resource, the limits to "greatest seasonal aggregation" would increase. The case in point may be best understood by examining the susceptibilities of the critical food resources to localized or sequential exploitation. Whitefish, for example, are apparently sensitive to repeated local harvesting. So if repeated local harvesting increased, the frequency of year-to-year resource failure for whitefish would theoretically rise. One could suggest that to some unknown point, seasonal aggregation and environmental condition are unrelated, but after that limiting point, they are negatively related. The threshold point undoubtedly varied from resource to resource. Lovis and Holman recognize this link implicitly within the general carrying capacity concept, whose limits controlled seasonal aggregation. The contention of the revised model is that seasonal aggregation itself may operate negatively on environmental condition, especially in relation to apparent key resources. The archaeological results of such a relationship might have been several: intra-regional movement, cyclical uses or periodic non-uses of some areas, and most critically the retention of diversity in site location and group size.

Population growth and technological repertoire

The standing models suggest that as technological repertoire increased, population sizes could also increase, because novel technologies increased available food supplies. Lovis and Holman suggested that this relationship included a negative loop; as population grew, the requisite technologies decreased in number because reliance on mass collection strategies replaced the individually-operated technologies of the past. In the case of the northern Great Lakes region, Cleland asserts the opposite, that novel technologies did not replace older ones, but co-existed. For Cleland, the archaeological result was more sites and larger sites in places that supported multi-season occupations.

The revised model depicts a positive link connecting these two elements, through the intervening factor of flexibility or diversity of capture technology. Historical data suggest that fish-related tasks did not require a critical labor mass for effective exploitation, except in the case of large weir construction. So the relationship between these elements is indirect.

Seasonal aggregation and technological repertoire

According to the standing models, seasonal aggregation and technological repertoire were positively related, because novel technologies became effective as their labor requirements were met. For Lovis and Holman this meant labor related to the extraction of the fish via

mass-collection devises, whereas for Cleland the labor increases derived from the support tasks of preserving abundant fish. Conversely, for Cleland at least, as technological repertoire increased, so could the tendency toward social aggregation because the labor organizational requirements of the critical preservation technology encouraged the seasonal co-residence of particular groups. However, it seems reasonable that seasonal aggregation might be damped by a diverse technological repertoire, which would provide a force for dispersal of aggregates. In the case in point these elements probably existed in some dynamic balance. Lovis and Holman close the loop between these two elements by the intervening factor of regional identity, which operates to create localized technologies and style pools, in other words, a negative link between aggregation and technological repertoire. The archaeological results of such relationships exist on two levels; for Cleland the result is again, larger sites in settings that allow settlements of growing permanence. For Lovis and Holman the result is the creation of localized style pools in ceramic decoration. The result in terms of site locational characteristics is not clear. The revised model suggests that diversity of setting and of technology provides and reflects essential system flexibility.

Population growth and environmental condition

In both of the standing models, long-term environmental changes do not act as causal agents in producing significant

system change during the Woodland time period. Likewise, short-term changes do not affect cultural systems in a significant way, other than to perhaps require a change in the scheduling of fish extraction. Some aspects of environmental condition, namely carrying capacity, offer a ceiling to growth, and indirectly to aggregation tendencies. But otherwise in both models the role of environment is exogenous and non-critical, and neither model takes into account the potential effects of cultural systems on environmental condition.

To cite environmental change as causal or as unimportant without reference to the particular resource(s) that the change will ultimately affect is incomplete. A growing population may have deleterious effects on environmental condition, at least as far as a key resource is concerned. The revised model looks at cultural systems as potentially damaging to environmental conditions, and concentrates on the characteristics of some key resources whose behavior creates conditions for culturally-induced and naturally-induced fluctuations. The characteristics of whitefish, for example, render them susceptible to local over-exploitation and temperature flux. Contrary to some other resources, long- or short-term warming trends are not ameliorating trends for whitefish, but tend to reduce fish population sizes rather decidedly, especially in marginal areas (Lawler 1965). The revised model depicts environmental condition and population growth in a negative loop that tends to keep these elements in internal balance.

This is, of course, hardly original. The results of such a balance would, in the case in point, suggest minimal growth over long reaches of time. Imbalance between these elements might be depicted by evidence of population surges, out-migration, or population crashes. The relationship between these elements and hypothetical population crashes in the area of study after A.D. 1350 (Fitting 1979) is particularly of interest.

Technological repertoire and environmental condition

In all probability these two elements existed in dynamic balance, although a broad technological range would probably minimize the impact of local exploitation on key resources. As technological repertoire decreased (i.e. became more specific to some resource), environmental condition may have deteriorated, and the susceptibility of a specialized technological system to eventual failure of the resource would increase.

If technological innovations were additive, the main archaeological result would first be greater site diversity, and later the use of site locations that accommodated all available technologies. If technological specialization (i.e. replacement of one technology with another) occurred, the frequency of very specialized site locations would increase.

In the preceding discussion I have attempted to account for relationships among key elements that may have counteracted forces of system change, and have internalized

the factor of environmental condition in discussing hypothetical system change. The revised model provides a framework around which to structure some competing hypotheses about system change, and to advance some predictions about their archaeological ramifications in terms of site locational tendencies. These hypotheses and their outcomes will be directly examined by the simulation procedure.

## Definition of the study area

A segment of the larger Upper Great Lakes region supplied the necessary data base for this research project. It seemed reasonable to center the study on a geographical area both familiar to the author and well-researched as far as archaeological survey and excavation were concerned. I considered it critical that the area exhibit the prehistoric fishing adaptation unaltered by the presence of other potentially significant storable resources such as wild rice, nuts, or grains. The study required reasonably good data on lake conditions and fish spawn information on a locality-by-locality basis. Finally, the study required reliable historical documentation for specific localities.

The study area for this project incorporated all of these qualities (Figure 5). The study concentrated on approximately 1200 kilometers of Great Lakes shoreline in the following Michigan counties: Charlevoix, Chippewa, Delta, Emmet, Mackinac, and Schoolcraft. The land area involved equalled approximately 5100 square kilometers, the

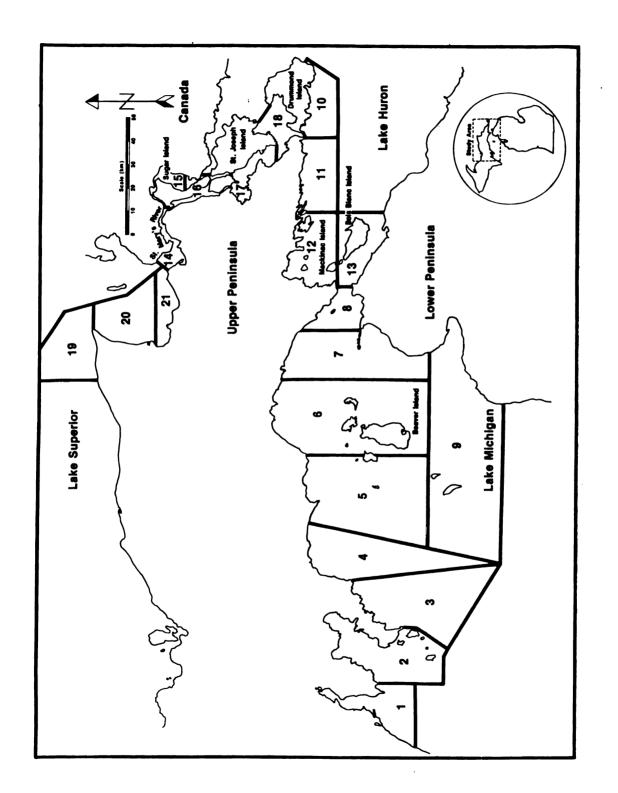


Figure 5. Analytical sectors of the study area (after Goodyear et al 1982)

total of all near-shore lands as well as the larger islands. The area included the arch of the northern shore of Lake Michigan, and extended as far south along the Lake Michigan shoreline as the southern end of Charlevoix County. This southern end represents a cultural boundary of some lengthy duration. The Straits of Mackinac, the northern shore of Lake Huron, the Saint Mary's River, and the shorelines of Whitefish Bay (Lake Superior) were also included. Figure 5 depicts the division of the study area into environmental sectors, derived from the Goodyear et al (1982) analysis of Great Lakes fishery spawn and nursery areas. These sectors provide comparisons of relative fish productivity across the study area's shorelines, and serve as the geographical units on which the simulation models will operate.

The research was regretably restricted to Michigan shorelines and water areas of the localities mentioned above, due in part to the meager data base available for some areas of contiguous Ontario shoreline. There is no doubt that archaeological data from adjacent states and most critically from the Canadian shorelines of Lake Huron and the Saint Mary's River are relevant to the problem at hand. They may stand as independent data for the examination of propositions generated by this study at such a time as adequate data are available.

By contrast, the Michigan area exhibited relatively steady archaeological research coverage, though the intensity of the coverage varied from place to place (Tables 1 and 2). Past research, conducted primarily by the

Michigan State University Museum and the University of Michigan Museum of Anthropology, suggested a widespread prehistoric adaptation to the fisheries resources. None of the areas included in the study had been the scene of highly successful prehistoric agriculture, nor were the rice-growing regions of northwest Michigan and the nut-producing regions of central Michigan included. land masses lay over Paleozoic bedrock; those areas that could be described as boreal enclaves rather than part of the Canadian Biome were excluded (Dice 1943:11). remained was, on a general scale, a rather uniform area geologically, biotically, and presumably, culturally. But within the area there existed quite varied conditions of availability of the target resource, the fish species, because the study area maintained a generous east-west and north-south spread within which local hydrologic and topographic features varied drastically. Finally, the study area included many localities that were well-described by early European witnesses to the methods and organization of the aboriginal fisheries.

### Past research levels

Tables 1 and 2 introduce data on relative component densities and levels of archaeological research intensities across the study area of interest. Coverage has been even with the following qualifications. In general, Mackinac, Emmet, and Charlevoix Counties received relatively greater attention than did the other counties. In 1984, Delta and

Chippewa Counties included large areas of unsurveyed coastline, though specific areas in these counties received careful attention in the past. Much of the southern island portion of Chippewa County remained unsurveyed, as did extensive areas of the Stonington and Garden Peninsulas in Delta County. Appendix A describes the eighty-four components included in this study, and Chapter Three includes an explanation of how the components were selected.

Table 1. Research intensity for archaeological components in the study area

<u>Highest level of research</u>	Frequency	Percent
Survey/collection (surface)	29	34.5
Controlled collection	2	2.4
Test excavation (less than 9 sq. m.)	10	11.4
Excavation (more than 9 sq. m.)	43	51.2
Total	84	100.0

# General environment

Despite stochastic fluctuations the environment of the area under study has presumably been stable for the past two-three thousand years. This is not to suggest that there has been no change, but simply to say that it has been limited to non-directional change. Drastic changes in lake ecology occurred during the nineteenth and twentieth centuries, a situation which places constraints on uses of contemporary data to reconstruct past environments relating

Table 2. Site density	and	survey i	intensity within	ty wi		sectors of the	study area
Sector/ Appro-	pprox.km. oreline 85	N Middle 0	of Lat	components	nts Total 0	Density (comp./km.)	Survey intensity
Garden Peninsula (w)	92	4	м	9	13	.14	m
Garden Peninsula (e)	42	0	0	0	0	00.	-
Manistique	40	0	0	0	0	00.	m
Seul Choix	37	-	4	0	2	.13	m
Beaver Islands	42	0	2		2	1.	m
Sturgeon Bay	09	<del>-</del>	4	8	7	.1.	4
Straits (w)	20	m	4	0	7	14	S
Little Traverse	99	m	ω	m	14	.21	ស
Drummond Island (s)	70	0	0	0	0	00.	-
Les Cheneaux	62	0	7	0	7	.03	. 7
St. Ignace (n)	78	4	ω	0	12	.15	7
Bois Blanc	88	<b>-</b>	m	0	4	.04	7
Whitefish Point	24	0	0	0	0	00.	
Whitefish Bay (n)	30	0	-	0	-	00.	2

Table 2 (cont'd.).

Sector/ locality	App	N Middle	N of components	ponent <u>Unk</u>	71	Density (Comp./km.)	Survey intensity
wniterisn Bay (s)	(S) 45	7	7	-		-	ท
Sault Ste. Marie	le 45	2	7	0	4	.08	2
Sugar Island (e)	e) 32	<b>-</b>	0	0	-	• 03	2
St. Mary's (cent'l)	ıt'l) 77	0	0	_	-	.01	7
St. Mary's (s)	99	0	0	0	0	00.	2
Drummond Island (n)	d (n) 65	ĸ	0	0	m	• 0 •	7
Total	1196	25	46	-	84		

\* estimate of area surveyed

to fish habits. However, these data are seen as adequate for a general discussion of relative areal fish potential and related archaeological site distributions. The following discussion, therefore, draws upon contemporary sources, but presumably represents the period under study as well.

According to the biogeographcal classification of Dice (1943:13), the study area falls into the Canadian biotic province, a region of climax deciduous forests that is characterized by long severe winters and brief mild summers. Local enclaves of the neighboring Hudsonian province, wherein boreal or taiga conditions prevail, are common particularly in the northern peninsula. Precontact period subclimax vegetation complexes varied with latitude and local conditions, dominated by conifer subclimaxes to the north and deciduous subclimaxes to the south. Precontact vegetation was apparently less varied in the southern portions of the study area than in the north (Veatch 1959), despite the fact that there is relatively greater topographic relief in the south.

Locally the effects of the major water bodies are quite pronounced in terms of buffering major temperature changes and producing great variation in snowfall; the entire study area exhibits a mean temperature in the 40-44 degree F. range. Slightly cooler averages prevail along the northern shore of Lake Michigan and at Sault Sainte Marie. Snowfall varies over the 80-120 inch range, with clines of heavier fall near but not at the lakeshore areas. Typically the

lakeshore receives less snowfall than adjacent upland areas. Likewise the length of the frost-free season decreases with distance from the moderating effects of the major water bodies. Cleland has suggested that the relatively longer frost-free season along the northern lower Michigan littoral may have affected the prehistoric attractiveness of the area for settlement (Cleland 1967). Overall the length of the frost-free season and the low degree-day status of the study area provided poor conditions for successful agriculture at any time in the past.

The study area's underlying bedrock varies from north The major configurations include deposits of Devonian and Silurian age. These deposits reach the surface in many areas, notably at the Garden Peninsula, at the northern Straits of Mackinac, and along the northern lower Michigan littoral, where they are associated with chert inclusions that were exploited by prehistoric inhabitants of the region (Luedtke 1976). Surface geological configurations of the study area include primarily lake-border plain, upon which are found deposits produced by inundation and recession of post-glacial lakes. At the Garden Peninsula, Drummond Island, and in the Little Traverse Bay areas, hill-lands produced by glacial morainic and drumlin deposits as well as differential erosion of resistant bedrock are exhibited (Sommers 1977:24). Drainage systems throughout the study area are relatively young in geologic terms; the entire area lacks large well-developed systems. Outlets and drainages to the large lakes in this

area are small. The largest river systems are to be found in the Little Bay de Noc region where there are several narrow watersheds.

#### Coastal contexts

There is great local variation in shoreline conditions within the study area, which no doubt related rather intimately to very specific places for prehistoric occupation. According to Goodyear et al (1982), the coast can be divided into four components or contexts, defined as follows. The mainland littoral is the area adjacent to mainland shorelines over which water depths are less than thirty feet. The tributary context is self-explanatory and includes all rivers and streams that enter the major lakes. Offshore refers to those areas removed from the mainland littoral over which water depths are greater than thirty feet, and offshore littoral refers to those places removed from the mainland littoral over which water depths are less than thirty feet. The offshore littoral includes offshore reefs, shoals and islands. These four components are a practical way of describing the structure of local coastal configurations because they incorporate discrete sub-environments for fish habitat and reproduction and provide reasonably distinct physical categories that render various technologies for fish capture appropriate or inappropriate.

Using these coastal system components as an organizing device, it is clear that the actual structure of the coastal

arrangements varies greatly across the study area. The

Little Bay de Noc sector includes a cluster of narrow

watersheds, all of which flow into a trench-like embayment

that is well-protected from the open lake while enjoying the

benefits of near-lake amelioration in weather conditions.

The bay itself has a rather wide shallow shelf that plunges

abruptly to about seven fathoms' depth. Outside of the

embayment there are large areas of offshore littoral and a

number of small tributaries.

In contrast, the west Garden Peninsula sector includes few large tributaries, exhibits a large and uniformly shallow lake embayment, and a rocky peninsula exposed to the major southwest to northeast trends in weather patterns. Yet this area is littered with islands and rocky offshore littoral structures. The east Garden Peninsula has a few tiny tributaries and a complicated rocky mainland littoral, and limited offshore littoral off Wiggins Point. The Manistique sector includes a sandy-rocky mainland littoral and a major tributary embouchure as well as limited areas of rocky offshore littoral.

The Seul Choix sector includes a very few small tributaries, and rocky uneven conditions along the relatively narrow mainland littoral. Offshore littoral areas are frequent though distant from the mainland. In the Beaver Island sector, offshore littoral areas are large, diverse and distant from the mainland. There are many large islands in this sector; the mainland has a rather broad littoral and numerous tiny tributaries.

The Sturgeon Bay sector exhibits a rather narrow mainland littoral except near Waugoshance Point. Along the south of this sector, the water attains great depth very close to shore. There are scattered areas of offshore littoral distant from the mainland. The Straits of Mackinac west sector includes a few small tributaries on the north and a relatively broad mainland littoral with gradual depth contours. The offshore littoral component is infrequent in this sector.

The Little Traverse Bay sector has a very narrow mainland littoral and the offshore littoral contexts are far from shore. The sector includes several lengthy though small tributaries. Little Traverse Bay itself is a narrow and relatively deep and exposed embayment accompanied by a narrow strip of mainland littoral. The Bois Blanc sector includes few tributaries but a number of large islands surrounded by relatively narrow littoral areas. Other areas of offshore littoral are infrequent and lie at great distances from the mainland and large islands. The sector includes one large tributary at Cheboygan.

The Saint Ignace sector includes a broad mainland littoral with gradually deepening conditions that are relatively even. There are large areas of offshore littoral near the Saint Martin Islands. In the Les Cheneaux sector the mainland littoral is rather extensive. Conditions offshore are very complicated because the entire sector is composed of a series of northwest to southeast trending shoals, islands, and peninsulas. Beyond this complex

mainland are numerous rocky reefs and other elements of the offshore littoral context. The sector includes numerous small tributaries. It is rather similar to the complicated Drummond Island (south) sector upon which there are numerous rocky points of land and an accompanying rocky narrow mainland littoral. There are relatively frequent occurrences of the offshore littoral context separated by moderately deep water. This is prime lake trout and whitefish reproduction habitat. Likewise, the Drummond Island (north) sector includes dozens of islands and some areas of shoal-like offshore littoral, but for the most part these islands are separated by very large areas of relatively shallow water of the mainland littoral context.

The Saint Mary's (south) sector includes vast areas of mainland littoral and numerous marshy tributaries. Some limited areas of relatively deep water occur in this sector as narrow channels within the Saint Mary's River system. There are few examples of the offshore littoral context in this sector. The Saint Mary's (central) sector consists exclusively of the mainland littoral context entered by a number of small tributaries. At this sector the Saint Mary's channel is very narrow and extremely shallow. The Sugar Island sector includes primarily shallow water of the mainland littoral type, but at this sector the river system broadens into Lake George and includes some of the offshore context. There are few tributaries and no offshore littoral at this sector. The Sault Sainte Marie sector includes large areas of mainland littoral, though prior to nineteenth and

twentieth-century alterations there were limited areas of water greater than thirty feet in depth. The river channel in this sector is littered with small islands and rocky rapids. At the west are extensive areas of mainland littoral through which a relatively deep natural channel passes.

At the Whitefish Bay (south) sector there are varying conditions. On the east the mainland littoral is very narrow and great water depths of the offshore context appear very close to shore. There are no occurrences of the offshore littoral context. But on the west of this sector there is a large and sporadically rocky shelf of mainland littoral, and numerous small tributaries. The Whitefish Bay (north) sector varies as well. The south end includes portions of a broad and very shallow mainland littoral shelf while on the north the mainland littoral is very narrow and the water gets deep abruptly. A major tributary enters the bay at this sector. Again the offshore littoral context is lacking in this sector. The Whitefish Point sector includes no offshore littoral, no tributaries, and an exceedingly narrow mainland littoral that is exposed to the main body of Lake Superior. The offshore context is well-represented here; on the north however the water is relatively shallow close to shore (ca. seven-eight fathoms), while on the south the water quickly attains depths greater than forty fathoms.

The presence and absence of these categories of shoreline structure helps to portray the varying opportunities for occupation by people, the local variation

in spawn/nursery habitat suitability for fish species, and varying physical conditions that influenced locally-appropriate fish capture methods. In general it seems that in the north of the study area there are relatively larger tributaries as well as a broader mainland littoral shelf. The frequency of small tributaries varies greatly over the study area but does not appear to display a geographic gradient in frequency. The largest areas of offshore littoral are to be found east and west of the Straits of Mackinac adjacent to the northern peninsula of In some sectors the absence of several of the structural contexts produced a very restricted setting in terms of habitat suitability and appropriate fishing technology (e.g. the Whitefish Point sector). Other sectors include diverse conditions that apparently allowed a range of suitable habitats and a diversity of capture techniques. The Whitefish Bay (south) sector is a good example of this situation. It is apparent that the aboriginal inhabitants recognized these differences and that the settlement distributions of the prehistoric period reflect them as well.

Relevant fauna and characteristics

Hinsdale first recognized the significance of fish habits for the distribution of aboriginal people in the area under study (Hinsdale 1932:16). The most abundant sources of animal protein were quite clearly the fish species, at least during some seasons of the year. Table 3 lists the

many species of fish upon which prehistoric capture efforts focussed, at least in the study area. The compilation of Table 3 derived from published sources of fish species identified from archaeological contexts within the study area. The table does not include all of the potentially exploited fish of the northern lakes, some of which may have contributed to the prehistoric diet, for example Coregonus artedi (lake herring). Some identifications existed on the generic or probable level and as such were not included in the table. Other species, represented by minute samples only, were not included either. Finally, some species were critically important in that they provided the forage base for piscivorous species, but because evidence is lacking that they served as human food they were not included. Thus this table represents the minimal relevant species to the prehistoric diet.

Varying fish behaviors and preferred habitats constitute critical factors in this study because these characteristics influenced directly the seasonal distribution of people in the coastal zone. Cleland (1982) succinctly summarized the bimodal spawn characteristics of the fish species of the northern Great Lakes, and suggested that the periodicity of the spawn was the primary organizing factor for Woodland-era coastal exploitation. Of prime importance were the lake sturgeon, several species of suckers, lake whitefish, and lake trout in terms of food yield, providing regularly abundant protein in the spring and fall spawn periods respectively. Many other species

Table 3. Species of fish analyzed

Table 3. Species of fish	analyzed
SCIENTIFIC NAME	COMMON NAME
Acipenser fulvescens	Lake sturgeon
Coregonus clupeaformis	Lake whitefish
Stizostedion vitreum	Walleye
Stizostedion canadense	Sauger
Salvelinus namaycush	Lake trout
Salvelinus fontinalis	Brook trout
Lepisosteus osseus	Longnose gar
Catostomus catostomus	Longnose sucker
<u>Catostomus</u> <u>commersoni</u>	White sucker
Perca flavescens	Yellow perch
Micropterus salmoides	Largemouth bass
Micropterus dolomieu	Smallmouth bass
Aplodinotus grunniens	Freshwater drum
Moxostoma anisurum	Silver redhorse
Moxostoma aureolum	Northern redhorse
<u>Ictalurus</u> <u>punctatus</u>	Channel catfish
<u>Ictalurus</u> <u>nebulosus</u>	Brown bullhead
Esox lucius	Northern pike
Ambloplites rupestris	Rock bass
Lota lota	Burbot
Roccus chrysops	White bass

<u>Prosopium cylindraceum</u> Round whitefish

were taken as well, judging from the faunal remains identified from cultural deposits. Fish populations existed at relatively constant levels of abundance through time and variation in quantity by periodic flux, by year, or by locality was portrayed as minimal by Cleland.

Because this study primarily examines the link between preferred habitats of fish and human use of those habitats as an adaptive response, the preferred habitats and particularly the spawning behaviors of the primary species of interest must be described. These habits rendered the species more or less susceptible to aboriginal capture methods. In addition their study sheds light on fluctuations in abundance to which the long-lived cultural adaptations of the study area were no doubt adjusted.

Lake sturgeon (Acipenser fulvescens) were plentiful prior to heavy exploitation by commercial fishing operations of the nineteenth century, during which time they were removed from the northern lakes as nuisances. Their slow growth and long maturation time rendered them susceptible to population imbalance produced by fishing exploitation (Smith 1972:720), yet these same characteristics probably were related to low year-to-year variation in abundance prior to severe depopulation. Preferred spawn habitat probably included shallow fast-moving tributaries or lakeshores over varied but hard bottom conditions. Movement upstream to spawning grounds begins before ice-out time. The spawn season probably began as early as April but peaked as water temperatures exceeded 10 degrees C. Goodyear et al (1982)

<13>:4) suggest that these fish were probably available in most Great Lakes tributaries. During the non-spawn seasons these fish were found in quiet shallow or shoal waters (Eddy and Underhill 1974:126-7). The sturgeon constituted a stable and widely-available seasonal resource in the tributary and mainland littoral coastal contexts.

The esocidae, of which the northern pike is an example, moved to tributaries or the mainland littoral very shortly after ice-out time for spawning. They preferred soft bottom areas with quiet water and were likely to begin spawning as temperatures rose to 10 degrees C. or above. Typical spawn depths were probably less than one fathom (Goodyear et al 1982 <13>: 44-46). In the non-spawn season they were found in diverse habitats, some in shallow moving inshore waters and others in weed beds or obstructed areas (Eddy and Underhill 1974:201, 208).

Spawning conditions for the perch family are similar to those described for the esocidae, with peak spawn occurring near 10 degrees C. in tributaries and nearshore areas. But these fish may be found at relatively greater depths, ca. three-five fathoms, for spawning, over a variety of bottom conditions. In particular, walleye may prefer hard bottom conditions and moving water (Goodyear et al 1982 <13>:131-137), while yellow perch seem to prefer quiet protected water and variable bottom conditions. Otherwise these fish, especially the walleye, may be found in shallow water or on bars or reefs in the early to mid-summer (Eddy and Underhill 1974:370). Yellow perch may be found at weed

borders or in schools in open water (Eddy and Underhill 1974:367). Some of these species, especially walleye and yellow perch, are quite susceptible to water temperature changes, as they require steady and rapid spring rises in temperature to insure successful spawning. When these conditions do not prevail a weak year class results (Hartman 1972:906). Otherwise there is evidence that cyclical percid population flux may occur on a three to five year schedule (Hartman 1972:909) at least in Lake Erie. It may also be true of the other Great Lakes (Smith 1972:721).

Several species of catostomids or suckers probably contributed a great amount of fish food to the prehistoric diet. Like the sturgeon these fish preferred shallow moving tributary or lakeshore waters for spawning. Migrations to preferred areas may have begun before ice left the tributaries but was at its peak when spring water temperatures rose to 10 degrees C. or greater. While most species preferred very shallow water (i.e. less than a fathom), the longnose sucker (C. catostomus), a lake fish, may have been found at depths to four fathoms. Typical spawn sites included gravel or sandy bottoms. These fish were abundant in the spring in the tributary and mainland littoral contexts. While the longnose sucker may have been found at great lake depths during the non-spawn seasons, the white sucker is generally found in tributaries and nearshore areas in the larger lakes after spawning (Eddy and Underhill 1974:292).

Several species of catfish have been recovered from

archaeological contexts in the study area. These fish prefer shallow tributary or littoral waters for spawning, over soft obstructed bottoms at depths that may range from one to two fathoms or greater. Generally these fish spawn as the water temperature rises above 12 to 15 degrees C., and several spawn peaks may occur (Goodyear et al 1982 <13>:92-94). Depending upon the species' preference, these fish may otherwise be found in fast-moving tributary waters or in quiet flooded backwaters (Eddy and Underhill 1974:300-302).

The bass family may spawn in tributaries and sheltered lake margins that feature protected but moving water.

Typical spawning temperatures probably exceed 12 to 15 degrees C., or early to mid-summer in the study area (Goodyear et al 1982 <13>:112-122). Most spawn in water less that a fathom in depth, over or near obstructions.

Otherwise they are to be found in clear water which for some species may include weedy shorelines, while others prefer deeper conditions over rocky or gravelly bottoms (Eddy and Underhill 1974:344-6). These fish are quite susceptible to temperature drops (Eddy and Underhill op. cit.), and such events may eradicate eggs in great numbers. Thus these species may be considered marginally adapted to some parts of the study area, and may exhibit strong year-to-year and place-to-place variations in abundance.

The trouts and whitefishes are particularly interesting because they spawn during the fall months when water temperatures are dropping to near-freezing levels. Rather

than hatching shortly after spawn, the eggs remain unhatched until the following spring when gradually warming water apparently triggers hatching. Some of these fish, especially lake whitefish (Coregonus clupeaformis), are very sensitive to water temperature fluctuations particularly in relation to the scheduling of successful spawning and hatching. Christie (1963) and Lawler (1965) have studied the relationship between water temperature characteristics and successful spawn and hatch years, and have determined that fluctuating or elevated fall or spring temperatures have significant negative effects on the strength of the subsequent year class. Recently, Henderson et al (1983), conducting research in the Manitoulin district of northern Lake Huron, have concluded that over a twelve year period of actual data collection there was no significant correlation between monthly temperature means and subsequent year class strength in Coregonus clupeaformis. However, Lawler (1965) held that monthly means obscured the very significant effects of daily temperature variation. He was able to profile a number of temperature-related variables that contributed to good or bad years for whitefish production. Lawler concluded that steady and late temperature rises in the spring and steady early temperature drops in the fall contributed to year class strength. Other environmental factors such as wind and turbidity were not critical. Temperature-induced fluctuations were most pronounced in areas that were, at least in terms of temperature regimes, marginal to successful whitefish propagation. The effect of

temperature on year class strength was apparently independent of the actual number of spawning fish, so that the <u>C. clupeaformis</u> has the ability to recover quickly from periodic drops in numbers (Smith 1972:720-721). Temperature flux is apparently related to year class strength and failure in <u>C. artedi</u> (lake herring) as well, but population rebound is probably not as rapid as in whitefish (Hartman 1972:907). Because the whitefish are rather territorial and may be subtlely adapted to local areas, they appear to be quite susceptible to local fishing pressure (Castelman et al 1981:1777, Christie 1972:904).

Whitefish generally move to spawning grounds early in the fall; spawning commences over a two to five week period as the temperature of the water drops and stays below 6 degrees C. (Goodyear et al 1982 <13>:15). Preferred sites include shoal or reef areas, and inshore shallows over rocks or hard bottom conditions. Tributary spawning for both C. clupeaformis and C. artedi was prevalent historically (Smith 1972:723), and has been documented recently at Little Bay de Noc (Goodyear et al 1982 <1>:57). Possible depths have a wide range but the preferred depths are apparently five fathoms or less (Goodyear et al op. cit. <13>:15). In the non-spawning season C. clupeaformis may be found in deep cool water over a range to thirty or more fathoms (Eddy and Underhill 1974:183-5). In cool spring waters they frequent inshore areas.

The other prime salmonid, the lake trout, appears to be more stable in terms of periodic, yearly, or locational flux

than the whitefish. Smith attributes this stability to a broad range of acceptable habitats and to a stable forage base composed to some degree of the lesser coregonines (Smith 1972:722). The question of resilience and population rebound is unsettled.

The resistance of the lake trout to intensive fishing is probably related to its unusual position in, and adaptation to, the Great Lakes ecosystem. The lake trout was the climax predator of the three Upper Great Lakes, and was the only predator that occupied the entire lake shore to shore and surface to bottom (Smith 1972:722).

Tributary runs onshore during spawning were common historically (Smith op. cit.) and have been documented in the present day by Loftus (1958) and Ryder (1972:625). Lake trout spawn in moving water over honeycombed rocky reefs or shoals close to shore in September to December as water temperatures drop to near 5-6 degrees C. Preferred depth ranges may vary from the surface to 100 fathoms (Goodyear et al 1982 <13>:38). They often use the same spawn grounds as whitefish, though the trout spawn generally occurs earlier in the fall. Otherwise these fish are found throughout the lake in cool water, e.g. less than 18 degrees C. Spring inshore movement to stream or river mouths is common (Eddy and Underhill 1974:165-7).

It is clear that many species possess rather constrained requirements for successful spawning and hatching and that flux in environmental conditions may

affect the success of spawn and hatch yields positively or negatively. Fish are most plentiful and most accessible during the periods of spawn, particularly at the tributaries and mainland littoral contexts. The spring period of spawn may, depending upon location and characteristics of the year's temperature regime, extend from ice-out time to mid-summer, whereas the salmonid spawn may extend from early fall to early winter. Cyclical variation in fish abundance is not clearly understood because for many species there are few baseline data about cyclical abundance in contexts free from culturally-induced high amplitude flux. Yet firsthand evidence and biological data would suggest that such flux is characteristic of many species. It is essential that a model of fishing adaptation over time take into account the probability of and the nature of fluctuations in abundance in terms of place-to-place and year-to-year variations.

#### Summary

The goals of the intended study include a description of aboriginal fishing methods vis a vis labor requirements, fish habitat, physiographic requirements, and implications for community organization. Secondly, the study will examine the locational evidence for shifting site distributions through the Woodland time period. Finally, the study will create a dynamic model relating resource distributions, technological repertoire and population growth to the process of settlement system transformation. The following hypotheses will be examined:

- 1) There are significant differences in the physiographic settings of Middle and Late Woodland sites.

  These differences exist both in the kinds of areas exploited and in the significance of offshore areas.
- 2) Middle Woodland site locations are variable in terms of physiographic conditions. Conversely, Late Woodland site locations are relatively uniform in terms of physiographic setting.
- 3) Balance among seasonally available fish resources is characteristic of Late Woodland site locations.

  Subsistence intensification, though implying specialization of means and localities, is evidenced by increasing efficiency of locations, where efficiency is a measure of maximum local resource availability taken over all seasons.
- 4) A simulated settlement pattern assuming the addition of the gill net to the Upper Great Lakes technological repertoire about A.D. 600-700 is consistent with real-world settlement distribution patterns.
- 5) A simulated settlement pattern assuming a low initial population, a low population growth rate, and a low resource failure rate is consistent with real-world settlement distribution patterns.

#### CHAPTER TWO

THE FISHERIES AND THEIR CONDITIONS: THE ARCHAEOLOGICAL, ETHNOGRAPHIC, AND HISTORICAL DATA BASE

This chapter examines the historical, ethnographic, and archaeological data related to warm-season fishing in the study area, and presents a description of the range of technologies, work groups, and physical remains of the prehistoric and early historic period fisheries. These data describe the known organizational variation within the fisheries adaptation, and provide the base for the simular model to be presented in Chapter Four. The study of methods in the light of labor organizational, species behavioral and geographic locational constraints or requirements is perhaps the key to understanding the fisheries-related functions of the archaeological sites within the area of study.

The archaeologically-derived artifactual, faunal and feature data appear to substantiate the presence of some historically-observed capture methodologies in the prehistoric period. The many species of fish captured by the prehistoric people of the area under study occupied a great range of aquatic environments and required a range of capturing technologies because their physiologies, spawning behaviors, and other social habits varied from species to species and from place to place. It is a fact that the historical and ethnographic records relating to the study area provide less than a perfect analogy with the conditions that probably operated in the prehistoric fisheries. There

are scanty data, for example, on the organization and division of labor within the fisheries themselves, and unfortunately the most detailed descriptions of capture methods and work group composition come from the later periods of time during which the American Indian subsistence fisheries operated, times centuries removed from the period of interest in prehistory. Additionally, the data come from a geographic area that certainly extends beyond the boundaries of the specific area under study. widespread time and space boundaries may make the use of direct analogies to the past methodologically unsound. However, the function of the historical and ethnographic data in this study is not to provide a direct analogue but a general view of the range of possibilities of fisheries variation, and to present evidence of the many ways in which subsistence fishing in the historic period took advantage of the habits of the fish to make their capture efficient and certain. I limited the review of the ethnographic and historical literature to those documents that assured first-hand observations of the area under study, or of the ethnic groups that occupied the area during the earliest historic times. I also reviewed some of the ethnographic literature documenting life among Ojibwa people who until recently practiced subsistence-level fishing in traditional These documents revealed many details about the methods and organizational requirements and results of such activities (Jenness 1935; Skinner 1911; Landes 1961; Dunning 1959a, 1959b). Though it is misleading to assume that all

of the prehistoric methodologies and their structures of use were somehow preserved within historic-period American Indian fishing habits, these observations can offer cross-cultural comparisons of variation in the solutions to common problems of fish capture and use in the Canadian Biome.

## The Methods of fish capture

There have been several attempts on the parts of fisheries historians to organize commonly-used methods of fish capture into a descriptive typology (Dumont and Sundstrom 1961, Brandt 1972). This study adapted the typology of Brandt to organize information on the study area fisheries. The division among capture types is drawn primarily on the basis of capturing tools and their structural and functional similarities. I found that most of Brandt's method types were in use at one time or another within the study area, with the potential exceptions of poisonings and trained animal retrieval, and the obvious exceptions of the methods developed by industrialized economies (Table 4).

Table 4. Typology of fish capture methods

Method	Includes
Without gear	Hands, rocks, sticks
Grappling/wounding	Spear, harpoon, bow and arrow
Lines	Hooks, gorges
Traps	Weirs, yokes
Bag nets	Dip net, scoop net
Seine nets	Long haul, stick, others
Gill nets	Free or set

## Without gear

Some fish may be caught with bare hands, with no special gear, or with ad hoc gear such as sticks or rocks. There are many ethnographic descriptions of such behavior, and its frequency could only have been constrained by the good and bad fortune of the fisher. Any individual could catch fish in this way; shallow water was probably the major requirement of this method. Fish behavior was also important as well, so this method would surely favor some species over others. Sturgeon and suckers stranded as they moved upstream to spawn would be particularly susceptible; this method would occasionally have been used at any small stream or backwater impoundment. Many of the archaeological sites within the study area probably provided the conditions for this method, but beyond local physiographic requirements the method leaves no archaeological trace.

Early reporters made some mention of this method.

Hennepin, despite his mendacity, suggested that the trout in

Lake Frontenac were so numerous that they were taken with

sticks at places where waterfalls entered the lake (Hennepin

1972:524). He also instructed his aboriginal friends in the

art of ad hoc fishing.

Whilst I was in the Mission of the Fort
Frontenac, I went to see this Leap, which comes
from a River in the North, and falls into a great
Bassin of the Lake Ontario, big enough to hold a
hundred Men of War. Being there, I taught the
Savages to catch Fish with their Hands; I caused

Trees to be cut down in the Spring, and to be rolled down to the Bank of the River; that I might lie upon them without wetting me; and after I thrust my Arm into the Water up to the Elbow; where I found a prodigious quantity of Fish of different species; I laid hold of them by the Gills, gently stroking them; and when I had at several times taken fifty or sixty large Fish, I went to warm and refresh me, that I might return fresher to the Sport: I cast them into a Sack which a Savage held in his hand. With these I fed above fifty Iroquese families of Ganneousse...(Hennepin 1972:523-4).

Likewise, Tanner described the capture of sturgeon stranded in shallow water during the spawning season; after hand capture they were dispatched with rocks (Tanner 1956:38,61). During the cold seasons a common practice was to visit holes in the ice around which small fish would gather; these fish were easily scooped out by hand in great number (Tanner 1956:73).

Reliable ethnographic and historical data suggest that some species were captured by hand or with non-specialized tools for a low risk-low return fortuitous payoff. Both Jenness (1935:14-17) and Skinner (1911:137) reported capture of sturgeon and suckers without gear (or with <u>ad hoc</u> gear such as sturgeon clubs) among relatively contemporary northern Ojibwa.

## Grappling/wounding methods

This is a highly variable methodology in terms of equipment for the actual killing of the fish, but all equipment variables reduce to the idea that the fish is pierced or grabbed and injured with some sharp object or tool. Included in this category are such diverse methods as harpooning, spearing, ginning and bow and arrow shooting. These highly diverse methods are useable amidst a number of species and are not constrained by conditions other than the depth of the water and the habits of the particular species. According to accounts of early historic period observers, these methods were most likely in use at the littoral (water depth less than thirty feet), or at tributaries.

There are many accounts of the methods of grappling and wounding throughout the literature pertaining to the study area; many of them refer to wintertime fishing for various species through the ice. Lahontan, however, described the taking of sturgeon with grapples in the summer (Lahontan 1905:361-362). Likewise LaPotherie described the spearing of sturgeon by parties of men in the early summer on Lake Huron (Blair 1911:280), and in the river of the Malhominis (Menominee River) during the ice-free seasons. The Menominee River fishery was conducted with spears on poles and took place from canoes, generally in the morning and in the evening. Charlevoix left a clear description of the methods of sturgeon fishing using the spear/canoe method.

The following is the way the Indians fish for them in the lakes. Two men place themselves in the

extremities of a canoe; the next the stern steers, the other standing up holding a dart to which is tied a long cord, the other extremity whereof is fastened to one of the cross timbers of the canoe. The moment he sees the sturgeon within reach of him, he lances his dart at him, and endeavours, as much as possible, to hit in the place that is without scales. If the fish happens to be wounded, he flies and draws the canoe after him with extreme velocity; but after he has swam the distance of an hundred and fifty paces or thereabouts, he dies, and then, they draw up the line and take him (Charlevoix 1923:221).

Hennepin left a description of some of the gear that was used for the wounding tactics; the implements were those of the Iroquois near Niagara Falls.

The first catch all sorts of Fish with Nets,
Hooks, and Harping-irons (i.e., harpoons), as they
do in Europe. I have seen them fish in a very
pleasant manner: They take a Fork of Wood with two
Grains or Points, and fit a Gin to it, almost the
same way that in France they catch Partridges
After they out it in the Water, and when the Fish,
which are in greater Plenty by far than with us,
go to pass through, and find they are entred into
the Gin, they snap together this sort of Nippers
or Pinchers, and catch the Fish by the Gills
(Hennepin 1972:522).

The wounding methods were applicable in other settings as well. Smith (1978:114-116) described the spearing of largemouth bass at a small weir, while Tanner (1956:11, 24-25) reported the spearing of fish in Saginaw Bay from a canoe. McKenney left a detailed account of flambeau fishing in the Saint Clair River area. The fishing took place one night in June of 1826 in calm inshore waters. It required three people, two to hold the trident on a pole, and one to hold the rolled flaming bark that served as the torch. The fish were attracted to the light and were speared. In the evening during which McKenney observed the fishing, bass, pickerel, pike, and sheepshead were taken (McKenney 1827:151-152). Lanman suggested that the method was equally useful for trout during the fall when the fish came nearshore (Lanman 1856:124).

Kohl left an extended account of spearing methods and equipment during all seasons of the year.

Of all the varieties of fishing, the one best suited to a hunting people appears most extensively used---namely, spearing. Most astounding are the many sorts of fish lances they have invented, and how cleverly they use them.

And we might draw the conclusion from this fact, that the people were at first exclusively hunters, and then at length applied their hunting operations to fishing, thus converting Diana's hunting spear into Neptune's trident.

They spear fish in winter and summer, by

night and by day. They spear the huge sturgeon and the little herring-often, too, even the smaller fish...I saw nearly all their varieties of fishing spears. They call them generically "anit" but have special names for the various sorts. They all appeared to be very neatly made, and admirably adapted for the purpose. Some had two prongs, others three. In the trident the center prong is shorter than the other two, which diverge slightly. At times they use several short central prongs, while they have all barbs on the outer sides. For catching larger fish they also have a species of spear head, which, on striking, comes loose from the pole, and is merely attached to it by a cord. The fish darts off, dragging the wooden bob after it, gradually becomes exhausted, and is captured without difficulty (Kohl 1956:328-331).

Some observers left a record of yet another elaboration of the wounding tactic. Schoolcraft reported that bows and arrows sometimes served as fising devices.

Sometimes fish are shot with an arrow, by a watcher sitting on the banks of the river, when the fish approach the land in their vernal track of migration...Fish are also speared from a canoe, usually in the morning, when they are close in-shore, lying under the leaves and rushes that grow on the banks of streams. An Indian woman or

boy paddles the canoe gently along the shore, while the man stands up in the bow or on the gunwales of the canoe, holding his spear ready to strike the fish when seen (Schoolcraft 1851-1856 <v.2>:52-53).

Judging from the historical accounts of wounding methods, the organization of labor for the warm season spear fishery involved, generally speaking, two or three people at the most (if from a canoe), and a single fisher if the fishing took place from the shore. Apparently the task of handling the canoe fell to a woman, a boy or a man, while the actual spearing was usually done by a man or two.

In terms of location and species susceptibility the wounding methods appear to be highly versatile. combination with the use of the canoe, and the use of a lure such as light to entice deep-swimming fish to the surface, these methods could apparenly dispatch many different species. There are few cases of the use of this method on whitefish, though Schoolcraft left one such account (1851-1856 <v.2>:53). As far as locational criteria were concerned, the attributes of the particular fishing location did not seem to make much difference, because the methods were equally useful in quiet streams, adjacent to weirs, or in the open water of the larger lakes. This versatility extended to the cold seasons as well; in fact the descriptions of Kohl, despite his claim to the method as one for all seasons, are largely those of cold conditions over ice cover. The methods are those of small labor forces,

composed of one-three persons, that return a relatively stable yield with relatively low risk of failure involved.

The various designs of harpoons commonly found in archaeological contexts in the study area constitute some of the best technological evidence for the use of any method of fish capture (Table 5 and 6). It is clear as well that items of technology such as projectile points, long assumed to play a role in land-based food capture, may be relevant to fish wounding. The components of gins, tridents and leisters are rarely reported in tool inventories from the study area, but commonly-observed artifact types such as copper awls may have played a role as elements of the grappling/wounding assemblage. It is interesting to note that Kohl explains differences in harpoon design by variations in prey size, whereas the standing archaeolgical explanation for this difference lies with temporal-cultural causes (Mason 1981).

Jenness (1935) found that the spearing of fish, in combination with small stone weirs, was a common practice among the northern Ojibwa of Parry Island, Ontario, especially in the spring. They prey were typically suckers and pickerel. Skinner (1911) documented the spearing of fish by pairs of men.

# Lines (methods of angling)

There are numerous accounts of fishing with hooks, lines, lures, and other equipment designed to take individual fish, but most of them describe ice-fishing

rather than warm-season open-water fishing. Whatever the case with the historical record, there remains a substantial inventory of angling equipment from the study area that verifies the prehistoric use of these methods. The following accounts focus on descriptions of open-water warm-season angling.

Sagard was probably the first observer to relate evidence of the angling method, recording the practice of dragging a line behind a travelling canoe, "putting on and fastening to the hook a piece of skin cut from a frog" (Sagard 1968:60). Lahontan, writing of seventeenth century Michilimackinac, was a bit more specific of prey and equipment. "Here the Savages catch Trouts as bigh as one's Thigh, with a sort of Fishing-Hook made in the form of an Awl, and made fast to a piece of Brass wire, which is joyn'd to the Line that reaches to the bottom of the Lake" (Lahontan 1905:148).

Other species were susceptible to the hook as well.

Tanner took dory (yellow perch) via hook and line near the

Source River and Rainy Lake (Minnesota) embouchure,

apparently during the season of the sturgeon spawn (Tanner

1956:61). Carver suggested that "trout might be taken at

all times with the hook" in Lake Superior and that herring

was a common bait for trout (Carver 1956:140). An

apparently naive H.R. Schoolcraft recorded that at

Michilimackinac, the whitefish, highly esteemed, was taken

with hook and line (1970:118-119). But after a residence of

thirty years, his assessment of the methods of fishing

altered.

The fish-hook is employed chiefly in deep waters, and is intended for the larger species. The white fish, so common to the whole line of lakes, never bites at a hook, and is captured solely by nets or spears. The ordinary trout and cod hook has been supplied by commerce since the discovery of America; but the ancient Indian hook of bone was shaped much like it, and its use was every way similar (Schoolcraft 1851-1856 <v.2>: 53).

Likewise Hearne left accounts of angling in open water, apparently as a method of last resort when fishing with nets was unsuccessful (Hearne 1958:12). Angling for trout was practiced by those with Hearne (probably Cree) during the summer of 1770. In this account Hearne left a lengthy description of angling magic.

The methods used, and strictly observed, when angling, are equally absurd as those I have mentioned; for when they bait a hook, a composition of four, five, or six articles, by way of charm, is concealed under the bait, which is always sewed round the hook. In fact, the only bait used by those people is in their opinion a composition of charms, inclosed within a bit of fish skin, so as in some measure to resemble a small fish. The things used by way of charm, as bits of beavers tails and fat, otter's vents and teeth; musk-rat's guts and tails, loon's vents,

squirrel's testicles, the cruddled milk of sucking fawns and calves, human hair and numberless other articles equally absurd.

Every master of a family, and indeed almost every other person, particularly the men, have a small bundle of such trash, which they always carry with them, both in Summer and Winter; and without some of those articles to put under their bait, few of them could be prevailed upon to put a hook into the water, being fully persuaded that they may as well sit in the tent, as attempt to angle without such assistance. They have also a notion that fish of the same species inhabiting different parts of the country, are fond of different things; so that almost every lake and river they arrive at, obliges them to alter the composition of the charm. The same rule is observed on broiling the first fruits of a new hook that is used for a new net; an old hook that has already been successful in catching large fish is esteemed of more value, than a handful of new ones which have never been tried (Hearne 1958:212).

One could conclude several facts from these brief accounts. First, the use of single-fish devices such as hooks, lures and lines seems a rare practice during the warm seasons. The abundance of accounts for ice-fishing, however, suggests that many large species, especially

piscivorous trout and pike, were taken using these methods during the cold seasons. There are no accounts of set lines or trot lines that derive from the study area.

The methods described seem constrained most by season and by the habits of the target species rather than the availability of some minimum work group. In some places supernatural beliefs conditioned angling methods. Perhaps the attempts to predict the preferences of the fishes may have randomized the bait and improved the overall chances of fish capture. Data on work group composition for angling during the warm seasons are sparse, but those few clearcut cases document that it was an activity for individuals rather than one in which large groups took part. Winter ice fishing was sometimes done by women and juveniles as well as by men. The production of gear and the actual procurement apparently did not depend on a work group of larger than one individual, and the gear was ultimately portable.

Hooks, gorges, lures and materials related to angling for fish have been recovered at a number of archaeological sites within the study area (Tables 5 and 6). Jenness' (1935) descriptions of angling refer solely to wintertime fishing, as do Dunning's accounts (1959b). In both cases the fishing was done by women and children.

## Traps (weir methods)

Traps include devices that enable the fish to enter but prevent them from leaving; accordingly Brandt's types depict a highly variable methodology (Brandt 1972:223-224). Within

the study area there is currently no archaeologically derived evidence for the use of weirs. But there are many detailed descriptions of such devices derived from the centuries after contact, including those of LaPotherie, Allouez, Henry, and Schoolcraft. From these descriptions it appears that there were many ways to build a weir, and that one of the more obvious constraints on the construction of a weir was the size of the stream that it was to close. Some weirs apparently extended across a tributary and provided a strategic perch for the individual fisher, who then dispatched the fish with some other device such as a spear. Other weirs channelled the fish into some device, such as a net, a basket, or a sieve. Still others were rather informal devices such as lines of rocks or brush.

Champlain's classic description is perhaps the earliest account of an Upper Great Lakes weir, which he observed one summer early in the seventeenth century among the Huron of Lake Simcoe, Ontario.

...passed along the shore of a small lake distant three leagues from the village, where they catch large quantities of fish, which they preserve for the winter. There is another lake, closely adjoining, which is twenty-five leagues in circuit, and flows into the small one by a strait, where the above-mentioned extensive fishing is carried on. This is done by means of a large number of stakes which almost close the strait, only some little openings being left where they

place their nets, in which the fish are caught (Champlain 1907:287).

Both Allouez and LaPotherie visited operating weirs at the close of the seventeenth century; the locales were directly adjacent to the area currently under study.

Allouez, visiting a group of Saks at the Fox River about April, 1667, described a weir located about four leagues upriver from the settlement. The Saks fished there for sturgeon as well as other species. "They call this contrivance mitihikan, and it serves them during the spring and a part of the summer" (Kellogg 1945:150). LaPotherie, citing neither the location nor the target species for fishing, left the following description. The location, judging from accompanying material, was probably northern or western Lake Michigan.

Although their rivers are deep, they close the stream with a sort of hurdle, leaving open places through which the fish can pass; in these spaces they set a sort of net which they can cast or draw in when they please; and several small cords are attached, which, although they seem to close the opening, nevertheless afford passage to the fish. The savages are apprised of the entrance of the fish into the net by a little bell which they fasten on the upper part of it; when this sounds, they pull in their fish. This fishery suffices to maintain large villages (Blair 1911:305).

The weir described by Henry and Schoolcraft near the

Ontonagon River embouchure to Lake Superior leaves no doubt about the relevance of this method to the general area under study. Henry, who visited it during August of 1765, described the sturgeon as abundant. He gave the location as three leagues from the river mouth (Henry 1969:186). Schoolcraft left a detailed account, both of the construction and the operation of the weir. His visit there took place during June, 1820. He gave the location as four miles upstream, describing the weir as extending from bank to bank at a rapid, where the water was about four feet deep.

This wier (sic) is constructed to saplings, and small trees, sharpened and drive into the clayey bottom of the river, with an inclination down stream, and, supported by crotched stakes bracing against the current. Against the sides of these inclined stakes long poles are placed horizontally and secured by hickory withes, in such a manner as to afford the Indians a passage from one end to the other, and at the same time allow them to sit and fish upon any part of it. The sturgeon are caught with an iron hook, fixed at the end of a long slender pole, which the Indian, setting on the wier holds to the bottom of the river, and when he feels the fish pressing against the slender pole, jerks it up with a sudden and very dextrous motion, and sledem (sic) fails to bring up the sturgeon. On one side of the wier, an

opening is left for the fish to pass up, which they do at this season in vast numbers, but in their descent they are hurried by the current against the hooks of the savages, who are thickly planted on every part of the wier (Schoolcraft 1970:172-174).

Schoolcraft left descriptions of other weir designs, but unfortunately the locations of such devices were not revealed.

During the low waters of the summer solstice, lines of stones are placed from each bank, where the river has a marked descent pointing downwards at an acute angle, until they meet, within three or four feet. This space is filled with stones of a less height, over which the pent-up and dammed water rushes and falls on a platform of poles. This platform, which performs the purpose of a gross longitudinal sieve, lets through the water, leaving the fish to flounder and be picked up - ad libitum. This contrivance is sometimes called namekowagon, or sturgeon's yoke (Schoolcraft 1851-1856 <v.2>:52).

Still other designs were recorded by some observers.

James Smith, while a captive of a mixed group of Wyandots,

Chippewas, and Ottawas, helped to build a small weir at a

creek near the Sandusky River. They then caught spawning

largemouth bass via spear and torchlight (Smith

1978:114-116). In Smith's account one sees the flexibility

of the weir technology, for it is not the case that the method involved a large group; a temporary weir for one season and the spawn of one species was a possibility.

These accounts lead to the following conclusions. methods were apparently most common in the early and middle warm seasons, and there were suggestions that weirs may have been used throughout the warm season. There were many ways to build a weir, that accomodated the size of the tributary, the depth of the water, the habits of the fish, and the method of the final kill. The size of the operating group varied as well, and the yield was sometimes a collective matter, sometimes an individual one. But with a large weir, a number of strategic choices were made. Unlike other fishing devices, weirs were not portable, and the large ones were quite obviously not the property of individuals nor of households. A local resource failure at a permanent weir would have had dire effects on a social group committed to its maintenance and use. The descriptions of weirs that derive from the historic period near the study area suggest that there must have existed some body of ideas about who had the right to a weir location, and who had the obligation to maintain the device year to year. Smaller weirs did not necessarily involve heavy outlays of labor nor decisions about future maintenance. Both Jenness (1935) and Skinner (1911) suggested that the use of informal stone weirs and brush traps was a common spring-season practice among relatively contemporary northern Ojibwa. The usual prey were spring-spawning suckers, pickerel, and sturgeon.

#### Bag nets

This method involved the catching of fish by snaring them in a small net attached to a pole by which an individual ladled the fish out of the water and onto the bank or into a boat. No fishery is as well described as the dip net fishery for whitefish that was conducted in the rapids of the Saint Mary's River near Sault Sainte Marie. Dablon left one of the earliest accounts.

It is at the foot of these rapids, and even amid these boiling waters, that extensive fishing is carried on, from Spring until Winter, of a kind of fish found usually only in Lake Superior and Lake Huron. It is called in the native language <a href="https://doi.org/li>
<a href

Dexterity and strength are needed for this kind of fishing; for one must stand upright in a bark canoe, and there, among the whirlpools, with muscles tense, thrust deep into the water a rod, at the end of which is fastened a net made in the form of a pocket, into which the fish are made to enter. One must look for them as they glide between the Rocks, pursue them when they are seen; and, when they have been made to enter the net, raise them with a sudden strong pull into the canoe. This is repeated over and over again, six

or seven large fish being taken each time, until a load of them is obtained.

Not all persons are fitted for this fishing; and sometimes those are found, who, by the exertion they are forced to make, overturn the Canoe, for want of possessing sufficient skill and experience (JR v.54:129-131).

Other observers leave essentially the same description of the fishing (Henry 1969; Schoolcraft 1851-1856; Lanman 1856). The net itself was described by LaPotherie as "a net which resembles a bag, a little more than half an ell in width and an ell deep, attached to a wooden fork about fifteen feet long" (Blair 1911:276). The season for whitefish lasted from May until November according to Johnston (Masson 1960:147-8). McKenney, however, observed that the season for whitefish occurred from May until August and again from September until November; brook trout were also taken there (McKenney 1827:193).

Carver, who observed the Sault Sainte Marie fishery in October of 1767, described the method without the canoe element.

At the bottom of these falls, Nature has formed a most commodious station for catching the fish which are to be found there in immense quantities. Persons standing on the rocks that lie adjacent to it, may take with dipping nets, about the months of September and October, the white fish before mentioned...(Carver 1956:142-143).

Fishing with dip nets was not, however, merely a local occurrence. Copway mentioned it as a feature of native life during the spring and fall at Fond du Lac (Lake Superior).

"Late in the fall, white fish ascend the rapids, and can be scooped up with nets. In the spring, fish of every kind, and in great abundance, ascend these rapids" (Copway 1847:125).

Judging from the accounts included here, the dip net fishery was prolific at certain seasons and given certain settings. All descriptions of this method of fishing restrict it to river rapids, shallow water, and either canoe or from-the-bank locations. From the banks the fishery could be carried out by a single individual, and from the canoe the fishery required two people. These fishers were usually two men, but pairs of juveniles and pairs of juveniles and adults are also reported (McKenney 1827:193).

Arguments concerning the pre-contact frequency of adfluvial fish spawning, especially in relation to the lake trout and whitefish, have partially been answered by the research of Loftus and others (Loftus 1958; Ryder 1972; Smith 1972). Prior to the destruction of habitat the adfluvial spawn was assuredly more common than at present. It apparently occurred at the Michipicoten River (Keating 1825:186-188), at the Little Dog River (Perrault 1910:618), and at the outlet of Lake Huron (Milner 1874). It is reasonable to expect that it occurred in the larger rivers and tributaries of the study area, and in all likelihood the dip net methods were widespread as well. There is, however,

no archaeological evidence, apart from the placement of some stratified habitation sites at river banks, to support this suggestion.

### Seine nets

Despite the frequency of mention in the historic record of the use of fish nets by the native residents of the Upper Great Lakes, there are relatively few instances in which a concise description of the configuration of the nets themselves is given. Even less information is available about the composition of the work groups for various modes of net fishing, and the characteristics of place, season and quarry that may have influenced net operation. Many of the clearest descriptions about net fishing pertain to winter seasons during which ice cover both facilitated and limited the kinds of fishing that could be carried out. Of course the discussions of net fishing associated with ice cover are peripherally interesting to the current discussion, but full discussion will be restricted to those methods in use during the warm seasons.

The seine net category of Brandt is extremely diverse, and includes all kinds of net configurations that involve the submersion of a net in the water to entrap, but not necessarily entangle the fish. Brandt describes a seine as

a gear in the genuine type with very long wings and towing warps with or without bag or bags. The mode of capture is by surrounding a certain area and towing the gear over this area with both ends

to a fixed point on the shore (Brandt 1972:225).

I have included the type "dragged gear" in the seine category. Brandt describes this gear as follows:

This group contains all netbags or netwalls which are towed through the water on or near the bottom or even pelagically for an unlimited time. The manner of capture is filtering the passive prey by the active moved gear (Brandt 1972:224).

Virtually every early observer of Upper Great Lakes fishing documented the use of nets by native people. Some of these nets were without a doubt of the seine variety. It is likely that the arrangement of nets varied with the characteristics of a particular location, the intended prey, and the labor force available. To our great misfortune, many of the accounts do not describe the organization of the labor involved nor details about seine operation. Seines were a flexible technology; they could be used at any season of the year, and were used under the ice as well as in open water. Champlain observed the winter seine fishery among the Huron some time during the years 1615-1618.

They make several holes in a circular form in the ice, the one where they are to draw the seine being some 5 feet long and three wide. Then they proceed to place their net at this opening, attaching it to a rod of wood from six to seven feet long, which they put under the ice. This rod they cause to pass from hole to hole, when one or

more men, putting their hands in the holes, take hold of the rod to which is attached an end of the net, until they unite at the opening of five to six feet. Then they let the net drop to the bottom of the water, it being sunk by little stones attached to the end. After it is down they draw it up again with their arms at its two ends, thus capturing the fish that are in it (Champlain 1907:331).

Sagard's early description of a seine operation told little about its particular shape, operation, or labor requirements, other than to remark that the fishing was done communally.

At another season they catch with the seine-net a certain kind of fish that seems to be a species of our herring, but smaller, and these they eat fresh and smoked. And as they are very clever, like our cod-fishermen, in knowing within one or two days the time when each kind of fish appears, they do not fail, when it becomes necessary, to go after this little fish, which they call <a href="Auhaitsiq">Auhaitsiq</a>, and they catch an immense number with their seines. The catching of small fish is done in co-operation; then the division is made by great bowlsful, and in this we had our share as fellow-townsmen and residents (Sagard 1968:231).

Among the Iroquois, Hennepin apparently observed a

river haul seine operation.

The Iroques in the fishing season sometimes make use of a Net of forty or fifty fathom long, which they put in a great Canow; after they cast it in an oval Form in convenient places in the Rivers. I have often admired their dexterity in this Affair. They take sometimes four hundred white Fish, besides many Sturgeons, which they draw to the Bank of the River with Nets made of Nettles. To fish in the manner, there must be two Men at each end of the Net, to draw it dexterously to the shoar (sic). They take likewise a prodigious quantity of Fish in the River of Niagara, which are extreamly (sic) well tasted (Hennepin 1972:522-523).

Sagard's description of a Huron haul seine is essentially the same; their prey included sucker, sturgeon and pike (Sagard 1968:60).

Yet another configuration of a seine net was reported by Grant, who observed its operation among the Sauteux early in the nineteenth century.

They fish with nets, lines and spears, but they have a method of taking sturgeon with a kind of drag-net or seine, which, I believe, is peculiar to themselves. The net use for this puspose (sic) is about 20 feet long by 6 feet deep, when shut double. It is dragged between two small canoes, having two men in each; while the bowmen paddle

gently down the stream, the men in the sterns hold the seines by meansof long cords, fixed to each end and which can be shortened or lengthened according to the depth of the water and the wish of the seineurs. Two stones are suspended from the lower ends of the seines, by which the nature of the bottom and the soundings are ascertained, a very necessary precaution to keep the whole clear of foul bottom. The course of the canoes must form an obtuse angle with the middle of the seine.

nets, with small knobs of cedar fixed to the upper border instead of cork. When, by the vibrations of the cords, they perceive that fish is taken, they instantly haul up and paddle with all their might to bring the canoes together and, thereby, shut up the fish in the seine. This method of fishing is, of course, practicable only in rivers, narrow channels and small bays, where the bottom is clear (Masson 1960:345-346).

These accounts suggest that a seine required a minimum of four people, if a long haul seine such as that described by Grant, or six people, if a river seine such as that described by Hennepin. Other possible configurations would have required fewer persons, such as a seine under ice. A stop seine, i.e. a staked fixed-position submerged net, probably allowed a solitary person to take fish. There are no accounts of such devices, however, recorded in the early

historic literature from the area of study. Sagard's account clearly depicts the communal nature of some forms of net fishing.

Different geographic locations required different seine configurations, but a general characteristic seemed to be a smooth and even river or lake bottom. Some configurations required additional tools, for example the canoe itself or perhaps a scoop net, spear or container. Depending upon location, configuration, and season of use, the seine was apparently effective against all economic species of fish.

Traditionally, the body of archaeological data that evidenced the use of nets as prehistoric fish capture devices was the appearance of the net sinker, a small flat, oval or circular notched stone that was tied to the bottom of the net either to carry it to the bottom or to cause it to hang perpendicularly. Net sinkers are frequently recovered from archaeological sites within the study area, but unfortunately there is no means of concluding on what configuration of nets they were used. It has been suggested by Weston (1978) that seine nets, presumably of every configuration, required sinkers of uniform size, whereas gill net sinkers varied greatly in size and weight. addition Weston devised formulae to predict the lengths of nets from clusters of sinkers recovered archaeologically. But because of the great variety of seines in the historical literature, some of which probably required no sinkers at all to function effectively (for example seines staked in place or short seines attached to sticks), there seems no

compelling reason to assume that all seines required a uniform number of sinkers, or that sinker frequency or presence was solely related to the length of the net. It seems plausible that sinker frequency or presence was related to bottom conditions and the particular shape of the seine in use at a place. With the use of gill nets, whose position in the water was critical for success, uniform sinker size would make good functional sense. Given, however, the wide range of possible seine shapes, one could logically expect a corresponding range of ways in which they were weighted.

Other lines of evidence used to support the suggestion of seine use have been the continuous size ranges of the fish faunas recovered archaeologically (for example Lovis 1973), and the presence of net impressions on prehistoric ceramics from sites in the area (Hamilton et al 1982). Additional support comes from the presence of netting tools such as shuttles within local assemblages. But these discoveries unfortunately fail to confirm that a particular net configuration was in use. Tables 5 and 6 include a number of examples of archaeologically-derived implements that are probably related to net manufacture.

#### Gill nets

These nets were suspended in the water (either on the bottom or floating), and were of a uniform mesh size which entangled the head and gills of a target fish and prevented its escape. There are limited accounts of the setting and

tending of gill nets during the warm seasons contained in the historic record of the area, but there are many clear descriptions of the use of this technology under the ice. I do not think that this fact means that gill nets were used exclusively in the cold seasons, but simply that it was relatively easy to observe the setting and tending of the net when ice cover rather than canoe travel provided the route to the net. The earliest account of the methods of open water gill net fishing was left by Joutel, who witnessed net fishing by Huron expatriates at Mackinac about 1687.

They are very skillful at fishing, and the fishing is very good in these parts. There are fish of various kinds which they catch with nets, made with a very good mesh; and, although they only make them of ordinary sewing thread, they will nevertheless stop fish weighing over ten pounds. They go as far as a league out into the lake to spread their nets, and to enable them to find them again they leave marks, namely, certain pieces of cedar wood which they call aguantiquants, which serve the same purpose of buoys or anchors. They have nets as long as two hundred fathoms, and about two feet deep. At the lower part of these nets they fasten stones, to make them go to the bottom; and on the upper part they put pieces of cedar wood which the French people who were then at this place called floats. Such nets are spread in the water, like snares among crops, the fish being caught as they pass, like partridges and quails in snares. The nets are sometimes spread in a depth of more than thirty fathoms, and when bad weather comes, they are in danger of being lost (Margry 1876-86 <v.3>:503).

Other detailed accounts of the methods and procedures of gill netting refer to conditions over ice. Hearne, witnessing the operation during the winter of 1770-71, described setting a net through the ice.

To set a net under the ice, it is first necessary to ascertain its exact length, by stretching it out upon the ice near the part proposed for setting it. This being done, a number of round holes are cut in the ice, at 10 or 12 feet distance from each other, and as many in number as will be sufficient to stretch the net at its full length. A line is then passed under the ice, by means of a long light pole, which is first introduced at one of the end holes, and by means of two forked sticks, this pole is easily conducted, or passed from one hole to another, under the ice, till it arrives at the last. pole is then taken out, and both ends of the line being properly secured, is always ready for use. The net is made fast to one end of the line by one person, and hauled under the ice by a second; a large stone is tied to each of the lower corners,

which serves to keep the net expanded, and prevents it from rising from the bottom with every waft of the current.

In order to search a net thus set, the two end holes only are opened; the line is veered away by one person, and the net hauled from under the ice by another; after all the fish are taken out, the net is easily hauled back to its former station, and there secured as before (Hearne 1958:11-12).

Schoolcraft's description, again of a gill netting operation conducted under ice, is essentially the same as Hearne's, but gives some insight into the particulars of fishing at Michilimackinac.

Another mode of taking fish in the winter, is by making a series of orifices, through the ice, in a direct line. A gill-net is then pushed, by its head-lines, from one orifice to another until its entire length is displayed. Buoys and sinkers are attached to it, and it is then let down into deep water, where white fish, and other larger species, resort at this season. The next morning the net is drawn up, the fisherman secures his prey, and again sets his net as before. By this mode, which is very common throughout the lakes where deep water abounds, these species are captured at the greatest depths, while sheltering themselves in their deepest winter recesses. Fish are sometimes

brought up in the immediate vicinity of
Michilimackinac, from a depth of eighty fathoms.

The Indians' ingenuity in capturing the finny
tribes during the prevalence of the severities of
winter, may be quoted as an evidence of their
resources, in sustaining themselves (1851-1856
<v.2>:51-52).

The adaptibility of this method to particular physical conditions is documented by Allen, touring the southern shore of Lake Superior in 1832. Speaking of Whitefish Point, he remarks

This point is remarkable and important as a fishery of whitefish---as affording more, and a better quality, of that excellent fish, than any other fishery of the southern shore of the lake yet explored. It has long been known as a point where this fish could be taken in gill nets at certain seasons of the year; but no use was made of it, more than is at present of several other fisheries of the lake, where a few Indians, or an individual trader, procure only what is necessary for the immediate subsistence....

The fishery, as at present developed, commences at the Shelldrake River, nine miles from the end of the point, on its eastern shore, and extends round the point and along the southern shore of the lake, as far as the Grand Marais, or the commencement of the Grand Sable, a distance of

fifty-four miles. The bottom along this part of the coast is sandy, and falls off gradually into deep water, and the shore is a sandy beach -circumstances favorable to the safety and easy working of the nets. The fish occur in equal numbers in every part of its whole extent, but the point is the most desirable locality, from its generally affording, on one or the other side, a lee, and smooth water, where the nets may be used during winds. The fish are taken by means of the gill net alone; the meshes of which are of a size adapted to the fish's head, so as to fasten in the gills when the fish attempts to withdraw its head, after having inserted it in an attempt to force its way in the direction of its movement. nets are generally eighty fathoms long, and from five to ten feet broad, according to the depth of the water; and are set in a vertical position by leads or sinkers that rest on the bottom, and floats of sufficient buoyancy to support the weight of the net and hold it up....

The fishing season commences here in the spring, (when the largest and best fish are taken), about the last of April, and ends about the last of June; and in the fall, occurs in October and part of November; making the whole season a little more than three months...It is remarkable that at no other known fishery of the

lake can the whitefish be taken in quantities in the spring...(Mason 1958:165-166).

The considerable length and detail of this passage gives a clear picture of the nineteenth century gill net fishery on Lake Superior, and of some of the physical conditions under which it operated.

There is, again, only limited information regarding the organization of labor within the gill net fishery. Allen's account stated that the tending of nets was accomplished by two people, presumably men (Mason 1958:166). Likewise, Tanner's account of net fishing for trout and whitefish at Moose Lake (Minnesota) during a summer night in the eigtheenth century stated that two boys tended the nets (Tanner 1956:22). Otherwise, constraints on the operation of gill nets probably related to weather and water conditions, and secondarily to physiographic conditions such as smooth sandy bottoms. However, it is certainly the case that smooth bottom conditions and sandy beaches were not necessary for the use of gill nets since gill net configurations varied and did not always rest on the bottom, nor were smooth sandy bottoms necessarily the preferred habitat of the target species.

Archaeological evidence for gill net use has in part centered upon the presence of net sinkers and related implements in an assemblage (for example Weston 1978), but stronger evidence exists in the presence of net-susceptible faunas such as the various salmonids in an assemblage (Cleland 1966). Discontinuous ranges of fish sizes also

suggest the use of uniform and deliberately selective net mesh sizes (Lovis 1973). Geographical and particularly offshore physiographic situations have also been suggestive of gill net use, especially in combination with one of the other categories of archaeological data. These data presume that gill nets enabled the exploitation of the offshore and offshore littoral situations over which no other prehistoric gears and methods were effective (Cleland 1982). Otherwise it is difficult if not impossible to distinguish archaeologically between the presence of seine net technology and that of gill net technology.

Landes (1961), Dunning (1959b), and Skinner (1911) all reported the use of the gill net during warm seasons among relatively contemporary northern Ojibwa people. In all cases the tasks associated with the warm season gill net fisheries were conducted to some degree by women as well as by men. Among Landes' informants a woman tended her nets alone, or with an adolescent, or occasionally with her husband. All accounts, including those of the historic period, attest to the low numbers of persons involved in tending the nets; the tasks associated with setting and collecting the nets typically involved two persons. Many historical accounts document the physical dangers involved in the setting and tending of the nets, and to the constraints presented by unfavorable winds and currents, which operated to produce both risk of life and risk of equipment failure in gill netting (JR 54:151; Lahontan 1905:147; Blair 1911:287). Accordingly, Cleland termed this a high-risk high-return strategy (Cleland 1982).

## Fish Preservation

Many historical accounts include details about the methods and frequency of fish preservation. Obviously the preserving of organic food items was one of the earliest technological developments of people and probably occurred in the distant past of the species' evolutionary history. But recent accounts of the processes of fish preservation suggest that there are certain characteristics of fish as potential food that set special requirements for the preservation process.

Fish however, is more susceptible to spoilage than certain other animal protein foods, such as meat and eggs. As part of the natural process by which organic matter is broken down and returned to the nitrogen cycle, fish flesh is rapidly invaded, digested, and spoiled by the micro-organisms which are abundant on the skin and in the intestines.

Ferments ('enzymes' to the scientist) also contribute to the dissolution, and oxidation by atmospheric oxygen is an additional process of deterioration, particularly an the case of natural fats. (Cutting 1956:1-2).

Cutting's descriptions of the procedures of fish

preservation in the non-industrialized world derived

primarily from inland Africa, but the details of the drying

and smoking methods were essentially the same as those from

the historical literature of the Upper Great Lakes.

According to Cutting's African data, the houses often provided convenient places in which to smoke fish. Wood ashes were sometimes used to cure fish in the absence of salt (Cutting op. cit.). The picture of aboriginal living structures as large smoke houses certainly follows from early descriptions of longhouse life in the Upper Great Lakes, in which the houses suggested visions of hell, filled with smoke and unspeakably stale air. Sagard remarked that among the Huron, the houses themselves served as places to smoke fish.

The savages cure fish in the following manner: they let them drip a little, and then cut off the heads and tails; they open them at the back, and having emptied them, they make incisions, to allow the smoke to penetrate them thoroughly; the perches in their huts are all loaded with them. When they are well buccaned, they bring them together, and make them into packages, each containing about a hundred (Rau 1884:270).

Skinner documents similar tactics among the northern Saulteaux (Skinner 1911:133-4). Additionally, Sagard described the drying and smoking of fish on racks in the out-of-doors.

They gutted them, cutting them open as one does cod, and then spread them out on racks made with poles set up for the purpose in order to dry them in the sun. But if the weather is unfavorable and

rain prevents and counteracts the drying of meat or fish they smoke it on hurdles or poles, and then pack it all into casks, for fear of dogs and mice...(Sagard 1968:185-6).

Other tactics of preserving, according to early accounts, included the pounding or pulverizing of fish (JR 51:259), and the drying of whitefish roe for food (Hearne 1958:143). Some methods, commonly used in the Upper Great Lakes, took advantage of the foul weather of the fall fishing season to assist in the preservation of the food. Alexander Henry described the procedure at Sault Sainte Marie.

In the beginning of October, the fish, as is usual, was in great abundance at the Sault; and, by the fifteenth day of the month, I had myself taken upward of five hundred. These, I caused to be dried, in the customary manner, by suspending them, in pairs, head downward, on long poles, laid horizontally, for that purpose, and supported by two stakes, driven into the ground at either end. The fish are frozen the first night after they are taken; and, by the aid of the severe cold of the winter, they are thus preserved, in a state perfectly fit for use, even till the month of April (Henry 1969:64).

Carver describes the same methods in use at Mackinac, for the winter fishing of trout (Carver 1956:148).

Both LaPotherie and Schoolcraft recorded the methods of preservation associated with the dip net fishery for

whitefish at Sault Sainte Marie. LaPotherie described the drying and smoking of whitefish on wooden frames for wintertime use (Blair 1911:280). Schoolcraft documented the drying/smoking procedure in other seasons as well. Travelling along the southern shore of Lake Superior during June of 1820, Schoolcraft recorded the drying of whitefish at the Shelldrake River by "several lodges" of Chippewas "who are drawn to this spot by the advantages of taking fish at the mouth of the river" (Schoolcraft 1970:144). Further to the west, the drying of sturgeon was described, this time at the Ontonagon River weir that Schoolcraft observed. sturgeon were cut into thin strips and dried over smoke. McKenney, travelling along the same route a few years later, left a partial description of the physical evidence of the drying/smoking methods. "I found the kind of frames on which the Indians dry their fish. It is built over a square hole in the ground, of about six feet by three, where the fire is built" (McKenney 1827:361).

These accounts give a summary description of some of the structures that may have been associated with the drying/smoking of fish. It is interesting to realize that the smoking of fish did not necessarily require any special architecture other than what was available as part of living structures and their associated fires. But drying racks and fire pits specific to the purpose of drying/smoking are also typical of past preservation methods. The literature of the contact period is silent on the question of the sex/age composition of work groups carrying out the tasks associated

with the preservation methods, nor was there any information about the time demands of such tasks. Apparently the preservation of fish was a common incident regardless of season or species. Both Sagard (1968:185-6) and Grant (Masson 1960:330) suggest that the methods of fish preservation were identical to those used to preserve meat, from which one could infer that the methods of preserving large amounts of protein were in place prior to innovations in procurement technology that might have increased the frequency of large landfalls of fish. There is, sadly, no information about the organization of labor associated with preserving periodic abundances of fish. Tables 5 and 6 indicate the presence of racks and pits potentially related to fish preservation activities from archaeological sites in the study area.

## Assemblage data

Tables 5 and 6 depict archaeologically-recovered artifacts and features associated with fishing technologies from sites in the study area. Several things are clear from these tables. First, some categories of technology have no archaeological basis to support their relevance to the prehistoric period. Secondly, differential excavation plays a role in producing patterns within assemblage inventories, for the more extensively excavated sites appear to have a more diverse fishing-related assemblage, especially in the Middle/mixed Woodland assemblages. Net sinkers, as an assemblage element, are most frequent in mixed Woodland

Table 5. Artifacts/features from Middle and mixed Woodland components

<pre>Item Grappling/Wounding</pre>	20 DE <u>3</u>	20 MK <u>53</u>		20 DE 17	20 DE <u>4</u>	20 CH 2
Toggling harpoon Unilateral harpoon	X	x		?	X	?
Leister Spear (wood)	x				?	
Lines (Angling)						
Gorge Hook	X		x		3 X	x
Seine/Gill Net						
Sinker Needle/Shuttle		x			x x	X
Preservation						
Scaler Rack Pit		?		? X	x x	X X
Excavated area (m.sq.)	13	45*	65	125	205	*260

<sup>\*</sup>includes other components

Table 6. Artifacts/features from Late Woodland components

<u>Item</u>	20 MK <u>22</u>	20 DE <u>4</u>	20 MK 82	CX	CX	20 MK <u>1</u>
Grappling/Wounding					•	
Toggling harpoon Unilateral harpoon Leister Spear (wood)	x		x			x
Lines (Angling)						
Gorge Hook	x	X X		x	?	?
Seine/Gill Net						
Sinker Needle/Shuttle	x	X X	x	x	X	?
Preservation						
Scaler Rack Pit	?	x x	?		?	?

Excavated area (m.sq.) 104 205\*205\*203 350 450

<sup>\*</sup> includes other components

settings whose faunas suggest spring/summer exploitation. The design and function of the harpoon, known to change through prehistoric time, does not apparently accompany alterations in species captured. Nor is there covariation among assemblage items and faunas present at a site. example, there is no site at which whitefish and net sinkers co-occur (Tables 7-9). On a presence/absence basis, the angling elements are represented through time, and many sites include features possibly related to fish preservation activities. But several of the places of intensive fishing such as 20MK1 and 20MK22 do not include varied assemblages of fishing-related tools despite heavy excavation and broad ranges of recovered species. This fact requires some explanation, and there are many possibilities. One is that the fishing-related assemblages of these places were perishable and failed to remain part of the archaeological record. Another possibility is that elements of land-based capture assemblages played a role in the fishing methods. There is also the possibility that sampling error is primarily responsible for observed assemblage patterns.

Tables 7-9 depict identified fish faunas on a presence/absence basis from archaeological deposits within the study area. Examining these tables leads one to conclude that the extent of excavation is partially associated with the diversity of the faunal assemblage recovered. Secondly, the older the component, the more reduced in species the faunal assemblage, and the less frequent the appearance of the coregonines. The trouts,

Table 7. Faunal identifications from Middle Woodland components

1st spawn month/ Species	20 DE 3	20 DE 17	20 DE 4	20 MK 51
MARCH A. fulvescens	x	X	X	X
S. vitreum		X	X	X
APRIL S. canadense	x			
Catostomus sp.	Λ			
C. commersoni	Х		X	X
Ictalurus sp.	••	X	21	21
I. punctatus				
I. nebulosus				
Esox sp.				
E. lucius			X	
MAY				
A. grunniens	X	X	Χ.	
R. chrysops		X		
Micropterus sp.			X	
M. salmoides	X			
M. dolomieui				
A. rupestris				
Perca sp. P. flavescens	х			
Moxostoma sp.	Λ			
C. catostomus				
JUNE				
L. osseus				
SEPT				
Salmonidae				
Salvelinus sp.			•	
S. fontinalis				
S. namaycush	X			X
OCT/NOV				
Coregoninae				v
Coregonus sp. P. cylindraceum		X		X
C. clupeaformis		Λ		
DEC				
L. lota	X			
Area exc. (m.sq.) * includes other components	13	125	205*	488

Table 8. Faunal identifications from mixed Woodland components

1st spawn month/ Species	20 MK 90	20 ST 1	20 MK 102	20 MK 53	20 EM 22	20 MK 61
MARCH A. fulvescens S. vitreum APRIL	x	x x	x x	X X	x	X X
S. canadense Catostomus sp. C. commersoni Ictalurus sp.						
<ul><li>I. punctatus</li><li>I. nebulosus</li><li>Esox sp.</li></ul>						X
E. lucius MAY A. grunniens		X				
R. chrysops Micropterus sp. M. salmoides M. dolomieui					?	X
A. rupestris Perca sp. P. flavescens Moxostoma sp.						
C. catostomus JUNE						
L. osseus SEPT Salmonidae						X
Salvelinus sp. S. fontinalis S. namaycush OCT/NOV		x		x		x
Coregoninae Coregonus sp. P. cylindraceum C. clupeaformis DEC L. lota						
Area exc. (m.sq.)	2	9	39	45	65	110

Table 9. Faunal identifications from Late Woodland components

1st spawn month/ Species	20 EM 25	20 MK 54	20 MK 22	20 DE 4	20 CX 19	20 CX 18	20 MK 1
MARCH A. fulvescens S. vitreum APRIL	x	x x	x x	x x	x	x x	X X
S. canadense Catostomus sp. C. commersoni Ictalurus sp.			X X			X X	x
I. punctatus I. nebulosus			X			X	X X
Esox sp. E. lucius MAY			X				x
A. grunniens R. chrysops							X
Micropterus sp. M. salmoides M. dolomieui			X X	x		X	X X
A. rupestris Perca sp. P. flavescens Moxostoma sp.					x	X X X	? X X
C. catostomus JUNE							X
L. osseus SEPT							X
Salmonidae Salvelinus sp. S. fontinalis			X			x	
S. namaycush OCT/NOV			x			^	X
Coregoninae Coregonus sp.		x					
P. cylindraceum C. clupeaformis DEC			x				x
L. lota			x				
Area exc. (m.sq.) *includes other com	sur ponent	75+ s	104+	205*	230	350	450

particularly <u>S. namaycush</u> (lake trout) are, on a presence/absence basis, relatively well-represented through time. Likewise, the sturgeon is represented at every locality. The late spring-summer spawners are not well-represented within Middle and mixed Woodland assemblages, and are most frequent in the Late Woodland assemblages.

Judging from fish assemblages alone, the Middle Woodland and mixed Middle/Late Woodland components suggest seasonal (i.e. spring and/or fall) uses, while the Late Woodland components suggest full warm-season uses. Given the presence of coregonines in some Middle Woodland assemblages such as 20DE17 and 20MK51, and the presence of S. namaycush at five Middle Woodland localities, it is apparent that fall-spawning fish played some role in Middle Woodland subsistence patterns. The dearth of coregonines at Middle Woodland components is compatible with two possibilities. These fish were rarely used during Middle Woodland times, or, these fish were rarely preserved in components dated to Middle Woodland times. Sampling error based on differential excavation is responsible, to some degree, for the patterns displayed by these faunal assemblages, and it is without a doubt the case that sampling error produced by differential preservation has an impact on these patterns. Site B-95, an exceptional locality in terms of organic preservation, displays how significant the preservation differences may be (Table 7). The 20MK1 and 20MK22 faunas are strong evidence for the

prominence of fall-spawning fish in Late Woodland economies.

## Conclusion

Several things are apparent from the data presented to this point. The historical data suggest that with the exception of a river-operated haul seine or a permanent weir on a large river, fish capture regardless of method required no more labor than that potentially associated with one extended family (Table 10). In fact most haul-seine methods themselves were probably less dependent upon large (i.e. five to six) numbers of persons than the historic record would suggest, if as stated by Cleland (1982), the tasks associated with most seine operations were rather unspecialized and could have been accomplished equally well by any available person in a co-resident group. This fact potentially includes haul seine operations within a group of methods viable for an extended family labor force.

Large permanent weirs offer a surprising contrast with other methods of fish capture documented historically and ethnographically (Table 11), especially when the contrast is drawn between other methods of mass capture such as nets. Weirs do not have the same or similar labor requirements as net operations. In fact, weirs appear to be, structurally and functionally, ultimately the opposite of nets in terms of labor requirements, locational requirements, and other constraints such as those imposed by the characteristics of target prey. The large fish weir was, quite clearly, a very sophisticated technological device appropriate for a stable

Work group composition by species, method and season Table 10.

M = adult male W = adult female m = boy	w = girl
= adult male W =	11
= adult male W =	female
= adult	11
	= adult

Species	Method	Age/Sex comp.	Season	Source
Whitefish	dip net	2 M	all	12
		1 m/1 w		McKenney 1827
		1 M/1 m		Carver 1956
	dip net	Z Z		Carver 1956
	seine (?)	2-3 m or 2 M	sum/fall	Tanner 1956
	haul seine	4-6 M	Ç	Hennepin 1972
Sturgeon	ad hoc	M L	<i>د</i> •	Tanner 1956
	spear	Σ [	spr/sum	Charlevoix 1923
	spear/canoe	2 M/1 m	spr	Tanner 1956
	weir/gaff	M L	spr/sum	Schoolcraft 1970
	haul seine	4 M	۰-	Masson 1960
E		;		
Trouts	angle angle	× ×	wns wns	Hearne 1958 Hearne 1958
	spear	Z Z	¢.	Charlevoix 1923
	spear/torch	2 3	fall	Lanman 1856
	net net	2-3 m 2 M	wns	Tanner 1956 Tanner 1956

Table 10 (cont'd.).

	1827	1827
Source	ley	McKenney 1827
Season	mns	s mm
Age/Sex comp.	3 W	2 M/1 m
Method	spear/torch	spear/torch
Species	Bass, etc.	

resource and for a group of persons attached to a fixed location. Large weirs appear to be incompatible with current understandings of prehistoric group organization and resource-related activities, which were presumed to have been conducted by self-sufficient and autonomous extended family groups. There is strong evidence for the use of this method during the earliest historic times in the study area.

Table 11. Comparison of weir methods with net methods

Large weirs
Construct new or make major
repairs each use with special
labor forces
Fixed location
Useable in warm seasons
Requires shallow water
Catch by individual
User stays at locale
Catch preserved in water/laborer

Collective ownership

Net methods
Can make at all times
with unspecialized labor

Portable
Useable year-round
Many depths, quiet water
Mass collection
User may follow quarry
Catch preserved by
laborer
Individual ownership

Some methods are versatile in the sense that they are applicable in numerous settings and successful with a broad range of species. Grappling/wounding methods and seining methods appear to be the most versatile methods. There are abundant accounts that would suggest that some species were more vulnerable to a range of methods than were others (Table 12). For instance, sturgeon and lake trout are recorded to have been taken by numerous and versatile configurations of gear, whereas whitefish are the most method-specific of any economic species. Quite obviously, Table 12 doesn't exhaust the possibilities of particular species susceptibility, but merely reports tendencies

apparent within the historic record.

Table 12. Susceptibility of fish species to gears Method Ad hoc Grappling Lines Traps Bag Seine Gill Sturgeon X X X X Χ Dory/pike X X ? ? Bass X X Sheepshead X Sucker ? ? Lake trout X X X X X Whitefish X(rare) X X X

? refers to fish that may have been netted by some unknown net configuration

Without a doubt, the archaeologically-derived tabulations presented in this chapter reflect several aspects of sampling error. The greater the area excavated on a site, the greater and in general the more diverse the fishing related inventories and faunas appear to be. The more recent the site, the more diverse its faunal assemblage appears to be.

In reference to the standing models of subsistence and population change, the preceding archaeological, historical, and ethnographic data suggest that aboriginal fishing methods were diverse in geographic setting, and that no one kind of location predominated. No body of data suggests a labor force larger than the household. Likewise, no particular technology nor fauna dominated over one or the other cultural-chronological period. The hypothetical model

of change discussed in the preceding chapter incorporates these conclusions.

#### CHAPTER THREE

### A STATISTICAL STUDY OF PREHISTORIC LAKESHORE LOCATIONS

## The Sites and their locational attributes

This chapter specifies the kinds of lakeshore microenvironments in which prehistoric archaeological occupations are found, and, more critically, outlines whether there are differences between groups of components that can be attributed to changing subsistence practices and/or changing technologies over time. The reason for examining the sites' locations is first of all to characterize and describe their specific environmental settings and secondly to analyze systematic and patterned changes in these characteristics through time.

There are a number of specific questions that need to be answered before one can successfully model changes in prehistoric subsistence in the Upper Great Lakes. Do the sites' settings change through time in a perceptible manner that can be explained in relation to other changes in cultural systems? Are certain kinds of microenvironments given differing emphases through time? Are new microenvironments added to a cultural repertoire of settlement choices as time passes? Or, are there no apparent changes, other than increased numbers of sites, that can be correlated with time passing? Are there clusters of similar sites within the data base, and can their similarities be explained? What role does sampling error play in the appearance of patterned regularities?

Does the data set reflect the bipolar Late Woodland coastal settlement use modelled by other researchers (Cleland 1982), and does it reveal expected Middle Woodland settlement modalities?

The set of analyses to follow will examine certain aspects of the locations of prehistoric components of the A.D. 0 to A.D. 1550 time range to describe their physiographic settings, to compare these settings, and to explain differences in settings in relation to assumed differences in subsistence practices. Specifically, the analyses will examine whether, through time, lakeshore locations became more specific in regard to offshore conditions such as spawning reefs, shoals, deep offshore countours, dropoffs, and other features thought to be associated with a highly-evolved fishing adaptation.

Chapter Two revealed a rich data base from which to model the historic period adaptation to the Upper Great Lakes fisheries resource, and to some extent provided archaeologically-derived data that supported the extension of the historic period model to the prehistoric period. Another way to assess the prehistoric adaptation to the fisheries is to examine the locations of prehistoric occupations across the time range of interest, vis-a-vis the environmental attributes at these locations, assuming that these locations are accurate in placement, are sampled in proportion to their actual occurrence and preservation, and that the environmental attributes chosen by the researcher are able to reflect past potential and use of the

prehistoric fisheries.

Case selection procedures

The study includes locational data on twenty-five Middle Woodland and forty-six Late Woodland components at ca. forty-five localities. An additional thirteen components of unqualified Woodland age were included as well, for a total of eighty-four components. Components were assigned to the categories Middle and Late Woodland based on the original researcher's assessment of the age and cultural affiliation of the components. When this information was unavailable, the assessments were made by the author. Middle Woodland components were those at which Laurel, Laurel-like, or other Lake Forest ceramic complexes were identified. Late Woodland components were those at which the ceramic complexes representative of the local sequence were present (McPherron 1967). This group included Mackinac phase, Bois Blanc phase, Juntunen phase and other related complexes such as Heins Creek, Blackduck and those complexes sharing attributes of Oneota affinity.

Ideally, the catalog of components for this analysis should include all Middle and Late Woodland habitations in the region under study (for a brief account of the region and its boundaries see Chapter One). But in practice there were a number of reasons why this goal could not be achieved. It was necessary to be selective about the inclusion of certain of the locations known for the area. First, only major lakeshore or relict major lakeshore areas

were under study. Components occupying other environments were excluded at the outset, and in fact solitary components in areas that were poorly surveyed or poorly known were also excluded. Thus, the lakeshore areas of Cheboygan County were not included, for these areas were unknown entities given the level of survey coverage that prevailed elsewhere in the study area.

Only those components with precise locations were included. If locations were precise to the level of the quarter-section (i.e. "correct" to within .8 km. of an "actual" location) then locational information was considered accurate enough to include the component in the study, provided that all other criteria were satisfied. Components presumed to be duplicates were counted once.

Selection procedures favored components from which collections of artifacts had been derived. Without a collection, it was generally not possible to meet the next condition for selection; there needed to be strong evidence that the component was Woodland in age. Acceptable evidence was clear association of ceramic complexes with the component in question. In the case of several non-ceramic components this stipulation was relaxed when, in the opinion of the original researcher, the component was of Woodland age. Aspects of location such as elevation were used to support the operational claim to Woodland status. Historic-period American Indian settlements were included only if there was artifactual evidence supporting the presence of a precontact component of Woodland age.

Finally, the study favored components that were assumed to be habitations as opposed to special-purpose areas such as quarries and cemeteries. The most difficult decisions of inclusion/exclusion in regard to the habitation criterion focussed on the elusive "lithic scatter." Unless there were strong extenuating conditions, the "lithic scatters" failed on both the habitation and chronological criteria. For a complete account of the components chosen for the study, and some detailed justification for the inclusion/exclusion of borderline cases, see Appendix A.

These selection procedures standardized the data set to some degree by factoring away a number of sources of uncontrolled variation. Obviously the data set reflects the shifting emphases of various researchers over time, and bears the stamp of uneven research intensity over the region in question. But for the most part it is systematically composed of the best-researched, most precisely-located habitations in the region, and it is believed to represent faithfully the range of variation present in the archaeological record of the relevent time range.

Components included in the analyses are listed in Appendix C.

## Variable design and measurement

The eighty-four cases chosen for inclusion in the statistical evaluation of prehistoric locations and their relation to major lakeshore microenvironments were measured on a total of thirty-three variables. These variables

included a wide range of information classes and expressed variability on differing levels of generality and measurement scale. Full documentation on variable definitions and coding values appears in Appendix B. The following section discusses the variables, particularly the reasoning behind their design, the methods of measurement, and their potential usefulness in discovering locational trends through prehistoric time.

Some of the variables functioned by providing nominal and ordinal scale values on which to partition the data set for further analysis. Table 13 displays these variable definitions. The values derived from published sources, institutional documents, and institutional site files but complete information was not available for all components. These variables provided a data base to determine whether there was a statistically significant association between components of Middle or Late Woodland age and multiple occupations, historic occupations, or occupations representative of particular phases within the regional ceramic sequence with some environmental condition. RESINT variable provided some control of the variation associated with differing levels of archaeological research at a place. Each site was placed in a category that described relative extent of research. By selecting components via the RESINT variable, an analysis could, for example, concentrate on excavated components only.

The DUP variable identified multicomponent sites, and allowed one to concentrate on locations rather than

Table 13. Nominal/ordinal scale variable definitions

Variable Name	Definition
DUP	Location duplicated in the files?
MULTIC	Is this a multicomponent site?
CCPER	Prehistoric cultural-chronological period of this component
NNOCCS	Number of additional components
CERCX	Ceramic complex represented by this component
NHIST	Is there a historic component at this location?
RESINT	Intensity of research at this location

components, because it was advantageous to vary the definition of what constituted a case from analysis to analysis. In some situations, each component was considered a separate analyzable entity, or case. In other situations, the analysis concentrated on locations as the analyzable entities. Obviously, the results and interpretations of the analyses depended on what entity constituted a case. necessary to control case definition, however, primarily because of the tendency for identical locations to be associated with components of varying ages. This situation offered contradictory information to some of the statistical models used in the analyses, particularly to the discriminant function techniques, in which the research question required a categorical distinction between cases of opposing cultural-chronological periods to be drawn. Obviously when a statistical analysis proceeds to distinguish between groups of cases that despite their temporal grouping share identical numeric values, failure is the result. This difficulty was corrected somewhat by the reduction of the data set to the level of unique locations rather than components as the analyzable entities.

Fourteen nominal and ordinal scale variables measured aspects of environmental setting and the presence of archaeological derived data related to the presence of faunal deposits, fishing-related implements, and settlement characteristics. The data for these variables were collected from published references, by examination of museum collections, and from NOAA navigational charts for

the Upper Great Lakes. The variables were designed both to partition the data set into segments based on various definitions of internal groups, as well as to provide the data for contingency-table association between cases of the two Woodland time periods of interest and the presence/absence of critical environmental features related in theory to fishing technologies.

The archaeologically-derived data portrayed by the variables USE through REGION provided comparisons of assumed or estimated seasonality, assumed/estimated intensity of use, concentration of components per subregion, and the presence/absence of faunas and fishing implements at components. The data on the rest of the variables, derived primarily from NOAA navigational charts, provided categorical summaries of the physical conditions at a location. NCTO attempted to categorize the shape of the shoreline configuration at/near a location. NWAT accounted for the kinds of water features associated with a place. OTOPO attempted to categorize the kinds of onshore topographic conditions near/at a place, and NGRAD categorized the gradient of the lake bottom. SHOAL, JSHOAL, JDRPOF, and DRPOF, were designed to record the presence/absence of shoals and dropoffs offshore from a location. An element of arbitrariness was necessary in defining how far distant from a place an offshore feature could lie before its location was insignificant. A distance of ten kilometers was chosen initially, large enough to include data on many of the locations within the study, but

restricted enough to eliminate cases for which the spatial association between features was doubtful. A complete description of value ranges for each variable is listed in Appendix B and definitions of the variables may be found in Table 14.

My attempts to collect systematic information on point-locational lake bottom compositional characteristics were unsuccessful. Despite occasional inclusions of general substrate or bottom compositional characteristics on the NOAA charts, there was no apparent broad-area source for such information nor was there information on bottom conditions at locations as specific as the precise locations required for this study. As a result this very critical characteristic of fish habitat (and related human locational strategies) was not included in this study. If adequate data were available it would be important to distinguish between the sort of bottom that dominated at a place as opposed to the quality of local variation in substrate, for these two kinds of variation are critical for assessment of habitat suitability for various species of fish. Currently the first kind of variation is generally accessible, but at a scale so general that I am convinced that it is not useful for the present study.

The final group of variables portrayed interval-scale variation in a number of physical conditions associated with the components in the study. Because these variables constitute the most useful and critical data set in terms of potential and actual analytical results, each variable's

Table 14. Nominal/Ordinal scale variables: environmental setting/Archaeological setting

	_	
Variable	Name	Definition
USE		Hypothetical use of this component
NSEA		Hypothetical season of use of this component
TECH		Technological items present in the collection?
FAUNA		Faunal sample present in the collection?
REGION		Intra-regional designator
ASPECT		Major directional exposure
NCTO		Coastal topography
NWAT		Watercourse present
ISL		Island location?
NISL		Islands within ten kilometers
SHOAL (JSHOAL)		Shoal within ten kilometers (within 4.8 kilometers)
DRPOF (JDRPOF)		Dropoff within ten kilometers (within 4.8 kilometers)?
ОТОРО		Offshore topography

Offshore gradient

NGRAD

design, measurement, and utility will be discussed in turn.

RCMEAN measured an age estimate in radiocarbon years associated with a component. When there was more than one estimate available, the estimate thought to express the central tendency in the components' age, or the estimate generally accepted in the literature was accepted for the study. Estimates that were considered to vary considerably from realistic assessments, or those that contradicted stratigraphic and other archaeologically-derived data were excluded. Published age estimates were substituted when radiocarbon estimates were lacking, and in cases for which there were no available estimates, a missing data code was substituted for the estimate. RCMEAN values primarily derived from published literature but in some cases unpublished estimates were obtained from the institution or individual responsible for the original excavation.

DISL measured the straight-line distance in kilometers between a location and the nearest island. This variable attempted to measure the degree to which components clustered at places where offshore topographic features were varied, assuming that for many fish species the presence of irregular shorelines and topographic variation provides optimal habitat. The DISL data derived from NOAA navigational charts.

DSHOAL measured the straight-line distance in kilometers between a location and an area of offshore shallow water. For the purposes of this study, a shoal was operationally defined as an offshore shallow area of any

bottom composition over which the water depth was ca. thirty feet or less. I made no attempt to distinguish between reefs and shoals; both kinds of features are grouped within the variable. NOAA charts provided excellent data on the point locations of shoals and reefs. These places are of critical importance for this study because they provide spawning habitat for many economic species of fish and they may have been associated with the placement of Middle and Late Woodland settlements on adjacent shorelines (Cleland 1982:729).

DDRPOF measured the straight-line distance in kilometers between locations and sharp depth increases offshore. This variable suffers from some subjectivity in regard to the definition of "sharp depth increase," for unlike SHOAL there is no convenient definition available. But in practice when great increases in depth over short distances were observed on the NOAA charts the decision about measurement points was not a problem. In other less clear cases some level of arbitrariness entered the measurement procedure. DDRPOF incorporates an important aspect of fish related topography because it is apparently the case that these steep contours accompany brisk water currents that are associated with high potential habitat for some species. There is no doubt that modern fixed-position fish nets are associated with such dropoff features, a fact rapidly discovered by examination of the navigational charts for the lakes.

MAXD1, MAXD2, and MAXD3 are variables designed to

measure maximum water depths in feet at distances of one, two and three kilometers offshore from prehistoric locations. These variables were again collected from data included on navigational charts. The choice of the distance intervals was arbitrary; they were chosen in order to utilize the available charts while preserving as much measurement precision as possible. Secondly, the study intended to make comparisons among locations and those comparisons were only possible through the standardization implicit in the restriction of relevant measurements to consistent areas. Finally, in an earlier study (Martin 1977), a similar attempt to compare point characteristics of archaeological sites using considerably larger radii (i.e. one, two, and three miles) actually sampled the characteristics of the background environment rather than the specific characteristics associated with the prehistoric locations themselves. The smaller radius design attempted to correct the problem of relevant scale. It was important to study local water depth characteristics because water depth was considered a causal factor both in the structure of appropriate fish habitat and the structure of related human settlement choice. There was, at the outset, some strong indication that late prehistoric components were associated in space with relatively deep water conditions, and that this difference was an important element of evidence related to changing subsistence practices and technologies (Cleland 1982:778).

SHALL, MED, DEEP, and THIRTY derived from the NOAA

These variables measured the percentage of total charts. water area near a location (within three kilometers) falling into four depth categories. SHALL measured the percent of shallow water (zero to three fathoms); MED measured the amount in the three - ten fathom range, and DEEP measured the greater than ten fathom percentage. THIRTY measured the percentage less than five fathoms in depth. Despite the redundancies inherent in the design of these variables, the depth categories portrayed how much the water depths varied from location to location, and moreso than the actual measurements of extremes, allowed some comparison of overall depth characteristics from place to place. It has been assumed that both seasonal fisheries potential and appropriate fishing methodologies are associated with the presence of certain depth ranges; therefore the design of depth categories based on the range of possible technologies and the preferred depth ranges for relevant fish species was Additionally, the THIRTY variable was designed reasonable. for compatibility with modern-day studies of fisheries potential and, in particular, spawning localities (Goodyear et al 1982).

The AREA variable measured the water area in square kilometers accessible within a three kilometer radius of a location. The rationale behind the choice of the radius size is consistent with that explained for the depth variables. The AREA variable allowed the comparison of overall fisheries potential from place to place, as well as providing some interval-scale assessment of the topographic

configuration of a place. Obviously, embayed locations possess smaller total water areas than do those locations on islands or peninsulas. Measurement of total available water area also allows one to observe whether the topographic placement of components varies through time and subregion.

All statistical routines operated on the Sperry 1100/80 installation at Michigan Technological University's Academic Computing Service, using widely-available statistical software. The univariate distributional analyses, student's t, discriminant function analyses, and chi-square analyses made use of the Statistical Package for the Social Sciences (SPSS) data analysis system (Nie 1975), and the cluster analyses derived from the University of California biomathematical (BMDP) software (Dixon 1981).

# Analysis of nominal/ordinal variables by the chi-square statistic

The variable CCPER representing cultural-chronological period was cross-tabulated with a number of nominal/ordinal scale variables over 71 cases representing all components of known age (n = 25, 46 respectively). The analysis examined whether statistically-significant relationships existed between the values of CCPER and those of the other variables. Using the subprogram CROSSTABS of SPSS (Nie 1975), chi-square statistics, measures of strength of association (Cramer's V or phi), and levels of significance were generated and used to examine the null hypotheses of no relationship between crosstabulated values. The internal

characteristics of the data set required that original categories on some variables be collapsed to fulfill the dimensional requirements of the chi-square test. The variable identifiers and ranges of values used for this exercise are to be found in Table 15.

Table 16 depicts the crosstabulations attempted, the chi-square value, a measure of strength of association, and the status of the null hypothesis for each variable. No contingency table analysis approached the rejection of the null hypothesis at any level lower than .15, but there is a tendency for Late Woodland sites to be positively associated with uniform offshore topography, though this association is not statistically significant at the .05 level (Table 16: OTOPO). All other relationships are very weak, judging from the magnitude of phi or Cramer's V.

In order to examine more carefully the very important hypotheses that shoals and dropoffs were associated with Late Woodland components in space, the definition of relevant distance between components and such features was examined in several ways. In the first instance, the distance of relevance was figured as ten kilometers (SHOAL, DRPOF variables); any offshore feature beyond this distance was assumed to be irrelevant and was recorded as a negative. Then, the relevant distance measure was recalculated as 4.8 kilometers, based on the historical accounts of Joutel who observed net-fishing among expatriate Huron at Mackinac in 1687, about which he wrote "they go as far as a league out into the lake to spread their nets" [Margry 1876-86

Table 15. Values on variables examined by the chi-square statistic

Variable/	label	Value/range
NNOCCS	Number of additional occupations	none, one, more
MULTIC	Multicomponent location?	yes/no
NHIST	Historic location?	yes/no
REGION	Subregion in study area	Sault Ste. Marie Straits area northern Lake
Michigan		northern lower Michigan
USE long	Est. length of use	none, short, med,
NSEA	Est. season use	<pre>none, spr/sum, sum/fall, spr/sum/fall</pre>
NCTO	Shoreline coutour	embayed, straight
NWAT	Watertype present	<pre>lake, lake/stream lake/river</pre>
ISL	Island location?	yes/no
NISL	Islands within 10 km.	none, one/two, many
SHOAL	Shoal within 10 km.	yes/no
JSHOAL	Shoal within 4.8 km.	yes/no
DRPOF	Dropoff within 10 km.	yes/no
JDRPOF	Dropoff within 4.8 km.	yes/no
ОТОРО	Offshore contour	uniform, med, varied
NGRAD	Offshore gradient	shallow, mixed, plunging

Table 16. Results of 2x2, 2x3, and 2x4 contingency table analyses, CCPER with categorical variables

<u>Variable</u>	<u>Chi-square</u>	V/phi	significar	ice E	1(o)=nc	rel.
NNOCCS	.08124	.03383	.960	)2		FTR
MULTIC	.00000	.00097	1.000	0		FTR
NHIST	.01754	.04760	.894	16		FTR
REGION	3.03302	.20668	* .386	56		FTR
USE	expected	cell free	quency too	low t	to test	;
NSEA	expected	cell free	quency too	low t	to test	:
NCTO	.08960	.07007	.764	17		FTR
NWAT	.45699	.08023	* .795	57		FTR
ISL	.01766	.05189	.894	13		FTR
NISL	.02482	.04957	* .874	18		FTR
SHOAL	1.46286	.17414	.226	55		FTR
JSHOAL	.07154	.04760	.894	16		FTR
DRPOF	.37876	.10267	.583	38		FTR
JDRPOF	.87746	.14073	.348	39		FTR
ОТОРО	3.68676	.22787	* .158	33		FTR
NGRAD	1.98585	.16724	* .370	)5		FTR

FTR = failed to reject at .05 level
 \* = Cramer's V

(3):503]. Assuming that for Joutel a league equalled roughly 4.8 kilometers, any feature more distant than this figure was presumably irrelevant and was recorded as a negative (JSHOAL, JDRPOF, variables). These variables are displayed in Table 16. With the smaller distance providing an historically-based way to assess relevant distance, there is still no significant difference between Middle and Late Woodland components on the matter of spatial association with shoals and dropoffs. I take this to mean that, like the Middle Woodland people, the Late Woodland people were sometimes selecting for locations that were convenient to these important features.

Obviously without the guarantee of independent random sampling it is foolhardy to state without reservation that no relationship exists between these variables and categories of components based on age. But, given the known data base measured on these variables, component locations are quite similar in the sense that their environmental characteristics do not necessarily co-vary with the age of the occupation.

## <u>Descriptive</u> <u>statistics</u> <u>for</u> <u>interval-scale</u> <u>variables</u>

Multivariate analyses of the 84 by 12 interval-scale data matrix were preceded by a review of univariate descriptive statistics on the matrix produced by SPSS subprogram CONDESCRIPTIVE. Evaluation of each variable's distribution characteristics was necessary because the reliability of further multivariate analyses was in part

dependent upon the degree to which the data set approximated a multivariate normal distribution.

Inspection of univariate measures of skewness and kurtosis revealed absolute values equal to or greater than 3.0 on two variables, DSHOAL and DDRPOF (Table 17). The degree to which the data set approaches the multivariate normal distribution may therefore be questioned. However, the data set's tendencies to non-normality were not considered great enough to attempt the creation of normality through drastic transformation exercises. The offensively non-normal variables were experimentally excluded from some analyses, but their exclusion rarely altered the results of the analyses to any significant degree.

An examination of correlation coefficients among the variables produced by SPSS subprogram PEARSON CORR revealed that several variables were moderately to highly intercorrelated. Of 72 unique coefficients, three equalled or exceeded the level of r = .80. Again, these multicollinear variables were excluded from some analyses to determine whether their effects actually altered results. In most cases they did not, but analytical results presented below are based on a set of reduced variables to offset multicollinear effects.

Identical univariate analyses were run for partitioned portions of the data set; that is, the subsets representing the time-restricted groups of cases, and the subsets representing the data set reduced to the level of unique locations and isolated components were examined using

10.86 15.29 Variance 201544.44 51.29 896.90 2977.43 6112.38 504.16 721.79 631.51 96.86 68.93 88.75 91.00 35.20 19.34 1600.00 58.19 102.00 224.00 393.00 Range Univariate statistics over all components (n = 84) 68.93 19.78 58.19 36.00 19.50 88.75 1650.00 105.00 230.00 399.00 100.00 100.00 Max 50.00 3.00 00.9 00.9 9.14 00. 9.00 10 .80 00. 4.33 .07 Min .61 Kurt -1.20 -0.90 96.0-2.56 -0.12 -1.15 -0.19 -0.73 1.04 5.01 7.93 .41 1.79 2.04 99. .82 . 74 1.43 .32 .03 .38 -0.15 Std Dev Skew 2.51 7.16 13.84 29.94 54.56 78.18 25.13 22.45 20.80 28.86 3.29 448.93 3.91 7.98 40.06 114.83 23.43 37.95 57.19 11.25 2.69 77.76 37.43 12.42 870.55 Mean Table 17. Variable RCMEAN DSHOAL DDRPOF THIRTY MAXD3 MAXD2 SHALL **MAXD1** DEEP DISL AREA MED

CONDESCRIPTIVE and PEARSON CORR. These analyses revealed similar patterns of non-normality and multicollinearity (Table 18). Within the set representing the Middle Woodland cases, the variable SHALL exhibited a positive skew and kurtosis value of greater than 5.0.

### Analyses of difference of means

These analyses compare age group means on all interval-scale variables to determine whether the difference between the two means was mathematically significant rather than attributable to chance. The technique used was student's t implemented via the SPSS suproutine T-TEST. This test assumes interval-scale variable measurement, independent error, normally distributed populations, and homogeneity of variance of populations. The question of independent error, while a major difficulty in most archaeological research, must be shelved, at least operationally, for this set of analyses to proceed. the size of the population of prehistoric sites in the region under study remains in question, the sample under study faithfully represents the range of variation that is known to occur among prehistoric habitations. The degree to which sampling has overlooked habitation site locational variation is assumed to be minimal.

The normality requirement is also a potential problem although in practice the severity of this problem in terms of affecting actual outcomes is slight, resulting in the enhanced probability of reporting a few too many significant

Table 18. (n = 25, n	Univari 1 = 46)	ate	statistics	measured	red over	Middle,	Late Wo	Woodland com	components
Var	Mean	Std Dev	Skew	Kurt	Min	Max	Range	Variance	C.V.
RCMEAN MW LW	347.05	156.13 260.72	-0.25	-0.61	50.00	630.00 1650.00	580.00 1050.00	24377.76 67979.05	44.98
DISL	11.20	14.67	1.97	3.57	.50	58.10 58.19	57.60 58.09	215.23	130.98 116.57
DSHOAL	8.07	7.82	2.21	5.89	1.30	36.00	34.70 35.20	61.17	96.90 88.77
DDRPOF	3.20	3.47	1.18	4.47	.10	11.00	10.90	12.04	108.43
MAXD1	37.76 42.82	32.21 28.83	.60	-1.18 -0.84	3.00	97.00	94.00 95.00	1037.69 805.88	85.30 66.27
MAXD2	65.84 86.45	50.77	.73	-0.54 -0.33	9.00	171.00	162.00 193.00	2577.55 2581.36	77.11 58.76
MAXD3	92.42	66.53 80.13	. 36	1.20	17.00	199.00	182.00 377.00	4427.44	71.98 59.85
SHALL	43.20	31.66 14.66	.97	-0.68 5.59	9.14	100.00	90.86 75.87	1002.36	73.28
DEEP	20.43 26.13	21.97	. 25	-0.88	000	68.93 68.93	68.93 68.93	482.72	107.53 83.69
MED	34.56 41.83	22.30 18.56	.04	-1.06	.00	71.76 88.75	71.76	497.66 344.61	64.52 44.37

Table 18 (cont'd.).

Var	Mean	Std Dev Skew	Skew	Kurt	Min	Max	Range	Variance	C. V.
AREA	12.01	4.87	60.	-1.23 -0.76	4.33	4.33 19.73 5.28 19.78	15.40	23.75 13.49	40.54
THIRTY	63.16 53.00	31.64	-0.03 -69		9.00	-1.48 9.00 100.00 .17 15.00 100.00	91.00	1001.30	50.09

findings (Thomas 1976:256). The problem is partially corrected by large sample sizes. The homoscedasticity problem, while potentially more damaging in terms of erroneous results than the normality requirement, is again solveable as sample size increases. The SPSS subprogram confronts this problem by calculating an estimate of t rather than the actual t in cases in which unequal variance exist.

## Comparison between all components

Table 19 displays the results of t tests among all Middle and Late Woodland components. Both two-tailed and single-tailed probability estimates are shown. The two-tailed estimate reveals the significance of the t-value when the null hypothesis of no difference between means is examined [H(o): u(1)=u(2)]. The single-tailed estimate reveals the significance of the t-value when a directional hypothesis is examined. Thus for instance in the RCMEAN case, both hypotheses are rejected. We are justified in concluding that Middle Woodland radiocarbon estimates are significantly different from Late Woodland estimates, and in addition we have faith in the conclusion that the direction of the difference is actually known and confirmed. single-tailed probability reports the status of the hypothesis u(1)>u(2) or u(1)<u(2), with u(1) and u(2)representing population means.

Three other variables exhibited a value of t substantial enough to reject the hypotheses under

Table 19. T tests between all Middle and all Late Woodland components (n=25, n=46)

<u>Variab</u>	le	Mean	t-value	df	2-tail prob.	1-tail prob.st	H(o) atus
RCMEAN	MW LW	347.04 1151.46	-15.29	60	.000	.000	R
DISL	MW LW	11.20 11.40	06	69	.954	. 477	FTR
DSHOAL	MW LW	8.07 7.57	.28	69	.783	.391	FTR
DDRPOF	MW LW	3.20 2.10	1.40	37	.169	.084	FTR
MAXD1	MW LW	37.76 42.82	68	69	.496	.248	FTR
MAXD2	MW LW	65.84 86.45	-1.63	69	.107	.053	R
MAXD3	MW LW	92.24 133.86	-2.21	69	.030	.015	R
SHALL	MW LW	43.20 30.84	1.78	27	.087	.043	R
DEEP	MW LW	20.43 26.13	-1.05	69	.299	.149	FTR
MED	MW LW	34.56 41.82	-1.42	65	.161	.080	FTR
AREA	MW LW	12.01 12.93	89	69	.375	.187	FTR
THIRTY	MW LW	63.16 52.00	1.57	36	.124	.062	FTR

significance level = .05
R = reject H(o)

FTR = fail to reject H(o)

under examination. The variable MAXD2 means are different enough to reject the hypothesis u(1)>=u(2), where u(1) represents the Middle Woodland population mean and u(2) represents the Late Woodland population mean. MAXD2's mean is significantly smaller for the Initial Woodland sample, and the same is true of the mean on the MAXD3 variable. In the case of the SHALL variable the hypothesis u(1)<=u(2) is rejected, and one concludes that the Middle Woodland mean is significantly larger in the variable than is the Late Woodland mean.

The components do not differ drastically on the variables that measure features offshore, nor on those that measure total water area and maximum available moderate or deep water. But the Late Woodland components on the average are closer to deep water and adjoin less shallow water.

#### Comparison between isolated components

This analysis holds constant the fact that some locations are shared by Middle and Late Woodland components, and compares those components that do not share their locations with components from the alternate cultural-chronological period. These I have termed isolated components. This collection of cases, then, tends to accentuate differences between the cases, and it is important to remember that the population of components to which this analysis generalizes excludes shared localities as well. Each component is counted as one case, meaning that stratified locations are redundant within the data set.

This situation approximates reality because each component may be considered to be at least one occupation or one use of a place. But the distinction distorts reality because contradictory information (i.e. information about shared locations) has been cleaned from the data set. In cases in which one component clearly dominated at a place, that place was included in the study. A listing of included components may be found in Appendix C.

Table 20 shows the results of the student's t difference of means test on all interval-scale variables. The by-now familiar cluster of depth-related variables gives a strong showing here (DDRPOF through MED, THIRTY). The Middle Woodland variable means are significantly greater on DDRPOF, SHALL and THIRTY, and significantly lower on MAXD1, MAXD2, MAXD3, DEEP and MED. But there are no apparent differences in the means of DISL, DSHOAL, and AREA.

When locations are not shared among components of opposing time units, there are significant differences on all of the depth-related variables. The differences between the offshore features are not significant with the exception of the DDRPOF variable.

# Comparisons of locations of known age

The following two analyses examine locations as the cases under study. Each location was entered once in the data file and identified as a Middle Woodland location, a Late Woodland location, or a mixed Woodland location. The comparisons of means were conducted between Middle and mixed

Table 20. T tests between isolated Middle and Late Woodland components (n = 14, n = 29)

Varial	ole	<u>Mean</u>	<u>t-value</u>	<u>df</u>	2-tail prob.	1-tail prob.	H(o) status
RCMEAN	MW LW	297.50 1125.19	-11.26	38	.000	.000	R
DISL	MW LW	9.24 8.77	.11	18	.915	.457	FTR
DSHOAL	MW LW	8.63 6.54	.79	16	.440	.220	FTR
DDRPOF	MW LW	4.55 2.08	2.22	17	.040	.020	R
MAXD1	MW LW	23.00 40.06	-2.04	41	.048	.024	R
MAXD2	MW LW	45.14 88.68	-2.86	41	.007	.003	R
MAXD3	MW LW	69.42 144.65	-2.92	41	.006	.003	R
SHALL	MW LW	56.28 31.34	2.38	13	.033	.016	R
DEEP	MW LW	13.60 25.79	-1.85	41	.072	.036	R
MED	MW LW	29.05 42.84	-2.00	40	.053	.026	R
AREA	MW LW	11.25 13.14	-1.29	18	.212	.106	FTR
THIRTY	MW LW	70.07 49.10	2.17	16	.046	.023	R

significance level = .05
R = reject H(o)
FTR = fail to reject H(o)

locations and between Late and mixed locations. Locations for which no general date of occupation was available were excluded from the analyses. It is again important to recall that the populations to which these tests extend are different for each exercise.

Table 21 illustrates the results of the analysis of Middle/mixed locations. The differences in sample sizes require what seem to be enormous differences in absolute value of the means before a difference is judged significant even at the .05 level. While the variables MAXD2 and MAXD3 pass the significance test at the .05 level, differences in distances to shoals, to dropoffs, and in area seem negligible. Subjectively, the cases differ on all of the other variables, but not to the degree necessary for the statistical rejection of H(o). In this analysis, the hypothesis u(1)>=u(2) is rejected for the variables MAXD2 and MAXD3.

Comparison of Late Woodland and mixed locations

Based on the examination of the t test results between groups composed of Late and mixed locations, there are essentially no significant differences between the groups (Table 22), nor are there any readily apparent subjective differences. No hypothesis can be rejected at the .05 level of significance, and the hypothesis u(1)=u(2) stands in every case. Assuming that one could predict the direction of the expected differences, the hypotheses u(1)>=u(2) and u(1)<=u(2) also fail the rejection test.

Table 21. T tests between Middle Woodland and mixed locations (n = 6, n = 17)

<u>Variab</u>	<u>le</u>	<u>Mean</u>	t-value	<u>df</u>	2-tail prob.	1-tail prob.	H(o) status
DISL	MW mix	5.85 13.32	-1.54	20	.138	.069	FTR
DSHOAL	MW mix	6.16 8.24	54	21	.593	. 296	FTR
DDRPOF	MW mix	4.10 2.26	1.23	21	.231	.115	FTR
MAXD1	MW mix	31.16 44.00	84	21	. 411	. 205	FTR
MAXD2	MW mix	42.00 80.35	-1.66	21	.111	.055	R
MAXD3	MW mix	51.33 115.29	-2.23	21	.037	.018	R
SHALL	MW mix	56.92 30.93	1.43	5	.204	.102	FTR
DEEP	MW mix	15.03 24.78	93	21	.365	.182	FTR
MED	MW mix	28.03 41.01	-1.28	19	.215	.107	FTR
AREA	MW mix	11.92 12.62	30	21	.767	.383	FTR
THIRTY	MW mix	68.50 56.94	.78	21	. 444	.222	FTR

significance level = .05
R = reject H(o)

FTR = fail to reject H(o)

Table 22. T tests between Late Woodland and mixed locations (n = 22, n = 17)

<u>Variab</u>	<u>le</u>	<u>Mean</u>	t-value	<u>df</u>	2-tail prob.	1-tail prob.	H(o) status
DISL	LW mix	13.89 13.32	.12	37	.908	. 454	FTR
DSHOAL	L <b>W</b> mix	9.31 8.24	.42	37	.678	.339	FTR
DDRPOF	LW mix	2.17 2.26	11	37	.915	. 457	FTR
MAXD1	LW mix	50.31 44.00	.63	37	.534	.267	FTR
MAXD2	LW mix	101.81 80.35	1.16	37	. 254	.127	FTR
MAXD3	LW mix	153.22 115.29	1.39	37	.174	.087	FTR
SHALL	LW mix	31.01 30.93	.01	35	.989	.494	FTR
DEEP	LW mix	30.28 24.74	.75	37	. 458	.229	FTR
MED	LW mix	38.69 41.01	35	35	.729	.364	FTR
AREA	L <b>W</b> mix	12.30 12.62	26	37	.806	. 403	FTR
THIRTY	LW mix	46.77 56.94	-1.26	37	.217	.108	FTR

significance level = .05

R = reject H(o) FTR = fail to reject H(o)

The dimension of variation on which the cases differ most is that related to water depth, with the Late locations favoring deeper water conditions. This characteristic of Late as opposed to mixed locations suggests that mixed locations are not specific but generalized locations in regard to some environmental characteristics.

In summation, the t tests have pointed out that several differences exist within the pooled file of components. While Late and mixed locations vary to a minor degree, isolated Middle and isolated Late components vary considerably. Despite the consistent variation in the depth-related variables, regardless of the definition of case as component or location, there is a remarkable consistency in the common variation in locations vis-a-vis total water area and offshore features such as islands and shoals, and to some degree with dropoffs. Interestingly, the variable DEEP is not one of the strongest variables in terms of exhibiting differences in depth between cases of different time periods, which suggests that the water area that exceeds sixty feet is not of great importance in telling cases apart (Table 19). It is tempting to suggest that perhaps the greater-than-sixty foot waters are unimportant to aboriginal settlement choice and that the critical depths are perhaps in the eighteen to sixty foot range. Within this range the MED variable does help to distinguish cases on age classes, as will be seen in subsequent discussions of discriminant function analyses.

## Discriminant function analyses

One question that is essential in this research has to do with the relative ability of some variable or dimension of variability to tell the difference between groups of components that vary on the category of cultural-chronological period. What differences exist between the groups, how strong are they, and what combination of variables measures these differences the most succinctly?

The technique of discriminant function analysis can be quite useful in analyzing such questions, for the technique requires the categorical division of the data set's cases on some dimension of variation. Then the interval-scale variables within the data set are mathematically weighted in an equation that provides the maximum level of distinction (or discrimination) between or among the a priori groups. Presumably the interval-scale variables in the data set are theoretically likely to provide distinguishing values on the categories within the study. The equations, or linear discriminant functions of powerful variables, provide an opportunity to analyze both the strength of the included variables as discriminators, and to examine the affinities among individual cases in the analysis for the linear discriminant functions. Finally, the linear discriminant functions (LDFs) are the source of a set of additional functions that evaluate the likelihood that individual cases actually match their a priori group assignment in mathematical terms. This classification stage of the

analysis can also be used to assign cases to groups if some condition such as missing data values prevented their original assignment to a group. Thus the technique provides precisely what is needed in this application: a method of distinguishing between groups on commonly-shared ranges of variablility, a means of telling what variables best differentiate between groups, a means of examining individual cases and their affinities for their a priori groups, and a way of assigning cases with missing data to groups with which they share the strongest similarities.

Discriminant analyses were conducted using the SPSS subprogram DISCRIMINANT. This routine provides graphic and statistical output to aid in the interpretation of the discrimination between groups. Because the mathematics of the routine attempt to project the internal differences between groups as severely as possible, it is convenient (and factual) to regard the groups as occupying axes of differentiation in mathematical space. Thus a graphical display of the group clustering in mathematical space and group differentiation (or, less optimistically, group overlap) in this space provides a visual impression of actual differences and similarities. Also, the routine provides mathematical expressions of similarity in the form of group means (or centroids). These centroids express group central tendencies as well as providing a baseline for assessing cases that are distant from the central tendency of their a priori group. In this way outlier cases can be identified and explanation as to their sources of unique

variation sought.

On the first attempt the discriminant function analysis operated on the entire file of cases (Appendix C) subdivided into groups based on age. There were three a priori groups to be analyzed: Middle Woodland components, Late Woodland components, and components of unknown age. At the outset this appeared to be a very important analysis, but in fact it was an abysmal failure. The equations generated by the routine were, judging from their weights and their relative abilities to distinguish between groups, very poor discriminators. The classification stage of the operation reclassified nearly every case to a new group. The graphics and associated statistics indicated an extreme level of overlap among the groups. Obviously something was very wrong. Because components rather than locations were analyzed, the many components that shared identical environmental settings (and thus identical values on the potential discriminating variables) were giving contradictory information to the routine. Based on all components as the cases to be analyzed, the variables were very poor discriminators and internal differences were negligible. The remedy for this was to base the analyses on locations rather than components as cases, thus reducing contradictory information to the routine. Another alternative was to divide the cases into groups based on some category that minimized rather than maximized group overlap, i.e. define an additional group variable.

Analysis of isolated components

This analysis was attempted on a file of cases that represented isolated components: Middle Woodland components, Late Woodland components, and components of unknown age. Those components that shared their locations with components of the opposite time range were excluded from the analysis, thus leaving only uniquely Middle or uniquely Late locations to be analyzed. Thus this analysis parallels the student's t analysis of isolated components. As in the former analysis, in some cases components that dominated the real situation (e.g. the Middle Woodland components at 20CH2 and 20DE4 and the Late Woodland components at 20MK1) were assigned to the dominant group despite the presence of a nominal component of the opposing chronological group. Thus this analysis sacrifices realism (to some degree) in pursuit of simplicity. Components included as cases in this analysis are listed in Appendix C.

A total of 56 cases entered this analysis. The method of choice was the Wilks' lambda option included in the SPSS subprogram, in which the step-by-step decision to enter a potential discriminating variable into the LDF is made when that variable causes lambda to be reduced by the greatest degree, where lambda is a reflection of the differences in group centroid magnitude. The analysis proceeded on a reduced set of variables in order to minimize redundant measures within the data set. Those variables included were the normally-distributed and the non-intercorrelated, including DSHOAL, MAXD1, MAXD2, MED, AREA, and THIRTY. The

linear discriminant function for this analysis, and related statistics appear in Table 23. The variables MAXD2, DSHOAL, MED and THIRTY contribute the most to differentiating between cases of isolated components. The standardized discriminant function coefficients, or measures per variable of relative contribution to the LDF, are displayed below.

MAXD2	1.37751
MED	1.02370
THIRTY	.73616
DSHOAL	46484

Table 23. LDFs and related statistics

Analysis	Function	<u>Eigenvalue</u>	% of <u>Variance</u>	Canonical Correlation
Components (isolated)	1	.47959	100.00	. 56
Locations (age)	1 2	.24203 .00812	96.75 3.25	.44 .08
Locations (subregion)	1 2 3	1.23297 .33507 .05874	75.79 20.60 3.61	.74 .50 .23

Thus, the function represents the dominance of MAXD2, and MED in providing discrimination between groups. The function itself, with a correlation level of .56, is moderately correlated with group distinctiveness, but quite obviously this measure still shows a great deal of ambiguity.

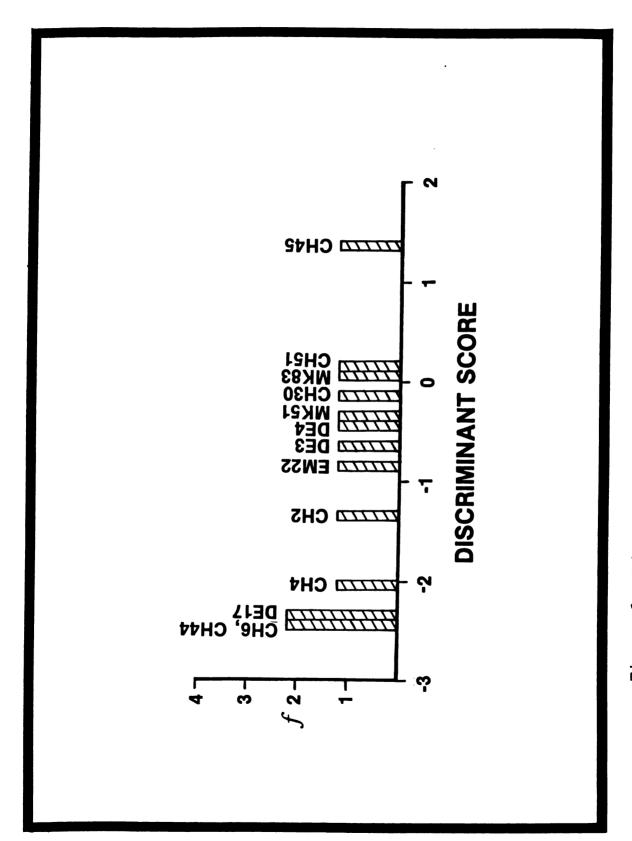
Predictably, Group I (Middle Woodland components) are negative on the function (centroid = -1.02726) and Late Woodland components are positive (centroid = .44515). The

routine correctly classified 84% of all cases, but among the Late Woodland cases it was able to match original class assignment for 93% of cases, and for the Middle Woodland group it was "correct" for only 64% of cases. An examination of the histograms in Figure 6-7 reveals the clustered character of the Late Woodland components as opposed to the spread of the Middle Woodland components.

Nearly 36% of Middle Woodland cases were mathematically more like the Late Woodland group than like the Middle Woodland group. These cases were 20MK51 (Gyftakis), 20MK83 (Arrowhead Drive), 20CH30 (Big Pine), 20CH45 (West Harbor), and 20CH51 (Fort Brady). All of these components occupy areas near moderately deep water and are relatively close to shoals; four of the five are stratified with Late Woodland components. But several components were reclassed as more Middle-like than Late: 20MK58 (White) and 20CH27 (Brown Fishery). While the site 20MK58 shares many affinities with the northern Middle Woodland locations, the reclass of 20CH27 is a bit harder to understand. Either geographical nearness to many Middle Woodland locations, or an extremely large value on the DSHOAL variable is probably responsible for this reclass.

The routine assigned Group 3 cases (the unknowns) to both time groups. Those cases close to relatively deep water were assigned to the Late Woodland group, and those adjacent to shallow water, moderately deep water, and tributaries joined the Middle Woodland group (Figure 8).

The histograms represent the cases arrayed along a



Histogram for Middle Woodland components Figure 6.

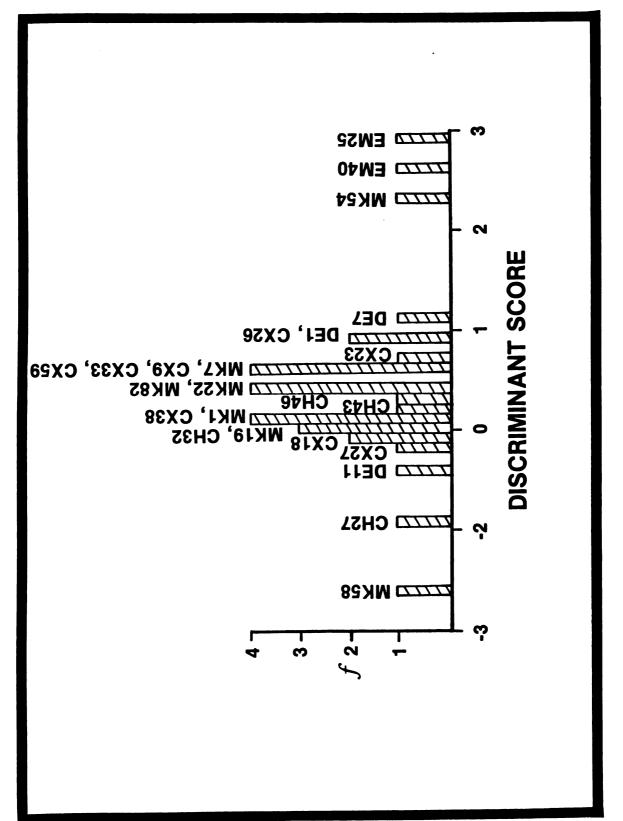
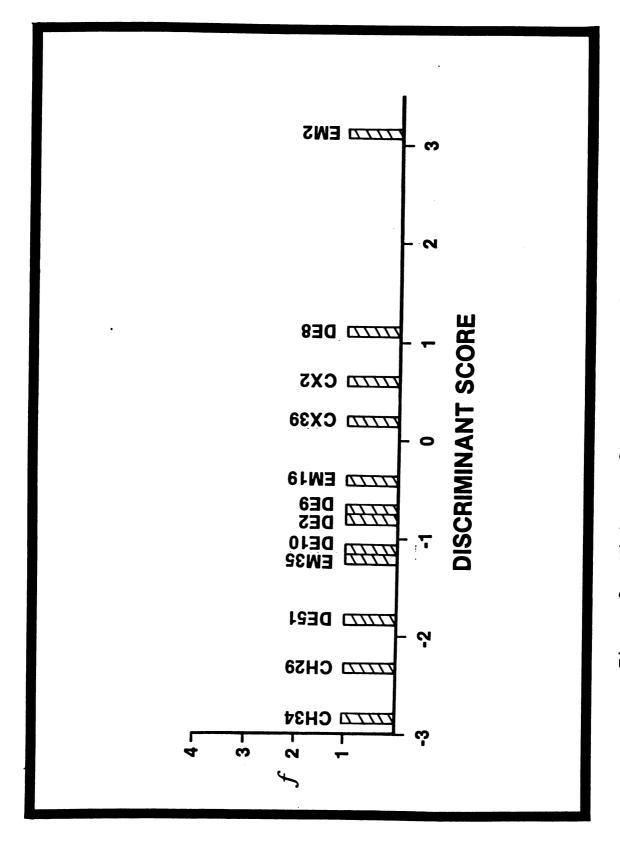


Figure 7. Histogram for Late Woodland components

dimension depicting standardized scores of each case on the LDF, i.e. how "correlated" each case is with the mathematical arrangement of variables in the LDF. It is noteworthy that in the case of the Middle Woodland sites (Figure 6) there appear to be clusters of values, one cluster occupying the -2/-3 range on the LDF, and the other occupying the -1/0 range. As might be predicted, the cases in the latter range are those that were reclassed as mathematically Late Woodland. This pattern is not repeated in the Late Woodland histograms (Figure 7), but there is again a cluster in the 0/+1 range. This creates the idea that much variation exists within the Middle Woodland group, and that the group as a whole is bipolar, whereas despite variation in the Late Woodland group, there is a strong single-pole tendency expressed. To some degree this is due to the presence of many re-occupied sites within the Late Woodland group. But there seems to be no Late Woodland counterpart to the -2/-3 Middle Woodland cluster although there are certainly Late Woodland outliers on both extremes. This situation suggests what was discovered during an early subjective assessment of the values of percentages of depth ranges; the reoccupied locations of all ages presented balanced or approximately equal potential for the use of various offshore depths, whereas the single component sites of all ages rarely exhibitied this potential and were far more "specific" for one set of dominant characteristics. The reasons for reoccupation can of course vary, but one simple hypothesis is that over time, places that are equally



Histogram for components of unknown age Figure 8.

viable for any season's use tend to be used more.

Analysis of locations by age

In this analysis the location file (Appendix C) was analyzed by age variation to answer the following questions: What combinations of variables best distinguishes among groups based on Middle Woodland, Late Woodland and mixed locations? What degree of overlap exists among these groups, and which groups are most alike? Do mixed locations lie somewhere between Middle and Late locations in mathematical space?

All operating conditions were identical to those used in the prior analysis. Table 23 lists the relevant statistics for this analysis; it is clear that the LDFs are not as correlated with the group variable (age) as in the former analysis. Note that two LDFs were calculated, for this analysis describes three a priori groups and requires more than one mathematical function to account for the space in which these three groups lie.

Two variables, MAXD2 and MED, entered the analysis before reduction in lambda was minimal and analysis ceased. Two LDFs were calculated, the first moderately correlated with the group variable at .44, and the second very weakly correlated at .08. The standardized discriminant function coefficients appear below.

	Func 1	Func 2
MAXD2	.98288	40269
MED	.71046	.78960

MED is equally expressed on both functions, whereas MAXD2 is

highly positively correlated with the first LDF and moderately negatively correlated with the second LDF. LDF 1 represents a moderate to deep dimension of water variation and LDF 2 represents a moderate to not-so-deep dimension. Predictably, Group 1 Middle Woodland locations are the most distinctive in their high negative association with LDF 1, whereas the Late and mixed locations are close in mathematical space and are positive on the first function. LDF 2 serves to distinguish between mixed locations and other locations, and is expressed primarily on the MED variable.

#### Group Centroids

	Func 1	Func 2
1 - Middle	-1.16266	03505
2 - Late	.25713	07065
3 - Mixed	.08794	.11764

A total of 44% of all cases were correctly classified, (i.e. their a priori classing was identical with the mathematical classing) but within the mixed group 64% were correctly assigned. The function did the poorest job in predicting Group 2 (Late) membership, with only 27% of cases correctly classified. A total of 50% of the Middle and Late cases were reclassed to Group 3 (mixed locations).

No Middle Woodland locations were mathematically assigned to the Late Woodland group, but 20DE3, 20CH30 and 20CH45 were reassigned to the mixed group. These reassignments are understandable given the composition of LDF 1 and the particular characteristics of the locations,

for they are all places at which there are, relatively speaking, sharper offshore contours than the mathematical function associated with the general tendencies of Group 1.

Five Late Woodland locations mathematically resembled Group 1: 20MK58, 20DE1, 20DE7, 20CX18, and 20DE51. site occupies a water area with a greater percentage of shallow water than the mathematical model associates with Late Woodland locational tendencies or with mixed component tendencies. Three of these locations are at tributaries. Eleven Late Woodland locations resembled Group 3 (mixed) 20MK7, 20MK19, 20MK22, 20CX9, 20CX26, 20CX38, 20CH27, 20CH32, 20CH43, 20EM19, and 20EM35. Though they do not generally represent Middle/Late mixes, some of these cases do represent locations that are frequently reoccupied. My impression is that these clusters of locations are very similar to the mixed set of locations in the sense that these places are of such great general fishing potential for all seasons that they are likely to be reoccupied over time, if not vertically stratified then horizontally!

I have already related that there seems to be a consistency among the Middle/Late shared locations; the stratified places exhibit roughly equal percentages of total water area in the various depth categories. The 20MK1/MK83 location, for example, showed ca. 33% total water area in the zero to eighteen foot range, 36% in the eighteen-sixty range, and 30% in the greater-than-sixty range. Other stratified sites, though not replicating the balanced numbers of the former location, do replicate the general

pattern. The most important point is that few stratified Middle/Late locations show the strong presence of one or another depth range. Those that do are quite unusual, given the contents of the data set.

The tendency to have equal proportions in all depth ranges, or, alternatively, to have equal and/or large measures in the first two depth categories is pronounced both in the case of the Middle/Late stratified sites and in the case of the group of Group 2 reclasses from the analysis. Because mixed locations 20ST1 and 20CH2 do not fall into this apparent pattern, one can perhaps conclude that these may be seasonally-specific or resource-specific locales, or that there is some intractable variation within any pattern and that there is no current explanation for why these two cases differ from the perceived pattern. Thus, 20ST1 and 20CH2 may be the ultimate sites in terms of a zero-eighteen foot or eighteen-sixty foot resource-specific locale, but they are both unusual given overall tendencies.

Four mixed locations were reclassified as similar to the Middle Woodland group: 20CH2, 201ST1, 20CH51, and 20CH77. Each includes large measures on the variables that quantify shallow or moderately shallow conditions.

Two mixed locations resembled Late Woodland group tendencies. These are 20MK61 and 20MK102, and each exhibits steep offshore contours and large measures on the deep water variables. Overall the similarities between the mixed group and the Late group were strong. Figure 9 portrays the spatial distribution of the groups with scores on each LDF

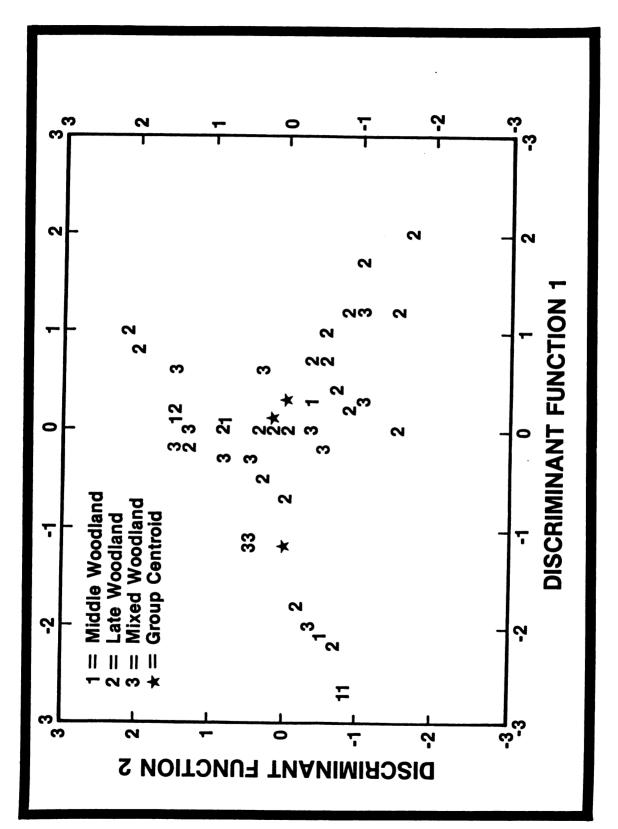


Figure 9. Scatterplot for locations, by age

serving as spatial coordinates for each case. The centroids representing the Group 2 and Group 3 cases are very close, and it is clear that there is a high degree of overlap in the mathematical characteristics of the three groups. The Group 3 cases do seem to occupy a middle position among the cases that depict Group 1 and Group 2 extremes.

## Analysis of locations by subregion

There is, of course, a tendency for some level of purely geographical similarity to enter all quantitative models of subsistence and settlement. After all, the environmental backdrops on which sites appear are integrated systems in themselves and it is beyond a matter of debate that components that are close physical neighbors will exhibit a high degree of similarity regardless of differences in age. Recognizing that this tendency exists is critical, for it is quite possible that one could confuse geographical propinguity with cultural similarity. As a group variable does geographical propinquity do as good a job or better than age in distinguishing between groups of components? I have assumed that the variables in this study are somehow sensitive to culturally-conditioned differences in subsistence and technological adaptation. Can this supposition in fact be supported?

The objective of this analysis is to compare the relative abilities of the variables to distinguish groups of cases on a new group variable. In essence, the identical data set (Appendix C) of locations was examined from a

differing perspective; what variables best distinguish among groups if the groups are based on geographical subregion?

Do differing categorizations allow the variables to do equal, poorer, or better jobs of distinguishing among cases?

And, as a final way of determining the differing powers of the functions to distinguish between groups, from which categorization do the most accurate classification functions derive?

The routine analyzed four a priori groups based on subregional spatial clusters of locations. Group 1 represented the Sault Sainte Marie area, Group 2 the Straits area, Group 3 the northern Lake Michigan area, and Group 4 the northern lower Michigan area. Three LDFs derived from the analysis, and the first function contributed a high level of explanatory power to the analysis (See Table 23). MAXD2 was the strongest positive contributor and DSHOAL was second. The second function showed a strong positive contribution from the MED variable and a moderately strong negative association with the MAXD1 variable. The first LDF seems to show a "deep and close to shoals" dimension and the second a "large area of moderately deep water" dimension. A third rather insignificant function may display a "deep but distant from shoals" dimension. The centroids suggest that Group 1 locations (Sault) are negative on all functions, that Group 2 (Straits) has a balanced affinity with all functions, that Group 3 has a strong affinity for LDF 2, and that Group 4 has a strong affinity with LDF 1 (Table 24).

Table 24. Measures associated with subregion-based discriminant function analysis of location

### Coefficients

Variable	LDF 1	LDF 2	LDF3
DSHOAL MAXD1 MAXD2 MED	.78462 71263 1.22516 .17941	.23312 61330 .24166 .98047	61899 .10547 .59930 .32670
<u>Centroids</u>			
Group	LDF 1	LDF 2	LDF 3
<pre>1 (Sault) 2 (Straits) 3 (NoLaMich) 4 (NoLowMich)</pre>	83930 35650 68874 1.78085	78865 30418 .73767 .00595	30081 .36429 .07038 06235

LDF 1 and its coefficients very clearly discriminate northern lower Michigan from the Sault and northern Lake Michigan subregions. Function 2 distinguishes between the Sault and northern Lake Michigan, and LDF 3 distinguishes between the Sault and the Straits region. There seems without a doubt to be a successful geographical discrimination of variation displayed by this analysis. The most distinctly different areas are those that are the most distant in space. Group 1/2 centroids are very close, and are very distant from the Group 4 centroid, although some degree of spatial overlap remains among the groups (Figure 10). Group 4 appears to be the most dissimilar of all the groups.

This group is highly correlated with the more powerful functions and enjoys the most success in the classing procedure with 81% of its members showing accurate

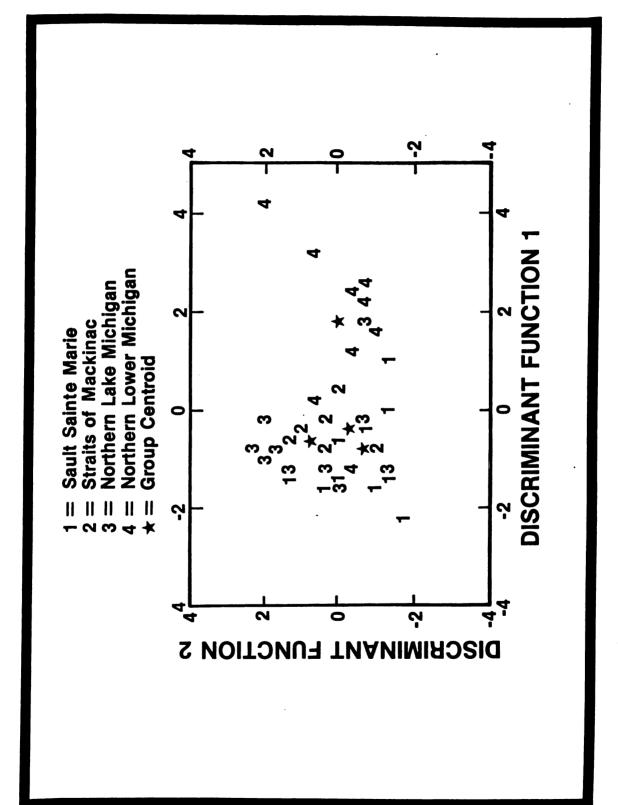


Figure 10. Scatterplot for locations, by sub-region

predictions of class membership. There seems to be a geographical gradient that accompanied the classing procedure. In Group 1, furthest north, only 40% were correctly classed. Group 2, the Straits, had 63% correct while the northern Lake Michigan subregion had 70% correct. The variables plainly do the best job in displaying Group 4 distinctiveness and are not so useful in describing the situation in the northern edge of the study area.

Many locations were reclassified to Groups 2 and 3 but few were reassigned to Group 4. The routine reassigned 20CH32, 20CH43, and 20CH46 to the Straits group as they are all adjacent to rather steep grades and include much moderately deep water, and are therefore rather atypical of the Sault subregion. Locations 20CH45, 20CH51, and 20CH77 were reassigned to Group 3. These are all locations at which there are Middle Woodland components, and the latter two include Late Woodland components as well. They all include shallow and moderately deep water, and thus belong mathematically with the Group 3 locations, which apparently include the greatest "shallow water" tendency of any group. Several other locations were reclassed to Group 3; 20MK51, 20MK7, and 20MK58 are all located near shallow embayments and have limited deep water nearby.

20MK1 has many statistical "attributes" that are apparently present in Group 4 locations, and it was reassigned to the Group 4 periphery. Those locations that were unlike the Group 4 tendencies included the O'Neill site (20CX18), which was reassigned to the Sault subregion, and

the Wycamp Creek site (20EM4) which was reassigned to the northern Lake Michigan subregion. But overall Group 4 locations can be distinguished from the others by these variables to a high degree, and its membership can be predicted with a high degree of accuracy. Group 4 is the most different group with the lowest degree of overlap with the others (Figure 10).

An examination of the canonical correlations between the derived functions and the variables (i.e. how correlated the group variables are with the functions) clearly shows that the subregion definer does a better job in explaining variation among the locations than does the group variable based on age. In the age-defined analysis the measure of correlation with the first LDF equalled .44, whereas in the subregion-defined analysis the measure of correlation equalled .74 (Table 23). But perhaps the clearest display of all is the degree to which the cases within the data set are "correctly" classified. In the age-defined analysis, a total of 44% of the cases matched statistical affinity with a priori group assignment. In the most successful group, a total of 64% were correct, and these were the locations at which mixed occupations were found. In the subregion-based analysis, 64% of all cases were correctly assigned, and in one subregion, the northern lower Michigan subregion, the figure jumped to 81%. There seems to be little question, according to these results, that geographic subregion makes a strong competitor with cultural-chronological affiliation for providing explanations of data set variability.

#### Cluster Analysis

The objective of this part of the analysis was to discover, both for descriptive and analytical purposes, whether significant subsets of similar cases existed within the interval-scale data set, and whether these subsets could be related to particular characteristics of environmental setting. Hypothetically these potential clusters of similar components and locations could be explained in reference to seasonal, technological and general subsistence-strategic aspects of Middle and Late Woodland adaptations, and internal variation could be compared between the major time-defined groups.

The clustering method selected was the BMDP.P2M case clustering subroutine. This routine groups cases on a measure of overall multivariate similarity, or Euclidean distance, where the distance between any two cases or clusters j and k is calculated as follows (Dixon 1981:459):

$$d j,k = [i(x_{ij} - x_{i})^{2}]^{1/2}$$

Those pairs of cases or clusters with the minimum between-case mathematical distance become larger clusters, and the routine proceeds to join cases to clusters until all cases are members of one large heterogenous cluster. The routine requires standardized data to overcome the potential problem of non-normality or out-of-scale values, to which Euclidean methods of clustering are sensitive. Accordingly, the transformation aspect of the BMPD software operated and mathematical distance measures between cases and clusters

were calculated on z-scores derived from the raw variable values. The centroid-linkage method of selecting minimum-distance pairs was used. In the centroid linkage method, a case joined any cluster whose centroid (mean) values were mathematically closest to case values. Therefore new cluster members resembled the average cluster values rather than the values of a particular cluster member.

As with other clustering techniques, there is some inherent subjectivity involved when interpreting the results of the clustering procedure. The object of clustering is to derive meaningful clusters, but in fact there is no clear criterion with which to judge when case additions to clusters are insignificant, or at what point excessive case additions obliterate meaningful detail. There is likewise no firm and objective means of determining at what point meaningful (i.e. interpretable and "honest") clusters do occur. Some element of subjectivity will always remain in the decision to cease clustering, guided by the desire to maximize detail while minimizing excess numbers of clusters and the desire to yield interpretable results based on clear statements of theory, assumption, and the inherent structure of the data.

Cluster cutoff points may be natural in the sense that an examination of the incremental distance to cluster among cases (i.e. how much the multidimensional distance between cases or clusters increases between successive cluster steps) may reveal points at which very large or sudden

increments as opposed to small steady changes occur. large increase in the amalgamation distance indicates a point at which relatively dissimilar entities become co-members of a cluster. Another characteristic of the clustering procedure that occurs near the end of the clustering is the advent of chaining, or the stepwise additions of all remaining outlying cases to an increasingly heterogenous cluster. In this exercise the cluster cutoff points (i.e. the points at which clustering ceased and interpretation began) signified the levels of maximum detail prior to the beginning of chaining. Early clusters, largely attributable to geographical nearness rather than cultural similarity, were overlooked in favor of large-scale or general trends. Validation of cluster tendencies derives in part from confirmation of general discriminant function results (Aldenderfer 1982).

Clustering of Middle Woodland components

Twenty-five cases representing the subset of Middle Woodland components were analyzed; the assumptions, requirements, and shortcomings of the method have been discussed above. The present analysis examined whether the Middle Woodland subset of cases was internally diverse, and, if so, challenged the investigator to explain the case clusters in reference to similarities and differences in environmental setting and subsistence strategies. Cases with missing data values, specifically those two cases from the Sault Sainte Marie area (20CH51, 20CH77) had to be

deleted from this analysis. The initial analysis included the variables DSHOAL, MAXD1, MAXD2, MED, AREA, and THIRTY. Non-normally distributed and redundant variables were deleted from the analysis. The sequence of cluster formation may be seen in Figure 11, and Table 25 displays centroid values on each variable for each cluster at its formation step.

By Step 15 four clusters had appeared; the most homogenous, at least as reflected by the incremental amalgamation distance, was the first cluster, formed at Step 8. Cluster I's four members are all stratified with Late Woodland components, and their locations share a relatively balanced composition of depths, gradients and offshore conditions recognized earlier among locations at which stratified occupations are to be found. They share high values on the MED variable. Most of these cluster members are in the western Straits-northern Lake Michigan area, where they are found in protected embayments.

Cluster II represents a group of five components similar due to geographic proximity and to the structures of environmental setting. All of these components occupy areas adjacent to extensive shallow water, a kind of setting that has been perceived as the modal sort of Middle Woodland early warm season location (Cleland 1982). With one exception the members of this cluster are close to the Saulte Sainte Marie area and border small tributaries. It is clear that the members of this cluster do not, in general, share their locations with Late Woodland components

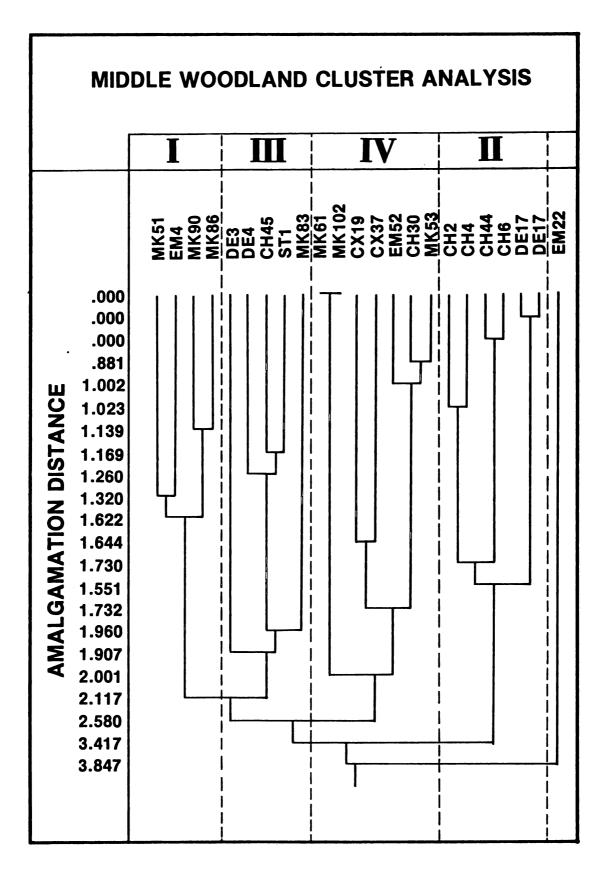


Figure 11. Middle Woodland cluster analysis

Table 25. Centroid values by cluster, Middle Woodland cluster analysis

<u>Variable</u>	Cluster <u>I</u>	Cluster <u>II</u>	Cluster <u>III</u>	Cluster <u>IV</u>
DSHOAL	287	.102	481	070
MAXD1	656	-1.049	.064	1.285
MAXD2	467	-1.077	153	1.203
MED	1.012	-1.257	.925	247
AREA	748	831	1.203	.388
THIRTY	.121	1.290	400	857

## Cluster Members

Cluster I	Cluster II	Cluster III	Cluster IV
*MK51	*CH2	DE3	*MK61
*EM4	CH4	*DE4	*MK102
*MK90	CH44	CH45	*CX19
*MK24	CH <b>6</b>	*ST1	*CX37
	DE17	*MK83	*EM52
			?CH30
			*MK53

# Unclustered

<sup>\*</sup>EM22

<sup>\*</sup>CH77

<sup>\*</sup>CH51

<sup>\*</sup> stratified with Late Woodland components

(Figure 11). This cluster is collectively the furthest from shoals of all the Middle Woodland clusters, and its members have relatively small local water areas.

Cluster III's five members, three of which occupy islands, include large total water areas and lie close to moderately deep water in the eighteen to sixty foot range. They frequently share their locations with components of later time periods. Overall these places are the closest of all clusters to shoals, and in general this cluster most closely resembles Cluster I.

Cluster IV represents components that occupy locations close to the deepest water, but they are, as a group, rather distant from shoals. They share low values on the variables that measure shallow and medium water depth characteristics. Each of these locations is stratified, or, for all practical purposes, shared wih one or more Late Woodland components. This cluster includes many components close to the Straits of Mackinac subregion.

The integrity of Cluster II is maintained nearly to the end of clustering, and at the only point in the routine where a visible natural break occurs in the incrementation of the amalgamation distance, the cases are divided among the large heterogenous stratified Cluster I/III/IV, the shallow water Cluster II, and the Portage site (Figure 11). The Portage site is clearly an outlier in this analysis. The essential bipolarity of the Middle Woodland case subset is visible, and the bipolarity reflects both the environmental situations and the clustering of stratified as

opposed to isolated components.

Clustering of Late Woodland components

It was then necessary to assess internal variation among the forty-six components representing Late Woodland occupations, to determine whether Late Woodland components were very similar in environmental terms, or whether there were identifiable groups of components that perhaps represented seasonal or resource-specific strategies of settlement and subsistence systems. Those components with missing data were excluded from the clustering procedure. The actual missing data were the percentage measurements on several depth categories, irretrievable due to the recent drastic alterations in the Saint Mary's River channel. other forty-four cases, each representing a Late Woodland component, were analyzed on the interval-scale variable set using BMPD.P2M clustering. All conditions of the clustering exercise were identical with those in operation during the cluster exercise on Middle Woodland components. results, clusters at several stages of stepwise clustering, and the amalgamation distances between steps are portrayed in Figure 12 and in Table 26. The amalgamation distances measured at .000 represent the "clusters" formed when two or more components sharing a locaton join, so potential meaningful clustering begins at Step 12.

Cluster I represents eleven locations, most of which are stratified with many occupations; there is a total of twenty components in Cluster I. This is a remarkable

Figure 12. Late Woodland cluster analysis

Table 26. Centroid values by cluster, Late Woodland Cluster analysis

<u>Variable</u>	Cluster <u>I</u>	Cluster <u>II</u>	Cluster <u>III</u>	Cluster <u>IV</u>
DSHOAL	392	289	1.255	718
MAXD1	529	951	1.294	1.422
MAXD2	461	-1.157	.923	1.278
MED	.209	1.175	700	766
AREA	.171	-1.245	319	. 988
THIRTY	.197	1.645	-1.153	699
Cluster Members	MK1* MK19 CH43 CH46 CX9,33, DE4* CX18 CX26 EM4* MK82* MK82*	MK90* MK86* MK7	CX19* CX38 CH27 CH32? CX40* CX23	MK61*,102* EM40 EM51* MK53* MK54
Unclustered	ST1* DE7 DE1 CX27 EM25 CH2* MK58 EM22*			

<sup>\*</sup> stratified with a Middle Woodland component

cluster for it represents what is thought to be the principal kind of large Late Woodland coastline settlement locality. Three of the members of Cluster I are island locations, each of which includes a Middle Woodland component. The members of this cluster share balanced conditions offshore, at least in relation to water depth areas, and these locations are, on the average, rather close to shoals. Careful examination of raw variable values reveals, however, that in fact the distance to shoals is quite varied for this cluster (range = .8 to 10.5 kilometers). Most Cluster I members include a rather local water area (but these values again appear highly varied) and commonly half is less than thirty feet in depth. Repeated multiseason occupations seem to be the norm for this cluster, and this is true not only of those places stratified with Middle Woodland components but the exclusively Late Woodland places as well.

In contrast, Cluster II's members are adjacent to shallower water (at three kilometers offshore ca. 65' range), and are further from islands, shoals, and dropoffs than are the members of Cluster I (greater than three kilometer range). These locations, rather than having equal proportions among all water depth categories, concentrate on the eighteen-sixty foot depth range. In general most of these places include as much as 70% of local water that is less than thirty feet in depth. These locations are characterized by relatively shallow, protected water with gradual offshore contours and few neighboring offshore

topographic features. All of these locations occupy embayed shorelines so they share relatively small local water areas. They are close in space, all occupying the northern Lake Michigan shoreline. Two of these locations include Middle Woodland occupations.

Cluster III's six members, three of which share their locations with Middle Woodland components, are furthest of any cluster from shoals, but share very deep water close to shore. There are limited areas of shallow and moderately deep water nearby. Some of these locations are exposed but relative to the other places analyzed in this exercise, these locations share small to average areas of total local water.

Cluster IV's six members are adjacent to very deep water, are closest of all clusters to shoals, and with one exception share locations in the Straits of Mackinac subregion. Four of the six include Middle Woodland components. As a rule they occupy exposed areas with relatively large local water areas, of which perhaps one-third is less than thirty feet in depth.

Judging from the tree diagram of the clustering procedure (Figure 12), there are two major cluster tendencies, that representing shallow to average or general depth characteristics (Cluster I/II) and that representing relatively deeper offsore conditons (Cluster III/IV). All clusters include members that share their locations with Middle Woodland components.

Though clustering did occur, my impression is that

there were few strong distinctions among the clusters, and the Late Woodland components are quite generally speaking rather more homogenous as a set than are the Middle Woodland This is not to say that all Late Woodland locations are the same but merely to say that they are, moreso than the other set, inherently less varied on the variables included in this study. This idea supports the conclusion from the discriminant function analyses, in which it was apparent that the Late Woodland cases clustered around a single pole of variability. But it is important to point out that there are relatively few Late Woodland components in the northern subregion of the study area, so that the Late Woodland set is less varied, as far as geographic spread is concerned, than the Middle Woodland set. It is also the case that the Late Woodland locations are more likely to be stratified with many occupations, and this a priori spatial clustering promotes the internal homogeneity of the set.

## Conclusion

When Middle Woodland components are compared to Late
Woodland components on nominal/ordinal variables included in
this study, there is no apparent association between the age
of the component and sets of values on the variables.
According to the outcome of the chi-square contingency table
analysis, there are no differences among subsets of
components based on age and environment characteristics, nor
are there distinctions between age and those archaeological

characteristics known in great enough detail to test.
Without the guarantee of independent random sampling this conclusion must remain a bit infirm.

When Middle Woodland components are compared to Late Woodland components on interval-scale measures of age group means, the variables that measure maximum water depth at two and three kilometers distance from a location are significantly larger for Late than for Middle Woodland components. Likewise, the variable that measures total water area less than eighteen feet in depth is significantly larger for Middle than for Late Woodland components. The components do not differ drastically on the variables that measure features offshore nor on those that measure total available water and total available deep water.

When isolated Middle Woodland components are compared to isolated Late Woodland components on interval-scale variables, there are significant differences between groups on all the depth-related variables and on the variable that measures distance to a dropoff. Other offshore variables and the variable that measures total available water area show no significant differences. Late Woodland components are closer to dropoffs, and are associated with water that is excessively deep when compared to that adjacent to isolated Middle Woodland components.

Likewise, isolated Middle Woodland locations differ from locations at which there are stratified occupations of both time periods on the variables that measure maximum depths at two and three kilometers' distance from the

locations. Offshore features do not co-vary with the group variable. Isolated Late Woodland locations at which there are occupations of both time periods are essentially statistically identical, though there is a tendency for the Late Woodland isolates to favor slightly deeper water areas.

Multivariate analyses replicate the results of the comparisons of group means to some degree. The variable that measures maximum depth at two kilometers and the variable that measures total water area in the middle range of eighteen to sixty feet do the best jobs in distinguishing between isolated components. When comparisons are based on locations themselves (rather than isolated components) the same two variables distinguish between Middle, Late and mixed locations. When the comparisons are among groups of components subdivided by subregion, the depth variables at one and two kilometers' distance and the variable that measures distance to shoal or reef contribute the most to group distinctiveness. The group variable based on subregion is more correlated with group distinctiveness than is the group variable based on age of the component, which supports the conclusion that there is a geographically-based gradient of difference running through the study area that may possibly be confused with differences that occur because of subsistence-settlement changes through time. All analyses conducted by discriminant function techniques support the conclusion that there is a high degree of overlap among subsets of components based on age.

Both the cluster analyses and the discriminant function

analyses support the conclusion that Middle Woodland components are more varied in location than are the later components. This is due in part to the wide geographic spread of the Middle Woodland components. Middle Woodland components occupy generalized locations that are almost always shared with Late Woodland components, as well as isolated shallow water locations that are often near tributaries. Late Woodland components cluster around a pole of generalized locations often shared with earlier occupations, but some Late Woodland places are close to relatively deeper water. These places are occasionally stratified but often reoccupied exclusively by Late Woodland people. Overall each age group occupies all kinds of available locations, but differing emphases rather than addition or replacement of locales seems to be the key to understanding change through time.

There are more Middle Woodland sites in the northern subregion than in the southern subregion, and this may represent the spread of Lake Forest Middle Woodland settlement into the study area. The earliest southerly Middle Woodland component is to be found at the Portage site (20EM22) in a rather unusual environment that truely may serve a function not primarily "caused" by the need to collect food. The distinctive character of this location has been apparent throughout the analyses of component locations. The people of the Middle Woodland cultures may have adapted earlier to the Sault Sainte Marie area and to the northern shore of Lake Michigan than to the southern

subarea, though they did make use of the Straits area very early. In the north they made use of a variety of environmental settings: river rapids (20CH51, 20CH77), islands (20CH45), shallow tributary mouths (20CH44, 20CH6), beaches with steep offshore contours (20CH30), and shallow water beaches (20CH2). In the Straits and northern Lake Michigan area they occupied islands (20EM4), shallow tributary mouths (20DE17, 20EM4), beaches with steep offshore contours (20MK53, 20MK102, 20MK61), shallow water beaches (20MK90, 20MK86, 20MK51), places close to moderately deep water and dropoffs (20ST1), and places of generally mixed conditions (20MK83, 20EM52). In the southernmost subregions, they occupied beaches with steep offshore contours (20CX37), and places of generally mixed conditions (20CX19). Their presence in the southern area is faint compared to that further north. Middle Woodland populations did not use the northern lower Michigan subregion heavily. Overall the Middle Woodland components suggest: 1) seasonal use of shallow-water environments and 2) seasonal use of general or deeper-water environments. The Middle Woodland adaptation may have involved mid-season movement from shallower to appropriate deep-water locations for the late summer/fall fishery. At neither kind of location were the camps very large, and with some exceptions it is suggested that the two sets of components were equal in occupational intensity, that is, both kinds of components represent temporary seasonal camps of equal population size.

The Late Woodland components are somewhat different.

Whereas the Middle Woodland adaptation tended to suggest relatively short-term seasonal occupation of places (with a few potential exceptions, e.g. 20CH2), locational evidence supports the repeated multiseasonal use of certain places by Late Woodland populations. These reoccupied places are markedly general compared to other occupied locations from all time groups; thus they may tend to be reoccupied not only as a matter of culturally-determined choice but because in a probabilistic sense they are always suitable for occupation. These places were occupied in many cases by Middle Woodland populations as well, but the depositional evidence would suggest short-term rather than entire warm-season use for the Middle Woodland levels. some exceptions to this generalization about co-use, specifically the O'Neill site (20CX18) which has no discernable Middle Woodland occupation level, and likewise the generalized localities in the Saint James Harbor area of the Beaver Islands.

Some of the Late Woodland components are small in size and were likely occupied seasonally or by small work groups for a short time. These places are rarely stratified with other occupation levels of any time period. The Late Woodland pattern suggests the elimination of major warm-season moves between localities of differing fish potential, and concentration on those areas that can be judged as reliable producers of fish throughout the warm season. These they occupy again and again, despite occasional short forays to "unusual" places, namely ones

such as 20CH2, 20MK58 and 20CX26, and the steep deep kind such as 20MK54 and 20CX27.

The major summation is the point that the Middle Woodland people occupied nearly all types of localities, even the steeply-graded deep-water types (e.g. 20CX37, 20CH45, 20MK61, 20MK102, 20CH30), but they may have done so one season at a time, and this fact seems to be the case for all places with the possible exception of 20CH2. By contrast the Late Woodland people occupy the same kinds of places but emphasize the generalized rather than the specialized localities. There is apparently greater diversity within the Middle Woodland set as a result, though this tendency is made a bit cloudy by a differing Late Woodland spatial spread. But if one can discount sampling error, then this tendency is apparent by examining coefficient of variation measures for groups (Table 18).

The situation in terms of relating specific intervalscale measures of environmental variation to site location,
and ultimately, to subsistence-settlement systematics, is
made complex and difficult by a backdrop of unrelated
environmental variation. Unless a study area is small
enough or uniform enough to filter out differences of scale,
this source of variation will remain confusing. In areas
where sites occur in low numbers the problem will generally
be large, for in order to make statistical sense of a region
with low site densities one must necessarily include a large
area. This analysis shows that there are clusters of sites
within the defined study area whose most prominent

characteristic of similarity is that they are geographical neighbors, and this tendency is apparently stronger than a tendency to cluster together because of cultural-chronological similarity.

It may be that one is able to measure site likenesses based on environmental location, without a mask of broad-scale variation obscuring the configuration of features, but that in this case the wrong variables have been chosen to contrast cultural-chronological differences. There are a number of ways in which this situation could have arisen: 1) the variables could be appropriate for one subregion but not for another, 2) the variables could be inappropriate for all subregions, 3) the variables were too crudely designed to reflect relevant differences between subsets of locations, 4) there are no measurable differences between subsets of locations.

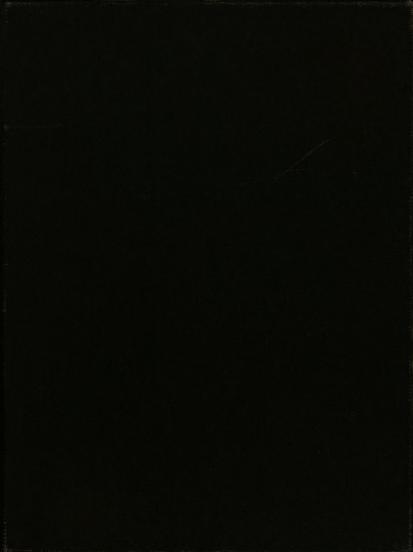
It is clear that for some reason there are relatively few Middle Woodland locations in the northern lower Michigan subregion. In this area there is support for the notion that adaptation to offshore features happened during the Late Woodland time period, but elsewhere these kinds of locations were occupied in numbers earlier in time. With the exception of the Portage site, a location that is by all accounts at odds with other Woodland locational tendencies, all radiocarbon evidence suggests relatively late uses of the northern lower Michigan subregion by Lake Forest Middle Woodland people. The earliest known Middle Woodland occupation of this region occurs at the Portage site

(20EM22) at A.D. 120 +/- 120 (DIC-652, unpublished), derived from the lower zone, Feature 2. Other Middle Woodland occupations in this subregion occur at 20CX19 (estimated at A.D. 600), and at 20EM4 [A.D. 630 +/- 120 (M-2065)].

Despite a relatively heavy use of this area in the years post-dating A.D. 1000, there are few indications of Middle Woodland use on the same scale as that which occurred in the other subregions under current study. There is quite clearly a gap of nearly 500 years' duration in the radiocarbon sequence for this sub-region that may eventually be connected with a gap in the sequence of occupation.

The foregoing analyses provided some estimates of actual locational differences among components of sequent time periods. In addition they suggested that some geographical areas were occupied earlier and more densely than others. Finally, they provided a pattern of site locational factors in time and space that will stand as a real-world model against which simulated distributional patterns may be compared.





3 1293 10792 3157

THESIS

107 THS

> LIBRARY Michigan State University

PLACE IN RETURN BOX to remove this checkout from your record. TO AVOID FINES return on or before date due.

	n on or before date due.	
DATE DUE	DATE DUE	DATE DUE
<b>L</b>		
SEP 2 8 222		
0 <u>00026</u>		
M419225 2104		
_:Uil 1 4 2008	"8	
Y <u>2 1 2012</u> 1½ 1 9 11		

MSU Is An Affirmative Action/Equal Opportunity Institution

c:\circ\datedue.pm3-p.1

#### CHAPTER FOUR

#### THE SIMULATION OF ARCHAEOLOGICAL SITE DISTRIBUTIONS

### Introduction

Chapter Three indicated that there were subtle differences in the environmental characteristics associated with Middle and Late Woodland settlements. The places chosen for settlement not only changed through time but across the space of the study area as well, with the southern-most areas evidencing their heaviest prehistoric occupations in the years that followed A.D. 1000. Additionally, as a group the Middle Woodland settlements in all areas were more diverse in character than were the Late Woodland places, whereas the Late Woodland places tended toward homogeneity as many were stratified with multiple occupations. In the light of these differences, a simulation of the establishment, growth and spread of settlement across the study area offers an opportunity to study the consequences of currently-held assumptions about the magnitude of population growth, the timing of technological innovations, and the significance of varying environmental characteristics for the distribution of archaeological remains.

It will be recalled from Chapter One that simulations are a particular kind of modelling exercise, whose uniqueness rests in the fact that they predict the changing states of systems through time. Some of the prime justifications for simulation research derive from the

clarity required for simular modelling, the replicability of simular results, and the revelation of counterintuitive relationships among important model elements. Another of the justifications for the simulation approach is its simplification of real world patterns to enhance comprehension of complex phenomena. Thus the simulation attempted in this study necessarily eliminates many aspects of detail that do have some consequences for the known distributions of the sites within the study area. patterns that emerge from the simulation exercise must be viewed in this light. Their similarity with the real world pattern will be distorted, for instance, because the simulation overlooks influences of site destruction and differential site discovery on the distribution of archaeological remains. The simulated patterns resemble a real world pattern that might exist were these differential processes held constant. The simulation's utility lies both with its ability to produce a facsimile of the real system's data (i.e. mimic the patterns of real world site distribution) and to predict the real system's behavior (e.g. to offer hypotheses about site distributions in unresearched areas). But its ultimate goal is structural validity: to reflect the structural relationships that produced the real system. Without the final qualification of structural validity, the simulation model falls short of its potential benefits. Unfortunately the methods of demonstrating model validity are undeveloped and imprecise for archaeological as well as other applications (Zimmerman

		•
		;
		:
		•

1978:35). The overall benefit of this sort of approach seems to lie in its potential to generate new hypotheses about system operation; traditional archaeological means may confront and settle the validity issue. System simulators in other fields have suggested the following validation criteria, for the job of model validation is no more easily accomplished in other disciplines than in the special world of archaeological applications. Shannon (1975), for example, poses the following series of validation criteria. First, does it make sense? Other key questions include whether the model performs predictably when fed test data, and whether the model performs in correspondence with expected outcomes of statistical tests. Finally, Shannon (1975:236) suggests that ultimately, models regardless of source must face rejection by field testing as a true assessment of validity.

The subject of detrimental aspects of the simular approach does not end with the validity dilemma.

Simulations are expensive, require some programming expertise, and are so much fun that manipulation of the model for its own sake may replace dignified scientific research! A structurally valid model that produces innovative results is liable to be as complex, and as intractable to understanding, as the real system that it simulates. Finally, the results may be pushed beyond their true significance. Shannon (1975:10-11) calls this tendency the "deification of numbers." It is somehow comforting to realize that these problems are not the private domain of

archaeological applications, but abound in all applications of the simular approach.

### Questions of scale

The aim of this simulation is the creation of a pattern of archaeological site distributions as a response to the distribution of fishing-related resources and people. a small scale model, that is, one that operates over a long range of time and a broad space. It assumes that every use of a place contributes, in some small way, to the cumulative archaeological record of that place. It disregards post-depositional processes that quite obviously alter the appearance of the archaeological pattern. The use of the simulation approach enables us to ask and answer certain questions about seasonal place use. The question is simply, what values of the critical variables (i.e. population growth rate, timing of the gill net innovation) create familiar site distributional patterns in the light of the other system variables (i.e. environmental failure rate, seasonal resource potential)? Within the simulation the actual operations that create the patterns of place use are quite simple. The simulation "creates" lodges and repeatedly seats each lodge at attractive locations as the seasons change. Lodges may "co-reside" with lodges that are recognized as agnates or affines. Co-resident kin may "exchange information" about resource distributions and technological skills. The simulation keeps a record of seasonal place use over "generations" of simulated time as

Ş
Ø
Ş
:
:
:
:
:
:
:

system environment changes and as resource-related events occur. Attractive locations are assumed to be the result of several interacting factors: the presence of kin, the diversity of seasonal resources, prior use, possession of the gill net technology, distance from present location.

Unlike Zimmerman's (1977) work, this attempt is not a simulation of a total society. Such an exercise, while theoretically possible, encompasses bodies of data that are not available for the problem at hand, nor are they particularly desirable, because the goal here is a very general one in terms of scale and output. It is quite possible, for example, that few of the individual nuances of locational behavior (a subsystem of enormous complexity) will ever be understood. Some are frankly beyond our capacity to understand, deduce, or appreciate. These very important levels of individual decisionmaking must be put aside in order to look at approximations of the overall pattern that such decisionmaking left behind. The model borrows certain principles of organization from the body of ethnographic information, but it is primarily a model of distributions rather than a model of how decisions about where to live were made.

## The Choice of a programming language

The simulation program was written in the PASCAL programming language, for the following reasons. First, the available computing facilities favored the choice of PASCAL.

Adequate programming assistance and consulting existed for

the PASCAL language, and it was cheaper to run and less restricted in terms of facility use than were the languages specifically designed for simulation. But the most critical factor had to do with the structures or world views of languages designed for simulation programming. simulation languages include a host of internal arrangements that enable the modelling of complex relationships among entities or events. The language GPSS, for example, is useful for sequencing an event-driven series of steps through a set of facilities. Thus GPSS might be useful for modelling the manufacture of some artifact type, in which there is interest in knowing what time-related characteristics the manufacturing process involved. same vein, the structure of the SIMSCRIPT language allows the modelling of complex relationships among people, objects, or entities. SIMSCRIPT would be useful for modelling the changing clan affiliations of a population through time, or modelling a hypothetical kinship system shift. One interesting elementary application might be the modelling of the debated Natchez social class/marriage system. Because the present problem did not include inordinately complex relationships among entities, nor did it dwell on event-driven aspects of model operation, the choice of PASCAL was efficient in terms of computer resource use. A program listing may be found in Appendix D.

## Critical assumptions of the model

In the simulation program, lodges are created to choose and occupy suitable locations, to exchange locational and technological information with kinsmen, and to produce a record of habitation. The total population may grow through time, subject to a set of environmental constraints. A timed technological innovation, the introduction of the gill net, may provoke changes in seasonal resource productivity, may allow novel alliances among affines, and may produce a response in the accumulated "history" of site locational patterning. The timing and intensity of the principal aspects of the simulation may be altered to depict competing hypotheses about system transition during the Woodland cultural-chronological period. Table 27 demonstrates the attributes possessed by the principal entities in the simulation, the individual lodge and the location cell.

Table 27. Attributes of principal entities

Lodge (n may vary)

identifier number	x-coordinate
<pre>identifier of agnate lodge(s)</pre>	y-coordinate
<pre>identifier of affine lodge(s)</pre>	spring resource score
list of former locations	summer resource score
current location	fall resource score
gillnet status	fall resource score (net)
	list of current occupants

Location cell (n = 21)

Quite obviously there are a number of assumptions

underlying this simulation that require clear articulation before the operation of the model may be understood. it is assumed that the lodge or household constitutes the appropriate social unit for studying the process of locational change. This assumption is made recognizing the autonomy of the lodge as the instigator of movement during the historic time period in the area under study (Henry The autonomy of lodge movement is a feature of 1969). recent northern Ojibwa life as well (Dunning 1959a, 1959b, Landes 1961, Jenness 1935). It is important to understand that within the simulation the lodges are basically immortal; that is, they do not leave the simulation as new lodges enter. Rather, they perpetuate themselves over the generations of the simulation. In this characteristic the simulation is no different than a real system of life viewed at the household level, for though the individuals making up a lodge may come and go, the lodge itself may be considered a permanent entity whose membership changes through time. During one season the lodge may contain a man and his family, thirty years later that lodge may include the man and his married son, and thirty years after that the lodge may include the man's son and grandsons. In this sense the lodges are not only immortal but are capable of carrying a family history of past decisions about where to live and about the probable distribution of resources. Most critically, in the real system it was at the level of the autonomous lodge that decisions about where to live were apparently made. This topic wil be covered more fully

below.

Likewise, it is appropriate to allow time to advance generation by generation in the simulation exercise. alternative arrangement would have allowed time to advance year by year. However, the lengthy time span of interest (1200 years) created a difficulty in terms of potential computational expense and of modelling accurately the scale of population growth. To some degree the lodge as the unit of analysis took precedence over other considerations and the time advance arrangments were considered less critical. Suffice it to say that, assuming realistic population growth rates for hunter/gatherer/fisher adaptations in temperate latitudes, a generation is about the minimum real time necessary to produce enough additional people to populate a new lodge. Otherwise, at a year by year time advance, the model would be required to add a fraction of a "new person" to the growing population every year. This seemed highly irregular, so lodges rather than persons are added generation by generation, assuming that other constraints on population growth were satisfied.

Within each "generation" or cycle of the simulation, there are three "seasons": spring, summer and fall. Once a season each lodge has an opportunity to choose an attractive place to settle. While it is obvious that in the real system, lodges did not adhere slavishly to a program of one move per season, it is assumed that changes in resource distributions related to seasonal changes exerted a timed effect on the seasonal distributions of prehistoric people

in the study area. The actual causes of lodge movement are another matter altogether, and probably related to such factors as stress management. For example, the group of people composing the lodge of Wawatam (Henry 1969) moved in response to a number of impulses (among them prescient notions, childbirth, illness, fear of enemies, lack of food) all within the general pattern of seasonal changes. Nor were all encampments of equal duration. In the Henry example mentioned above, the duration of encampments varied from one to fifty days per place. For the purposes of this simulation it is assumed that every lodge has at least one opportunity per season to move, and that all encampments are of equal duration.

It is also assumed that the possession of the gillnet changes the attractiveness scores of some of the location cells, and that the overall population growth rate may change after the acquisition of the device. If some spatial clusters of lodges have the gillnet, populations may change most quickly in those areas. The model assumes that stochastic environmental problems not only occur but operate to stop population growth. The rate of environmental flux may vary as the number of lodges with the gillnet increases, reflecting the vulnerability of a specialized system of capture to random changes in local conditions. After a threshold percentage of lodges acquire the gillnet, both the population growth rate and the rate of environmental flux may change.

Lodges may always co-reside with those lodges

		·
		;

designated as agnates, and may sometimes co-reside with their affines. They may never co-reside with lodges that are neither agnates nor affines unless the lodges share a common co-resident relative. Co-resident relatives of all types may exchange information about past locations and may "give" one another the gillnet technology. Finally, the acquisition of the gillnet may change the likelihood that people will choose to live with their affines. Figure 13 offers a general view of these activities within the simulation program.

### Resource score calculations

Resource scores are critical to this study, but unfortunately the available data on resource scores does not allow the fine grain of resource distributions that one would hope. Considerably finer estimates of resource distributions would improve the precision with which predictions about prehistoric occupations could be made. As it stands, the grain of the resource information (i.e. how precise one can be about the attributes of a location cell) ia rather large, and the study area contains 21 cells that are potentially available for occupation. Each cell has four resource scores: spring, summer, fall, and fall with gillnet technology. Resource scores related only to fish fauna and forsook all other kinds of resources. The reader will recall from Chapter One that the study area was divided into sectors representing the statistical fishing districts of the Upper Great lakes (Goodyear et al 1982). Each sector

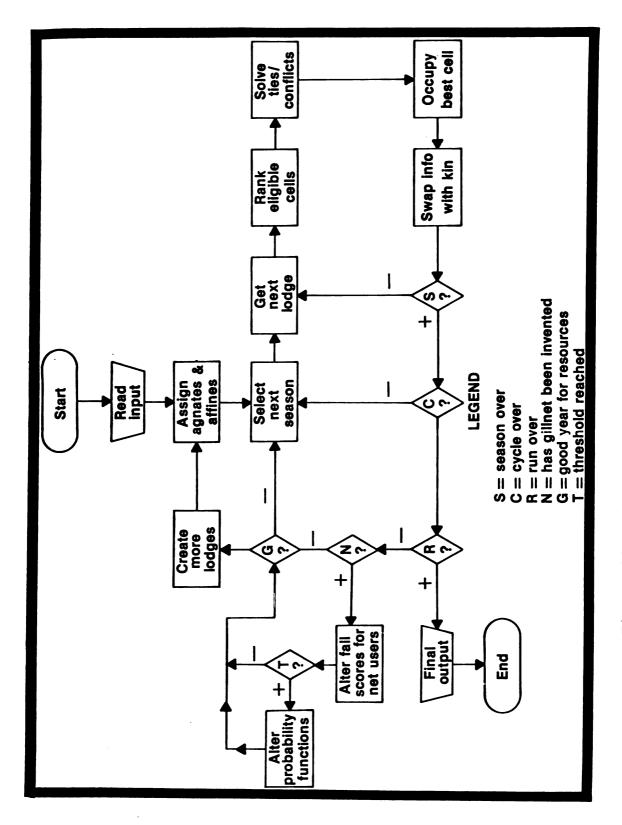


Figure 13. Flow diagram of simulation program

was described in terms of coastal contexts, including tributaries, mainland littoral, offshore, and offshore littoral. The seasonal resource score is a composite figure that represents the sum of favorable physiographic conditions for fishing as well as the presence of individual species. Each score included points for total number of species by context, boosted by additional points for tributaries, islands and shoals. Thus, areas with many major and minor tributaries, shoals, and islands, and with sightings or probable sightings of many spawning species in multiple coastal contexts have high resource scores, whereas areas with no tributaries, shoals, or islands, and a solitary species sighting in one coastal context have low resource scores. Low-scoring areas can be profitable to exploit, however, in the sense that they may be highly specialized. The "fall score with gillnet technology" was the only score that took into account the distribution of fall-spawning species over both offshore and offshore littoral contexts, since these were the contexts to which the gillnet technology presumably gave special significance. All spawn and contextual data were taken from Goodyear et al (1982).

The evaluation of differing potential by season takes into account the two major constraints on appropriate technology, namely, the characteristics of the hydrologic features associated with an area, and the particular habits of the target species. By virtue of the gillnet resource score, the values include an expression of the importance of

the hypothetical key technology in relative estimates of local resource potential. Because there are few data from which to obtain spawning locality information about the lake sturgeon, it was assumed that all shallow inshore contexts were suitable for lake sturgeon spawning. But otherwise the use of contemporary spawn-sighting data and its applicability to the prehistoric situation cannot be accepted without some critical commentary. It is true that these data may not locate accurately the actual places of spawning known prehistorically. These places must have changed somewhat during the 1200 years of interest anyway. There remains on a general level some correspondence between current spawning areas and those areas known to have been actively occupied during the prehistoric era. Moreover, these are the only data now available that can give any indication of relative differences in habitat suitability for spawn on a species basis, despite the degradation of specific places and of ecological relationships of the recent centuries. The objective is to create a general picture of areal resource variation, and for this purpose the contemporary data seem adequate. For a description of the methods by which the spawn data were gathered, the reader is referred to Goodyear et al (1982). The additional criticism that the resource areas themselves are of unequal size and therefore unequal resource potential is well taken, but again the need for a general picture of areal resource variation overrides this complaint. Seasonal resource scores by locality (sector) may be found in Table 28.

Table 28. Seasonal resource scores by locality

Resource score

<b>Locality</b>	Spring	Summer	<u>Fall</u>	Gillnet
Little Bay de Noc	20	11	. 15	21
Garden Peninsula (west	) 15	10	10	17
Garden Peninsula (east	) 4	2	5	8
Manistique	10	6	11	15
Seul Choix	8	6	10	16
Beaver Islands	17	11	14	21
Sturgeon Bay	13	12	12	20
Straits (west)	11	13	7	13
Little Traverse	11	9	10	17
Drummond Island (south	) 11	9	12	16
Les Cheneaux	14	9	12	19
St. Ignace/St. Martin	15	9	12	17
Bois Blanc	15	9	12	18
Sault Ste. Marie	13	5	9	9
Sugar Island (east)	10	3	3	3
St. Mary's (central)	15	5	8	8
St. Mary's (south)	13	6	9	9
Drummond Island (north)	) 14	10	12	12
Whitefish Point	2	0	3	3
Whitefish Bay (north)	10	6	8	10
Whitefish Bay (south)	13	6	12	15

Sources for specific program operations and input values

Table 29 lists the values that are required as input to each run of the simulation program. In addition, internal alterations in the values of program constants and logical conditions provide the means to structure in greater detail the hypotheses to be examined by the simulation. The following discussion pertains to the sources for input data as well as to specific aspects of the program's operations and the ethnographic and archaeological assumptions that underlie them. The findings of Chapters Two and Three provide some justification for these assumptions.

Table 29. Input data for the simulations

<u>Variable</u>	Values
Number of generations	1-n
Output report frequency	Cycle, period, run
Random number seed choice	1-3
Search radius (fixed option)	
Number of original lodges	Brose 1970, Janzen 1967,
	MSUM site file, Fitting 1979
Locations of original lodges	
	Fitting 1979, MSUM site file
Lodge size	Henry 1969
Population growth rates	Cowgill 1975, Mosimann and
	Martin 1975
Resource failure rates	Lawler 1965
Time of gillnet appearance	Cleland 1982, Lovis and
	Holman 1976
Place of gillnet appearance	Cleland 1982

From this list one can see that the simulator has a great deal of control in structuring the values with which to experiment with various models of settlement change. It is important, as well, to understand that the number of possible combinations of input values is enormous, and that the stochastic elements within the model require repetitive

examination to identify random effects in model behavior. This situation presents the need for a <u>clear</u> and specific research design and the elimination of non-essential combinations of values.

The number of generations may be specified at the beginning of each run. It is assumed that "time zero" means the first century A.D. in the study area. A value of forty generations, for example, allows a time reach of 1200 years for the simulation. An alteration in this value would allow the examination of any specified period of time: for instance a timespan of twenty generations would give output that, barring other changes, would depict the cumulative settlement situation at A.D. 600. Earlier I discussed the time scale plan for the simulation and the fact that time advances generation by generation rather than year by year. The program will not, therefore, tell one how many sites each sector should have. Rather, it will indicate the relative seasonal use level of each sector as though sampled through time.

The output report value specifies what intervals of output are requested. The program keeps track of locations and other features throughout the time of the run, but the actual periodicity of output may be specified. The reports of output can be cumulative, i.e take into account all settlement since the beginning of the run, or cyclical, i.e. take into account the settlement of the particular period only. In this fashion the settlements of a simulated "archaeological phase" may be followed.

There are three random number seeds available to the program. If the same seed is selected run after run, the same random number stream will result. This situation is inadequate for distinguishing between random and non-random results of the program, i.e. the effect of stochastic variables on program output. Using multiple random number streams allows one to investigate whether the perceived results are truely a reflection of program operation or whether they derive from a particular random number stream.

The search radius variable may operate in two ways. the program takes control of this value, the search radius is calculated as the largest distance necessary to allow every lodge to interact with at least one additional lodge. It is recalculated every cycle as lodge movement occurs. The program-controlled search radius value is designed to emulate the real world search for mates, kin, information and resources. It is dependent on the total number of lodges within the simulation, and on their spatial spread. Consequently at the start of the simulation, assuming a situation in which population densities are low and lodges widely dispersed, the search radius is large. As the area fills with lodges, the radius of search will contract if they are evenly dispersed and expand if they are aggregated. If all lodges somehow attempt to occupy the same cell, the program responds by recalculating the search radius to equal the distance to the closest cell space boundary. As a fixed value or constant, the search radius will remain unchanged throughout the run of the program. The value specifies that

		Ņ
		]:
		\$:
		a
		,
		Ţ
		Í
		ŧ
		;
		:
		:
		;
		!
		:
		,

part of the cell space that is available to an individual lodge during the locating process. This is seen as a way of simplifying program operation and isolating the effects of change in the radius on settlement distributions.

The number of original lodges may be specified in each run of the program, as long as it is a number larger than (Four lodges is the minimum number required to assign every lodge an agnate lodge and an affine lodge, barring the possibility that one is kin to oneself or that one's agnates and affines are the same lodge.) One value might be the number of statistically contemporary lodges present in the study area at A.D 100. On the basis of archaeological data, researchers have estimated lodge numbers at several relevant sites (Brose 1968, Janzen 1967, MacPherron 1967). Judging from archaeological deposits that are statistically contemporary, the study area may have contained as many as fourteen lodges at A.D.100; at least this value may stand as a order of magnitude estimate. Total population of the area, assuming that lodges averaged eight persons, would have been on the order of 110 persons.

The location of the lodges at the start of the run is a variable value. One reasonable way to provide these data is to use the actual (archaeological) locations as original locations. These locations are read into the program as Cartesian coordinates. Alternatively, the origin of settlement could be depicted as a front from a cardinal direction, or as a cluster of lodges in one central location, thus allowing a number of possible origins for

occupation of the study area. Zimmerman reported that the origin points of lodges left an imprint on the final outcomes of his GLENWOOD I simulation (Zimmerman (1977); it is expected that this set of values is influential in the present case.

The lodge population mean and variance provide the numerical base upon which to calculate total population and to generate new lodges as a reflection of growing population. Alexander Henry's experience with the lodge of Wawatam is exemplary in this regard (Henry 1969). members of Wawatam's lodge were the following: the headman, two adult sons, one visiting Englishman, the headman's wife, the son's wife, the son's wife's adolescent daughter, and the son's wife's infant. A total of eight persons over three generations made up this family, with a work force of six adults. It is assumed that this work force could provide all necessary fishing-related labor. I have assumed that the Wawatam lodge represented a central tendency in lodge size, and in age/sex composition for the prehistoric period. With eight persons representing mean lodge size, a variance of four might account for likely size ranges. This means that a lodge is assumed, for the purposes of this simulation, to have been composed of between four and twelve persons. Other researchers have suggested comparable lodge sizes for the area and time under discussion. Brose, for example, estimated that thirty persons representing two extended familes and two nuclear families occupied the Summer Island site (1970:165). Janzen reckoned that four

families or lodges with a total of 15 persons lived at the Naomikong Point site, but suggested that his estimate was perhaps exceeded by an order of magnitude (1967:91).

Estimates for the Juntunen site co-resident population range between 25-50 persons (McPherron 1967:288). Given these estimates and the Henry data, the choice of lodge size as uniformly distributed over the range of four to twelve seemed reasonable. The program will, however, accept other mean and variance values for this particular variable set.

The growth rate and post-gillnet growth rate can be supplied per run, with a range of values derived from worldwide ethnographic sources. Mosimann and Martin (1975), Cowgill (1975), and others have provided long-term population growth rates for hunter/gatherer adaptations. Cowgill suggests a mean rate of between one and three persons per thousand population per year, or a rate of from .001 to .003 for the mid-Holocene. It is interesting to note that at a population growth rate of .003 assuming an initial population of 150 persons and a starting date of A.D. 100, the population of the study area will double by A.D. 370. At the lower rate it will double by A.D. 820. These are very conservative estimates of growth when compared to those selected by Mosimann and Martin, who used figures that were allowed to vary between .65 and 3.5 per cent per year in simulations of Paleoindian growth rates (Mosimann and Martin 1975). Cowgill suggests the range of 4-7 per thousand per year as a rarely-observed leap in growth rates associated for a short time with some very

productive areas. These figures will provide a base for run-specific population growth rates for the post-gillnet era.

The time and the place of the gillnet appearance may be controlled by specifying cycle number and cell number in which the technology appears. On its appearance all current occupants of the cell obtain it. If the cell is empty, those occupants of the closest occupied cell receive the net technology. Those in possession of the net may then use the gillnet resource score as the basis for finding a fall location. By using variable locational input, the diffusion of the technology from some cardinal direction could be simulated. It is tempting in this regard to take Ojibwa mythology literally and specify the Straits of Mackinac as the origin point for the net technology! The timing of the gillnet's appearance may be varied from run to run, or it may fail to appear.

Resource failure rates are order of magnitude estimates derived both from the ethnographic literature and the literature describing the frequency of key resource failure in response to stochastic environmental flux. The resource failure rate can change as some specified percentage of the total population of the area possesses the gillnet. It is absolutely true that these figures are not clearly known. But a glance at the ethnographic and historical literature makes it clear that equipment failure, human error, fish behavior, and adverse weather combined to cause fishing failures. In this simulation, the failure rates are

expressed as probabilities that describe the likelihood that a resource failure will occur. I used Lawler's excellent article (1965) to estimate the frequency of whitefish year class failure as the basis for a resource failure rate. frequency for Lake Erie, according to his data, appears to be above one year in ten and closer to one year in five. Though specific to Lake Erie (an area marginal to successful whitefish adaptation) the Lawler rate may estimate the frequency of such events in the northern lakes. The Jesuit Relations suggest resource failures, due to one or another causes, at Huronia in 1650, at Keweenaw in 1661, at Sault Sainte Marie in 1665-6, at St. Esprit in 1667, at Mississagi and Nipissing in 1670, and at Menominee in 1673. Whatever the case in the seventeenth century with its special culturally-induced circumstances, the resource failure event surely occurred and the resource failure rate provides some estimate of its frequency. The rate may change after a threshold percentage of lodges within the simulation possess the gillnet.

### Choosing a location cell

The choices of location cell by each lodge are primarily organized by the current search radius, which dictates those cells potentially available for occupation. The available cells are ranked by resource score for the current season and each lodge in sequence occupies the highest-scoring cell. This arrangement presumes that the lodges have perfect knowledge about the distribution of

resources close to their current locations. In the event of resource score ties between two or more cells within the search radius, a tie-breaking rule is envoked that favors any cell occupied previously by the subject lodge (each lodge keeps a log of cells occupied previously.) In the event that there are still resource score ties, the lodge chooses that cell closest to its current location. Further ties are solved randomly with equal likelihood. But that is not the end of the selection process. Suitable cells must either be empty of lodges, or contain kin lodges in order to be occupied by the subject lodge. Cells that contain lodges recognized neither as agnates nor as affines may not be occupied. In this case the lodge must select the next most suitable cell from the ranked list. Lodges that identify one another as agnates may always co-reside, and affine lodges may co-reside or not based on the outcome of a random number draw and some specified probability. After the gillnet appearance, this probability factor may change and co-residence with affines may increase or decrease in likelihood. The reader will recall that agnate and affine lodges are assigned randomly at the start of the simulation, and that through time a lodge may have more than one agnate lodge and more than one affine lodge. While co-residing with kin of any category, lodges may exchange information about previous locations occupied and may "infect" their kin with the gillnet technology. Obviously the exchange of information about prior occupations will influence future selections of cells to be occupied, since prior occupation

			:
			:
		•	;
			:
			:
			:

is the key to tie-breaking rules.

In the event that a "good year for resources" occurs, the population will be incremented by the current population growth rate, divided into lodges of sizes uniformly distributed between four and twelve persons and assigned locations to "live" in areas that are the most heavily populated at the current time. When the next cycle begins, each lodge's current location will be that cell which it occupied in the prior fall.

These rules serve to mimic the real world arrangement of prehistoric people both in regard to the distribution of resources and the distribution of relatives and others. The tie-breaking rules serve to correct the resource-dependent view of settlement distributions that has occupied so many archaeologists, and replace it with a view of settlement processes that is at least partially reflective of simple social rules and demographic facts.

### Program verification/validation procedures

An essential task of the simulation process is the assurance that the model constructed for the computer matches the conceptual model's design, and operates in the same manner as the conceptual model. This process is termed program verification and validation. There are several important yet different aspects of these stages in model development, for different processes of program operation require different checkout procedures. First, the program must be in a condition to run properly according to the

rules of the language in which it is written. Secondly the program must exhibit the same structural relationships among model elements as does the conceptual model. This procedure involves the painstaking comparison of manually generated outputs with machine results and subsequent corrections to the program and to the logic of the conceptual model. Finally, the randomness of some operations must be verified. In the current model, a procedure checked whether the distribution of random number values mimicked a uniform distribution by means of the chi-square test for homogeneity of distribution. Chi-square values for homogeneity of distribution tests were far below critical values for all of the random number seeds used by the program, particularly as the sample size of random numbers increased.

## Research design

Three models of settlement growth and distribution will be examined by the simulation approach. To as great a degree as possible, they correspond in structure, assumptions and input values to those three models described in Chapter One. These models I have called the ZPG model, the STRESS model, and the GILLNET model. The ZPG model is designed to emulate what might be termed classic hunter-gatherer-fisher arrangements, encompassing low population growth (expressed as zero growth herein), flexible co-residence possibilities (i.e. equal probabilities of residing with agnatic and affinal kin), zero resource failure rates, and technological diversity.

This model assumes the acquisition of the gillnet at time zero, or A.D. 100, from an unknown source east of the study area. The locations of the original lodges and their numbers derive from actual site locations as documented by archaeological research within the study area. In this model the relative attractiveness scores of the location cells do not change in response to the acquisition of the gillnet because the model presumes that diversity of conditions is an essential feature of hunter-gatherer-fisher subsistence regimes.

The STRESS model adopts variable values and operating conditions that depict the ideas of Lovis and Holman (1976). This model includes high population growth rates, high resource failure rates that emulate attainment of carrying capacity, the appearance of the gillnet following A.D. 400, and a shift in residence from an agnatic emphasis to equal likelihood of affinal co-residence. In this model, the relative attractiveness scores of the location cells change in response to the appearance of the gillnet. Other conditions are identical to those in the ZPG model.

The GILLNET model depicts values that derive from the thinking of Cleland (1982). In this model, the gillnet appears about A.D. 650, and the population growth rate increases following the spread of the new technology. Resource failure rates are low prior to the introduction of the gillnet and high thereafter. After the spread of the gillnet, co-residence with affines increases in likelihood. Post-gillnet resource scores are used for fall locations.

Other operating conditions are identical to those active in the other models.

The growth models ran for 40 cycles of time using each of the three random number seeds in sequence and a constant search radius of 3.0. The ZPG model ran until a stable state was reached (10 cycles). The values for input variables and internal constants are listed in Table 30.

Table 30. Input values for ZPG, STRESS, AND GILLNET models

<u>Variable/Model</u>	ZPG	STRESS	GILLNET
Basal pop. growth rate	.000	.001	.000
Gillnet pop. growth rate	.000	.003	.001
Basal failure rate	.000	.300	.000
Gillnet failure rate	.000	.300	.300
Cycle for gillnet appearance	1	10	19
Affine co-residence %	100	50	50
Gillnet affine co-residence %	100	100	100
Gillnet resource score	nil	yes	yes
Search radius	3.0	3.0	3.0
N of original lodges	14	14	14
Location of original lodges	real	real	real
Lodge size mean, variance	8,4	8,4	8,4
Gillnet appearance cell	10	10	10

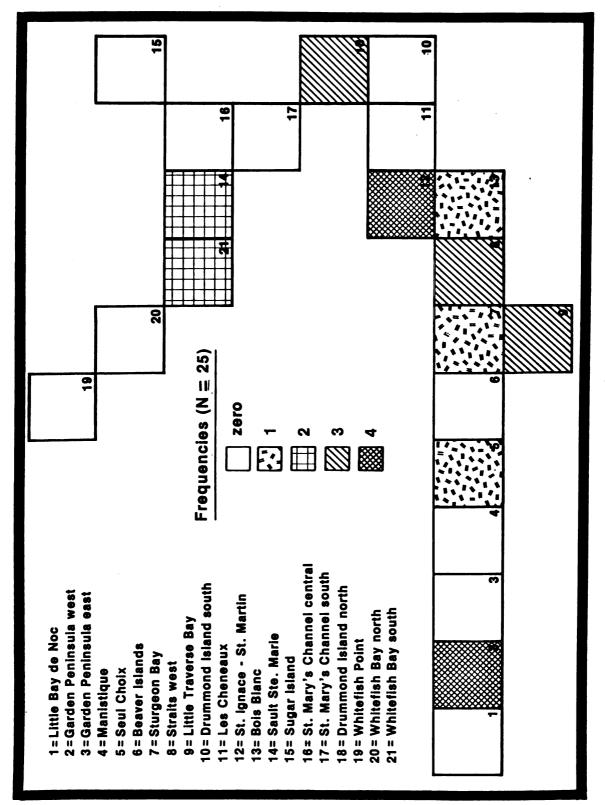
### Outcomes of the simulations

Outcomes of the simulations, in the form of artificial site distributions and related measures of aggregation and run performance, must be compared both with one another and with the real world distributions of archaeological sites in

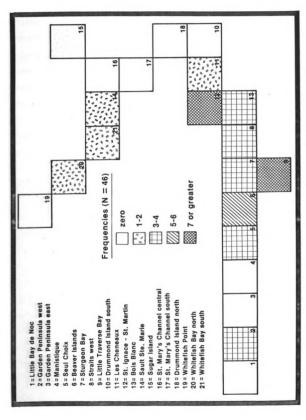
order to conclude whether any or all of the three models reveal insights about prehistoric settlement processes. There are several obvious problems with making these comparisons. First, each set is autocorrelated through time and space. This situation makes mathematical comparisons somewhat difficult. Secondly, the real pattern suffers from distortions created by site destruction, site multicomponencies and differential research. It is safe to say that the comparisons among simulated and real site distributions are plagued by difficulties that transcend the current study. However, formal conparisons can be made among the simulated patterns by use of the variance to mean ratio, or coefficient of dispersion (Sokal and Rohlf 1973:73). The ratio compares the spatial (or other) dispersion of frequencies across a cell space with a theoretical Poisson or random dispersal. Values of the ratio near 1 fit the random dispersal while those above and below 1 are respectively aggregated or uniformly dispersed. The ratio is scale dependent; thus the direct comparison of real world site distributional patterns with those of the simulations, which are lodge distributional patterns, may be questionable. For the same reason direct comparisons among models may be erroneous. However, the ratio may be used to compare run to run differences within the same models, and to compare models when absolute numbers of lodges are roughly equal. The ratio has been calculated for the actual site distributional pattern so that some sense of how this set appears in dispersal characteristics may be gathered.

Predictably the ratio for the data set was clustered, with a value of 4.9 (n = 84). Figures 14 through 16 demonstrate the baseline distributional characteristics of the Middle Woodland, Late Woodland and total sample spatial arrangements, adjusted to a standard grid. From these diagrams one can observe that in the Middle Woodland time period some clustering occurs at the Straits of Mackinac and at Sault Sainte Marie, while in the Late Woodland the clustering at the Straits is pronounced.

Researchers have suggested a number of techniques for analysis of simular results, including the use of visual inspection of patterning (Chadwick 1978). Although there is no mathematical basis for this practice, this is apparently the way in which many simulated patterns are evaluated. Chadwick also suggested the use of chi-square analysis for pattern comparisons; however, in the current study the problem of autocorrelation ruled out the use of techniques that assume statistical independence. Chadwick's last technique involved the spatial analysis of non-zero residuals from simulated distributions. This technique compares the areas of non-correspondence between real and predicted locations and mathematically analyzes their spatial properties to detect systematic (autocorrelated) biasses. In this analysis, I made informal comparisons between areas in which the simulations had overpredicted occupancy and areas in which occupancy was underestimated, and attempted to account for the erroneous predictions.



Component distribution, Middle Woodland period Figure 14.



Component distribution, Late Woodland period Figure 15.

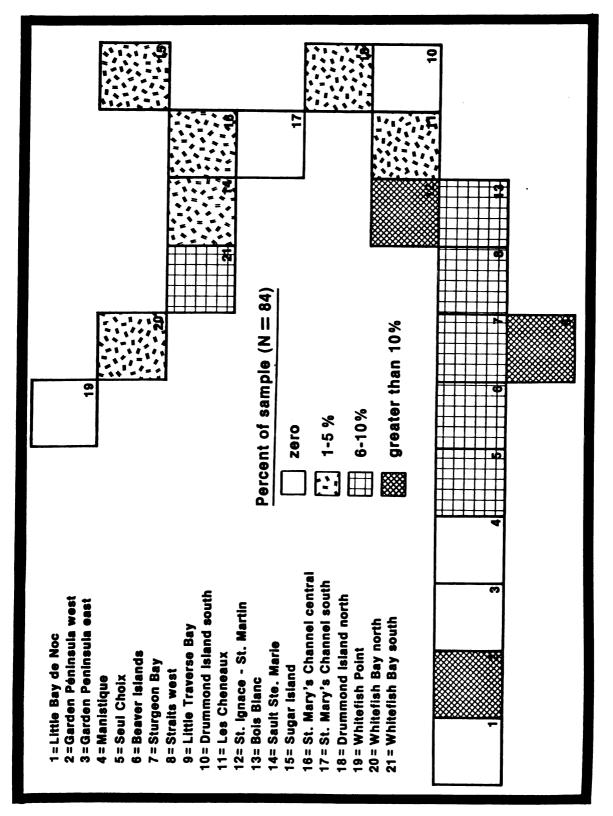


Figure 16. Component distribution, all periods

	II.
	_
	\$. •
	İ
	:
	•
	<u>:</u>
	;
	1

#### ZPG Model results

The ZPG Model ran for ten cycles at which point no further changes occurred and analysis ceased. The three iterations were quite variable in terms of aggregation tendencies, which given uniform operating conditions otherwise, is directly attributable to the effects of the random number seeds. Table 31 depicts some different characteristics of the three outcomes. In the table, mean tries refers to how many times a lodge attempted to enter a potential cell. Mean distance refers to how far (i.e. across how many cells) a lodge moved per season.

The use of RNS #3 resulted in very clustered large groups who tended to occupy two particular areas throughout all seasons. This tendency is visible in the small mean tries value (Table 31). Little Bay de Noc and the Beaver Islands/Sturgeon Bay area supported the "family territory" occupations (Figure 17). Lodges dubbed "family members" always co-resided. There were areas with no occupation that separated the "family territories". All families occupied territory exclusively; that is, families never exchanged member lodges as the seasons changed. Family #1 never obtained the gillnet as it never shared an occupation area with other lodges. Figure 18 depicts the final occupation distributions after pattern stabilization. This pattern depicts a number of interesting characteristics. First, Family #1 occupies an area that, according to resource scores, represents a uniformly favorable place over all seasonal changes. It is one of those places that is always

Table 31. ZPG model results

	RNS #1	RNS #2	RNS #3
Var/mean ratio			
Cycle 1	2.50	4.00	2.50
Cycle 3	3.90	3.70	7.60
Cycle 8	4.05	5.10	12.50
Cycle 10	4.05	6.50	13.50
Mean tries made	1.69	1.50	1.05
Mean dist. moved	.69	1.07	1.00
Perc. with gillnet	78.00	78.00	85.00
Population	90	111	114
N of lodges	14	14	14
N groups	6	5	2
Group size range			
Spring	1 – 4	1-7	3-11
Summer	1 – 4	1 – 8	1-10
Fall	2-3	1 – 7	1-10

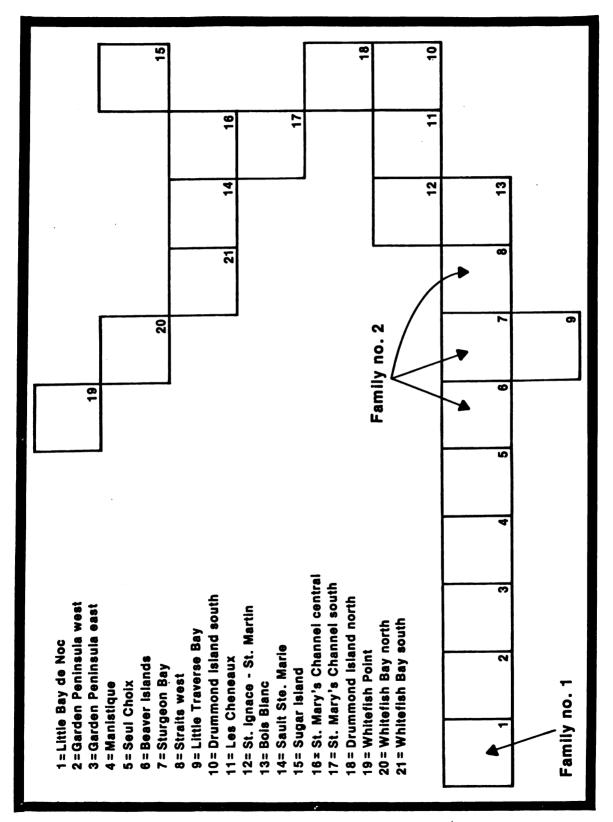


Figure 17. Lodge aggregates, ZPG model/RNS #3

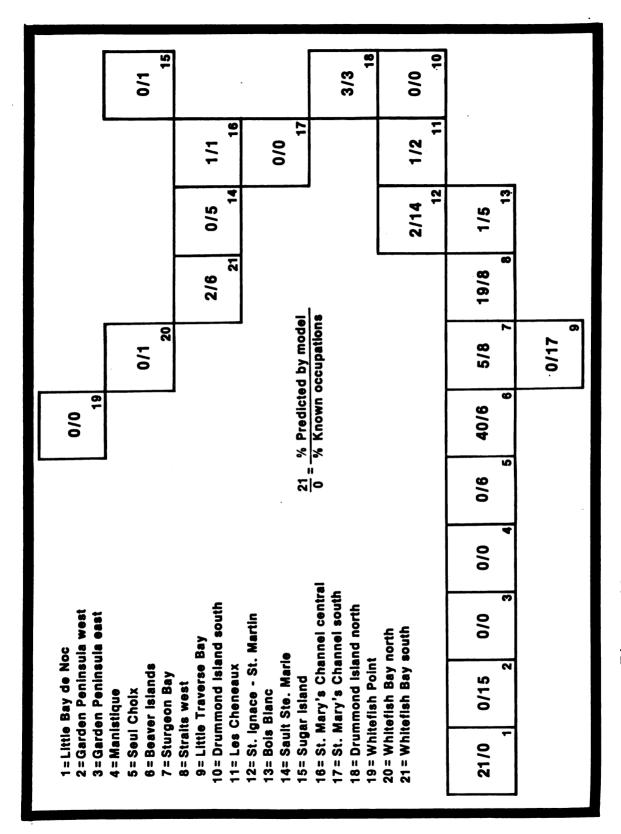


Figure 18. Lodge distributions, 2PG model/RNS #3

that is always highly suitable for occupation. In contrast, Family #2 occupies several adjacent areas that differ markedly in seasonal resource scores. As a result this large family moves east and west as the seasons change, with spring/fall occupations in cell 6 and summer occupations in cell 8. In the real system these areas correspond to the Beaver Island sector and the Straits west sector respectively. The interesting thing here is that two separate patterns of areal use are present at the same time given the input conditions and resource distributions.

The pattern produced by this analysis deviated from the actual distribution of sites (Figure 18). The model underpredicted occupancy levels for the Whitefish Bay/Sault Sainte Marie area, the Saint Ignace area area, the Garden Peninsula, and Little Traverse. It overpredicted the occupancy levels of Little Bay de Noc, Beaver Islands, and the Straits west. In other areas, the model predicted the occupancy levels "accurately", or within +/- 3 percentage points of actual levels.

The analysis run with random number seed #2 was quite different. Family groups were more varied in size than in the other runs. Co-use occurred among linked lodges in cell 8 with the largest aggregates in the summer. Like the run with RNS #3, the lodges in cell #1 were isolated and stationary while those in the Straits and adjacent cells moved in response to differential resource availability. The Straits-area lodges formed the largest "family". Family 5 was small, isolated and used a spring territory and a

ş
:
-
3
Ÿ.
3
<u>:</u>
•
ï
:
:
:
Ţ
:
3
\$
;
:
;
:

summer/fall territory (Figure 19). Families 1 and 5 never obtained the gillnet. Overall lodge distributions are depicted in Figure 20. Off-predictions for RNS #2 point to Little Bay de Noc and the area west of the Straits, with the most erroneous underpredictions occurring at Sault Sainte Marie, Little Traverse Bay, the Garden Peninsula and Seul Choix.

The activity of the model with RNS #1 is most interesting (Figure 21). Families 2, 3, and 4 co-use cells 7, 8 and 11; families exchange member lodges and co-reside with different combinations of lodges every season. Spring and summer are the seasons in which the largest groups gather. Family 1 is sedentary while the other families move at least once a cycle. The overall distribution of lodges is displayed in Figure 22. Predictions of actual percentages were off in familiar places; the model overpredicted the uses of Little Bay de Noc, Beaver Island, Les Cheneaux, and Saint Ignace. Underpredicted areas included Garden Peninsula west, Seul Choix, Little Traverse Bay, and Sault Sainte Marie/Whitefish Bay. Families 1 and 6 never obtained the gillnet.

# General conclusions for the ZPG Model

It is apparent from looking at the results of these runs that random effects contribute much to the aggregation tendencies exhibited by the lodges. But there were some consistent features of the ZPG model outputs that bear study. First, it is interesting to note that several

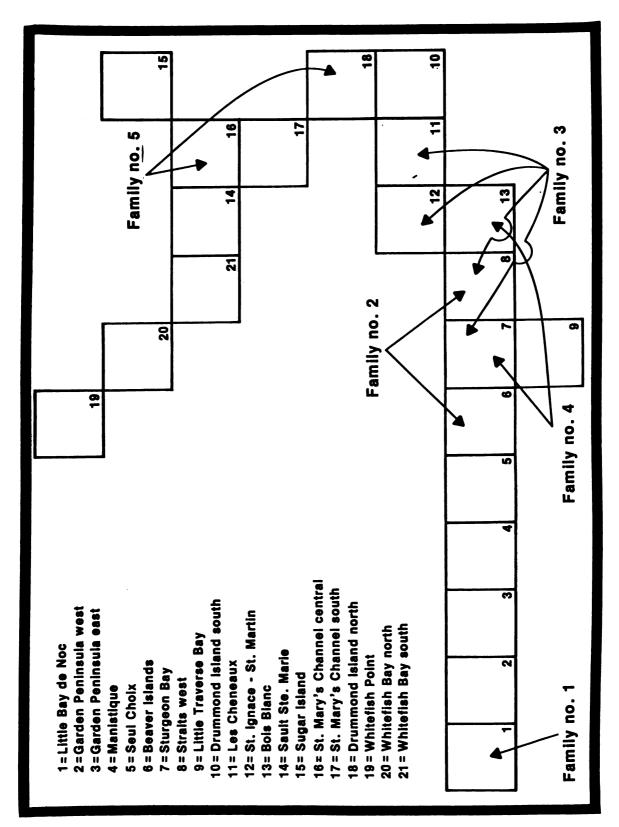


Figure 19. Lodge aggregates, ZPG model/RNS #2

0/1 0/1 0/0 1=Little Stay de Nod 2±Garden Peninaula west 3=Garden Peninaula east 4 = Manistique 5 = Seul Cholx 6 = Beaver Islands

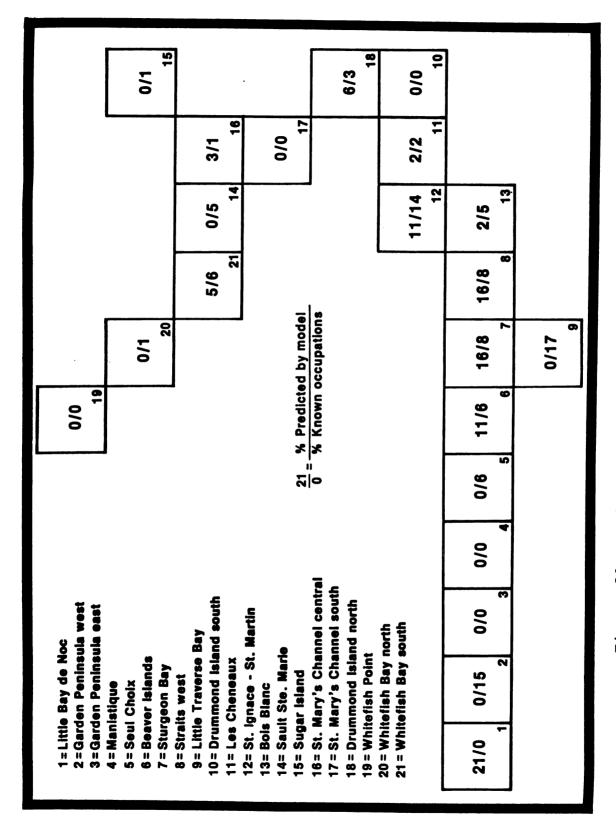


Figure 20. Lodge distributions, 2PG model/RNS #2

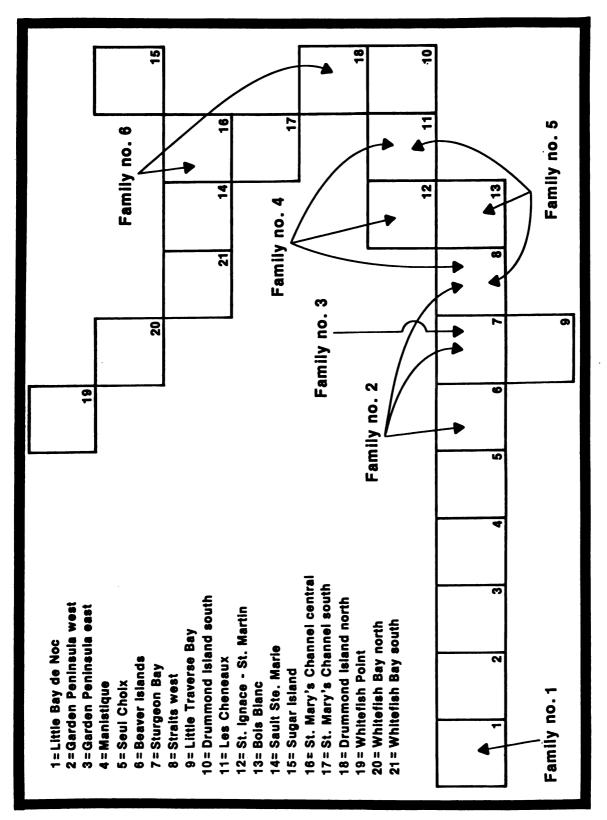


Figure 21. Lodge aggregates, ZPG model/RNS #1

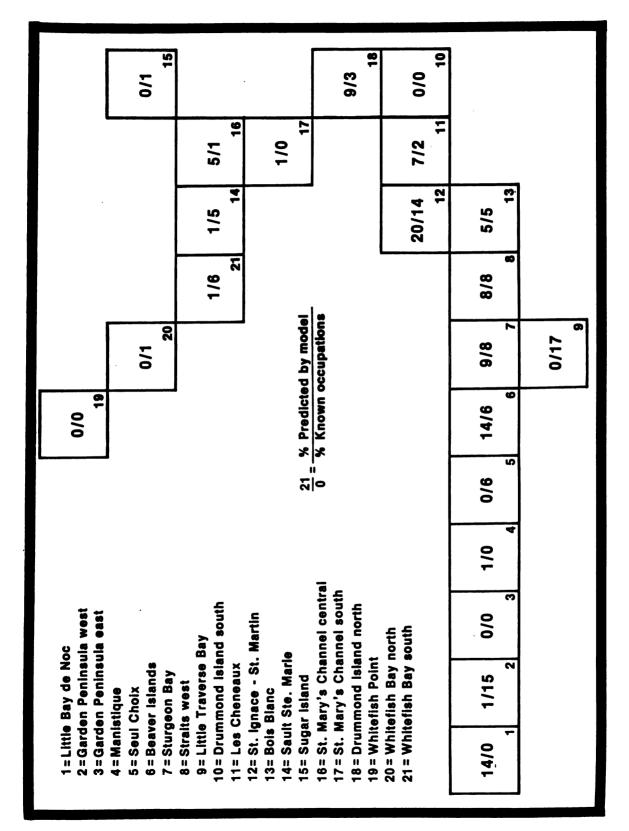


Figure 22. Lodge distributions, ZPG model/RNS #1

seasonal styles of residence can occur at the same time, at least at the spatial scale of the simulation. In the same cycle we see present a no-move situation, a tri-seasonal move pattern, and a bi-seasonal move pattern. Spring and summer occupations, though variable, were the largest aggregates in terms of numbers of lodges. Both of these results relate largely to the characteristics of resource distributions, particularly because the summer fish resources are limited and isolated when compared to those of the other seasons. Places that possess high resource levels thoughout the seasons tend to have stationary occupants. Co-use of areas is most common when families are small and dispersed. Areas of non-use and isolated groups seem to accompany the aggregation process, at least at the scale viewed in the model.

The models consistently over-and underpredicted the same areas in terms of percentage scores for total uses.

Little Traverse Bay, the Whitefish/Sault Sainte Marie area, and the Garden Peninsula were consistently underpredicted, while the Little Bay de Noc, Beaver Islands and Garden Peninsula areas were overpredicted. The model was very precise, however, in its ability to predict areas that have been shown to be archaeologically sterile. The poor areas, in terms of resource distributions, seem to be eliminated quite quickly. Occupation itself, however, is not so consistent; it appears to be subject to random choices after the poor areas are eliminated. This is particularly interesting because productive areas tend to be neighbors

and tend to be the subjects of poor prediction, for instance the Little Bay de Noc-Garden Peninsula west cells. These off-predictions can, to some degree, be explained by differential research intensities, but random choices among equally attractive areas may also be responsible for the apparent patterning, both in the real and simulated worlds.

In every case of ZPG model activity, the lodges abandon the northern portions of the cell space after a few (ca. 5 or 6) cycles of seasonal change. They then concentrate and stabilize near the Straits cells and the Little Bay de Noc cell. The isolated lodges do not obtain the gillnet, whereas the roving and dispersed lodges quickly spread it. In this sense the exchange of the gillnet operates as a "biassed" network, which of course it is, as potential recipients have different likelihoods of obtaining the device (Wyman 1970:173-174).

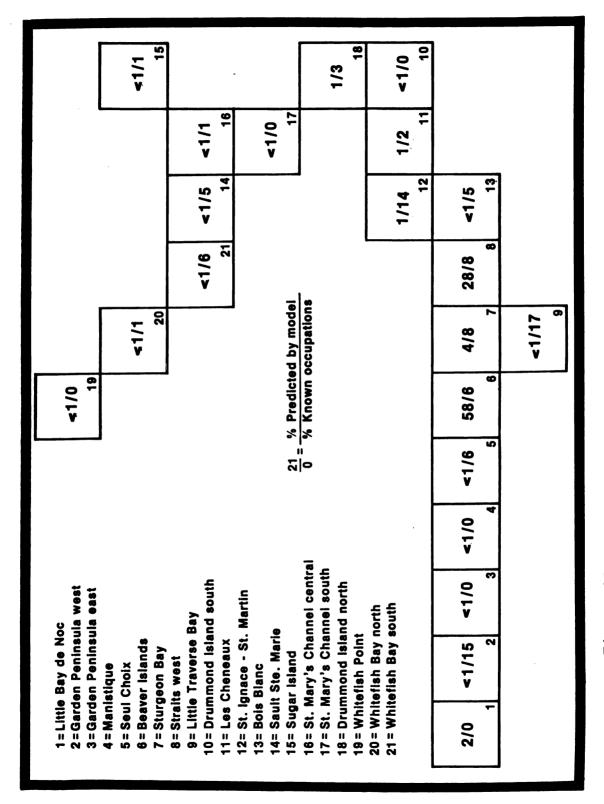
### STRESS Model results

The STRESS model assumed the introduction of the gillnet at cycle 10, rapid population growth, and frequent environmental failure. Table 32 depicts some final values for three iterations of the STRESS model. Variance/mean ratios were not calculated for these iterations as disparate sample sizes and the large size of the grid in relation to the size of the sample would have made such information useless.

The first iteration resulted in extremely aggregated conditions (Figure 23), with nearly 90% of lodges per cycle

Table 32. STRESS model results

	RNS #1	RNS #2	RNS #3
Mean tries	1.04	1.05	1.33
Mean distance	1.28	1.22	1.41
Perc. with gillnet	99.00	98.00	98.00
Beginning pop.	90	111	114
N of lodges	14	14	14
Final pop.	2161	4125	5232
Final n of lodges	293	564	695



Lodge distributions, STRESS model/RNS #1 Figure 23.

using just three cells of the cell space. However, every cell had some minimal occupation. The use of cells other than 6, 7, and 8 occurred both early and late in the 40 cycles of the simulation. The early uses occurred before cycle 5 and represented the initial adjustment of the lodges to the "resource distributions". The latter uses occurred as population growth pushed newly-created lodges to the peripheries of the cell space. The introduction of the gillnet did not cause a visible change in lodge distributions. Virtually all lodges obtained the gillnet, however. By cycle 40, aggregates were largest in spring and summer and slightly dispersed in the fall. Population levels reached 2161, most of whom "resided" in cells 6 and This population concentration appears to be out of scale with known aggregations of occupations. As a predictor of actual site distributions this model iteration operated poorly (Figure 23). Apparently cells 2, 5, 9, 12, 14, and 21 included some aspects of attractiveness which are overlooked in this model configuration.

The second iteration of the STRESS Model essentially duplicated the first, although population levels were higher. By 25 cycles, 90% of lodges were residing in cells 6, 7, and 8. Occupation elsewhere was minimal and occurred both early and late in the simulation. Figure 24 depicts final distributional characteristics of this iteration; it is clear that off-predictions follow a familiar pattern.

Iteration 3 included the largest population, and the distribution of lodges followed that of the other two

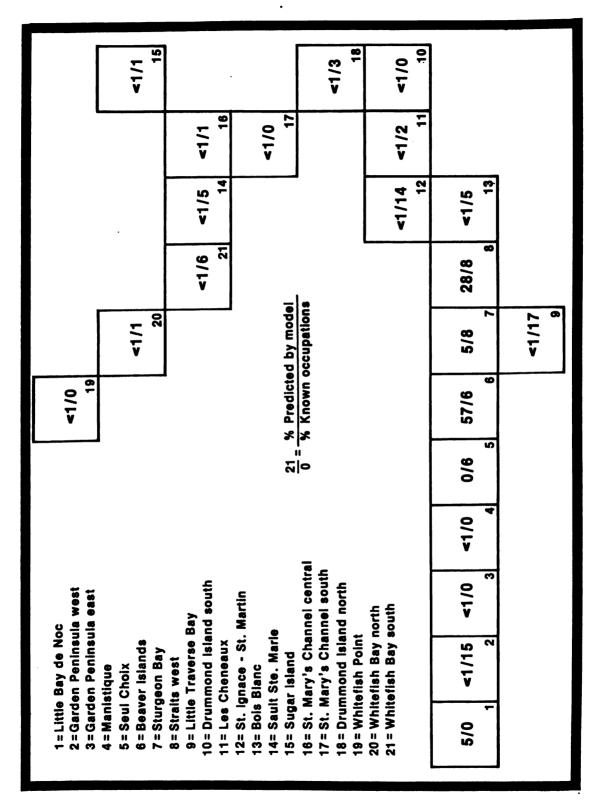


Figure 24. Lodge distributions, STRESS model/RNS #2

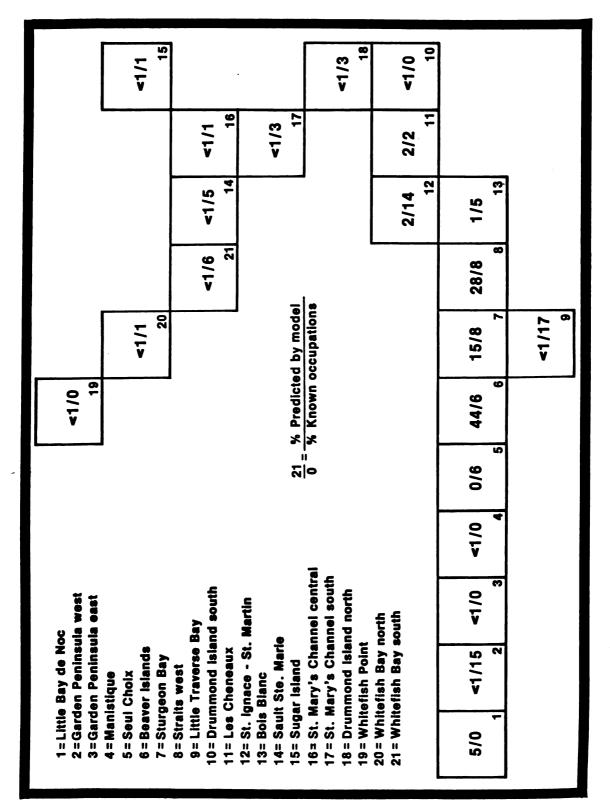
iterations (Figure 25). As seen earlier, overall seasonal aggregates were largest in the spring and summer and relatively dispersed in the fall.

# General conclusions about the STRESS Model

The STRESS Model did not create a familiar site distributional pattern nor did it develop a comprehensible pattern of lodge-territory use. The permanent concentration of thousands of occupants in a small cell space is clearly out of scale with assumed sizes and distributions of populations in the real system. The STRESS Model performs adequately in pointing to areas of light or non-occupation. But many suitable areas remained unoccupied or lightly occupied in favor of cells 6, 7, and 8. If the distribution of resources alone were responsible for the distribution of archaeological components, then perhaps real-world distributional characteristics would resemble those of the STRESS Model. But at the high levels of population assumed by these iterations, the operation of the model is unable to disperse aggregates of lodges.

## GILLNET Model results

The GILLNET Model assumed the introduction of the gillnet at cycle 19, low population growth, and a variable resource failure rate. Three iterations of the model ran, each for 40 cycles. Characteristics of these three iterations are presented in Table 33. The introduction of the gillnet reduced the mean distance value considerably, probably due to the relaxation of conditions related to



Lodge distributions, STRESS model/RNS #3 Figure 25.

Table 33. GILLNET model results

	RNS #1	RNS #2	RNS #3
Mean tries	1.35	1.95	1.64
Mean dist.	.76	.89	1.37
Perc. with gillnet	996.00	100.00	100.00
Beginning pop.	90	111	114
N of lodges	14	14	14
Final pop.	179	261	247
Final n of lodges	26	34	32
N groups	3	6	4
Group size range			
Spring	1-23	1 – 1 4	2-26
Summer	1-12	1-25	1-27
Fall	1-21	1-25	2-21

co-residence. This reduction quite clearly accompanies the aggregation process. During the first iteration of the model, occupation at some level occurred in 18 ot the 21 cells available. All of the occupation of the areas that correspond to Sault Sainte Marie and the Saint Mary's Channel occurred early in the simulation. The last occupation of the Saint Mary's Channel occurred at Cycle 19; after the gillnet introduction occurred these areas of the cell space were abandoned and were never reoccupied.

Figure 26 displays the lodge aggregates at Cycle 40 of the first iteration. In fact this use pattern established itself at Cycle 21 and remained unaltered until the simulation ceased. Families 2 and 3 exchanged members and shared territory during the summer and fall seasons. The individual lodges within these aggregates moved one, two or three times per cycle. The largest aggregates occurred during the spring and fall, and the most dispersed conditions occurred during the summer season. Each season saw a different combination of lodges in all occupied cells with the exception of cell 1. Family #1 was isolated and stationary in cell 1 and never obtained the gillnet as it had no interactions with families 2 and 3. The spread of the gillnet occurred rapidly; by three cycles following its appearance ca. 85% of lodges had obtained it.

Figure 27 displays the cumulative distribution of lodges for the GILLNET model's first iteration. Clustering occurred at cells 6, 7, and 8 but to a lesser degree than in the STRESS Model. All occupation of the northern areas

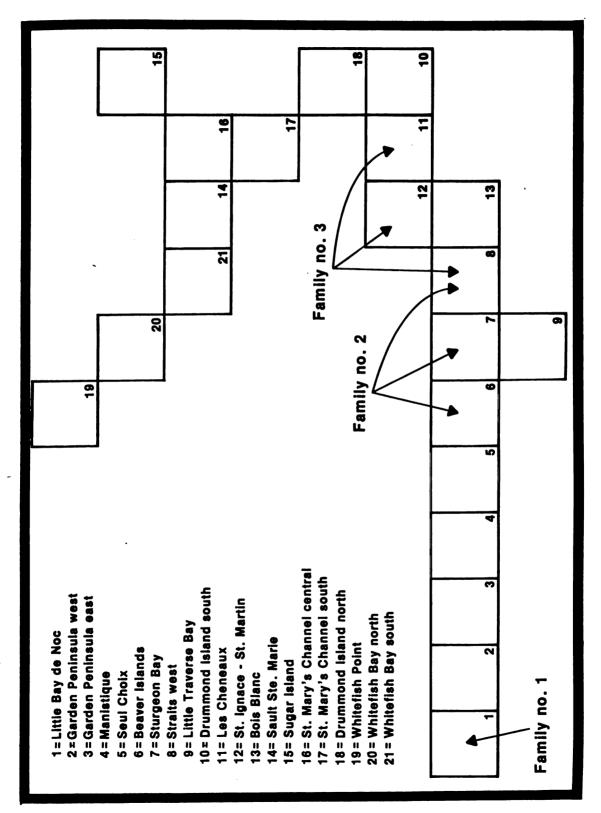
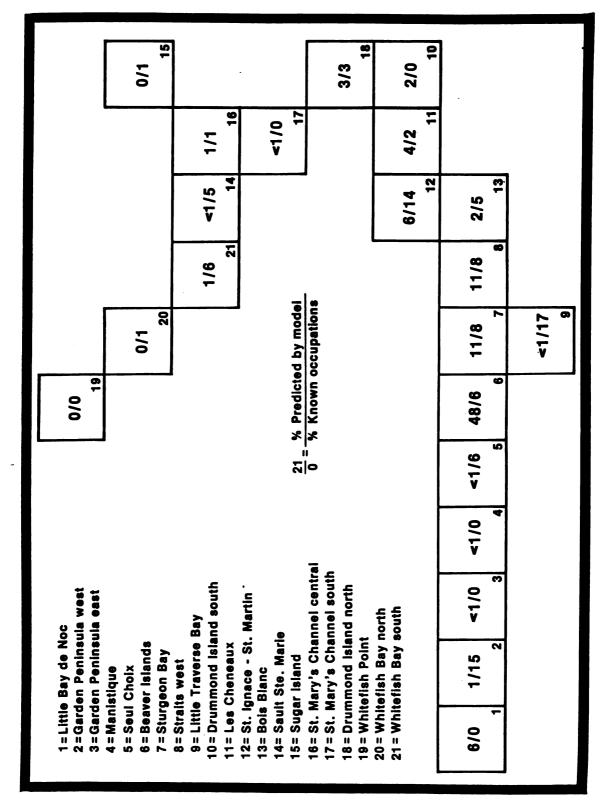


Figure 26. Lodge aggregates, GILLNET model/RNS #1



Lodge distributions, GILLNET model/RNS #1 Figure 27.

ceased following the introduction of the gillnet. The model underpredicted the use levels of cells 2, 9, 14, and 21; overpredictions occurred at cells 1 and 6.

The second iteration of the GILLNET Model resulted in a different pattern of lodge aggregation, interaction and distribution. This model included a higher population level and a greater number of lodges. Co-resident groups varied greatly in size with the largest aggregates occurring in summer and fall. There was great variation in lodge mobility. Some lodges were stationary while others moved frequently and shared cells with a combination of other lodges (Figure 28). The occupation of the northern parts of the cell space ceased <u>prior</u> to the introduction of the gillnet, though light active occupation continued in the lower Saint Mary's Channel to the close of the simulation. All lodges obtained the gillnet, as this iteration exhibited more lodge interaction than the previous one.

Figure 29 displays cumulative lodge distributions.

Though familiar off-predictions occcurred, they seemed to occur to a lesser degree than with earlier iterations of growth models. The largest error producers were higher-than-real predictions for cells 1, 6, and 7 and lower-than-real predictions for cells 2, 9, 12, 14, and 21.

This model iteration depicted vigorous seasonal movement and interaction in the Straits of Mackinac-northern Lake

Michigan area to a greater degree than did the other growth model iterations.

The third iteration of the GILLNET Model exhibited its

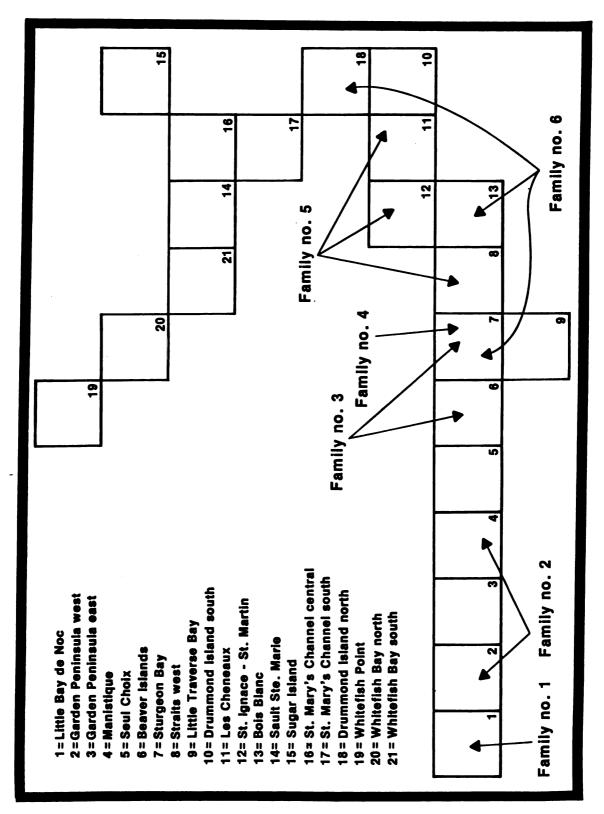


Figure 28. Lodge aggregates, GILLNET model/RNS #2

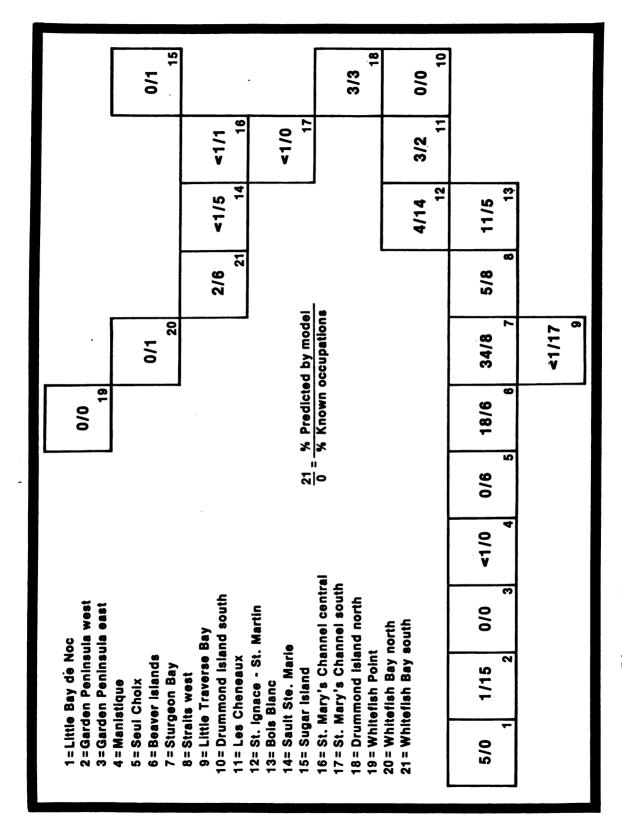


Figure 29. Lodge distributions, GILLNET model/RNS #2

largest aggregates in the spring/summer and relative dispersal of aggregates during the fall. This iteration included an isolated stationary "family" in cell 1 and vigorous interaction among the three "families" at the Straits and northern Lake Michigan shoreline (Figure 30). This tendency is apparent in the large values for mean distance and mean tries (Table 33). Figure 31 shows that the interactions of this iteration tended to even out the relative use levels of adjacent cells in the area of most heavy occupation. As seen in the iterations examined earlier, occupation of the Sault Saint Marie area ceased early in the operation of the model, and the abandonment of the Saint Mary's Channel coincided with the introduction of the gillnet at Cycle 19. However, errors in prediction occurred in familiar places: cells 1-2, cell 9, cell 14 and 21.

## General conclusions for the GILLNET Model

Several interesting conclusions can be extracted from the iterations of the GILLNET Model. First, the early and persistent abandonment of the northern reaches of the cell space occurs in every iteration of this model. After an initial adjustment to resource distributions, this area is never reoccupied. Secondly, the northern Saint Mary's Channel is virtually abandoned as the gillnet introduction occurs. This event seems to draw simular occupations to the south. Family aggregates and territories are variable in size throughout iterations of this model, and the tendency

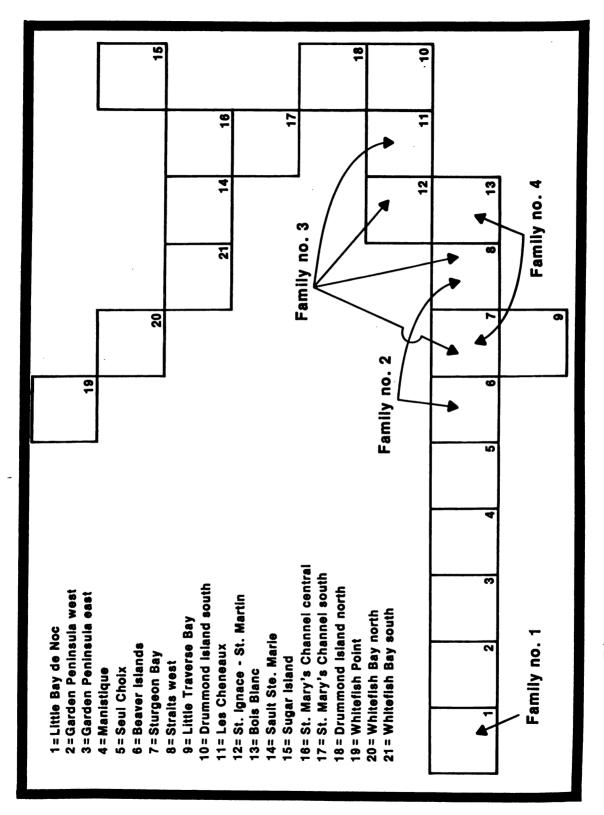
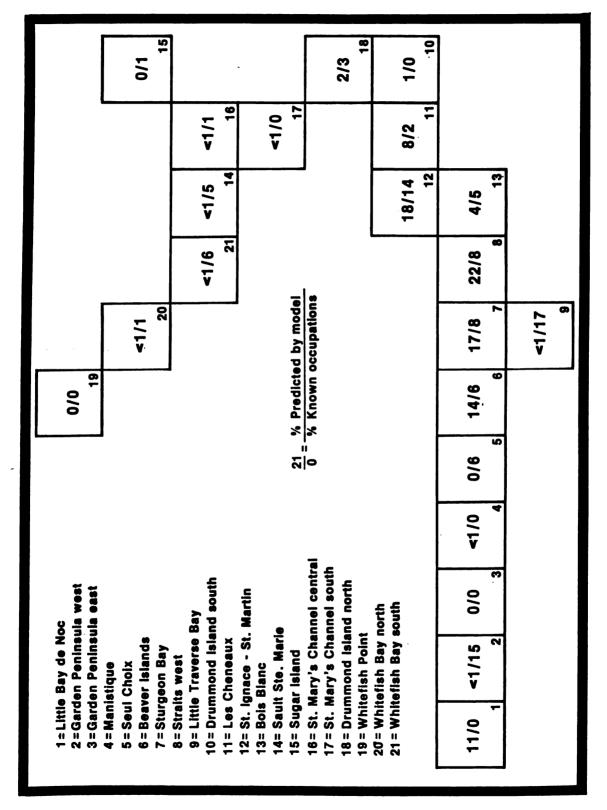


Figure 30. Lodge aggregates, GILLNET model/RNS #3



Lodge distributions, GILLNET model/RNS #3 Figure 31.

cycles of the model iterations. Likewise, seasonal aggregational tendencies are not ordained by resource distributions but appear to be altered by random factors within the simulation. Off-predictions continue to occur but are of less magnitude than seen earlier in the STRESS model. Of the model configurations examined here, this model most clearly shows movement on a large scale in response to the introduction of the gillnet, but fails to depict long-lived use of the northern cells.

#### CHAPTER FIVE

## CONCLUSIONS AND SUGGESTIONS FOR THE FUTURE

### Introduction

This study has re-examined the historical and ethnographic evidence for the development of the fisheries adaptation in the Upper Great Lakes, has analyzed the locational differences among Woodland sites of sequent time periods, and has constructed a simular experiment to investigate the consequences of technological change and population growth on the distribution of archaeological locations. Two standing models of system change were examined, and a novel model that derived from the ethnographic, historical and locational evidence was advanced. General support was given to the ZPG and GILLNET models because they corresponded both in scale and in distributional characteristics to the known pattern of site placement.

The general operation of the simular models derived from an ethnographic and historical data base presented in Chapter Two, and in the case of the ZPG model these data provided justification for specific model assumptions. A re-analysis of historical and ethnographic data suggested that the multi-generation household supplied enough laborers to conduct all chores related to fishing with the exception of the large permanent weir technology. Large permanent weirs appeared to be a technology inconsistent with current understandings of prehistoric group organization and

resource-related activities. There was support for the presence of all other historically documented methods of fish extraction in the prehistoric period. Some methods were versatile and could be used in a number of settings and against a number of species, while others were specific to a particular set of fish characteristics and/or physical settings. The dip net technology was likely more widespread in the prehistoric period than the archaeological evidence would suggest. Sampling error from a number of sources contributed to the apparent patterning of the archaeological record, especially due to the incomplete preservation of fish remains and fragile organic elements of most technological assemblages.

Chapter Three suggested that Middle and Late Woodland site locational differences were very subtle. The differences were most pronounced among locations that were never co-occupied by people of both time periods, and related mostly to water depth characteristics, with Late Woodland sites occupying places at which deeper water occurred close to shore. This was particularly true of sites in the northern lower Michigan area, and it was unclear whether these differences related solely to technological change or to contrasting spatial distributions of sites. The components did not differ drastically on the variables that measured relationships with offshore features such as shoals, reefs and islands. Overall more contrasts were related to geographical position than to temporal period, and in all cases a high degree of overlap existed

among the components of the two time periods. The Middle Woodland sites were more variable in their locations than were the Late Woodland sites. The Late Woodland locations were markedly general in physical characteristics; that is, no particular set of physical characteristics appeared to dominate from location to location. Again, these conclusions about site locational characteristics provided a general framework for the design of the simular experiments, as well as to provide specific information critical for the design of the ZPG model.

## Conclusions to the simular experiments

Chapter Four described three contrasting models of Late Woodland intensification and subjected them to simular experimentation. The ZPG model appeared compatible with current understandings of site distributions and created -familiar patterns of site dispersal in space. While the STRESS model results appeared to be out of scale with assumed population levels and aggregation tendencies, this model did depict some aspects of the known pattern of site distributions because northern occupations occurred both early and late in the simulations. Otherwise, the ZPG and the GILLNET models depicted the most appealing scale of population levels and aggregation tendencies. The early abandonment of the northern area prior to the introduction of the gillnet was a consistent feature of all iterations of the latter two models. All of these iterations showed variation in group size, lodge movement frequency and

interaction tendencies, which was an unexpected but realistic aspect of actual system operation. The ZPG and GILLNET models failed to simulate Late Woodland use of the northern parts of the study area.

High population growth spurts appear to be unlikely in the prehistory of the Upper Great Lakes. Mean population growth rates over the Woodland cultural-chronological period in this region surely lay below a rate of one per thousand per year; at least such a rate is consistent with other operational features of the simulations and with some aggregational aspects of the real system. All models allowed more aggregation than is apparent in the known system over the time of its operation. Real system operation was apparently not as dependent upon the distribution of fish resources as the simulation models depict, and must have included some important processes of lodge dispersal. Occupation of space appears to be more random in the real system than in the simular system, despite the fact that all of the choice-making operations of the simular models were no doubt important. The simular system probably provided more constraints on choice than in the real system.

The dependence of the models on a simple distribution of resources resulted in the non-use of many profitable areas. The tendency to alter locations after the introduction of the gillnet was a bit hidden by the fact that the places of greatest advantage for the use of the gillnet tend to be the best places for fall-season use

anyway, particularly in the Straits of Mackinac and the northern Lake Michigan area. But the special character of sectors such as Little Traverse and Sault Sainte Marie was overlooked by the resouce-dependent world view of the simulation. In the case of Sault Sainte Marie the reliability of the fish resources was not modelled by the simulation. For the Traverse area, the attractiveness may have related less to fish than to another local resource, for instance chert. The Traverse area would have been a favored place for deep-water fish exploitation had directional change in the region's environment occurred. Assuming a warming climatic trend, the deep waters of the Traverse sector immediate to the shoreline would have offered a margin of protection to cold-adapted lake fauna and to their prey, as well as to provide possibilities for human exploitation.

The consistent off-predictions of some areas by the models reflect biasses produced by differential research. Those cells most frequently underpredicted do coincide with areas that have received the highest levels of research attention: Garden Peninsula west, Little Traverse, Saint Ignace, and Sault Sainte Marie. The areas of overprediction could well use additional emphasis: Sturgeon Bay, Beaver Islands and adjacent northern shoreline, and Little Bay de Noc. The Little Bay de Noc occupations appeared isolated and stationary compared to other lodge occupations in the simulations. On a hypothetical level this area may, in the real system, have represented a permeable but noticeable

cultural boundary. The cause of this boundary probably related to the nature of resource distributions, since the Little Bay de Noc area is isolated by neighboring resource-poor sectors. Overall the simular results support the assumptions of the ZPG and the GILLNET models as well as to suggest that Fitting's assumptions about intra-regional movement have some basis in fact (Fitting 1979).

The sociological results of the simulations are interesting and informative. Both the ZPG and the GILLNET models suggested that groups of related lodges tended to separate in space, and that at any one time variety in residential styles was possible in a rather restricted area. For instance a stationary style, a tri-seasonal movement style, and a bi-seasonal movement style were contemporary within both models. Groups of related lodges tended to exchange members for some seasons of the year, which was an unexpected but very realistic aspect of lodge behavior. Some family aggregates and territories were variable in size and composition throughout iterations of the GILLNET model. Seasonal aggregational tendencies were not ordained by resource distributions but appeared to alter in response to the distribution of relatives.

The results of the study lend general support to some of the hypotheses advanced at the outset. Hypothesis One, which stated that there were significant differences in the physiographic settings of Middle and Late Woodland sites, was tested and rejected. The analyses concluded that a high degree of overlap existed among the sites when their

geographical characteristics were compared. Both groups of locations were associated with favorable offshore conditions such as shoals and reefs, and sites of both periods tended to share identical locations. Hypothesis Two, that Middle Woodland locations appeared more variable in character than Late Woodland locations, was tested and supported by the results of the locational analyses. Hypothesis Three, that multi-resource locations were typical of Late Woodland site placement choices, was supported though not directly tested by the analyses of Chapter Three. Hypothesis Four, which stated that the simular introduction of the gill net at A.D. 600-700 was consistent with real world settlement distributional patterns, was supported though not tested during the simular experiments. Hypothesis Five, which stated that a simular model assuming low levels of population growth, resource failure, and initial population size was consistent with real world site distributional patterns, was supported though not tested via the simular experiments. The best-of-fit situation between real and artificial site distributional patterns existed when zero or minimal population growth was assumed, as demonstrated by the final results of the ZPG and GILLNET simular models.

In summary, the study has shown that variability in site location and fish capture technologies is characteristic of the Upper Great Lakes adaptation throughout the prehistoric period. The variability existed in the nature and location of the fish prey, in the decisionmaking about settlement location, and in the use of

a technological system. One could characterize the area as one of multiple fisheries, or sets of activities that were alike in the sense that they exploited common prey, but different in the particulars of prey habits, locational preferences and items of technology. Aggregational tendencies among the people of the prehistoric era related more to the pleasures of company than to the acquisition of a critical labor mass. Nucleation appeared to be related to the distribution of resources, relatives, and increasing density of population rather than to labor requirements alone. Some aspects of the adaptive process took a rather counterintuitive turn in this case; site locations did not become more and more alike in response to a specific Rather, efficient utilization of diverse settings resource. seemed to be the hallmark of adapation, as multi-resource locations increased in number through time.

Archaeologically visible aggregation of people may be a simple result of population growth. As the density of persons increased, the effective and necessary social network may have existed in a continuously reducing space. This process, seen archaeologically, may be evidenced by increasing similarities in local style pools in ceramics and in other tool manufacturing styles. Logically, this process in space could eventually result in local perceptions of territoriality that fit the band-localized clan organization of the early historic period in the northern Great Lakes.

## Suggestions for the future

The simular experiments were adequate to review the impact of the gillnet technology on the northern Great Lakes settlement distributional patterns. The limits to the simular results were in this case imposed by the large grain of the resource-related information available. At this particular point the modelling exercises have pointed to the need for fundamental archaeological research on a number of related topics. Information is needed about demographic processes at work throughout northern Great Lakes prehistory. It is necessary to learn more about basic archaeological formation processes in this setting, that is, how to distinguish between small group palimpsests of repeated occupation, and large group single occupations. Future research may show that the large sites of the Late Woodland period are not duplicated by those of the Middle Woodland period, but that the cause of the large sites relates more to consistent year-in year-out occupation rather than to large group occupation. In effect these sites are stratified on a scale for which we have limited perception.

Active survey and excavation programs will complete the site distributional data base. Simular models can help with the organization of these activities but there is no substitute, in any discipline, for basic field data. Specifically, the Little Bay de Noc area, the Les Cheneaux area, and the Drummond Island area need intensive field survey. Despite some coverage in the past, the Beaver

Islands and the adjacent northern Lake Michigan shorelines need additional work. This information may turn current descriptions of site distributions in a new direction.

These field data are the only data with which to test the simular models for logic and accuracy of prediction.

In addition to those data it is important that excavation research be conducted at some of the carefully described locations of historic period weir operation.

These data could reveal much about the social and labor organizational aspects of weir operation, shed light on its viability in the prehistoric period, and improve understanding about the process of acculturation and its interaction with settlement location, social form, and appropriate technology.

A developmental view of the Middle Woodland occupation and use of the northern Great Lakes area should replace the current monolithic view of the Middle Woodland adaptive style. Examination of stratified Middle Woodland locations such as the Portage site (20EM22) can do much to improve the perspective on Middle Woodland cultural development, and will allow the eventual understanding of how diverse Middle Woodland cultural systems made use of the area.

Unlike many other simulations of subsistence or settlement system change, this set of experiments attempted to deal with hunter-gatherer-fisher systems, albeit in a simplified way. The success of such an attempt is in part measured by how accurate the environmental data for the prehistoric period are. In other examples, (Zimmerman 1977,

Chadwick 1978) where agricultural systems were the objects of inquiry, general environmental information such as soil composition and land elevation was available for the locational simulations, and it was possible to recreate areal settlement shifts that provided very close approximations to real world site distributions. Both of these studies operated on a relatively large scale; Zimmerman's for example, operated on a study area that totalled only forty square miles. But the surprisingly close fit of two of the models studied here to the real world distributional pattern suggests that the subtleties of hunter-gatherer-fisher systems are not beyond the reach of simular modelling, despite problems in deriving useful measures of prehistoric fisheries productivity over a large and complex region.

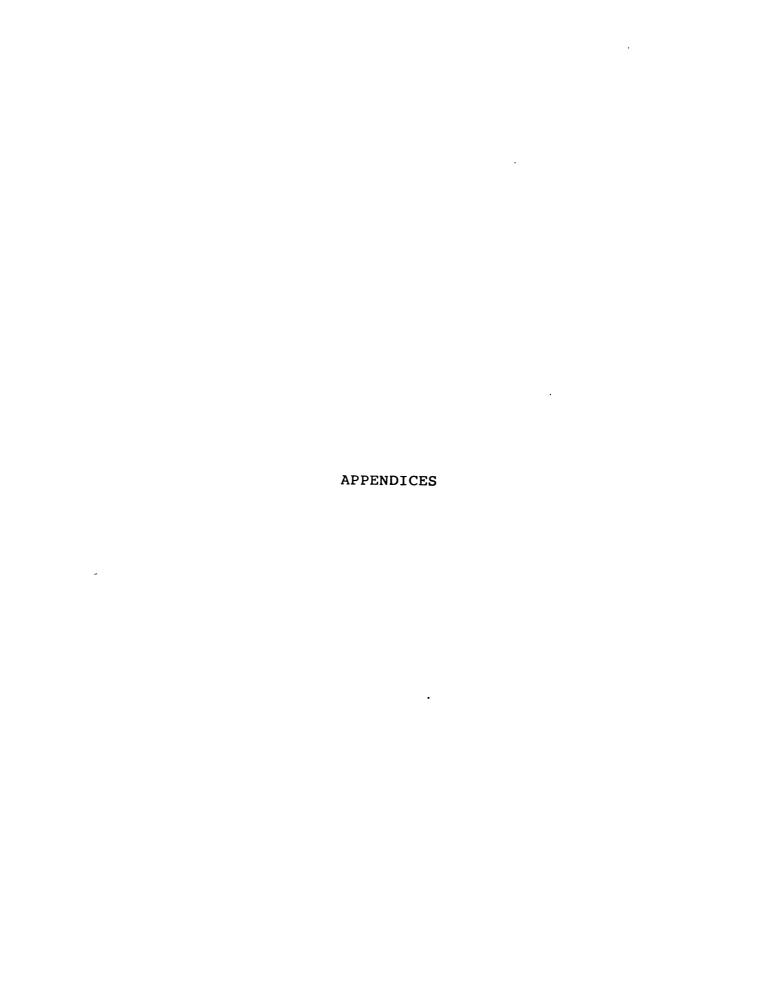
Other simular researchers have discussed the problem of simulation specificity, the situation in which a simulation may examine a very bounded and tightly defined system and have little or no applicability to other complementary systems. This specificity is perceived to be a general fault of much of simular research, because models were either applied to situations beyond their capacities, or were useless in other applications (Aldenderfer 1982:20). The simular model developed as part of this study can be used with other data bases. Most appropriately these data would derive from areas adjacent to that studied here, as the model is general enough to accommodate other Upper Great Lakes locational, ethnographic and technological

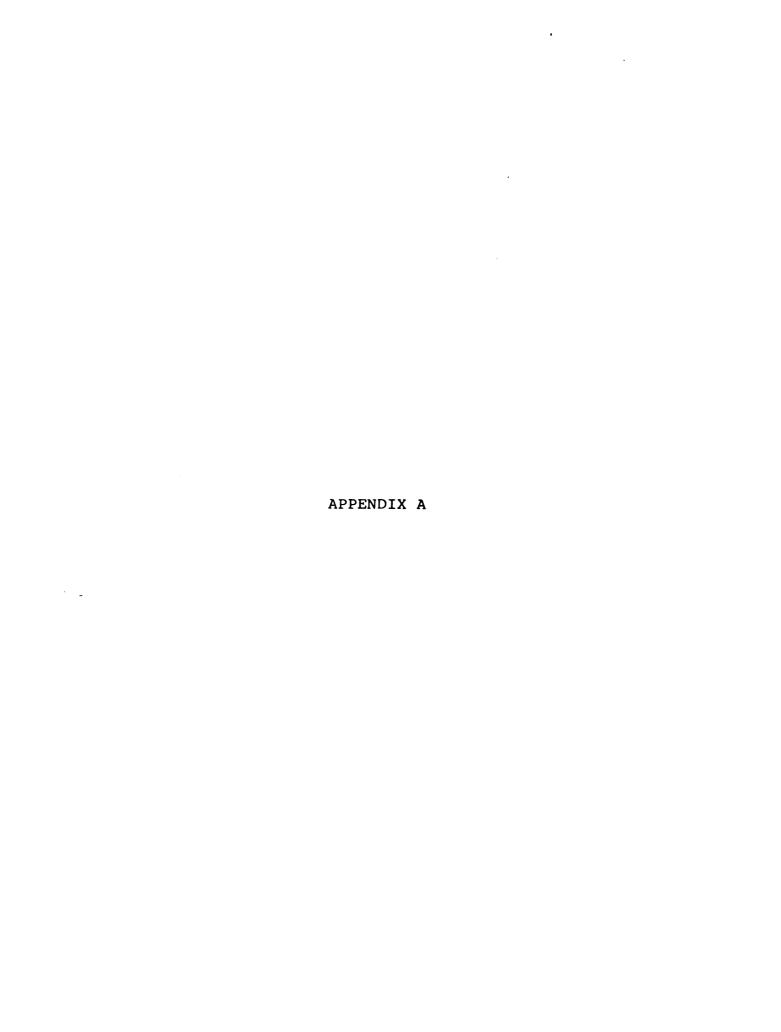
information. Only additional experimentation can determine whether this simular model is useful for understanding settlement and locational change in other geographic areas.

This study may be contrasted with other subsistence-settlement system simulations because of the emphasis here on multiple hypothetical models of change, rather than a display of changes associated with a single set of model assumptions. The utility of this simulation lies in comparing the relative fit of different models, and the effects of their differing assumptions on system development and change, rather than the demonstration of perfect fit with a solitary model. As suggested by Hodder (1978:136), the utility of the simular approach may be realized not only in the search of goodness-of-fit as described by some statistical standard, but also in achieving relative measures of fit as an aid to data interpretation. In some cases, simulations that display close fit with the real world are successes that fail because they do not inform (Cordell 1982:122). current study, the simular results, while not testing for goodness-of-fit beyond the informal level, do leave questions to be answered that are more interesting than those with which the study began. For example, why are there territories persistently used by some families in isolation from others? Why do the occupants, real or simular, of some territories interact with those of some neighboring territories and not with others? Is the simular appearance of multiple areal use strategies a feature of the real world, and can it be demonstrated archaeologically?

The true success of the simular method apparently lies not merely with statistical demonstrations of pattern identity but with the increasing sophistication of the questions that may be asked of the pattern itself. Despite this study's shortcomings, this aspect of simular utility has been achieved.

The current study presented a progression of models, from the use of historical and ethnographic models, to locational/environmental modelling, to simular/cybernetic modelling. As in the cycle of scientific inquiry, all of these attempts feed back and inform one another in a reiterative way, and all have intrinsic utility. Communication among modellers who practice the cybernetic/simular approach may eventually allow the development of general hunter-gatherer models on the scale of the WORLD model developed by system dynamicists (Forrester 1971). The WORLD model underwent continual change and improvement with contributions over several decades by many researchers. This reiterative technique can benefit archaeological and anthropological understandings of hunter-gatherer cultural dynamics as researchers begin to appreciate the evolutionary nature of the modelling process.





#### APPENDIX A

## Catalog of Sites Used in the Analyses

## Charlevoix County

The Burgess site was reported by crews from the University of Michigan Museum of Anthropology in 1948, at which time a small surface collection was made. An examination of collections from this site at the University of Michigan Museum revealed several thick rimsherds with cord-wrapped-object impressions and bevelled lips; the cord-wrapped-object impressions left an interior boss. No age estimate has been made for this site other than to say that the ceramics derived from a Late Woodland occupation of the place.

20CX2 occupies a moderately well-drained beach on a slight embayment of Lake Michigan. The site is very close to 20CX40 and 20CX37, and shares their general environmental characteristics.

Reference: UMMA Site Information File

20CX9 The Saint James #1 site was recorded by J. E. Fitting during the 1973 survey of Beaver Island, and a collection of materials exposed on the surface was made. The collection is in the possession of the Michigan History Division. Fitting noted the presence of both shell- and grit-tempered pottery and suggested a date of A.D. 1330 +/- 100 for the site, which he described as a short-term camp.

The site is located at Saint James Harbor on well-drained beach sands at a prominent embayment of Lake Michigan. Offshore, the harbor includes a very accessible steep dropoff to depths of eight-ten fathoms, but there is also an area of extensive shallows wherein depths rarely exceed one fathom. In general it shares these characteristics with sites 20CX33 and 20CX59. Reference: Fitting 1973

20CX18 Researchers from the Michigan State University Museum tested and excavated portions of the O'Neill site in 1969 and in 1971, for a total of ca. 350 meters. It was a partially stratified Late Woodland-Historic habitation site. The Area A Juntunen Phase Occupation III provided dates of A.D. 1210 +/- 100(M-2406) and A.D. 1290 +/- 100 (M-2405). Area B's major occupation layer dated to A.D. 1455 +/- 100 (M-2398). Lovis interpreted Occupation III as a spring fish camp and Occupation II as a warm-season hunt camp, based on comparative faunal data (Lovis 1973:217-220).

The site occupies a well-drained sandy beach on a slight embayment of Lake Michigan, at a place where a small

stream enters the lake. The site shares the general offshore characteristics of 20CX39. Reference: Lovis 1973

20CX19 The multicomponent Pine River Channel site was excavated by crews from the Michigan State University Museum in 1971 and 1972. A total of ca. 230 square meters was excavated. There were Middle Woodland (Laurel) and Late Woodland (Straits area sequence) components present, but the main component was the Mackinac phase occupation based upon what is currently known (Holman 1978). Holman reported a radiocarbon estimate of A.D. 1110 +/- 80 (N-1266) relating to the Mackinac phase occupation (Holman 1978:427). The Mackinac phase occupation was probabaly a warm-season fishing camp.

The site occupies a high terrace above Lake Michigan at a point where a small natural channel connects the large lake with an inland lake network. Before channelization the Pine River was somewhat different in character than at present; according to Milner "the mouth of the Pine is a sharp rapids" (Milner 1874:633). This condition without a doubt affected the kinds of fishing methods appropriate for the place, as well as the favored habitat for various species. The water conditions in the present day are highly varied near the site. In the large lake the water depth contours are moderately steep to depths of greater than seven fathoms rather close to shore, while in the smaller lake depth contours are steeper and maximum depths are in the thirteen-fifteen fathom range. Reference: Holman 1978

Crews from the Michigan State University Museum discovered the Mt. McSauba site during the survey of 1969, and test excavations totalling ca. 16 square meters were conducted there during 1972 (Cleland, personal communication). The site was located on an eroding exposed beach and dune area and consisted of many buried living floors, all probably dating to the Late Woodland time period. It is on the Lake Michigan shoreline. No fish remains were recovered at the site (Smith 1983).

Northwest of the site the depth contours offshore are rather gradual and as much as ca. one kilometer offshore the depths are in the three-four fathom range. Beyond this distance the water deepens rapidly to depths of ten-fifteen fathoms.

Reference: MSUM Site File; Smith 1983

20CX26 The Cable Bay prehistoric site was also recorded by Fitting's 1973 survey of Beaver Island, at which time a small collection was made from an exposed portion of a deeply-buried soil horizon. The collection included cord-marked and cord-wrapped-object punctated over

cord-marked ceramics that reminded Fitting of those excavated at Heins Creek (Mason 1966). Based on that similarity Fitting suggested a date of A.D. 720 +/- 150 for Cable Bay, and called it a short-term camp.

The site lies deeply buried in unstable dune sands on the south side of the embouchure of Cable Creek into Lake Michigan. Offshore, this is an area of relatively shallow water and gentle gradients, but one and one-half kilometers beyond the shore are waters exceeding fifteen fathoms in depth.

Reference: Fitting 1973

20CX27 Fitting discoverd the Martin Point site during his 1973 survey of Beaver Island. He collected materials exposed in a sand blowout, including cord-marked, braced-rim ceramics that he dated between A.D. 900 +/- 75 and A.D. 1130 +/- 120 based on similarities with Bois Blanc braced-rim ceramics excavated at the Juntunen site and elsewhere (McPherron 1967). Based on what was collected, Fitting called the site a short-term camp.

The site is located in a sand blowout in a clearing south of Martin Point on the eastern shore of Beaver Island. Relative to other locations along this shoreline, the Martin Point site adjoins a larger-than-average area of water less than five fathoms in depth. About one and one-half kilometers offshore the depth increases rapidly to greater than fifteen fathoms.

Reference: Fitting 1973

Saint James #3 is another site discoverd by Fitting during his 1973 survey of Beaver Island. collected lithic materials from the surface, but no ceramics. However, large private collections existed from this site, and these probably included ceramics. The general location was the same as that given for 20CX5, a site collected by Albert Spaulding of the University of Michigan Museum of Anthropology in 1953. I examined the ceramics from the earlier survey collection during 1982 at The collection included several thick rimsherds the UMMA. reminiscent of late prehistoric or protohistoric materials, as well as at least one Late Woodland cord-wrapped-object impressed sherd. Based on the presence of three Late Woodland-early Historic projectile points and three possible gunflints, Fitting suggested that Saint James #3 represented a Late Woodland or contact-period village, a time estimate consistent with the stylistic attributes of the ceramics I examined.

The site lies at a large embayment of Lake Michigan at Saint James Harbor, and shares the microenvironmental setting of 20CX9 and 20CX59.

Reference: Fitting 1973; UMMA Site Information File

20CX37 The Fish 'n' Chips site was reported by survey crews from the Michigan State University Museum in 1967, and in the same year a total of ca. nine square meters was excavated there. This site was an extension of 20CX40 (Cleland, personal communication). Presumably it was a Middle Woodland fishing camp. Mammalian remains dominated the faunal assemblage (Smith 1983).

The site lies on a slight embayment of Lake Michigan at a beach east and north of 20CX40 and Nine Mile Point. The site shares the microenvironmental setting of 20CX40. Reference: MSUM Site File; Smith 1983

20CX38 Archaeologists from the Michigan State University Museum test-excavated the Charlevoix City Park site in 1969. A total of ca. thirteen square meters was excavated there (Cleland, personal communication). Additional materials were collected from the surface during 1968, 1969, and 1976. The collection included Late Woodland ceramics representing a short-term occupation about A.D. 1350 (MSUM Site File).

This site lies on the Lake Michigan shoreline at a slight embayment on well-drained sands. 20CX38 shares the general environmental characteristics of 20CX19, but is a bit further from the Pine River and the interior lakes. Reference: MSUM Site File

20CX39 Crews from the Michigan State University Museum discoverd the N.A.H.M. site in 1969, and made a subsequent visit there in 1976. During the first visit the crews presumably collected all of the cultural material exposed on the surface. The presence of ceramics verified a Woodland age estimate. Mammalian remains dominated the faunal assemblage (Smith 1983).

20CX39 occupies a well-drained sandy beach at a slight embayment of the Lake Michigan shoreline, at a point where an intermittent drainage feature dissects the dunes. Relative to other parts of the Charlevoix County shoreline, the water near this site is shallow over broad areas, and does not exceed three-four fathoms until several kilometers from the shore.

Reference: Lovis 1976; Smith 1983

20CX40 In 1967, field crews from the Michigan State University Museum discovered the Wood site, and excavated a total of sixty-three square meters (Cleland, personal communication). This site was interpreted as a multicomponent gathering and fishing camp with the main occupation representing the Juntunen phase of the Straits area ceramic sequence. Radiocarbon age estimates of A.D. 1510 +/- 100 (M-2058) and A.D. 1020 +/- 120 (M-2057) were reported (Lovis 1973:281-2). In addition to Juntunen wares, ceramics suggesting of Laurel culture and expecially Mackinac phase occupations were found. Mammalian remains

dominated the fauna recovered at 20CX40 (Smith 1983).

The site is located on the west side of Nine Mile Point on an exposed gravel beach, at a place where an intermittent stream dissects the beach areas. There is accessible shallow water offshore but in general the water depths increase rapidly to greater than ten fathoms. Another site, the Wood site extension or Fish 'n' Chips site, shares this general location (20CX37), and this was the locality from which Laurel Middle Woodland wares were recovered.

Reference: Lovis 1973; MSUM Site File; Smith 1983

20CX59 The Saint James #2 site was another prehistoric site discovered as a result of Fitting's 1973 survey of Beaver Island. It is 400 meters distant from Saint James #1. Fitting's surface collection included cord-marked sherds; he suggested that the site functioned as a short-term camp and was earlier in time than Saint James #1, i.e. earlier than A.D. 1330 +/- 100. The site is located on a large embayment of Lake Michigan at Saint James Harbor, and shares the microenvironmental setting of 20CX9 and 20CX33.

Reference: Fitting 1973

## Chippewa County

20CH2 A total of ca. 260 square meters had been excavated at the predominantly Middle Woodland Naomikong Point site by the end of the 1967 season. The extensive excavations of 1966-67 were managed by the University of Michigan Museum of Anthropology. In addition to the major Middle Woodland series of occupations, the Naomiking Point site included a Late Woodland and a nineteenth century historic component.

The ceramic collections from the Middle Woodland component demonstrated that Laurel culture people occupied the eastern Lake Superior basin early in the first millenium A.D. Radiocarbon assays suggested an occupation date of A.D. 430 + -400 (M-2055). Though the acidic soils at the site played havoc with organics, resulting in the recovery of few faunal remains, the site location and technological inventory demonstrated an undeniable reliance on fish at the site.

The Late Woodland ceramic materials represented many archaeological cultures, including the Straits area sequence of ceramic phases, of which Mackinac Ware dominated in number. But other affinities included Blackduck Oneota and Huron-Iroquoian materials.

The Naomikong Point site lies at the water's edge near a point of land extending into the shallow waters of Whitefish Bay in southern Lake Superior. The immediate site area faces a small embayment of the larger bay. Offshore conditions include shallow water areas close to the site, and deeper water somewhat close, but the contours are very gradual and in general depths do not exceed three-four fathoms for some distance away from the site.

Reference: Janzen 1968

The initial report of this mound on Sugar Island in the Saint Mary's River occurred prior to the 1931 publication of Hinsdale's \$Archaeological Atlas of Michigan\$. Subsequent efforts to relocate the mound in 1968 and in 1974 failed (Bigony 1968; Franzen 1975) but at some point in the past a small collection of ceramics from this site made its way to the University of Michigan Museum of Anthropology. I was able to examine this collection in 1982 at the Museum. I assumed that this collection derived from surface materials or from limited excavations. The sherds that I examined included several with stamped, incised, and cord-wrapped- object impressed decorations, presumably derived from a late Middle Woodland ceramic tradition.

The site lies on Sugar Island about 2 kilometers west of Lake George, at an elevation of ca. 192 meters above sea level, or ca. 15 meters above the level of Lake George. At places nearest the alleged location, Lake George is extremely shallow, i.e. less than one fathom in depth over broad areas.

Reference: UMMA Site Information File

Numerous early accounts suggested that a mound and/or village site existed at the mouth of the Potaganisssing River on Drummond Island. Franzen recovered a small ceramic collection from the vicinity and suggested that it represented a Laurel-like ceramic tradition, but there also existed several cord-marked ceramics that Franzen attributed to a non-Laurel or perhaps non-Middle Woodland component. There was also artifactual evidence of a historic component at this location.

The Cloudman site is located at the mouth of the Potaganissing River and at the head of a bay of the same name on the western shore of Drummond Island in northern Lake Huron, in an area dotted with small islands and shoals with water three fathoms deep or less.

Reference: Franzen 1975

20CH27 The Browns Fishery site at Whitefish Point was discovered during a 1968 survey by crews from the University of Michigan Museum of Anthropology. They collected a small amount of prehistoric material exposed on the surface. The ceramic materials included several sherds with cord-marking and horizontal cord-decoration that were attributed to an occupation in the latter part of the Late Woodland cultural-chronological period (Luedtke 1976:435).

The site occupies a sandy beach facing a large bay of southern Lake Superior, adjacent to relatively steep and deep offshore conditions.

-Reference: Luedtke 1976

20CH29 The Silver Creek site was investigated by crews from the University of Michigan Museum of Anthropology during the 1968 field season. A small collection was made that included several undiagnostic water-rolled ceramic sherds. On this basis the site was known to be Woodland in age. No evidence of this site was detected during a visit by archaeologists in 1978 (Franzen 1979).

The site occupies a sandy beach at a place where a small creek enters Whitefish Bay of southern Lake Superior. The offshore topography suggests extensive shallow water with maximum depths in the one-two fathom range.

Reference: Bigony 1968; Franzen 1979

20CH30 The Big Pine site was discovered in 1974 by crews working in association with the Chippewa County Historical Society. A surface collection was made from materials exposed in an eroding dune.

The site consisted of scattered concentrations of fire-cracked rock and objects of aboriginal manufacture along the shore of Pendill's Bay and within the deflated

dunes close to the water's edge. Franzen suggested a Laurel Middle Woodland cultural affiliation for ceramics of this site, based on the presence of dragged-stamped and other stamped decorative attributes of the pottery.

The site occupies an exposed and eroding sandy dune area close to the waters of Pendill's Bay on the south shore of Lake Superior. Offshore contours are rather steep at this location, and depths of greater than fifteen fathoms are relatively close to the shore.

Reference: Franzen 1975

20CH32 The Pendill's Bay site was very close to 20CH30 and was also recorded during the 1974 Chippewa County Historical Society survey. Like 20CH30, the collection derived from materials exposed on the surface of eroding dunes and intra-dune depressions, at a distance of about 50 meters from the edge of Pendill's Bay on southern Lake Superior. It shares the microenvironmental setting of 20CH30.

The Pendill's Bay surface collection evidenced the occupation of the area during the early part of the Late Woodland cultural-chronological period. Franzen noted a similarity between some 20CH32 ceramics and those derived from Early Late Woodland occupations at the Juntunen site (20MK1) and at sites in Door County, Wisconsin (Franzen 1975:24). These affiliations suggested a date of ca. A.D. 700-1000 for this component (McPherron 1967). Reference: Franzen 1975

-20CH34 This multicomponent Woodland and historic site was discovered by investigators working for the Chippewa County Historical Society during the 1974 season. The surface collection derived from eroding riverbanks for some distance along both sides of the Charlotte River. The collections included aboriginal ceramics and a historic period gunflint. Sadly the ceramics were body sherds of which nothing could be said of cultural affinity.

The site occupies the river mouth of the Charlotte at the place where it enters Lake Nicolet of the St. Mary's River, northern Lake Huron, on a passage of very shallow water between the mainland and a large island.

Reference: Franzen 1975

20CH43 The Albany Harbor site was investigated by archaeologists associated with the Chippewa County Historical Society during the survey of 1974. A surface collection was made of objects in a wind-deflated sandy area very close to the shore of Lake Huron.

Some of the ceramics collected in 1974 resembled those associated with the Bois Blanc components at the nearby Juntunen site (McPherron 1967). Others were impressed with a dentate decorative technique but were not typed (Franzen

1975:25). Franzen suggested that 20CH34 was a Late Woodland seasonal camp of the A.D. 1000-1200 time range, i.e. the Bois Blanc phase of the local ceramic sequence.

20CH43 occupies a small sandy bay on the north shore of Lake Huron in an area of may offshore shallows and shoals. Reference: Franzen 1975

20CH44 The Klamerus-Bucht site was also discovered and surface-collected by archaeologists sponsored by the Chippewa County Historical Society during the 1974 field season. Ceramics from 20CH44 resembled dentate-stamped, dragged-stamped, linear-stamped, and pseudo-scallop shell impressed pottery generally attributed to Laurel or late Laurel culture manifestations.

The site is located very close to 20CH6 at the head of Potaganissing Bay ca. 75 meters from the shoreline, in an eroded and partially disturbed area. In general it shares the microenvironmental setting of 20CH6.

Reference: Franzen 1975

20CH45 The West Harbor site occupies the west harbor of Harbor Island in Potaganissing Bay, northern Lake Huron, in an area sprinkled with islands, shoals, and shallows. Crews from the Chippewa County Historical Society archaeological survey of 1974 discovered the site in a sandy blowout a few meters from the lake, and collected prehistoric materials from the surface.

20CH45 was a Laurel Middle Woodland-age temporary habitation site, from which Franzen identified a variant of Laurel linear-stamped ceramics (Franzen 1975:26). He also recovered a small collection of chipping debris at 20CH45, a seasonally-occupied, functionally-specific campsite. Reference: Franzen 1975

20CH46 Crews from the 1974 Chippewa County Historical Society survey discovered the Late Woodland Slater site. The site shares its location with a nineteenth-century habitation site alleged locally to be an Indian villige. A surface collection included items from both components.

This site occupies a sandy blowout at Albany Harbor on northern Lake Huron, and is very close to site 20CH43, a Bois Blanc-phase Late Woodland camp a few hundred meters to the west. Ceramics from 20CH46 bore resemblance to those associated with the type Juntunen Drag-and-Jab excavated at the nearby Juntunen site (Franzen 1975:27). The survey crews also collected a small amount of chipping debris, and a small collection of nineteenth-century domestic materials including glass, modern ceramics, nails and other metal objects.

This site shares the microenvironmental setting of site 20CH43, one of may offshore shallows and shoals. Reference: Franzen 1975

20CH51 The site complex of Fort Brady at Sault Ste. Marie included several prehistoric components. These were investigated by Lyle Stone in 1967 and by Lee Minnerly in 1976-77 in conjunction with excavations of historic features associated with European and American occupations of the place. As a result there is no way to depict how much of the prehistoric portion of the site complex has been excavated. The excavated material is housed at the Michigan State University Museum.

Aboriginal ceramics recovered at Fort Brady suggested a Middle Woodland Laurel occupation followed by a Late Woodland Mackinac phase occupation, but the Laurel materials predominated in number and in the appparent extent of the associated remains.

Fort Brady occupies sandy ground next to the St. Mary's rapids, an area long known for its unique and prolific fishery.

Reference: Fitting 1974

20CH77 Excavations at the Schoolcraft House in Sault Ste. Marie in 1974 conducted by archaeologists from the Michigan History Division revealed the presence of a partially-stratified multi-component prehistoric site of Woodland age. In the course of excavations to uncover nineteenth-century habitation features associated with the house, some evidence of Laurel Middle Woodland and Juntunen-Oneota Late Woodland camps was discovered. Because the excavations included a predominance of historic-era-materials, one cannot estimate the area of prehistoric remains actually removed.

James Fitting, who analyzed the prehistoric materials from 20CH77, labelled it a low-density site representing Lake Forest Middle Woodland use in the A.D. 200-400 time range, with ceramics similar to those removed from other Mackinac and Chippewa County Middle Woodland sites (Fitting 1975). The Late Woodland ceramic materials included types associated with the Juntunen phase of the Straits-area sequence, and with Oneota ceramics derived from the Green Bay region.

The site occupies a sandy area south of the rapids of the Saint Mary's River.
Reference: Fitting 1975

### Delta County

20DE1 This Late Woodland site at Fayette was discovered by survey crews from the University of Michigan Museum of Anthropology during the 1956 season, and an additional survey occurred there in 1963. The investigators made surface collections on both occasions.

I examined some ceramics from this site during 1982 at the UMMA. They were exfoliated body sherds with heavy tempering particles and cord-marked surfaces. On the basis of the cord-marking it was assumed that there was at least a Late Woodland component at 20DE1.

The site lies on the west shore of the Garden Peninsula on well-drained sands. Its location, on a lengthy gentle bay of Big Bay de Noc, is protected toward the northeast by a bluff formation. Close to shore the water is very shallow over a shelf formation, which then abruptly drops to depths of five-six fathoms. In general, the site shares this microenvironmental location with 20DE7 and 20DE8. Reference: UMMA Site Information File; Peske and Kent 1963

The Puffy Bay site was surveyed in 1956 by crews from the University of Michigan Museum of Anthropology, and a small surface collection was made. Ceramics from this site were examined at the UMMA during 1982, but because no time-related attributes were recorded, the site was assumed to be one of general Woodland age.

The site lies on an intermittent stream that ends at a small protected embayment, Puffy Bay, that joins the waters of Big Bay de Noc. The bay is well-protected by land masses in all directions but north, and is less than a fathom in depth. Beyond the small bay the water is generally less than three fathoms deep all the way to the head of Big Bay de Noc, several kilometers to the north.

Reference: UMMA Site Information File

20DE3 This site complex was investigated in 1963 and 1965 by crews from the University of Michigan Musuem of Anthropology. There were many caves that were of interest, but for the purposes of this study only caves B-95 and B-10 were of great interest.

The cave B-95 was investigated in 1965, and a total of ca. 13 square meters was excavated. The occupation included seven human burials, and yielded a radiocarbon estimate of A.D. 375 +/- 130 years (M-1795) on wood associated with two of the burials. A relatively large faunal assemblage consisting of fish, birds, and some mammalian remains accompanied the other material remains. Excellent conditions for organic preservation prevailed, and wooden and fiber artifacts were recovered. One woolen fiber artifact is without a doubt derived from the historic period, yet the site, based on excavated data, is apparently

a single-component site. This situation calls the radiocarbon date into question, unless the woolen material was secondarily deposited.

The site occupies a bluff at the eastern shore of Big Bay de Noc, Lake Michigan. The closest area offshore is one of sharp topographic contours, but to the north is a vast area of shallows.

The cave B-10 was excavated by archaeologists in 1963 for a total of ca. 16 square meters. In addition, extensive surface collections were taken from this cave by the archaeologists and by the cave's owner. Collections from the cave B-10 consisted of many projectile points as well as several antler harpoons, most of which supported a typologically-derived Middle Woodland date for the cave's use. It has been suggested that the cave was primarily a site of ritual activity related to the food quest as opposed to one for habitation. No habitation debris was recovered from the cave, but a small faunal sample suggested use of the area by predatory birds and other animals. The site shares the microenvironmental setting of cave B-95. Reference: Fitting 1968

The Summer Island site has a long history of many surveys and excavations by a number of archaeologists, but the most significant were the excavations of 1967 conducted by David Brose and the University of Michigan Museum. A total of ca. 205 square meters was excavcated during 1967, but earlier researchers performed test excavations not included in this total.

The site was a stratified multicomponent Laurel Middle -Woodland and Upper Mississippian occupation with a lengthy radiocarbon record to support its age. The Middle Woodland materials suggested occupations as early as A.D. 70 +/- 280 (M-2073); the Upper Mississippian occupations occurred at the end of the thirteenth century, or A.D. 1290 +/- 200 (M-2072). The site included a protohistoric component.

Summer Island lies ca. three kilometers south of the Garden Peninsula in Big Bay de Noc, Lake Michigan. This site occupies a relict beach on a small sandy harbor on the east side of the island. Between Summer Island and the closest mainland landfall, the water depths exceed six-seven fathoms and the contours offshore are sharp. However, adjacent to the site and extending toward Little Summer Island and landfalls to the northwest are extensive areas of shallow water not exceeding one-two fathoms in depth. Reference: Brose 1970

20DE7 Survey crews from the University of Michigan Museum of Anthropology reported this site's location during the 1965 survey of the Garden Peninsula, at which time a surface collection was made. The collection included ceramics and projectile points dated on stylistic grounds to the eighth and ninth centuries A.D. (Luedtke 1976), i.e. an

early phase of the local Late Woodland sequence. Stylistically the materials suggested affinities with Heins Creek ceramics from Wisconsin (Fitting 1968:117).

The site occupies a dune area about one hundred meters or so away from the current shoreline of Big Bay de Noc. A small stream drains the area, entering a small bay of the lake.

The Port Bar site probably represented a single-component early Late Woodland warm-season hunt camp (Fitting 1968). The site shares the environmental setting of 20DE1 and 20DE8.

Reference: Fitting 1968; Luedtke 1976

The Janowski site adjoined 20DE7 to the south, 20DE8 and survey crews from the University of Michigan Museum discovered this site during the 1965 season. A surface collecting expedition revealed that the site 20DE8 was similar to 20DE7, although no ceramics were found at 20DE8. It was apparently closer to the modern beach than is 20DE7, but occupied the same environment otherwise. Fitting suggested that 20DE8 represented a short-term, warm-season hunt camp though no age estimate was made (Fitting 1968). For the purposes of this study I assumed that 20DE8 was of Woodland age, because it adjoined a site that is without a doubt a Woodland creation, and in terms of correlations between lake levels and site locations, 20DE8 was likely more recent than is 20DE7. Reference: Fitting 1968

James Fitting of the University of Michigan Museum during the 1965 season, at which time a surface collection of lithics was made. A local resident reportedly collected a ceramic vessel from this site but several professionally-directed surveys of the place have yielded only lithics. Based on the amateur report, the site was assumed to represent a short-term occupation of Woodland age, though Fitting called it a probable Archaic-age winter hunt camp (Fitting 1968:132).

The site lies on a sandy beach along a well-protected shallow embayment of Big Bay de Noc on the western shore of the Garden Peninsula. The waters of the narrow embayment do not exceed three fathoms in depth, but at the western side of the bay are sharp contours over which the water depths drop rapidly to greater than five fathoms.

Reference: Fitting 1968

20DE10 Survey crews from the University of Michigan Museum investigated the Fairport Store site during the 1965 season. They collected the surface and found exclusively lithics, and this situation prevailed over several subsequent surface collecting expeditions there.

There was no way to be precise about the age of the Fairport Store site. Fitting argued that the site functioned as a warm-season, short-term hunt camp, but made no age estimate on the place. For the sake of the present study I assumed that the site belonged to the Woodland cultural-chronological period, but as a caution, analyses were conducted both with and without the lithics-only sites to control for their presence in the sample.

The site lies on an exposed sand ridge several hundred meters from a rough limestone cobble beach adjoining the waters of Big Bay de Noc on the southern shore of the Garden Peninsula. Offshore there are extensive shallows and gradually-increasing depth contours.

Reference: Fitting 1968

20DE11 The Point Detour Bay site (also known as the Sand Bay site) lies on the southern end of the Garden Peninsula on a small embayment of Big Bay de Noc. It was a single-component Late Woodland flintknapping workshop based upon the assessment of the original research team (Binford and Quimby 1963).

Field crews from the Chicago Natural History Museum discovered the site in 1962. They conducted limited test excavations there during 1962, and in 1963, researchers from the University of Michigan conducted additional tests. The total area excavated is unknown, but it probably was limited to a few square meters.

For the purposes of this study I assumed that this site was Woodland in age, though there was no artifactual evidence to bolster this assumption. The elevation of the site approximately five and one-half meters above current mean lake level was comparable to the elevation of the upper levels of mixed Late Woodland-Upper Mississippian deposits at the nearby Summer Island site. But perhaps more serious was the likelihood that this was not a habitation site at all and therefore its locational character relative to fish/food potential were irrelevant, at least to the current study. So as far as assessing locational qualities was concerned, this site was deleted from the sample at appropriate places.

The site lies on the shore of Sand Bay at the southern end of the Garden Peninsula. The water depth does not exceed three fathoms for some distance from the shore, and in general offshore topographic contours are gentle.

Reference: Binford and Quimby 1963; Fitting 1968

20DE17 The Winter site was partially excavated by field crews from the Western Michigan University Department of Anthropology during the 1972 field season for a total of ca. 125 square meters.

The Winter site was a multicomponent Middle Woodland campsite in the A.D. 150-250 age range based on comparative ceramic data. Both Havana-influenced wares and North Bay

wares were represented there. Both components included ceramic types similar to those excavated at the Mero and Porte de Morts sites by Mason, but there were other ceramic types present that were thought to be distinctive to the northern Lake Michigan area (Richner 1973).

This site lies several hundred meters east of Big Bay de Noc on the west shore of the Garden Peninsula, along the bank of a small stream. At the time of its occupation the site area probably bordered the large lake, and offshore conditions were likely extensive marshy shallows. At present, water depths offshore are generally less than one fathom over extended areas.

Reference: Richner 1973; Martin 1980b

20DE51/52 This probable Late Woodland site at the village of Nahma was discovered and recorded by the amateur archaeologists Collins and Holmquist during 1942, at which time they collected a number of triangular projectile points and at least one Late Woodland ceramic vessel. All of the finds derived from the surface, and in some cases human skeletal remains were apparently associated with the artifacts.

The site lies on wet sandy soils on both banks (hence the two site numbers) of the Sturgeon River at its embouchure into Big Bay de Noc of Lake Michigan. Offshore the water is exceptionally shallow over large areas. For the purposes of this study I assumed that one site of Late Woodland age existed in this location.

Reference: Collins 1942

#### Emmet County

20EM2 This site at Goodhart was visited by Emerson Greenman during a 1927 survey for the University of Michigan Museum of Anthropology. A collection, presumably from the surface, was made at that time.

My examination of the information about this site at the University of Michigan Museum revealed that the general area of 20EM2, 20EM40, and the Goodhart environs was the source of a large private collection containing historic materials (including some from burials) as well as some prehistoric materials. The body and rimsherds were decorated with deep punctates produced by a cord-wrapped-object that left a pronounced interior boss. On this evidence the site was dated as Late Woodland, but it was obvious that there was a historic component here (or nearby) as well.

The site 20EM@ occupies an exposed well-drained level sandy beach, and shares the general microenvironmental setting of 20EM40.

Reference: UMMA Site Information File; Greenman 1927

20EM4 The Wycamp Creek site was surveyed in 1899 by Emerson Greenman of the University of Michigan Museum of Anthropology. The Michigan State University Museum conducted excavations there in 1967, for a total of ca. 101 square meters (Cleland, personal communication).

The site contained horizontally and vertically stratified Middle and Late Woodland components that represented a bewildering variety of local ceramic wares ranging in age from Laurel culture materials to Iroquoian ceramics of the protohistoric period. Radiocarbon estimates derived from Wycamp Creek showed evidence of occupations in the 13th-16th centuries [A.D. 1220 +/- 110 (M-2059) and A.D. 1605 +/- 100 (M-2060)].

The site lies at the embouchure of a small stream into Lake Michigan on a well-drained level sandy beach. Relative to other coastal conditions in this area, the site is very close to large areas of shallow water, i.e. water in the one-three fathom range.

Reference: Luedtke 1976; Lovis 1970

20EM19 The Waugoshance Point site was reported and test-excavated by crews from the Michigan State University Museum during the 1970 season. A total of ca. twelve square meters was excavated (Cleland, personal communication). The site has been estimated as Late Woodland in age based on ceramics in the MSUM collection (Luedtke 1976). Because the ceramics were unavailable for my assessment, the dating of this site as Late Woodland was based on Luedtke's designation.

The site occupies an exposed beach on the north side of

Waugoshance Point, at the western extremity of the Straits of Mackinac. The site adjoins extensive shallows but there is an accessible dropoff to depths of greater than ten fathoms a kilometer or less from shore.

Reference: Luedtke 1976

The Portage site was discovered in 1974 and 20EM22 partially excavated during 1974 and 1975 by crews from Michigan State University Museum. A total of ca. sixty-five square meters was excavated. This site was a partially-stratified multicomponent Laurel Middle Woodland and Mackinac phase Late Woodland site, but the Laurel component was by far the larger. There were many spatially separated concentrations of habitation debris at this site that represented multiple occupations within the Woodland time range, particularly in the Laurel culture time span. A Laurel stratum yielded an estimate of A.D. 120 +/- 120 (DIC-652, unpublished). The Late Woodland strata included ceramics of the post-Mackinac ceramic phases (i.e. Bois Blanc and Juntunen) as well as some Blackduck materials. Spring-spawning fish and many mammalian remains make up the faunal assemblage (Smith 1983).

The site occupies several sandy well-drained hollows within steep Nipissing-age dunes ca. 100 meters east of a deep embayment of Lake Michigan. Offshore there are large areas of shallow water, at least in comparison to other offshore contours within the bay. For a kilometer or more offshore the water is less than five fathoms in depth. Reference: Lovis and Holman 1976; MSUM Site File; Holman 1978; Smith 1983

The Zuber site was discovered by survey crews from the Michigan State University Museum during 1966 and a small surface collection was made. The materials collected included a faunal sample and several castellated rimsherds with closely-spaced cord-wrapped-object decorations reminiscent of Late Woodland post-A.D. 1000 Bois Blanc phase ceramics (Holman, personal communication).

The site occupies an exposed well-drained sandy beach on the shores of Lake Michigan. Offshore depth contours are moderately steep with depths of ca. thirty fathoms within two and one-half kilometers offshore.

Reference: MSUM Site File

20EM35 The Foster Otto I site was reported to a 1974 Michigan State University field survey crew. A surface collection was made, and though most of the cultural material was apparently of historic origin, there was a scatter of prehistoric material that included some cord-marked Late Woodland ceramics and a small chipped stone sample.

The site lies on an exposed sandy beach at the Lake

Michigan shoreline. Offshore contours are relatively steep close to the beach.

Reference: MSUM Site File

20EM40 The Johnston site was discoverd by survey crews from the Michigan State University Museum during 1966, and a small uncontrolled surface collection was made. The collection included "at least one clearly Late Woodland vessel of an unknown ware category" (Holman, personal communication). Human skeletal remains were apparently recovered from this location at an earlier time (MSUM Site File).

The site occupies an exposed beach at the shore of Lake Michigan on deep well-drained level sands. Depth contours offshore are relatively steep with water in the fifteen fathom range less than a kilometer offshore.

Reference: MSUM Site File

20EM51 The site MCS-4 was discovered by Lyle Stone during the 1972 survey of Mackinaw City and the Straits area. It was reported to be a single-component Late Woodland site of the Mackinac phase (Stone 1975:170). A total of ca. nine square meters was excavated there during 1972.

The site lies at a slight embayment on the south shore of the Straits of Mackinac on deep, well-drained level sandy soils, adjacent to a shallow bay with water depths in the one-three fathom range. But water deeper than ten fathoms is less than a kilometer distant from the shore.

Reference: Stone 1975

20EM52 This extensively-excavated site complex is a multicomponent prehistoric and historic manifestation collectively referred to as Fort Michilimackinac. Collections from this site included prehistoric materials that represented virtually every ceramic ware category of the region, with the apparent exception of Juntunen wares (Lovis and Mainfort 1971). The excavations within the fort area, though not specifically directed toward the recovery of prehistoric remains, occasionally encountered features of prehistoric age, and yielded Late Woodland (especially Mackinac phase) and protohistoric as well as Ontario Iroquoian ceramics (Lovis, personal communication). Excavations in 1973 of a French row house complex in the parking lot area south of the Mackinac bridge uncovered a Laurel culture site which was then partially excavated. The Laurel component was estimated at A.D. 450 + /- 120 (M-2489)and A.D. 650 + / - 130 (M-2490) (Lovis and Holman 1976).

The site complex occupies an exposed well-drained level sandy beach at the south side of the Straits of Mackinac. It shares the microenvironmental setting of 20EM51. Reference: Lovis and Mainfort 1971; Stone 1975; Maxwell 1964; MISPC Field Notes

### Mackinac County

20MK1 The Juntunen site was excavated by crews from the University of Michigan Museum during the 1960-61 and 1963 field seasons. A total of ca. 450 square meters was excavated. The site is a large partially-stratified series of Late Woodland habitations ranging in time from A.D. 835+/-75 (M-1142) to A.D. 1330+/-100 (M-1391). In addition there is a Middle Woodland manifestation (20MK83) adjoining 20MK1. It is from the Juntunen site with its elaborate stratification and lengthy radiocarbon catalog that the local prehistoric ceramic sequence was derived.

McPherron suggested that the site was occupied from spring until fall, based on floral, faunal, and environmental data that depicted a reliance on fishing as well as the use of a great variety of other plants and animals.

The site is located on well-drained level beach sands at the western end of Bois Blanc Island in the Straits of Mackinac. Offshore conditions are variable from the vantage point of this site (or sites, since these characteristics also apply to 20MK83). On the north and west, the water is relatively shallow (i.e. less than five fathoms) to points beyond the adjacent island and ca. one-two kilometers offshore. On the south shore of the island, water deeper than eight fathoms is rapidly encountered, because depth contours drop abruptly very close to shore.

Reference: McPherron 1967

20MK6/7 Test excavations in the area of 20MK6 and 20MK7 by Michigan Technological University field crews during 1978 and 1979 revealed an extensive protohistoric and historic habitation site associated with the Gros Cap cemetery. But in addition to the historic materials recovered, a number of grit-tempered Juntunen-like aboriginal ceramics, several shell-tempered sherds, as well as a small chipped-stone assemblage was collected, supporting the possibility that the place included a disturbed Late Woodland component. The MTU collections derived from intensive uncontrolled surface collecting over a broad disturbed area as well as controlled subsurface testing of a total of ca. five-ten square meters.

The site complex lies on well-drained sandy soils on an embayment of the north shore of Lake Michigan at a place where a small river enters the lake. The waters of this small bay do not exceed five fathoms in depth for one-two kilometers offshore.

Reference: Martin 1979

20MK19 The multi-component Halberg site was discovered and test-excavated by crews from the Michigan State University Museum during the 1967 season. A total of ca. forty square meters was excavated (Cleland, personal

communication). The site was Late Woodland in age based on the presence of ceramics similar to those excavated at the nearby Juntunen site, particularly those of the Bois Blanc phase. The crews excavated some faunal materials but no estimate of season of occupation, nor its duration has been made. The assemblage included many mammalian remains, most of which appear to be domestic (Smith 1983).

The site lies exposed on the north shore of the Straits of Mackinac at Point St. Ignace. Offshore conditions include extensive shallow water because of the closeness of the North and South Graham Shoals, over which water depths of ca. three fathoms are common for several kilometers offshore. Immediately beyond the shoals the water rapidly attains depths of greater than fifteen fathoms. Reference: MSUM Site File; Smith 1983

20MK22 The Scott Point site has a lengthy history of archaeological investigation, including surface collecting, test excavation, and, most recently, full-scale excavation. At the close of the 1980 season, at least 104 square meters had been excavated by crews from Northern Michigan University.

This stratified multicomponent Late Woodland site included materials representing the entire local sequence of Late Woodland phases, as well as Blackduck and Oneota ceramic wares, but the major occupation seemed to be that (or those) associated in time with the Juntunen phase. A faunal analysis of materials recovered by the NMU excavations of 1980-81 revealed a shift in proportions of classes present from the earliest to the latest occupations, specifically that the Juntunen phase occupations included greater numbers of the mammalian class than did the earlier phases. Additionally, remains of fall-spawning fish occurred more frequently in the later occupations than in the earlier, with a corresponding decrease in sturgeon for the later occupations.

The site lies on a small sandy bay west of Scott Point on the north shore of Lake Michigan, amid deflated dunes. The offshore topography is rather complicated. In general the water is less than four-five fathoms in depth for the first kilometer or so offshore, but the configuration of depths is highly irregular, with the frequent appearance of very small rocky shoal-like features, over which the waters depths are more likely to be in the one-two fathom range. Reference: Martin 1981; Buckmaster 1980

20MK51 The Gyftakis site was tested by crews from the Mackinac Island State Park Commission in 1972 and excavated by the Michigan History Division in 1973. The excavated area totalled 488 square meters, but some of this area pertained primarily to historic-period features. A radiocarbon assay from a hearth feature accompanying an ossuary burial yielded an age estimate of A.D. 170 +/- 80

(N-1723). The Gyftakis site represented a Lake Forest Middle Woodland (Laurel) camp occupied by people gathering a variety of fish species from the waters of Lake Huron, including both spring and fall spawners.

The site occupies a well-drained level sand beach at an embayment of Lake Huron on the north side of the Straits of Mackinac. This site shares the locational characteristics of 20MK82.

Reference: Fitting 1979

20MK53 Crews from the Mackinac Island State Park Commission discovered and tested the Norge Village site during the 1972 survey of St. Ignace. A total of ca. forty-five square meters was excavated by the end of the 1973 season.

The site is a mixed prehistoric and historic deposit with a sparse but clearly Laurel Middle Woodland component topped by a contact-period component dated at A.D. 1640 +/-85 (N-1727). No estimate of group size, site permanence nor seasonality was made, though both spring and fall-spawning fish species were present. The small faunal sample is dominated by mammalian remains.

The site lies on an embayment of Lake Huron on the east shore of Point St. Ignace opposite the islands of the Straits. Offshore, conditions are similar to those at 20MK51: extensive shallow water in two-three fathom range. Reference: Fitting 1974

The Beyer site was discovered and tested by crews 20MK54 from the Mackinac Island State Park Commission in 1972. Further excavations there in 1973 uncovered a total of greater than seventy-five square meters. Excavations at the Beyer site revealed a scatter of diverse ceramics and other remains as well as a burial. Radiocarbon assyas from materials collected in a hearth-like feature yielded an estimate of A.D. 1270 +/- 90 (N-1726). Ceramic wares recovered included the familiar Mackinac-Bois Blanc-Juntunen local sequence as well as Huron-like wares, Oneota wares, and Dumaw Creek-like wares, documenting occupations throughout the A.D. 1000-1650 time range. Fitting suggested that the Beyer site was a short-term camp over its 600-year sequence of occupations, and posited that it may have served as a warm-season satellite camp to the Juntunen site, at least in its earliest years of occupation. The later occupations, expecially those associated with Level One, suggested a short-term protohistoric camp with a mixed hunter-fisher subsistence base.

The site lies on a low-elevation beach terrace on the eastern shore of Point St. Ignace facing the islands of the Straits of Mackinac. There has probably been some recent sand deposition offshore near this location due to the construction of several large piers. But there are otherwise quite variable conditions offshore. Directly east

the water quickly exceeds eight-ten fathoms in depth, but to the south are large areas of water in the one-four fathom range.

Reference: Fitting 1974

20MK58 Crews from the Mackinac Island State Park Commission reported and tested the single-component White site during the St. Ignace survey of 1972, at which time a total of ca. fourteen square meters was excavated. Fitting dated the site, on the basis of Juntunen ceramic wares recovered there, to A.D. 1300 (Stone 1975:47). Fitting called 20MK58 a low-density, short-term camp based on ratios of artifact types recovered.

The site lies at the entrance of the Pine River into St. Martin's Bay at the north shore of Lake Huron. St. Martin's Bay includes large areas of shallow water, with depths gradually increasing to four fathoms at a distance of three kilometers or so from shore.

Reference: Stone 1975

20MK61 The Sposito site and accompanying Steiner burials were discovered by crews from the Mackinac Island State Park Commission in the summer of 1972. Both localities are adjacent to the McGreggor site, and it is very likely that Steiner, McGreggor, and Sposito are so complex and so disturbed that the actual relationships among the many occupations in this area may never be clearly understood. But for the purposes of this study Sposito/Steiner will be described as one site area and McGreggor as a separate entity.

The Steiner burials were discovered when cut through by a sewer trench. Ceramics accompanying the burials suggested a late Middle Woodland date with stylistic similarities to later Mackinac and Blackduck wares, "a late Middle Woodland assemblage of perhaps 600 A.D." (Stone 1975:47).

In 1973 the bulldozing of a house site revealed the Sposito site about fifty meters south of the Steiner locality. Nearly 110 square meters was quickly excavated by bulldozer and a salvage crew. Ceramic materials suggested a tentative date of A.D. 200-500 for a Middle Woodland occupation bearing stylistic similarities to later Mackinac and Blackduck wares. There was also a small Late Woodland assemblage that Fitting dated to the fifteenth or sixteenth century.

The site shares the locational characteristics of 20MK102.

Reference: Fitting 1974; Stone 1975

20MK82 The Marquette Mission site in St. Ignace was excavated by crews sponsored by the Mackinac Island State Park Commission during the 1971 and 1972 seasons. A total of 205 square meters was excavated by the end of 1972. 20MK82 is a multicomponent aboriginal habitation site and seventeenth century mission site with materials representing the prehistoric era as well as more recent centuries. A quantity of faunal remains and a scatter of aboriginal habitation debris was excavated; most came from an archaeological area thought to represent a seventeenth-century Tionontate Huron longhouse. The Michigan State University Museum began additional excavation at 20MK82 during 1983.

Shell- and grit-tempered ceramic materials excavated in the course of the historic investigations represent short-term prehistoric camps dating in the post-A.D. 1000 time range (Stone 1971). Fitting suggested that these sparse occupations represented winter hunting camps (Fitting 1976:244-45).

The site 20MK82 lies on a sheltered embayment of Lake Huron on a level well-drained sandy beach on the northeastern shore of the Straits of Mackinac. The waters of East Moran Bay, to which the site is neighbor, do not exceed three fathoms in depth, but beyond the bay depths increase to greater than twenty-five fathoms. Reference: Stone 1971; Fitting 1976

20MK83 The Arrowhead Drive site was excavated during the extensive excavations of the neighboring Juntunen site (20MK1) in 1960-62 by crews from the University of Michigan Museum of Anthropology. Feature 45, an ossuary burial of at least eight individuals dated to A.D. 50 +/- 120 (M-1392) dominates the collection from this site. A total of ca. forty-nine square meters was excavated at 20MK83; most test pits apparently clustered near Feature 45. Surface evidence suggested that there was a light scatter of Middle Woodland-age debris over a broad area (ca. one-two kilometers) along a ridge slightly higher in elevation than the site 20MK1 (McPherron 1967:27).

The site lies on the west end of Bois Blanc Island in the Straits of Mackinac on a well-drained sand ridge about 1000 meters from the lakeshore and about five meters above current lake levels. This site occupies the same microenvironment as 20MK1.

Reference: Bettarel and Harrison 1962; McPherron 1967

The Ferrier-Tamlin site was surface-collected and tested in 1973 by crews from the Michigan History Division. A total of ca. thirty-seven square meters was excavated. This site is probably the same as that discovered and tested by field crews from the University of Michigan Museum in 1963, which they called the Roger's Restaurant Site (20MK24).

Excavations at the site revealed a disturbed horizontally-stratified Middle and Late Woodland situation with a historic component overlying both prehistoric strata. The Late Woodland Tamlin component yielded a radiocarbon sample estimated at A.D. 1050 +/- 85 (N-1725). The Middle Woodland Ferrier component yielded a date of A.D. 930 +/- 90 (N-1724), a date that Fitting believed to be at odds with the ceramic styles recovered there. He suggested that the materials on which the Ferrier date was obtained probably derived from the Late Woodland Tamlin component.

The ceramics from the Ferrier component bore some resemblance to those excavated at the Middle Woodland Naomikong Point and Winter sites, but the Tamlin ceramics showed strong southern Michigan influence and did not resemble their contemporaries excavated at the nearby Juntunen site according to Fitting. Fitting suggested a warm-season occupation for both components; the small faunal sample suggested a subsistance pattern balanced between mammals and fish. Corn kernels were also recovered from Feature 10 at the Tamlin site.

The site lies on a small well-protected bay of the north shore of Lake Michigan about twelve kilometers west of the Straits of Mackinac. There are large areas of shallow water in the bay and depths do not exceed five fathoms for one-two kilometers or so beyond the shore.

Reference: Fitting 1974

20MK90 The Pointe Aux Chenes site was surveyed by crews from the Michigan History Division in 1972. Test excavations totalling ca. two square meters were conducted by crews from Michgan Technological University in 1978, and a small uncontrolled surface collection was made in deflated areas.

Based on ceramic materials recovered during the test of 1978, this site represented a multicomponent Laurel Middle Woodland and Mackinac Phase Late Woodland occupation. Later ceramic phases were represented as well. Faunal samples recovered from this site included sturgeon, Canis sp., and Castor sp. (Martin 1980a).

20MK102 The McGreggor site is a multicomponent Middle-Late Woodland occupation at the Straits of Mackinac. An amateur archaeologist made an intensive surface collection here in 1975 and 1976, and she conducted salvage excavations totalling 30 square meters prior to the destruction of the site.

The site represented a Laurel Middle Woodland manifestation with pronounced Late Woodland characteristics in its pottery. Fitting suggested a date of A.D. 300-500 for the proposed McGreggor phase (Fitting 1979:112). In addition some Late Woodland Mackinac and Juntunen ceramic wares were collected and excavated there, but the site was primarily a Middle Woodland occupation. No seasonality

estimates were advanced although the small faunal assemblage included a predominance of mammal bone. Sturgeon and walleye, both spring spawning fish species, were also identified from the site.

The site occupies a stony beach ridge on loamy sand on the northeastern shore of Point St. Ignace facing Lake Huron. The conditions immediately offshore may have changed since prehistoric times because the emplacement of several large piers along the shoreline has probably caused some local sand deposition. But in general water in the six-twelve fathom range is very close to shore at this location. These characteristics also apply to the Sposito/Steiner site complex (20MK61). Reference: Fitting 1979

## Schoolcraft County

20ST1 The Ekdahl-Goudreau site was first reported by George Quimby during surveys conducted by the Chicago Natural History Museum in the 1959-62 seasons. In later years (1965, 1968) Earl Prahl of the University of Michigan Museum of Anthropology and later of the University of Toledo excavated ca. nine square meters of the site and sampled materials exposed on the surface via controlled collecting. The collections from the latter episodes are currently housed at the Michigan State University Museum. Both spring and fall-spawning fish were recovered (Smith 1983).

This site was an extensive multicomponent Middle and Late Woodland camp that appeared to be partially horizontally stratified. Ceramics from the Late Woodland-age components suggested Juntunen phase relationships, while the Middle Woodland occupations were similar to the many Laurel culture and North Bay complex occupations that dotted the north shore of Lake Michigan. The upper stratum appeared to be associated with Juntunen-like ceramics and yielded a radiocarbon date of A.D. 1080 +/- 120 (M-2311) (Crane and Griffin 1972:163). The lower stratum yielded a date of A.D. 660 +/- 130 (M-2312) in stratigraphic association with Laurel-like ceramics (Crane and Griffin op.cit.). 20ST1 is also the site of a nineteenth-twentieth century fishery.

The site lies on a very small sandy embayment of lake Michigan, surrounded by rocky limestone outcrops. Offshore, the topography is varied. East of the site are vast areas of shallow water with maximum depths of three to five fathoms, while to the west, offshore contours are more abrupt, dropping rapidly to depths exceeding ten fathoms. Reference: Binford and Quimby 1963; MSUM Site File; Smith 1983

APPENDIX B

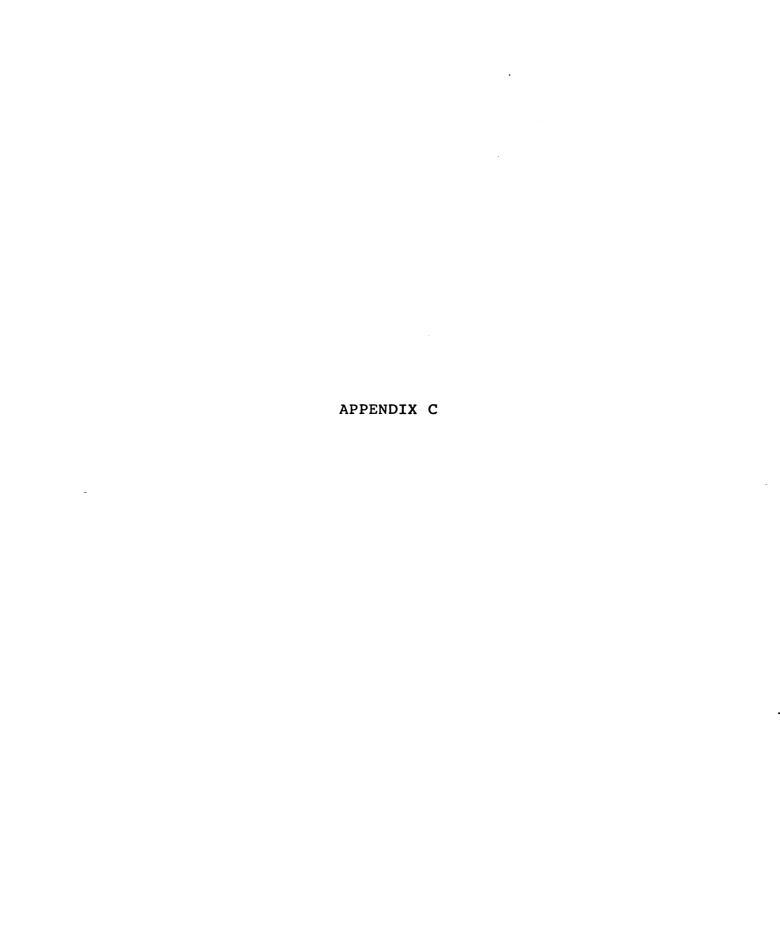
## APPENDIX B

# Codebook of Variables and Values

COL.NO.	NOMINAL/ORDINAL MEASURES
1-3	Case number: 000nnn
4	DUP Location duplicated in the files? 0 = no 1 = yes
5	MULTIC Is this a multicomponent site? 0 = no 1 = yes
6	CCPER Prehistoric cultural-chronological period of this component: 0 = unknown 1 = Middle Woodland 2 = Late Woodland 3 = general Woodland
7	NNOCCS Number of additional components:  0 = none 1 = one 2 = two 3 = three  4 = more than three
8	CERCX This component represents the ceramic culture known as: 1 = Laurel 2 = other Middle Woodland 3 = Mackinac, Pine River, Heins Creek 4 = Bois Blanc 5 = Juntunen 6 = other Late Woodland
9	NHIST Is there an historic component at this location? 0 = no 1 = 17th-18th century 2 = 19th-20th century 3 = 17th-20th century
10	RESINT Intensity of research at this location:  0 = unknown 1 = survey report  2 = survey/collection 3 = controlled collection  4 = excavation (less than 9 m. sq.)  5 = excavation (greater than 9 m. sq.)
11	REGION Intra-regional designator:  1 = Sault Sainte Marie/river  2 = Straits of Mackinac  3 = northern Lake Michigan  4 = northern Lower Michigan  5 = other
12	<pre>USE Hypothetical use of this component: 0 = no estimate 1 = short term 2 = medium 3 = long term</pre>

13	NSEA Hypothetical season of use of this component: 0 = no estimate 1 = spring 2 = spring/summer 3 = summer 4 = summer/fall 5 = fall 6 = spring/summer fall		
18	TECH Technological items present in the collection: 0 = no 1 = yes		
19	FAUNA Faunal sample present in the collection:  0 = no 1 = yes		
20	ASPECT 1 = north 2 = northeast 3 = east 4 = southeast 5 = south 6 = southwest 7 = west 8 = northwest		
21	<pre>NCTO Coastal topography; 1 = small bay 2 = large bay 3 = exposed and straight 4 = peninsula 5 = irregular</pre>		
22	<pre>NWAT Watercourse present; 1 = lake 2 = lake/small stream 3 = lake/large stream 4 = lake/river 5 = lake/river/lake or lake/stream/lake</pre>		
23	ISL Island location: 0 = no 1 = yes		
24	NISL Islands within 10 km. 0 = none 1 = one/two 2 = many		
30	SHOAL Shoal within 10 km. $0 = no 1 = yes$		
	JSHOAL Shoal within 4.8 km. 0 = no 1 = yes		
36	DRPOF Dropoff within 10 km. 0 = no 1 = yes		
	JDRPOF Dropoff within 4.8 km. 0 = no 1 = yes		
51	OTOPO Offshore topography: 0 = unknown 1 = uniform, smooth 2 = medium 3 = varied		
52	NGRAD Offshore gradient: 0 = unknown 1 = very shallow 2 = shallow 3 = varied 4 = sharp 5 = plunging		

COL. NO.	INTERVAL MEASURES
25-29	DISL Distance to closest island (km.)
31-35	DSHOAL Distance to closest shoal (km.)
37-41	DDRPOF Distance to closest dropoff (km.)
42-44	MAXD1 Maximum water depth at a distance of 1 km.
45-47	MAXD2 Maximum water depth at a distance of two km.
48-50	MAXD3 Maximum water depth at a distance of three km.
14-17	RCMEAN Radiocarbon mean for this component (rc years)
71 – 75	AREA Area of water within three km. (km. sq.)
53-58	SHALL Area of water less than three fathoms in depth (pct.)
77-79	THIRTY Area of water less than five fathoms in depth (pct.)
65-70	MED Area of water between three-ten fathoms in depth (pct.)
59-64	DEEP Area of water greater than ten fathoms in depth (pct.)



# APPENDIX C

## Data Tables

Table 34. Case identification

Case	Site	Component
number	number	identifier
101	20MK51	Gyftakis
102	20MK53	Norge Village (lower)
103	20MK61	Sposito (lower)
104	20MK83	Arrowhead Drive
105	20MK86	Ferrier-Tamlin (lower)
106	20MK90	Pointe aux Chenes (lower)
107	20MK102	MacGreggor (lower)
108	20DE3	Burnt Bluff (B-95)
109	20DE4	Summer Island (lower)
110	20DE17	· · · · · · · · · · · · · · · · · · ·
111	20DE17	Winter (upper)
112	20ST1	Ekdahl-Goodreau (lower)
113	20CX19	
114	20CX37	
115	20EM4	Wycamp Creek (Laurel)
116	20EM22	Portage(lower)
117	20EM52	Ft. Michilimackinac (Laurel)
118	20CH2	Naomikong Point (Laurel)
119	20CH4	Sugar Island mound
120	20CH6	Cloudman
<sub>-</sub> 121	20CH30	Big Pine
122	20CH44	Klamerus-Bucht
123	20CH45	
124	20CH51	
125	20CH77	Schoolcraft House (Laurel)
201	20MK1	Juntunen (Mackinac)
202	20MK1	Juntunen (Bois Blanc)
203	20MK1	Juntunen (Juntunen)
204	20MK7	Gros Cap (Juntunen)
205	20MK19	Halberg (Mackinac)
206	20MK19	Halberg (Bois Blanc)
207	20MK22	Scott Point (Mackinac)
208	20MK22	Scott Point (Bois Blanc)
209	20MK22	Scott Point (Juntunen)
210	20MK53	Norge Village (upper)
211	20MK54	Beyer
212	20MK58	White
213	20MK61	Sposito (upper)
214	20MK82	Marquette Mission (Juntunen/Oneota)
215	20MK86	Ferrier-Tamlin (upper)

## Table 34 (cont'd.).

```
216
            20MK90
                      Pointe aux Chenes (upper)
 217
            20MK102
                      MacGreggor (upper)
            20DE1
 218
                      Fayette
 219
            20DE4
                      Summer Island (upper)
 220
            20DE7
                      Port Bar
                      Ekdahl-Goodreau (upper)
 221
            20ST1
 222
            20CX9
                      Saint James #1
            20CX18
 223
                     O'Neill (Occ. 3)
 224
            20CX18
                     O'Neill (Occ. 2)
                     Pine River Channel (Mackinac)
 225
            20CX19
 226
            20CX23
                     Mt. McSauba
 227
            20CX26
                     Cable Bay
                     Martin Point
 228
            20CX27
 229
            20CX33
                     Saint James #3
 230
           20CX38
                     Charlevoix City Park
 231
                     Wood (Mackinac)
           20CX40
 231
           20CX40
                     Wood (Juntunen)
                     Saint James #2
 233
           20CX59
 234
           20EM4
                     Wycamp Creek (Mackinac)
 235
                     Wycamp Creek (Juntunen/Oneota)
           20EM4
 236
           20EM22
                     Portage (Mackinac/upper)
 237
           20EM25
                     Zuber
 238
           20EM40
                     Johnston
 239
           20EM51
                     MCS-4
240
                     Naomikong Point (upper)
           20CH2
241
           20CH27
                     Brown Fishery
242
                     Pendill's Bay
           20CH32
243
                     Albany Harbor
           20CH43
244
           20CH46
                     Slater
-245
           20CH51
                     Fort Brady (upper)
246
           20CH77
                     Schoolcraft House (upper)
301
           20DE2
                     Puffy Bay
302
           20DE8
                     Janowski
303
           20DE9
                     Sac Bay
304
                     Fairport Store
           20DE10
305
           20DE11
                     Point Detour Bay
306
           20DE51
                     Nahma
307
           20CX2
                     Burgess
308
           20CX39
                     NAHM
                     Goodhart
309
           20EM2
310
           20EM19
                     Wauqoshance
           20EM35
                     Foster Otto I
311
                     Silver Creek
312
           20CH29
313
           20CH34
                     Charlotte River
```

Table 35. Case members, TEMPDATA file

Case	Site	Component
number	number	identifier
101	20MK51	Gyftakis
104	20MK83	Arrowhead Drive
108	20DE3	Burnt Bluff (B-95)
109	20DE4	Summer Island
110	20DE17	Winter (lower)
111	20DE17	Winter (upper)
116	20EM22	Portage
118	20CH2	Naomikong Point
119	20CH4	Sugar Island
120	20CH6	Cloudman
121	20CH30	Big Pine
122	20CH44	Klamerus-Bucht
123	20CH45	West Harbor
124	20CH51	Fort Brady
201	20MK1	Juntunen (Mackinac)
201		Juntunen (Bois Blanc)
202	20MK1 20MK1	Juntunen (Juntunen)
203	20MK7	Gros Cap
204	20MK7 20MK19	Halberg (Mackinac)
205	20MK19	Halberg (Bois Blanc)
206 207	20MK19 20MK22	Scott Point (Mackinac)
207	20MK22	Scott Point (Bois Blanc)
208	20MK22	Scott Point (Juntunen)
211	20MK54	Beyer
212	20MK54	White
212	20MK82	Marquette Mission (Juntunen/Oneota)
- 21 <del>4</del> - 218	20MR02 20DE1	Fayette
220	20DE7	Port Bar
222	20CX9	Saint James #1
223	20CX18	
224	20CX18	_
226	20CX23	Mt. McSauba
227	20CX26	Cable Bay
228	20CX27	Martin Point
229	20CX33	Saint James #3
230	20CX38	Charlevoix City Park
233	20CX59	Saint James 32
237	20EM25	Zuber
238	20EM40	Johnston
241	20CH27	Brown Fishery
242	20CH32	Pendill's Bay
243	20CH43	Albany Harbor
244	20CH46	Slater
		- cc -
301	20DE2	Puffy Bay
302	20DE8	Janowski
303	20DE9	SacBay
304	20DE10	Fairport Store

Table 35 (cont'd.).

305	20DE11	Doint Dataum Dan
		Point Detour Bay
306	20DE51	Nahma
307	20CX2	Burgess
308	20CX39	NAHM
<b>309</b>	20EM2	Goodhart
310	20EM19	Waugoshance
311	20EM35	Foster Otto I
312	20CH29	Silver Creek
313	20CH34	Charlotte River

Table 36. Case members, LOCDATA file

Case	Site	Component	
number	number	identifier	Age
101	20MK51	Gyftakis/Marquette	mixed
102	20MK53	Norge Village	${\tt mixed}$
103	20MK61	Sposito/Steiner	mixed
104	20MK1/83	Juntunen/Arrowhead	mixed
105	20MK86	Ferrier-Tamlin	mixed
106	20MK90	Pointe aux Chenes	mixed
107	20MK102	MacGreggor	mixed
108	20DE3	Burnt Bluff	MW
109	20DE4	Summer Island	mixed
110	20DE17	Winter	MW
112	20ST1	Ekdahl-Goodreau	mixed
113	20CX19	Pine River Channel	MW
114	20cX37/40	Wood/Fish 'n' Chips	mixed
115	20EM4	Wycamp Creek	mixed
116	20EM22	Portage	mixed
117	20EM51/52	MCS-4/Michilimackinac	mixed
118	20CH2	Naomikong Point	${\tt mixed}$
119	20CH4	Sugar Island mound	MW
120	20CH6	Cloudman	MW
121	20CH30	Big Pine	MW
123	20cH45	West Harbor	MW
124	20cH51	Fort Brady	mixed
125	20CH77	Schoolcraft House	mixed
204	20MK <b>7</b>	Gros Cap	LW
204	20MK19	Halberg	LW
203	20MK22	Scott Point	LW
- 20 <i>1</i> - 211	20MK54	Beyer	LW
212	20MK54 20MK58	White	LW
212	20MK30 20DE1	Fayette	LW
210	20DE7 20DE7/8	Port Bar/Janowski	LW
222	20CX9/33/59	Saint James #1,2,3	LW
222	20CX3/33/33	O'Neill	LW
225	20CX10	Mt. McSauba	LW
227	20CX26	Cable Bay	LW
228	20CX27	Martin Point	LW
230	20CX27 20CX38	Charlevoix City Park	LW
237	20EM25	Zuber	LW
241	20EM23 20CH27	Brown Fishery	LW
242	20CH32	Pendill's Bay	LW
242	20CH43/46	Albany Harbor/Slater	LW
306	20DE51	Nahma	LW
307	20CX2	Burgess	LW
309	20EM2	Goodhart	LW
310	20EM2 20EM19	Waugoshance	LW
311	20EM35	Foster Otto I	LW
J 1 1	201133		

Table 37. FISHDATA data file

0101121152060170012110206.50104.80102.5001806009523026.04020.44053.5107.683030 0201111152009999112110105.50103.30101.0007509519023034.42038.08027.4912.843040 0800102053110375117310106.40102.40100.1006706306025009.14035.41055.4416.631009 0901138153220160112111204.80105.90102.0004304712733031.84016.62051.5217.683043 1001111053220150116130017.70017.10010.0000601802111088.27000.00011.7209.891100 112011111033000160016110030.00007.50000.5005006008623015.20013.02071.7619.733064 11701141152000450118410211.00105.40100.8007009912423022.27051.57026.1516.523031 11900101021000500002321101.30001.30106.0000400902411098.56000.00001.4310.471100 2100101021100400001310106.80007.00100.7507811414014012.26054.78032.9411.171025 2200101021100430008141200.90007.50007.0000301001711100.00000.00000.0004.331100 20211224052361070112111200.75105.60101.4002110019023033.44036.00030.6017.970045 20311225052361330112111200.75105.60101.4002110019023033.44036.00030.6017.970045 20401216142061400116150105.00108.00003.2001803206212035.66002.88061.4409.002078 20501213052100835014110202.40101.70102.0001706016833038.07022.46039.4515.890056 0301121052060400012110204.20102.00100.3009717118035024.02046.94029.0318.153040 0401141052000050002111200.75105.60101.4002110019023033.44036.00030.6017.973045 0501111053130430017110102.00105.40102.0001803604812034.40000.00065.5907.473070 0601111043019999016110103.80104.00007.2001602603111036.10000.00063.8910.473098 0701121052120400012110204.20102.00100.3009717118035024.02046.94029.0318.153040 11111111053220250116130017.70017.10010.0000601802111088.27000.00011.7209.891100 2401111351000430117540200.50101.75003.2000502104133999.99000.00999.9906.683099 20101223052360835112111200.75105.60101.4002110019023033.44036.00030.6017.970045 11400101044109999011310038.50018.70000.5007713919915009.84068.93021.2214.423014 11501141054240630018150112.50010.50001.2502206008913033.78020.63045.5810.423057 11601111054100120117210058.10036.00002.0002710414613024.79027.27047.9308.473053 2300101021100400006111201.00101.70100.8002903804533033.33000.00066.6619.081077 2501111341100300007540202.00005.00001.5002102703033999.99000.00999.9905.283100 3.3613.9 214052101070014110202.40101.70102.0001706016833038.07022.46039 .50006.5002004006033029.77001.86068 223053360850116120115,60109 20611 0701

Table 37 (cont'd.).

22901216123001455004211103.10105.00100.0705505906433044.75001.02054.2214.680059 23000207054001350001250037.00017.00001.0007913017714028.09034.29037.6009.362032 23101213054101020011120038.50018.70000.5007713919915009.84068.93021.2214.420015 20811224053361050116120115.60109.50006.5002004006033029.77001.86068.3613.942059 20911225053361250116120115.60109.50006.5002004006033029.77001.86068.3613.942059 21011216152001640112110105.50103.30101.0007509519023034.42038.08027.4912.842041 21401216152171200112110206.50104.80102.5001806009523026.04020.44053.5107.680051 21511217053131050017110102.00105.40102.0001803604812034.40000.00065.5907.470070 21611217043019999016110103.80104.00007.2001602603111036.10000.00063.8910.470098 22111215033001330016110030.00007.50000.500500608623015.20013.02071.7619.730064 22200206023101330004211103.10105.00100.0705505906433044.75001.02054.2214.680059 22411226154231555011130101.00100.80100.9002306016032042.80023.80033.3911.050060 22601233044009999017310030.00013.80100.8006015139933016.61048.28035.1015.472028 22700203023100720005151022.50102.00002.0003508030023025.04033.33041.6110.262063 22800204023101130003311018.00009.00102.0001814023022030.43059.85009.7813.472039 23211215054101510011120038.50018.70000.5007713919915009.84068.93021.2214.420015 21311223052060600012110204.20102.00100.3009717118035024.02046.94029.0318.152040 21711227052121000012110204.20102.00100.3009717118035024.02046.94029.0318.150040 21800200023009999007110103.00104.40100.3004505005224017.19000.00082.8011.630030 21911236153231290112111204.80105.90102.0004304712733031.84016.62051.5217.680043 22301225154211455111130101.00100.80100.9002306016032042.80023.80033.3911.050060 22511233054221110111250037.00017.00001.0007913017714028.09034.29037.6009.360044 23411243054241220018150112.50010.50001.2502206008913033.78020.63045.5810.420057 23511246054241650018150112.50010.50001.2502206008913033.78020.63045.5810.420057 21101215152111270012410104.00101.40100.2509717119035023.66044.69031.6419.780037 21200205045101330004240108.20014.50009.5001302002211078.63000.00021.3607.632100 22000203023100850007210104.00105.50101.0004004805124011.24000.00088.7508.890031 23300200023101000004211103.10105.00100.0705505906433044.75001.02054.2214.68005 23611233054101000117210058.19036.00002.0002710414613024.79027.27047.9308. 3701214224111070107310109.00108.80102.4003120026724029.26039.52031.2111. 60109. 20811224053361050116120115.

Table 37 (cont'd.).

24100206025001300003310119.00120.00101.1010013020015020.90057.27021.8111.000032 24200203025000720001320109.50009.00100.7509313015714013.84058.91027.2311.052025 24300204025101130006110200.10102.00101.0004009010533022.85040.05037.0813.430039 24511213351000835117540200.50101.75003.2000502104133999.99000.00999.9906.680099 24611215341101300007540202.00005.00001.5002102703033999.99000.00999.9905.280100 30400300023109999007310101.40104.80103.5000602503512082.02000.00017.9712.570093 30600300023009999005340108.00109.50007.2001502002711091.25000.00008.7413.842100 30700300024009999001110038.00016.50002.0007319024014018.59062.56018.8411.942020 3901243142000800018410211.00105.40100.8007009912423022.27051.57026.1516.520032 24011237251129999112110105.10105.30111.0001202002711085.71000.00014.2813.300100 24401215225101300005110200.60101.00101.0003709010633025.68040.16034.1413.120043 0100300023009999001120105.50107.50102.5001502204312058.98000.00041.0111.680076 30200300023109999007210104.00105.50101.0004004805124011.24000.00088.7508.890031 30300300023179999006110104.10104.20101.5002006806733020.96043.04035.9909.780036 30800300024009999017120100.50100.90003.5001604820012040.15014.80045.0313.100063 31000300052109999008110105.40112.40101.1006007208423033.78020.63045.5810.420044 31101310224009999008310050.00031.00000.7010513717214011.62055.52032.8412.212022 31200300025009999002330106.50106.50119.5000701201411100.00000.00000.0009.000100 30900300124009999007310012.80103.20002.0007023018434018.31059.33022.3514.362027 31059. 23801210224009999007310012.80103.20002.0007021318434018.

Table 38. TEMPDATA data file

20101223052360835112111200.75105.60101.4002110019023033.44036.00030.6017.970045 20211224052361070112111200.75105.60101.4002110019023033.44036.00030.6017.970045 20311225052361330112111200.75105.60101.4002110019023033.44036.00030.6017.970045 20401216142061400116150105.00108.00003.2001803206212035.66002.88061.4409.002078 5152111270012410104.00101.40100.2509717119035023.66044.69031.6419.780037 504510133000424010R.20014. 2401111351000430117540200.50101.75003.2000502104133999.99000.00999.9906.683099 20501213052100835014110202.40101.70102.0001706016833038.07022.46039.4515.890056 20701223053360850116120115.60109.50006.5002004006033029.77001.86068.3613.942059 21200205045101330004240108.20014.50009.5001302002211078.63000.00021.3607.632100 2140121615217120011210206.50104.80102.5001806009523026.04020.44053.5107.680051 20011111021100400007141201.90007.50007.0000301001811100.00000.00000.0004.331100 2100101021100400001310106.80007.00100.7507811414014012.26054.78032.9411.171025 2200101021100430008141200.90007.50007.0000301001711100.00000.00000.0004.331100 2300101021100400006111201.00101.70100.8002903804533033.33000.00066.6619.081077 0800102053110375117310106.40102.40100.1006706306025009.14035.41055.4416.631009 1001111053220150116130017.70017.10010.0000601802111088.27000.00011.7209.891100 1001111053220150116130017.70017.10010.0000601802111088.27000.00011.7209.891100 800200023009999007110103.00104.40100.3004505005224017.19000.00082.8011.630030 0901138153220160112111204.80105.90102.0004304712733031.84016.62051.5217.683043 20611214052101070014110202.40101.70102.0001706016833038.07022.46039.4515.89005 20701223053360850116120115.60109.50006.5002004006033029.77001.86068.3613.94205 20701223053360850116120115.60109.50006.5002004006033029.77001.86068.3613. 0401141052000050002111200.75105.60101.4002110019023033.44036.00030.6017. 1601111054100120117210058.10036.00002.0002710414613024.79027.27047.9308. 50104.80102. 0101121152060170012110206. 2110121

23801210224009999007310012.80103.20002.0007021318434018.31059.33022.3514.36002724100206025001300003310119.00120.00101.1010013020015020.90057.27021.8111.0000322420020302500720001320109.50009.00100.7509313015714013.84058.91027.2311.052025243002040204025101130006110200.10102.00101.0004009010533022.85040.05037.0813.430039 22000203023100850007210104.00105.50101.0004004805124011.24000.00088.7508.890031 
22200206023101330004211103.10105.00100.0705505906433044.75001.02054.2214.680059 
22301225154211455111130101.00100.80100.9002306016032042.80023.80033.3911.050060 
22411226154231555011130101.00100.80100.9002306016032042.80023.80033.3911.050060 30200300023109999007210104.00105.50101.0004004805124011.24000.00088.7508.890031 30300300023179999006110104.10104.20101.5002006806733020.96043.04035.9909.780036 30400300023109999007310101.40104.80103.5000602503512082.02000.00017.9712.570093 24401215225101300005110200.60101.00101.0003709010633025.68040.16034.1413.120043 30100300023009999001120105.50107.50102.5001502204312058.98000.00041.0111.680076 23701214224111070107310109.00108.80102.4003120026724029.26039.52031.2111.312048 22601233044009999017310030.00013.80100.8006015139933016.61048.28035.1015.472028 22700203023100720005151022.50102.00002.0003508030023025.04033.33041.6110.262063 30500200043109999006110103.30106.50102.0002003105922056.64000.00043.3513.840075 0600300023009999005340108.00109.50007.2001502002711091.25000.00008.7413.842100 30700300024009999001110038.00016.50002.0007319024014018.59062.56018.8411.942020 30800300024009999017120100.50100.90003.5001604820012040.15014.80045.0313.100063 30900300124009999007310012.80103.20002.0007023018434018.31059.33022.3514.362027 31000300052109999008110105.40112.40101.1006007208423033.78020.63045.5810.420044 31101310224009999008310050.00031.00000.7010513717214011.62055.52032.8412.212022 01310121109999002540201.50011.50002.8000400600611100.00000.00000.0006.7701 31200300025009999002330106.50106.50119.5000701201411100.00000.00000.0000.

(cont'd.).

38

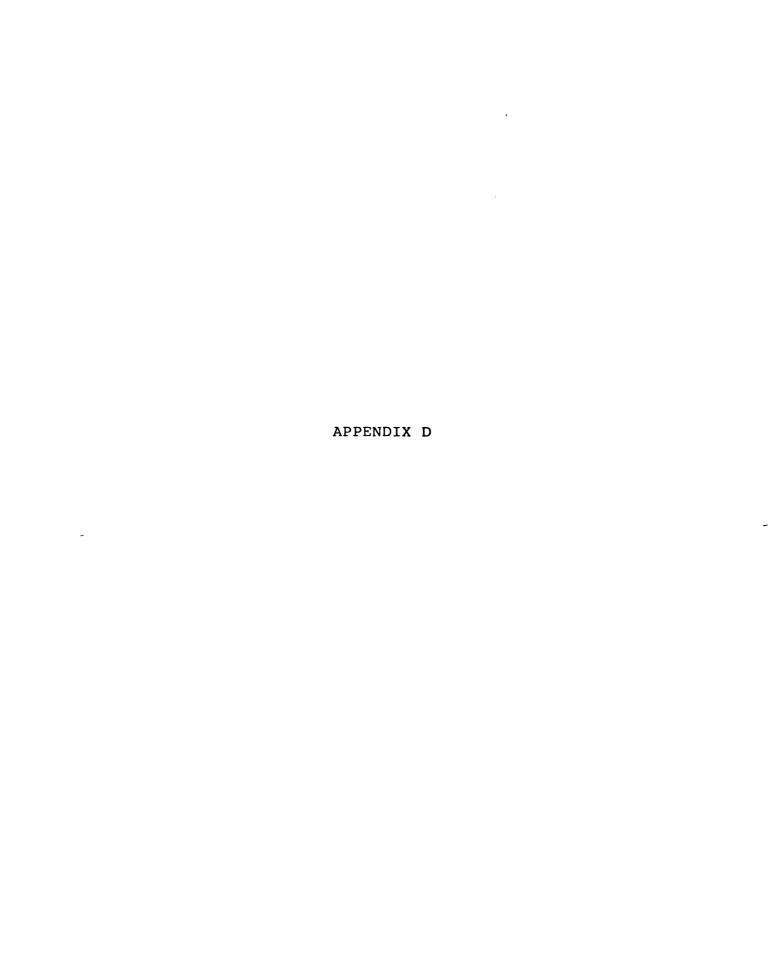
rable

Table 39. LOCDATA data file

0800102053110375117310106.40102.40100.1006706306025009.14035.41055.4416.631009 0901138153220160112111204.80105.90102.0004304712733031.84016.62051.5217.683043 10011111053220150116130017.70017.10010.0000601802111088.27000.00011.7209.891100 114001010044109999011310038.50018.70000.5007713919915009.84068.93021.2214.423014 11501141054240630018150112.50010.50001.2502206008913033.78020.63045.5810.423057 11601111054100120117210058.10036.00002.0002710414613024.79027.27047.9308.473053 22000203023100850007210104.00105.50101.0004004805124011.24000.00088.7508.890031 22200206023101330004211103.10105.00100.0705505906433044.75001.02054.2214.680059 11801131251320430112110105.10105.30111.0001202002711085.71000.00014.2813.303100 2401111351000430117540200.50101.75003.2000502104133999.99000.00999.9906.683099 20401216142061400116150105.00108.00003.2001803206212035.66002.88061.4409.002078 20501213052100835014110202.40101.70102.0001706016833038.07022.46039.4515.890056 20701223053360850116120115.60109.50006.5002004006033029.77001.86068.3613.942059 21800200023009999007110103.00104.40100.3004505005224017.19000.00082.8011.630030 0301121052060400012110204.20102.00100.3009717118035024.02046.94029.0318.153040 0401141052000050002111200.75105.60101.4002110019023033.44036.00030.6017.973045 0501111053130430017110102.00105.40102.0001803604812034.40000.00065.5907.473070 0601111043019999016110103.80104.00007.2001602603111036.10000.00063.8910.473098 0701121052120400012110204.20102.00100.3009717118035024.02046.94029.0318.153040 11201111033000160016110030.00007.50000.5005006008623015.20013.02071.7619.733064 1301131054220600111250037.00017.00001.0007913017714028.09034.29037.6009.363044 1701141152000450118410211.00105.40100.8007009912423022.27051.57026.1516.523031 2001111021100400007141201.90007.50007.0000301001811100.00000.00000.0004.331100 2100101021100400001310106.80007.00100.7507811414014012.26054.78032.9411.171025 2300101021100400006111201.00101.70100.8002903804533033.33000.00066.6619.081077 2501111341100300007540202.00005.00001.5002102703033999.99000.00999.9905.283100 21200205045101330004240108.20014.50009.5001302002211078.63000.00021.3607.632100 21101215152111270012410104.00101.40100.2509717119035023.66044.69031.6419.78 0101121152060170012110206.50104.80102.

Table 39 (cont'd.).

11130101,00100,80100,9002306016032042,80023,80033,3911,050060	472028	262063	472039	362032	312048	000032	052025	05037.0813.430039	842100	362027	420044	.8412.212022
,3911.	,1015.	,6110.	.85009.7813.47203	.6009	,2111.	.8111.	,2311.	.0813.	.7413.	.3514.	.5810.	,8412.
80033.	28035.	33041,	85009.	29037.	52031,	27021,	91027,	05037	00008	33022.	63045,	.52032.
80023.	61048.	04033.	43059.	09034.	26039.	90057.	84058.	85040.	25000.	31059.	78020.	62055.
32042.	8006015139933016.61048.28035.1015.47202	0003508030023025.04033.33041.6	22030.	14028.	4003120026724029.26039.52031.2111.31204	.00120.00101.1010013020015020.90057.27021.8111.00003	14013.	0004009010533022.85040.	2001502002711091.	07310012.80103.20002.0007023018434018.31059.33022.3514.36202	1006007208423033.78020.63045.	08310050.00031.00000.7010513717214011.62055
091090	151399	080300	140230	130177	200267	130200	130157	090105	020027	230184	072084	137172
90023	80060	000350	00018	00019	40031	10100	75093	00040	20015	00000	10060	70105
80100.	3.80100.	102.00002.	.00102.	.00017.00001.	.00108.80102.	00101.	00100.	10102.00101.	50007.	20002.	.40112.40101.	00000
00100.	00013.	50102.	600	00017.	00108.	00120.	50009.	10102.	00109.	80103.	40112.	00031.
30101.	0030.0001	05151022.50	03311018.	50037.	107310109.	03310119.	001320109.	0200	05340108.	0012.	08110105.	0020
5511113	901731	2000515	3000331	000125	010731	000331	000132	000611	900534	900731	900811	90
421145	40099	31007	310113	2300020705400135000	411107	500130	500072	510113	300999	3090030012400999900	210999	40099
223012251542114	3304	302	20402	20705	21422	20902	20302	20402	30002	30012	30005	022
22301	22601	2270020	22800	23000	23701	24100	24200	24300	30600	30608	3100036	31101



#### APPENDIX D

#### Program Listing

### by Kamal Bijlani and Susan Martin

may be undertaken by the user. The input data required by the program are first category represents a specific type of data. Each category is assumed to begin [The documentation presented here represents the information needed to provide input data to the program and as an aid to a few possible modifications which consist of more than one data line. If there are many values to be given on same line, there should be at least one blank to separate the given values. One category may described. The input data description is divided into categories. on a new line in the input data unless stated otherwise. categories are:

- 1. The values of flags for the season output, the cycle output and the period indicates output is required, e.g. 0 1 1 given as the data indicates that the season output is not required, whereas the cycle output and the period output output. A value of 0 indicates that output is not required. A value of are needed.
- the search radius is not constant and may be altered each cycle depending upon the locations of all the lodges. Otherwise, the given radius value is assumed A given value of 0 indicates that to be the constant search radius to be used for the simulation. The radius value can be a real number.
- particular seed is chosen for the random number generation. Presently, the seed index may be 1, 2, or 3 for a choice among three seeds which are declared as constants. Note 6that is the user wants to test a different seed, this can be 3. The seed index is an integer value. Depending upon the given seed index, a easily achieved by changing the value of the seed in the constant declaration, and then choosing the appropriate seed index.

- 4. Specify the total number of cycles (integer) and the period time (integer). The period time is a time unit; after every period time number of cycles a summary of the output is produced.
- Specify the growth rate (real number) before the gill net and after the gill (real number).
- 6. Specify the population mean, population variance, and the population threshhold (all integers).
- Specify the population failure probability density factor before and after the gill net (both reals).
- 8. Indicate the gill net introduction information, i.e. the cycle number (integer) and the cell number (integer). Note that a gill net introduction can The constraint is the declared constant NUM OF GILL INVENS whose values may be altered to reflect a change in the number of gill net introductions required. have as many lines of cycle number and cell number combinations as required.
- should X-position, the Y-position, the spring resource score, the summer resource score, the fall resource score, and the post-gill net resource score (all shobe integers). Note that the number of lines of cell information must be the same as the value of the declared constant MAX\_CELLS, whose values can be information should be on a new line. It must include the following, the Specify information about the location cells. Each location cell changed.
- specified for must be 10. Specify information about the initial locations of the lodges. lodge its unique name (an integer) and its location cell (integer) There is no constraint on the number of initial lodges the simulation.

Note: PASCAL users must replace comment brackets and superscript "2s" with correct PASCAL characters.]

## PROGRAM SIMULATE\_FISH (INPUT,OUTPUT);

```
1000; [an upper limit for the search radius]
                                                                                                 [a lower limit for the search radius]
[gill % used for location movement]
                                                                                                                        [for movement of affine lodge];
                                                                                                                                                                                                                                                                                                                                     INTEGER;
                                                                                                                                                                                                                                                                                                                                                INTEGER;
                                                                                                                                                                                                                                                                                                                                                             INTEGER;
                                                                                                                                                                                                                        INTEGER;
                                                                                                                                                                                                                                    INTEGER;
 21; [max number of location cells]
                                                                                                                                                                                    SEASON, CYCLE, PERIOD, TOTAL);
                                                                                                                                                                                                                                                           REAL;
                                                                                                                                                                                                                                                REAL;
                                                                                                                                                                                                                                                                                     REAL;
                                                                                                                                                                                                                                                                        REAL;
                                                                                                                                                                                                                                                                                                                                   SEASON SCORE
CYCLE SCORE
TOTAL SCORE
PERIOD SCORE
                                                                                                                                                            (SPRING, SUMMER, FALL); (KINS, AFFINES);
                                                                                                                                                                                                                                                           MEAN_DIST
MEAN_TRIES
GILL_KNOWN
END;
                                                                                                                                                                                                                                                RADIUS
                                                                                                                                                                                                                                    LODGES
                                                                                                                                                                                                                        POPUL
                                                                       342965391;
                                                1956987325;
                                                            763481529
                                                                                                                                                                                                            RECORD
                                                                                                                                                                                                                                                                                                                         = RECORD
MAX_CELLS = 2
NUM_OF_GILL_INVENS =
MAX_INTERVALS =
                                                                                                                                                               H H H
                                                    H
                                                               11
                                                                           11
                                                                                                                                                         SEASON_TYPS
RELATIVE_TYPS
PRINT_TYPS
                                                                                  MAX_RAD
MIN_RAD
GILL_PERC
AFF_MOV_PERC
                                                                                                                                                                                                                                                                                                                        RESULTS TYP
                                                                                                                                                                                                           FINAL_INFO
                                                            SEED2
                                                                       SEED3
                                                SEED1
```

END;

```
RESULTS_TYP;
RESULTS_TYP;
                                                                                                                                                                                                                                                                             LODGE_PTR;
                                                                                                                                                                          REL_PTR;
LOC_PTR;
BOOLEAN;
                                                                                                                                                                                                                                                           INTEGER;
INTEGER;
                                                                                                                                                                  INTEGER;
         CYCLE_NUM, CELL_NUM : INTEGER;
                                                                                                                                                                                                                       REAL;
                                                                                                                                                                                                                                REAL;
                                                                                                                                                                                                                                                   REAL;
                                                                                                                                                                                                                                         REAL;
                                                                                                                                                                  NAME, CURRENT LOC
                                                                                                            INTEGER;
                                                      INTEGER;
                                                                LOC_PTR;
                                                                                                                                                                         KIN, AFFINE
LOCS KNOWN
GILL STATUS
NO OF TRIES
NO OF SUCC
SEASON DIST
CYCLE DIST
PERIOD DIST
TOTAL DIST
                                                                                                                                                                                                                                                                  BEST_CELL
NEXT
                                                                                                           FAMILY
                                   LOC_REC;
                                                                                         = REL_INFO;
REL_INFO = RECORD;
                                                              NEXT
                                                                                                                    NEXT
END;
                 END;
                                                                        END;
                                                      LOC
                                                                                                                                                      LODGE = RECORD;
= RECORD
                                                                                                                                             = LODGE;
                                      11
GILL INVENS
                                                                                                                                              LODGE-PTR
                                                                                         REL_PTR
                                   LOC_PTR
```

POT\_CELLS\_INFO = RECORD

```
RESULTS TYP;
RESULTS TYP;
OCC PTR;
BOOLEAN;
                                                                                                    INTEGER;
INTEGER;
                                                                                                                                              INTEGER;
                                                                                                                             INTEGER;
                                                                                                                                      INTEGER;
                                                                                                                                                                                                BOOLEAN;
                                                                                                                                                                                                          BOOLEAN;
                                                                                                                     INTEGER
                                                                                                                                                                                                                                                     INTEGER;
                                                                                                                                                                                                                                                                     INTEGER;
                                                                                                                                                                                                                                                                             TIE_PTR;
                                                          : INTEGER; OCC_PTR;
                                                                                                                                                                                                                                                            REAL;
                                                                                                  CELL NUM
X_POS,Y_POS
SUM_SCORE
SPR_SCORE
GILL_SCORE
GILL_SCORE
SPR_OCCS
SUM_OCCS
FALL_OCCS
CURRENT_OCCS
CVRRENT_OCCS
CYCLE_OCC
PERIOD_OCC
                                                                                                                                                                                                                                                            RADIUS
                                                          LODGE
INTEGER;
REAL;
                                                                                                                                                                                                                                                                    INDEX
                INTEGER;
                                                                 NEXT
END;
                                                                                                                                                                                                                                                                                     END;
                                                                                                                                                                                                                                                    LOCT
                                                                                                                                                                                                                                                                             NEXT
                                                                                                                                                                                                                  END;
                                                                                                                                                                                                                           = TIE_INFO;
TIE_INFO = RECORD
                                       = 2OC_INFO;
OCC_INFO = RECORD
                                                                                           LOC_CELL = RECORD
CELL_NUM
DIST
                RESOURCE
                        END;
                                                                                                                                                                                                                                 TIE_PTR
                                        OCC_PTR
```

```
ORDERED_CELLS = ARRAY [1..MAX_CELLS] OF INTEGER;
NEW_HOMES = ARRAY [1..MAX_CELLS] OF POT_CELLS_INFO;
```

Some of them The following is a list of all the global variables used in the program. use of most of these vairables is evident from their names. explained further.]

```
VAR
```

```
ARRAY [1..NUM_OF_GILL_INVENS] OF GILL_INVEN;
ARRAY [1..MAX_CELLS] OF LOC_CELL;
LODGE_PTR; [head of the linked list of all the lodges]
                                                                                                                                                                                                                                                                      INTEGER;[variable to produce output periodically]
                                                                                                                                                                                                                                                                                                                                                                                                REAL; [indicates % of lodges aware of gillnet]
                                                                                                                                          REAL; [sum of radius for each period] REAL; [sum of the radius for all cycles]
                                                                                                                                                                                                        INTEGER; [used to indicate chosen seed]
FILE OF FINAL INFO; BOOLEAN;
                                                                                                                                                                                                                                       SEASON TYPS;
INTEGER;
                                                                                                                                                                         INTEGER;
                                                              BOOLEAN;
                                                                                                                                                                                        INTEGER;
                                                                                                                                                                                                                                                                                                                                   INTEGER;
                                                                                                                                                                                                                                                                                                                                                  INTEGER;
                               BOOLEAN;
                                                                                                                                                                                                                        INTEGER;
                                                                                                                                                                                                                                                                                                                                                                INTEGER;
                                              BOOLEAN;
                                                                                                                                                                                                                                                                                    INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                 INTEGER;
                                                                                                                                                                                                                                                                                                   REAL;
                                                                                                                           REAL;
                                                                                                                                                                                                                                                                                                                   REAL;
                                                                                                                                                                                                                                                                                                                                                                                                               REAL;
                                                                                                                       SEARCH RADIUS
PERIOD RADIUS
TOTAL RADIUS
TOTAL LODGE SUM
PERIOD LODGE SUM
             SEASON OUTPUT
CYCLE OUTPUT
PERIOD OUTPUT
CONSTANT RADIUS
GILL LIST
LOC TABLE
                                                                                                                                                                                                                                                                                                                    RATE
                                                                                                                                                                                                      SEED_INDEX
CYCLE_COUNTER
CURRENT_SEASON
TOTAL_CYCLES
PERIOD_TIME
                                                                                                                                                                                                                                                                                                                                 POPUL_MEAN_POPUL_VARPOPUL_THRESHCURRENT_POPUL
                                                                                                                                                                                                                                                                                                GROWTH RATE
                                                                                                                                                                                                                                                                                                              GILL GROWTH
                                                                                                                                                                                                                                                                                                                                                                                               MEAN GILLS
FAILURE PDF
                                                                                                         LODGE_HEAD
```

```
OCC_PTR; [head of list to save occupied lodges of cells]
ARRAY [1..MAX_INTERVALS] OF INTEGER; [list of randoms]
                                                                                          ARRAY [1..MAX_CELLS] OF INTEGER;
                                                                                                                                                NOON
GILL_FAIL_PDF : REAL;
TOTAL-LODGES, SEED : INTEGER;
                                                                     INTEGER;
                                                                                                                                                 ĸ
                                                                                                                                                回
                                                                                                                                                G
                                                                TOTAL RANDOMS
CELL OCCUP
                                  HEAD OCC PTR
RANDOM LIST
```

Store the value of the number for a chi-square [Generate a random number. analysis.]

FUNCTION GET\_RANDOM : REAL;

CONST MULTIPLIER = 65533; DIVISOR = 2147483648; VAR

: INTEGER;

LIST NO

BEGIN

```
[Get the maximum distance between the only occupied cell from its
                                                                                                                                                                                                 : INTEGER) : REAL;
                                                                                           := TRUNC (POPUL_VAR + POPUL_MEAN *GET_RANDOM);
                                                                                                                                                                                                                                       (SQR(LOC_TABLE[CELL1].X_SQR (LOC_TABLE[CELL1].Y_
                                                                                                                                                                                                                                                                                                                      ഗ
                                                                                                                                                           U
                                                                                                                                                                                                                                                                                                                      Þ
                                                                                                                                                                                                                                                                                                                      Δ
                                                                                                                                                           Z
                                                                                                                                                                                                 FUNCTION DISTANCE_BETWEEN_CELLS (CELL1, CELL2
                                                                                                                                                                                                                                                                                                                       Ø
                                                                                                                                                           臼
U
                                                                                                                                                           回
                                                                                                                                                            3
 Д
                                                                                                                                                                                                                                                                                                                      X
X
 0
                                        : INTEGER;
                                                                                                                                                            臼
                                                                                                                                                                                                                                       DISTANCE_BETWEEN_CELLS := SQRT (SQR LOC_TABLE[CELL2].X POS) + SQR LOC_TABLE[CELL2].Y POS));
END (* DISTANCE_BETWWEN_CELLS *);
                                                                                                                                                                                                                                                                                                                        Σ
                                                                                                                                                                                                                                                                                                                       H
                                                                                                                                                            回
 回
                                                                                                                                                                                                                                                                                                                       വ
വ
                                                                                                                                                            U
                                                                                                                                                             Z
 Ω
                                                                                                                    END (* GET LODGE POPUL*);
                                        FUNCTION GET LODGE POPUL
 0
                                                                                                                                                             Ø
                                                                                                                                                             H
                                                                                                                                                            I
Q
 H
                                                                                           GET_LODGE_POPUL
 臼
```

END (\* GET RANDOM \*);

: REAL;

FUNCTION GET MAX RADIUS

boundaries]

```
If all lodges
                                                                                                                                                                              TEMP_RAD := DISTANCE_BETWEEN_CELLS (COUNT1, COUNT2);
IF TEMP_RAD > MAX_RADIUS THEN
MAX_RADIUS :=TEMP_RAD;
END (* IF *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                [Get the maximum distance between two occupied cells.
                                                                                   COUNT1 := 1 TO MAX_CELLS DO
FOR COUNT2 := 1 TO MAX_CELLS DO
IF LOC-TABLE[COUNT2].CURRENT_OCCS <>NIL THEN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  are in the same cell, call GET_MAX_RADIUS.]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   FUNCTION GET_LODGE_RADIUS : REAL;
: INTEGER;
                                                                                                                                                                                                                                                                           GET_MAX_RADIUS := MAX_RADIUS;
                                                                                                                                                                                                                                                                                                                                                                                                LODGE
                                                                                                                                                                                                                                                                                                                       END (* GET MAX RADIUS*);
                      TEMP_RAD, MAX_RADIUS
COUNT1, COUNT2
                                                                                                                                                                                                                                                                                                                                                                                                H
                                                                  MAX_RADIUS :=0;
FOR_COUNT1 :=
                                                                                                                                                                                                                                                                                                                                                                                              <u>ម</u>
                                            BEGIN
```

VAR

VAR
COUNT1, COUNT2 : INTEGER;
TEMP\_RAD, LODGE\_RADIUS : REAL;

EGIN

LODGE RADIUS :+ 0;

```
FOR COUNT1:= 1 TO MAX_CELLS DO
FOR COUNT2:= 1 TO MAX_CELLS DO
IF (LOC_TABLE[COUNT2].CURRENT_OCCS <> NIL) THEN BEGIN
(LOC_TABLE[COUNT1].CURRENT_OCCS <> NIL) THEN BEGIN
TEMP_RAD:= DISTANCE_BETWEEN_CELLS (COUNT1, COUNT2);
IF TEMP_RAD > LODGE_RADIUS THEN
LODGE_RADIUS:= TEMP_RAD;
                                                                                                                                                                                                                                                                                             [Update globals to find totals for required periods]
                                                                                                                                                                                                                                                                                                                                         := PERIOD_RADIUS + LODGE_RADIUS;
:= TOTAL_RADIUS + LODGE_RADIUS;
US := LODGE_RADIUS;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                PROCEDURE SAVE_OCC_PTR (VAR AN_OCC : OCC_PTR);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     [Save the occupation pointer in linked list]
                                                                                                                                                                                                                                           LODGE_RADIUS := GET_MAX_RADIUS;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               U
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 := HEAD OCC_PTR;
:= AN_OCC;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              ပ
၀
                                                                                                                                                                                                                    IF LODGE-RADIUS = 0.0 THEN
                                                                                                                                                                                                                                                                                                                                                                                        GET_LODGE_RADIUS := LODGE END (* GET_MAX_RADIUS*);
                                                                                                                                                                        END (* IF *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               ഠ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        END (* SAVE OCC PTR*);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              SAV
                                                                                                                                                                                                                                                                                                                                         PERIOD REDIUS
                                                                                                                                                                                                                                                                                                                                                            TOTAL_RADIUS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               AN OCC .NEXT
HEAD OCC PTR
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  BEGIN
```

```
[The variable HEAD OCC PTR is the head of the linked list of the occupation pointers used inside the location cells. Done to save memory space.]
                                                                                                                                                                                                                                                                                                                                                                                                                                      [Insert a given lodge into its current location. Update statistics in the LOC_TABLE about the lodges occupied by location cell first.]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 PROCEDURE INSERT_LOC_IN_CELL (A_LODGE: LODGE_PTR);
                                                                                                                                                                                                                                                                                                                                                                                    田口
                                                                                                       PROCEDURE NEW_OCC_PTR (VAR AN_OCC : OCC_PTR);
 2
H
                                                                                                                                                                                                                                  := HEAD_OCC_PTR;
PTR := HEAD_OCC_PTR2.NEXT;
 C
                                                                                                                                                                                                                                                                                                                                                                                    1
0
1
U
 0
                                                                                                                                                                              IF HEAD_OCC_PTR = NIL THEN NEW (AN_OCC)
  3
                                                                                                                                                                                                                                                                                                                                                                                     H
                                                                                                                                                                                                                                                                           := NIL;
                                                                                                                                                                                                                                                                                                                                                                                     24
                                                                                                                                                                                                                                                                                                                             (* NEW OCC PTR*);
                                                                                                                                                                                                                                                                                                                                                                                     臼
                                                                                                                                                                                                                                                                                                                                                                                     ഗ
                                                                                                                                                                                                                                                                                                                                                                                     z
                                                                                                                                                                                                                                  AN_OCC := H
HEAD_OCC_PTR
AN_OCC .NEXT :
                                                                                                                                                                                                                                                                                       (* ELSE *);
                                                                                                                                                                                                                 ELSE BEGIN
                                                                                                                                             BEGIN
                                                                                                                                                                                                                                                                                                                              END
                                                                                                                                                                                                                                                                                             END
```

EGIN EGIN

PREV\_OCC, THIS\_OCC : OCC\_PTR;

```
IF_LOC_TABLE[CURRENT_LOC].CURRENT_OCCS = NIL THEN BEGIN
NEW OCC PTR (LOC TABLE[CURRENT_LOC].CURRENT_OCCS);
LOC_TABLE[CURRENT_LOC].CURRENT_OCCS*.LODGE := NAME;
LOC TABLE[CURRENT_LOC].CURRENT_OCCS*.NEXT :=NIL;
END (* IF *)
WITH LOC_TABLE[A_LODGE2.CURRENT_LOC] DO BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                           := CYCLE SCORE +
:= PERIOD SCORE +
:= TOTAL SCORE +
                                                                                                                                                                                                                                                                                                         := CYCLE SCORE + '
:= PERIOD SCORE +
                                                                                                                                                                                                                                                                                                                                                     := TOTAL_SCORE +
                                                                                                                                                                                              := PERIOD SCORE +1
:= TOTAL SCORE + 1
                                                                                                                                                                                                                                                            ELSE IF CURRENT SEASON = SUMMER THEN WITH SUM OCCS DO BEGIN

CYCLE SCORE := CYCLE SCORE +
PERIOD SCORE := PERIOD SCORE +
TOTAL SCORE := TOTAL SCORE +
END (* WITH *)
                                                                                                                                                                           := CYCLE SCORE
                                                                                                                                                                                                                                                                                                                                                                                              ELSE IF CURRENT SEASON = FALL THEN
                                                                                                                              IF CURRENT SEASON = SPRING THEN WITH SPRING OCCS DO BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                               WITH FALL OCCS DO BEGIN
CYCLE SCORE := CYCL
PERIOD SCORE := PERI
TOTAL SCORE := TOTA
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       := NIL;
                                                                                                                                                                          CYCLE SCORE :
PERIOD SCORE :
TOTAL SCORE :
D (* WITH *)
                                                                                 PERIOD OCC := TRUE
                                           EVER OCC := TRUE;
CYCLE OCC := TRUE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 WITH A LODGE DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      PREV_OCC
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     END (* WITH *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 ELSE BEGIN
                                                                                                                                                                                                                                           END
```

```
*
:= LOC_TABLE[CURRENT_LOC].CURRENT_OCCS;
                                                                                                                                                                                                                       Н
                                                                                                                                                                                                                       D
                                                                                                                                                                                                                       Д
                                                                                                                                                                                                                       Z
                                                                                                                                                                                                                                                                                                                                                                                                                          READLN (SEASON_BIT, CYCLE_BIT, PERIOD_BIT);
READLN (RADIUS_VALUE);
                                                                                                                                                                                                                                                                                                                                                         LODGE_PTR;
                                                                                                                                                                                                                      H
                        WHILE THIS OCC <> NIL DO BEGIN
PREV OCC :=THIS OCC;
THIS OCC :=THIS OCC *.NEXT;
END (* WHILE *);
                                                                                                                                                                                                                                                                                                                 INTEGER;
                                                                                                                                                                                                                                                                                                      INTEGER;
                                                                                                                                                                                                                                                                                                                                INTEGER;
                                                                                                                                                                                                                       Z
                                                                                           NEW_OCC_PTR (THIS_OCC);
PREV_OCC^2.NEXT := THIS_OCC;
THIS_OCC^2.LODGE := NAME;
THIS_OCC^2.NEXT := NIL;
END (* ELSE *)
                                                                                                                                                                                                                                                                                                                                            REAL;
                                                                                                                                                                                                                      24
                                                                                                                                                                                                                                                           PROCEDURE READ_AND_PRINT_INPUT;
                                                                                                                                                                                                                      Д
                                                                                                                                                                                                                                                                                                  CELL_COUNT,GILL_COUNT
SEASON_BIT,CYCLE_BIT
PERIOD_BIT
RADIUS_VALUE
A_LODGE
BEGIN_
                                                                                                                                                                          END (* INSERT LOC IN CELL
                                                                                                                                                                                                                     Ω
                                                                                                                                                                                                                     Z
                                                                                                                                                                                                                      Ø
                                                                                                                                                                                                                                                                                                                                                                                                                                                                   IF SEASON_BIT = 0 THEN
                                                                                                                                                                                                                                                                                                                                                                                                HEAD_OCC_PTR := NIL;
READLN;
                                                                                                                                                                                                                      Ω
                                                                                                                                                                                                                     Ø
THIS OCC
                                                                                                                                                                                                                     臼
                                                                                                                                                                                                                     24
                                                                                                                                                                                                                   *
                                                                                                                                                                                                                                                                                      VAR
```

```
The initial values of the program variable are:');
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 ', TOTAL_CYCLES:4);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                            ANTHROPOLOGY SIMULATION PROGRAM
                                                                                                                                                                                                                                                                                                                                    (SEED_INDEX);
(TOTAL_CYCLES, PERIOD_TIME);
(GROWTH_RATE, GILL_GROWTH_RATE);
(POPUL_MEAN, POPUL_VAR, POPUL_THRESH);
(FAILURE_PDF, GILL_FAIL_PDF);
                                                                                                                                                                                                                                                             CONSTANT_RADIUS := TRUE;
SEARCH RADIUS := RADIUS_VALUE;
(* ELSE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                WRITELN ('TOTAL_CYCLES',' ':8,':
                                                                                                                                                PERIOD OUTPUT := FALSE;
ELSE PERIOD_OUTPUT :=TRUE;
SEASON_OUTPUT := FALSE
ELSE SEASON_OUTPUT := TRUE;
                                                   IF CYCLE_BIT = 0 THEN
    CYCLE_OUTPUT := FALSE;
ELSE CYCLE_OUTPUT :=TRUE;
                                                                                                                                                                                                      IF RADIUS_VALUE = 0 THEN
CONSTANT_RADIUS := FALSE;
                                                                                                                               IF PERIOD_BIT = 0 THEN
                                                                                                                                                                                                                                            ELSE BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     WEITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      WRITELN:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                          WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            WRITELN
                                                                                                                                                                                                                                                                                                                                                         READLN
READLN
                                                                                                                                                                                                                                                                                                                                        READLN
                                                                                                                                                                                                                                                                                                                                                                                                                READLN
                                                                                                                                                                                                                                                                                                                                                                                               READLN
                                                                                                                                                                                                                                                                                                   END
```

```
,GROWTH_RATE:6:3);
. ',GILL_GROWTH_RATE:6:3);
,FAILURE_PDF:6:3);
. ',GILL_FAIL_PDF:6:3);
',POPUL_MEAN);
                                                                                                                                                                                                                                                                                                       SEARCH RADIUS: 6:3);
                                                                                                           ', POPUL_THRESH);
 PERIOD TIME:3)
                                                                                          POPUL VAR)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                READLN (CYCLE NUM, CELL NUM);
WRITELN (' ':3,CYCLE NUM:4,' ':9,CELL NUM:3);
                                                                                                                                                                                                                                                                                                                                                                                                                                                FOR GILL_COUNT := 1 TO NUM OF GILL INVENS DO WITH GILL LIST [GILL COUNT] DO BEGIN
                                                                                                                                                                                                                                                                                                                                                                                  (' Gill net will be introduced in :
                                                                                                                                                                                                                                                       'SEED);
                                                                                                                                                                                                                                                                                                                                                                                                                  CELL NO.');
                       ('GILL GROWTH RATE', ': ''
('FAILURE PDF', ': '9', ': 'GILL FAILURE PDF', ': '1
) ('POPULATION MEAN', ': ''
('POPULATION THRESHHOLD: ', 'SEED_INDEX', ': 10', ':
                                                                                                                                                                                                                                                                                IF CONSTANT RADIUS THEN WRITELN ('CONSTANT RADIUS
                                                                                                                                                                                                                                                    WRITELN ('CHOSEN_SEED',' ':9,
'PERIOD TIME'
           GROWTH RATE
                                                                                                                                                                                                                                                                                                                                                                                                                 CYCLE NO.
                                                                                                                                                                   SEED: SEED1;
                                                                                                                                                                                                       SEED:= SEED3;
                                                                                                                                                                                       SEED:= SEED2
                                                                                                                                                        SEED INDEX OF
                                                                                                                                                                                                                      * CASE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                               READLN
                                                                                                                                                                                                                                                                                                                                                                                  WRITELN
                                                                            WRITELNN
                                                                                                                                                                                                                                                                                                                                                                                                  writeln;
                                                                                                                                                                                                                                                                                                                                    WRITELN;
                                                                                                                                                                                                                                                                                                                                                    WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                writeln
                                                                                                                                                                                                                                                                                                                                                                  WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                 WRITELN
                                                                                           WRITELN
                                                                                                                           WRITELN
                                             WRITELN
                                                            WRITELN
                                                                                                          WRITELN
              WRITELN
                             WRITELN
WRITELN
                                                                                                                                                       CASE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                END
```

```
[Read and print all information about the location cells]
```

WRITELN;

```
FALL
                                     1 1 1
                          SUMMER
        WRITELN ("A listing of the location cells is:');
                           SPRING
                          Y POS
                          X POS
                                                                                      SPR SCORE:2, END (* WITH *);
                          ('CELL NO.
                                   -----
                          WRITELN
                                   WRITELN
WRITELN;
                 WRITELN;
```

# [Read and print the names and current locations of all initial lodges]

```
READLN (NAME, CURRENT LOC);
WRITELN (' ':4, NAME:4,' ':16, CURRENT LOC:4);
                                                                     Current Location')
                                Writeln (' The initial lodges are
                                                                                                                    NEW(LODGE_HEAD);
A_LODGE := LODGE_HEAD;
TOTAL_LODGES := 1);
WITH A_LODGE2 DO BEGIN
                                                                   (' Lodge Name
                                                                                         1 1 1
                                                                                                                                                                                                                             (* MITH *);
                                                                                     Writeln
                                                   Writeln;
                                                                   Writeln
Writeln;
                Writeln;
                                                                                                       Writeln;
```

```
PROCEDURE GET_LODGE (LODGE_NAME : INTEGER; VAR A_LODGE : LODGE_PTR);
                                                                                                           NEW (A_LODGE<sup>2</sup>.NEXT);
A_LODGE := A_LODGE<sup>2</sup>.NEXT;
WITH A_LODGE<sup>2</sup> DO BEGIN
READLN (NAME, CURRENT_LOC);
WRITELN (' ':4,NAME:4,' ':16,CURRENT_LOC:4);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        A_LODGE := LODGE_HEAD;
WHILE A-LODGE NEXT.NAME >> LODGE_NAME DO
A_LODGE := A_LODGE NEXT
                                                                                                                                                                                                                                                TOTAL_LODGES := TOAL_LODGES + 1);
A LODGE *.NEXT :+ NIL;
INSERT_LOC_IN_CELL (A_LODGE);
(* WHILE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        [Given a lodge name, get the lodge]
                                                                                                                                                                                                                                                                                                                                                                                                                                       LODG
A_LODGE 2.NEXT := NIL;
INSERT_LOC_IN_CELL (A_LODGE);
                                                                                                                                                                                                                                                                                                                                                                                                                                       H
                                                                                                                                                                                                                                                                                                                                                                                                                                       臼
                                                                                                                                                                                                                                                                                                                                                                 END (* READ INPUT *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   END (* GET LODGE *);
                                                                                                                                                                                                                           END (* WITH *);
                                                                 WIHILE NOT EOF DO
                                                                                           BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              BEGIN
                                                                                                                                                                                                                                                                                                                      END
```

```
FUNCTION NO_COLLISIONS (A_LODGE :LODGE_PTR; REL_NUM : INTEGER; RELATIVE_TYPS) : BOOLEAN;
                                                      [Make sure that the random lodge picked as a relative is not already
                                                                                                                                                                                                                                                                                                                                                                                                     IF TEMP_REL 2. FAMILY = REL NUM THEN NO COLLISIONS := FALSE;
TEMP_REL := TEMP_REL 2.NEXT;
END (* WHILE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       IF TEMP_REL 2. FAMILY = REL_NUM THEN NO_COLLISIONS := FALSE;
 2
0
                                                                                                                                                                                                                                                                                                                                              TEMP_REL := KIN;
WHILE TEMP_REL <> NIL DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              TEMP_REL := AFFINE;
WHILE TEMP_REL <> NIL DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    ELSE IF RELATION = AFFINES THEN
COLLI
                                                                                                                                                                         : REL_PTR;
                                                                                                                                                                                                                                                                                      IF RELATION = KINS THEN WITH A_LODGE DO
                                                                                                                                                                                                                                                NO_COLLISIONS :=TRUE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      WITH A LODGE DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   END (* WITH *)
                                                                        a relative]
0
Z
                                                                                                                                                                     TEMP_REL
                                                                                                                                                                                                                                                                                                                                                                                      BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     BEGIN
                                                                                                                                                                                                                                                                                                                           BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             BEGIN
                                                                                                                                                    VAR
```

```
PROCEDURE ADD_RELATIVE (VAR A_LODGE : LODGE_PTR; RELATION : RELATIVE_TYPS; REL_NAME : INTEGER);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       ELSE IF (RELATION = AFFINES) AND (AFFINE = NIL) THEN BEGIN
                                                                                                                                                                                  [Add the relative to the list of relatives of a lodge]
                                                                                                                               回
                                                                                                                                                                                                                                                                                                                                                                                                           IF (RELATION = KINS) AND (KIN = NIL) THEN
                                                                                                                               Е
                                                                                                                                Ø
TEMP REL := TEMP_REL *.NEXT;
END (* WHILE *);
END (* WITH *)
                                                                                                                               回口
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            := REL_NAME;
:= NIL;
                                                                                                                               æ
                                                                                                                                                                                                                                                                                                                                                                                                                                               NEW(KIN);
KIN<sup>2</sup>.FAMILYY := REL_NAME;
KIN<sup>2</sup>.NEXT := NIL_
                                                                                                                               Ω
                                                                                                                                                                                                                                                                                                : REL_PTR;
                                                                        END (* NO COLLISIONS *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     NEW (AFFINE);
AFFINE * FAMILY
AFFINE * NEXT
END (* ELSE IF *)
                                                                                                                                                                                                                                                                                                                                                                     WITH LODGE DO BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                ELSE BEGIN
                                                                                                                                                                                                                                                                                               PREV, TEMP
                                                                                                                                                                                                                                                                                                                                                                                                                               BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      END
                                                                                                                                                                                                                                                                                                                                     BEGIN
                                                                                                                                                                                                                                                                              VAR
```

```
PROCEDURE GET_RELATIVE (RELAT : RELATIVE_TYPS; VAR A_LODGE : LODGE_PTR);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        Otherwise
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           The other
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     affine.
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    Get a relative for a lodge of the kind desired, i.e. kin or If lodge already has such a relative, ignore the request.
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         get a relative randomly from among all the other lodges. lodge cannot already be a relative.]
                                                                                                                                                                                                                                                                                                                                                                                                                                                             TIVE
                                   ELSE IF RELATION := AFFINES THEN TEMP := AFFINE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                             L A
                                                                                                                                                                                                                                                                  := REL_NAME;
:= NIL;
IF RELATION = KINS THEN
                                                                                                                                                 := TEMP2.NEXT
                                                                                           WHILE TEMP <> NIL DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                               H
                                                                                                                                                                                                                                              TEMP := TEMP .NEXT;
TEMP .FAMILY := RE
TEMP .NEXT := NI
                                                                                                                                                                                                                                                                                                                                                                                                                                                             臼
                                                                                                                                                                                                                                                                                                                                                                (* ADD RELATIVE *);
                                                                                                                                 := TEMP;
                                                                                                                                                                                                                            NEW (TEMP 2.NEXT);
                                                                                                                                                                     END (* WHILE *);
                                                                                                                                                                                                                                                                                                       (* ELSE *)
                   TEMP := KIN
                                                                                                                                                                                                          TEMP := PREV;
                                                                       PREV := NIL;
                                                                                                                                                  TEMP
                                                                                                                                                                                                                                                                                                                      (* HIIM *)
                                                                                                                                 PREV
                                                                                                              BEGIN
                                                                                                                                                                                                                                                                                                        END
                                                                                                                                                                                                                                                                                                                                                                END
                                                                                                                                                                                                                                                                                                                           END
```

LABEL 100;

```
REL_NAME := TRUNC (GET_RANDOM * TOTAL_LODGES + 1);
GET_LODGE (REL_NAME, REL_LODGE);
UNTIL (REL_NAME <> A_LODGE .NAME) AND (NO_COLLISIONS(A_LODGE,REL_NAME,REL1))
AND (NO_COLLISIONS (REL_LODGE,A_LODGE .NAME,REL2));
                                                                                                                                                                                                                                                                                             ELSE IF (RELAT = AFFINES) AND (AFFINE <> NIL) THEN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     GLOBA
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       ADD_RELATIVE (A_LODGE, RELAT, REL_NAME);
ADD_RELATIVE (REL_LODGE, RELAT, A_LODGE 2. NAME);
                                                                                                                                                                                                                                               IF (RELAT = KINS) AND (KIN <> NIL) THEN
GOTO 100
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     臼
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      Ы
                                                                                                             LODGE PTR;
INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      INI
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               END (* GET RELATIVE *);
                                          REL2 = AFFINES;
                                                                                                                                                                                                                                                                                                                  GOTO 100;
                                                                                                                                                                                                                          WITH A LODGE DO
                     REL1 = KINS;
                                                                                                             REL_LODGE
REL_NAME
                                                                                                                                                                                                                                                                                                                                                                  REPEAT
CONST
                                                                                         VAR
```

[Initialize the values of all the global variables. This includes setting all variables to nil, 0 etc., ajd giving a relative (kin

This includes

```
and affine) to each lodge.]

PROCEDURE INITIALIZE GLOBALS

VAR

A LODGE : LODGE PTR;

REL TY : RELATIVE TYPS;

CELL COUNT : = 0;

PERIOD RADIUS := 0;

TOTAL LODGE_SUM := 0;

TOTAL LODGE_SUM := 0;

FOR CELL COUNT : 1 TO MAX CELLS DO

CELL COUNT : 1 TO MAX CELLS DO

CELL COUNT : 1 TO MAX CELLS DO

FOR CELL COUNT : 1 TO MAX CELLS DO

FOR CELL COUNT : 1 TO MAX CELLS DO

FOR CELL COUNT : 1 TO MAX CELLS DO

GEL TOUGE SUM := 0;

A LODGE SUM := 0;

A LOD
```

```
:= GET_LODGE_POPUL;
:= CURRENT_POPUL + PEOPLE;
FALSE;
                                                                                                                                                                                                                                                                          A_LODGE := A_LODGE 3.NEXT;
                                                                                                                                .; ;
0 0
                                                                                                                                                                                                                                                   END (* WITHIN A LODGE *);
                      0 = 0
                                            0 =
                                                                    CURRENT POPUL := CUNITH NO OF TRIES DO
                                                                                                                                                                WITH NO_OF_SUCC_DO BEGIN
              !!
                                                                                                                                                                                        SEASON SCORE
CYCLE SCORE
PERIOD SCORE
TOTAL SCORE
D (* WITH *);
                                                                                                       SEASON SCORE
CYCLE SCORE
PERIOD SCORE
TOTAL SCORE
END (* WITH *);
        SEASON DIST
CYCLE DIST
PERIOD DIST
TOTAL DIST
PEOPLE
GILL STATUS
                                                                                            BEGIN
                                                                                                                                                                                                                                        END
```

FOR CELL\_COUNT := 1 TO MAX CELLS DO
WITH LOC\_TABLE[CELL\_COUNT] DO BEGIN

END (\* WHILE \*);

CYCLE\_OCC := FALSE; PERIOD\_OCC := FALSE; EVER\_OCC := FALSE;

```
[Exchange the contents of the two given positions in the array
POT_CELLS>
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         Н
                                                                                                                                                                                                                                                                                            WITH FALL OCCS DO BEGIN
SEASON SCORE := 0;
CYCLE SCORE := 0;
PERIOD SCORE := 0;
TOTAL SCORE := 0;
END (* WITH *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       F 0
WITH SPR OCCS DO BEGIN

SEASON SCORE := 0;

CYCLE SCORE := 0;

PERIOD SCORE := 0;

TOTAL SCORE := 0;

END (* WITH *);
                                                                                                                                                      WITH SUM OCCS DO BEGIN

SEASON SCORE := 0;

CYCLE SCORE := 0;

PERIOD SCORE := 0;

TOTAL SCORE := 0;

END (* WITH *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                        END (* INIT GLOBALS *);
                                                                                                                                                                                                                                                                                                                                                                                                                                   END (* WITH *);
```

```
PROCEDURE SWAP_POT_CELLS (VAR POT_CELLS : NEW_HOMES;INDEX1,MAX_INDEX:INTEGER);
                                                                                                                                                                                                                                                                                                                                E
L
                                                                                                                                            U
                                            : INTEGER; REAK;
                                                                                                                                                                                                               WITH POT_CELLS[MAX_+INDEX] DO BEGIN_
                                                                                                                                                                                                                                                                                                                                Z
                                                                                                                        := CELL_NUM;
:= RESOURCE;
                                                                                                                                                                                                                                     CELL_NUM := TEMP_CELL;
RESOURCE := TEMP_SCORE;
DIST := TEMP_DIST;
) (* WITH *);
                                                                                                                                                                                                                                                                                                                               臼
                                                                                                  WITH POT_CELLS[INDEX1] DO
                                                                                                                                                                                                                                                                                                                               H
                                                                                                                                                                                                                                                                                                                               0
                                                                                                                                                                                                                                                                                           END (* SWAP POT CELLS *);
                                          TEMP_CELL, TEMP_SCORE TEMP_DIST
                                                                                                                                                                                                                                                                                                                               Д
                                                                                                                                                                                                                                                                                                                               K
                                                                                                                      TEMP_CELL
TEMP_SCORE
TEMP_DIST
CELL_NUM
RESOURCE
                                                                                                                                                                                                                                                                                                                               0
                                                                                                                                                                                          END (* WITH
                                                                                                                                                                                                                                                                      END (* WITH
                                                                                                                                                                                                                                                                                                                               ഗ
                                                                            BEGIN
                                VAR
```

[Use bubble sort to sort the potential cells]

```
(VAR POT_CELLS : NEW_HOMES; VAR
                                                                                                                                                                                                                                            FOR INDEX2 := (INDEX1 + 1) TO NUM_OF_CELLS DO OF_POT_CELLS[INDEX2].RESOURCE > MAX_SCORE THEN
                                                                                                                                                                                                                                                                                          := POT CELLS[INDEX2].RESOURCE;
:= INDEX2;
                                                                                                                                                                                                                                                                                                                                                    SWAP_POT_CELLS (POT_CELLS, INDEX1, MAX_INDEX);
END (* FOR *);
                                                                                                                                                                                                                                                                                                                                                                                                    TO MAX CELLS DO
                                                                                                                                                                                           := POT_CELLS[INDEX1].RESOURCE;
:= INDEX1;
                                                                                                                                                                    := 1 TO (NUM OF CELLS - 1) DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              Ω
                                                                                         INTEGER;
                                                             INTEGER;
                                                                            INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              Ø
                                                                                                         REAL;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              臼
                                                                                                                                                                                                                                                                                                                                                                                                 FOR INDEX1 := (NUM OF CELLS = 1)
POT_CELLS [INDEX1].RESOURCE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               H
                                                                                                                                                                                                                                                                                                                                                                                                                                               END (* SORT POTENTIAL CELLS *);
PROCEDURE SORT_POTENTIAL_CELLS
              : INTEGER);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              Д
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               D
                                                                          TEMP_CELL, TEMP_SCORE
MAX_SCORE, MAX_INDEX
TEMP_DIST
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               H
                                                                                                                                                                                                                                                                                        MAX_SCORE :
MAX_INDEX :
END (* IF *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              回
                                                          INDEX1, INDEX2
             NUM_OF_CELLS
                                                                                                                                                                                              MAX_SCORE MAX_INDEX
                                                                                                                                                                                                                                                                            BEGIN
                                                                                                                                                                   FOR INDEX1
                                                                                                                                      BEGIN
                                                                                                                                                                                  BEGIN
                                             VAR
```

```
: POT_CELLS_INFO;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    resolve the tie on the basis of distance from the cell.
[ Set up the head of the linked list to resolve the location cells tie on basis of distance from location cell.]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      PROCEDURE INSERT_NODE (VAR HEAD_LOC : TOE PTR; POS CELL CELL_INDEX : INTEGER);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                [A location cell with its distance from the lodge is
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         inserted as a node in the linked list created to
                                                                           PROCEDURE SET_UP_HEAD (VAR HEAD_LOC : TIE_PTR);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     团
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     о
О
                                                                                                                                                                                                                                                                       HEAD LOC<sup>2</sup>.RADIUS := MIN RAD;
HEAD LOC<sup>2</sup>.LOCT := 0;
HEAD LOC<sup>2</sup>.NEXT := NIL;
TEMP LOC
TEMP LOC
TEMP LOC<sup>2</sup>.NEXT);
TEMP LOC<sup>2</sup>.RADIUS := MAX_RAD;
TEMP LOC<sup>2</sup>.RADIUS := MAX_RAD;
TEMP LOC<sup>2</sup>.NEXT;
TEMP LOC<sup>2</sup>.NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     24
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       臼
                                                                                                                                                   TEMP_LOC : TIE_PTR;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         (* SET UP HEAD *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       ഗ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       Z
                                                                                                                                                                                                                                                            NEW (HEAD LOC);
                                                                                                                                                                                                             BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         END
                                                                                                                              VAR
```

VAR

```
INDEX := FIRST_INDEX TO LAST_INDEX DO
WHILE LOCATS <> NIL DO BEGIN

IF LOCATS*.LOC = POT_CELLS[INDEX].CELL_NUM THEN
    INSERT_NODE (HEAD_LOC,POT_CELLS[INDEX],INDEX);
    LOCATS*.NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           IF HEAD LOC2.NEXT2.RADIUS <> MAX RAD THEN BEGIN
RESOLVE RADIUS CONFLICT (HEAD LOC, GOOD CELL);
TIE BROKEN := TRUE;
END (* IF *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      IF NOT TIE_BROKEN THEN BEGIN
    SET_UP_HEAD (HEAD LOC);
    FOR_INDEX := FIRST_INDEX TO LAST_INDEX DO
     : TIE PTR;
                                                                                                                                  := CELL INDEX;
                                                                                                                                                                                                                                                                                                                                                                                              := A_LODGE 2. LOCS_KNOWN;
                                                                                                                                                := CELL_NUM;
S := DIST;
:= NIL;
NEW_LOC, PREV_LOC, THIS_LOC
                                                                                                                                                                                                                                          := HEAD_LOC;
:= NIL;
                                                                                WITH POS_CELL DO BEGIN

NEW [NEW_LOC2.INDEX := C

NEW_LOC2.LOCT := CE

NEW_LOC2.RADIUS :=

NEW_LOC2.NEXT := N
                                                                                                                                                                                                                                                                                                                                                SET_UP_HEAD (HEAD_LOC);
TIE_BROKEN :- FALSE;
LOCATS := A LODGE<sup>2</sup>.L(
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    END (* WHILE *);
                                                                                                                                                                                                                                        THIS LOC
PREV LOC
                                                                                                                                                                                                                                                                                                                                                                                                                                         FOR INDEX
                                                                                                                                                                                                                                                                                                        BEGIN
```

```
PROCEDURE SELECT_BEST_CELL (VAR POT_CELLS : NEW_HOMES; VAR A_LODGE : LODGE_PTR; CELL_NUMB : INTEGER; TRY_MORE : BOOLEAN);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                 or more cells have the same resource potential, use procedure RESOLVE_TIE. If the best cell is empty, occupy it. If not, check for relatives in that cell. If kin lives there, occupy it. affine lives there and knows about the gill net, occupy it. If af is ignorant of the gill net, occupy cell based on a random draw with probability of AFF MOV_PERC.

Continue until cell is found. If nothing is found, call the
                                                                                                                                                                                                                                                                                                                                                                                                                                     Select the best cell from the list of potential cells. If two
                                                                                                                                                                                                                                 PROCEDURE GET_MORE_POT_CELLS (VAR POT_CELLS : NEW HOMES;
VAR A_LODGE : LODGE_PTR); FORWARD;
INSERT NODE (HEAD LOC, POT CELLS[INDEX], INDEX);
RESOLVE_RADIUS_CONFLICT (HEAD_LOC, GOOD_CELL);
                                                                                                             SWAP_POT_CELLS (POT_CELLS, FIRST_INDEX, GOOD_CELL);
                                                                                                                                                                                                                                                                                                                                                                                        CELL
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         LODGE PTR;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              OCC_PTR;
REL_PTR;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    INTEGER
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           BOOLEAN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      REAL;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          procedure GET_MORE_POT_CELLS.
                                                                                                                                                                                                                                                                                                                                                     Н
                                                                                                                                                                                                                                                                                                                                                    BES
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   RELAT_FOUND, PLACE FOUND
CELL_INDEX, TIE_INDEX
CELL_OCCS
TEMP_RELT
AFF_LODGE, REL_LODGE
TEMP_RANDOM
                                                                                                                                                                                                                                                                                                                                                    H
                                                                                                                                                                                                                                                                                                                                                  ບ
                                                                                                                                                                        END (* RESOLVE TIE *);
                                                                                                                                                                                                                                                                                                                                                    SELE
                                                  (* IF *);
```

```
CELL_OCCS := LOC_TABLE[POT_CELLS[CELL_INDEX].CELL_NUM].CURRENT_OCCS;
                                                                                                                                                                                                                                                                                                                                                                                                                                                             WHILE (NOT RELAT_FOUND) AND (CELL_OCCS <> NIL) DO BEGIN
TEMP_RELT := A LODGE * KIN;
WHILE (TEMP_RELT <> NIL) AND (NOT RELAT_FOUND) DO BEGIN
IF TEMP_RELT * FAMÍLY = CELL_OCCS * LODGE THEN BEGIN
RELAT_FOUND := TRUE;
PLACE_FOUND := TRUE;
                                                                                                          <= CELL NUMB) DO BEGIN
                                                                                                                                                     TIE_INDEX := CELL_INDEX;
WHILE POT_CELLS[TIE_INDEX].RESOURCE = POT_CELLS[TIE_INDEX
1].RESOURCE DO
                                                                                                                                                                                                                              TIE INDEX := TIE INDEX + 1;
IF TIE INDEX <> CELL INDEX THEN
RESOLVE TIE (POT CELLS, CELL INDEX, TIE INDEX, A LODGE);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       := TEMP RELT'.NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      := A_LODGE 2. AFFINE;
                                                                                                    WHILE (NOT_PLACE_FOUND) AND (CELL_INDEX
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             IF NOT RELAT FOUND THEN TEMP RELT := TEMP
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      END (* WHILE *);
                                                                                                                                                                                                                                                                                                                                                                                      IF CELL OCCS = NIL THEN PLACE FOUND := TRUE
                            := FALSE;
                                                     := FALSE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       TEMP RELT
CELL_INDEX : PLACE FOUND RELAT_FOUND
```

BEGIN

```
(TEMP_RELT <> NIL) AND (NOT RELAT_FOUND) DO BEGIN
(TEMP_RELT<sup>2</sup>.FAMILY = CELL_OCCS<sup>2</sup>.LODGE) THEN BEGIN
GET_LODGE(TEMP_RELT<sup>2</sup>.FAMILY,AFF_LODGE);
TEMP_RANDOM := GET_RANDOM;
IF (AFF_LODGE<sup>2</sup>.GILL_STATUS) OR
(TEMP_RANDOM < AFF_MOV_PERC) THEN BEGIN
PLACE_FOUND := TRUE;
RELAT_FOUND := TRUE;
                                                                                                                                                                                                                                                                                                                                                                                             .DIST:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           BEST_CELL := POT_CELLS[CELL_INDEX].CELL_NUM;
                                                                                                                                                                                                                                                                                                                                                                                           POT_CELLS[CELL_INDEX] .DIS'
CYCLE_DIST + SEASON_DIST;
PERIOD_DIST + SEASON_DIST;
TOTAL_DIST + SEASON_DIST;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  := SEASON SCORE +
:= CYCLE SCORE +
:= PERIOD SCORE
                                                                                                                                                                                                                                  := TEMP RELT : NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              - TOTAL SCORE
                                                                                                                                                                                                                                                                                            := CELL_OCCS .NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              WITH NO OF SUCC DO BEGIN
SEASON SCORE := SE
CYCLE SCORE := CY
PERIOD SCORE := TOTAL SCORE := TC
END (* WITH *);
                                                                                                                                                                                                                IF NOT RELAT-FOUND THEN
                                                                                                                                                       *(*
                                                                                                                                                                                                                                                                                                                                                                          BEGIN
                                                                                                                                                                                                                                                                     *(*
                                                                                                                                                                                                                                   TEMP_RELT
                                                                                                                                                                                                                                                                                                                                                                                                                 11
                                                                                                                                                                                                                                                                                                                                                                      IF PLACE_FOUND THEN
                                                                                                                                                                                                                                                                                                                                                                                               11
                                                                                                                                                                                                                                                                                                                                                   WITH A LODGE DO BEGIN
                                                                                                                                                                                                                                                                        END (* WHILE
                                                                                                                                                                                                                                                                                                                                                                                           SEASON DIST
CYCLE DIST
PERIOD DIST
                                                                                                                                                        END
(* IF
                                                                                                                                                                                                                                                                                                                                                                                                                                                 TOTAL DIST
                                                                                                                                                                                                                                                                                          CELL OCCS
                                                                                                                                                                          END
   WHILE
```

```
IF RELAT_FOUND AND (CURRENT_SEASON = FALL) THEN BEGIN
GET LODGE (TEMP_RELT*.FAMILY, REL_LODGE);
IF REL_LODGE**.GILL_STATUS THEN
A_LODGE*.GILL_STATUS := TRUE;
(* IF *);
                                                                                                                                                                                                                                                                                                                                                                                              回
                                                                                                                                                                                                                                                                             IF TRY_MORE THEN
GET_MORE_POT_CELLS,A_LODGE)
             := SEASON SCORE
:= CYCLE SCORE
:= PERIOD SCORE
:= TOTAL SCORE
                                                                                                                                                                                                                                                                                                                                                                                              H
                                                                                                                                                                                                                                                                                                                                                                                              0
                                                                                                                                                                                                                                                                                                                                                                                              Д
  DO BEGIN
                                                                                                                                                                                                           CELL_INDEX :+ CELL_INDEX + 1;
                             11
                                             H
                                                                                                                                                                                                                                                                                                                                                                                              24
                                                                                                                                                                                                                                                                 IF NOT PLACE_FOUND THEN BEGIN
                                                                                                                                                                                                                                                                                                                                                  (* SELECT_BEST_CELL *);
                                                                                                                                                                                                                                                                                                                                                                                              0
                                                                                · ( *
WITH NO OF TRIES I
SEASON SCORE
CYCLE SCORE
PERIOD SCORE
TOTAL SCORE
                                                                                                                                                                                                                                                                                                                                                                                              Σ
                                                                                                           *(* HIIM *)
                                                                                  HTIM *)
                                                                                                                                                                                                                                                                                                                                                                                             H
                                                                                                                                                                                                                                                                                                                                                                                             臼
                                                                                                                                                                                                                                                                                                                                                                                            ტ
                                                                                                                                                                                                                                      END (* WHILE *);
                                                                                   END
                                                                                                                                                                                                                                                                                                                       (* IF *);
                                                                                                                                                                                                                                                                                                          ELSE
                                                                                                                                                                                                                                                                                                                                                   END
                                                                                                                                                                                                                                                                                                                        END
```

END (\* IF

the current search radius, get the rest of the location cells

Since the lodge could not pick any location cell within

```
IF CURRENT_SEASON + SPRIUNG THEN
RESOURCE := LOC_TABLE[CELL_COUNT].SPR_SCORE
ELSE IF CURRENT_SEASON = SUMMER THEN
RESOURCE := LOC_TABLE[CELL_COUNT].SUM_SCORE
ELSE IF (CURRENT_SEASON = FALL) AND A_LODGE *.GILL_STATUS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    RESOURCE := LOC_TABLE[CELL_COUNT].GILL_SCORE
ELSE IF CURRENT_SEASON = FALL THEN
RESOURCE := LOC_TABLE[CELL_COUNT].FALL_SCORE
ELSE WRITELN ('ERROR IN FINDING LOCATIONS');
                                                                                                                                                                                                                                                                                                   NUM_OF_CELLS := 0;

FOR_CELL_COUNT := 1 TO MAX_CELLS DO BEGIN
CELLS_DIST := DISTANCE_BETWEEN CELLS
(CELL_COUNT,A_LODGE<sup>2</sup>.CURRENT_LOC̄);
IF CELLS_DIST >= SEARCH_RADIUS THEN BEGIN
NUM_OF_CELLS := NUM_OF_CELLS + 1;
WITH POT_CELLS[NUM_OF_CELLS] DO BEGIN
CELL_NUM := CELLS_DIST;
DIST_ := CELLS_DIST;
                                                  PROCEDURE GET_MORE_POT_CELLS;
to make a choice from.]
                                                                                                                                                       INTEGER;
                                                                                                                                                                                 INTEGER;
                                                                                                                                                                                                         BOOLEAN;
                                                                                                                                REAL;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           END (* WITH *)
                                                                                                                          CELLS DIST
CELL COUNT
NUM OF CELLS
TRY MORE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 END (* IF
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           END (* FOR *);
                                                                                                   VAR
```

```
:= TEMP_OCC2.NEXT
                                                                                                                                                                                                       [ A lodge is going to find a new location. Therefore, delete its existence from the previously occupied location cell.
                                                                                                                                                                     * 1
TRY_MORE := FALSE;
SORT_POTENTIAL_CELLS (POT_CELLS, NUM_OF_CELLS);
SELECT_BEST_CELL (POT_CELLS, A_LODGE,NUM_OF_CELLS, TRY_MORE);
                                                                                                                                                                                                                                                                    PROCEDURE DELETE_FROM_PREV_CELL (A_LODGE : LODGE_PTR);
                                                                                                                                                 U
                                                                                                                                                 ഗ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                IF TEMP_OCC = PREV_OCC THEN
   LOC_TABLE[A_LODGE*.CURRENT_LOC].CURRENT_OCCS
ELSE BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             A_LODGE 3.NAME DO BEGIN
                                                                                                                                                 0
                                                                                                                                                EVI
                                                                                                                                                                                                                                                                                                                                                                                                            WITH LOC_TABLE [A LODGE . CURRENT_LOC] DO TEMP_OCC := CURRENT_OCCS;
                                                                                                                                                <u>Р</u>
В
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              := TEMP_OCC .NEXT;
                                                                                                                                                                                                                                                                                                                                 : OCC_PTR;
                                                                                                                                                Σ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            PREV OCC := TEMP OCC;
TEMP OCC := TEMP OCC NEXT;
(* WHILE *);
                                                                                                                                               0
                                                                                 END (* GET MORE POT CELLS *);
                                                                                                                                                æ
                                                                                                                                                ഠ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                         PREV OCC := TEMP OCC; WHILE TEMP OCC2.LODGE
                                                                                                                                               ۲
                                                                                                                                                                                                                                                                                                                             PREV_OCC, TEMP_OCC
                                                                                                                                               回
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       PREV_OCC2.NEXT
TEMP_OCC2.NEXT
                                                                                                                                               ы
П
```

```
[Exchange information about the locations known and the
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    WHILE (NOT LOC FOUND) AND (TEMP LOC2 <> NIL) DO BEGIN
IF TEMP LOC12.LOC = TEMP LOC22.LOC THEN
LOC FOUND := TRUE;
                                                                                                                                                                                                                                                                                                                               PROCEDURE ADD_INFO (VAR REL_LODGE, A_LODGE : LODGE_PTR);
                                                                                                                                                                                                                                                                             gill status between REL_LODGE and A_LODGE]
                                                                                                                                                                                                                                                                                                                                                                                                       TEMP_LOC1, TEMP_LOC2, PREV_LOC : LOC_PTR;
LOC_FOUND : BOOLEAN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    WHILE TEMP LOC1 <> NIL DO BEGIN
TEMP_LOC2 := A LODGE LOCS_KNOWN;
PREV_LOC := NIL;
LOC_FOUND := FALSE;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     := REL_LODGE .LOCS KNOWN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        PREV_LOC := TEMP_LOC2;
TEMP_LOC2 := TEMP_LOC2*.NEXT;
                                                                                                                                                                               INFO
                                                                                                 END (* DELETE FROM PREV CELL *);
                                               SAVE_OCC_PTR (TEMP_OCC);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       END (* WHILE '*);
END (* ELSE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  TEMP_LOC1
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   BEGIN
                                                                                                                                                                                                                                                                                                                                                                                 VAR
```

```
[ Check if the lodge RELAT (a relative of A_LODGE) lives in the same location. If so, exchange information]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         PROCEDURE CHECK_RELAT (VAR A_LODGE : LODGE_PTR; RELAT : INTEGER);
IF NOT LOC FOUND THEN BEGIN
    NEW (PREV LOC<sup>2</sup>.NEXT);
    PREV LOC := PREV LOC<sup>2</sup>.NEXT;
    PREV LOC<sup>2</sup>.LOC := TEMP_LOC1<sup>2</sup>
    PREV LOC<sup>2</sup>.NEXT := NIL;
END (* IF *);
                                                                                                                                                                                                                   TEMP_LOC1 := TEMP_LOC1 2.NEXT;
                                                                                                                                                                                                                                                                                                                                        IF (REL_LODGE 3. GILL_STATUS) TTRUE;
                                                                                                                                                                                                                                                                                                                                                                                                IF (A LODGE GILL STATUS) THEN REL LODGE GILL STATUS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          回
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              END (* ADD_INFO *);
                                                                                                                                                                                                                                                                              END (* WHILE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         VAR
```

LOC\_PTR; OCC\_PTR; LODGE\_PTR; TEMP\_LOCS TEMP\_OCC REL\_LODGE

```
*
                                                                                                                                                                                                                                                                                                                                                   [ Move a lodge to its new best location for the new season.
:= LOC_TABLE[A_LODGE2.CURRENT_LOC].CURRENT_OCCS;
                                                                                                                                                                                                                                                                                                                                                                                         : LODGE_PTR);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                LODGE LOC: + A LODGE CURRENT LOC;
A LODGE CURRENT LOC := A LODGE BEST CELL;
INSERT LOC IN CELL (A LODGE);
                                  WHILE TEMP_OCC <> NIL DO BEGIN

IF TEMP_OCC<sup>2</sup>.LODGE = RELAT THEN BEGIN
   GET_LODGE (RELAT, REL_LODGE);
   ADD_INFO (REL_LODGE, A_LODGE);
   ADD_INFO (A_LODGE, REL_LODGE);
   END (* IF *);
                                                                                                                                                                                                                                                                                            Ы
                                                                                                                                                                                                                                                                                           G
                                                                                                                                                                                                                                                                                            LOD
                                                                                                                                                                                                                                                                                                                                                                                    PROCEDURE MOVE_A_LODGE (VAR A_LODGE
                                                                                                                                                                      TEMP_OCC := TEMP_OCC *.NEXT;
END (* WHILE *);
                                                                                                                                                                                                                                                                                             Ø
                                                                                                                                                                                                                                                                                                                                                                                                                                              LOC_PTR;
LOC_PTR;
REL_PTR;
BOOLEAN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          INTEGER;
                                                                                                                                                                                                                                                                                           MOVE
                                                                                                                                                                                                                              END (* CHECK RELAT *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                        PREV_LOC
TEMP_REL
LOC_KNOWN
LODGE_LOC
                                                                                                                                                                                                                                                                                                                                                                                                                                            TEMP LOC
TEMP_OCC
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                           VAR
```

```
[ Add the new location to the lodge's list of locations known. Also, exchange information if any relatives (kin or affine) currently
[ Check if the lodge found a different (from the last one) new location.]
                                                                                                                                                                                                                                                                                                                                              WHILE (TEMP_LOC <> NIL) AND (NOT LOC KNOWN) DO BEGIN

IF TEMP_LOC².LOC = A_LODGE².BEST_CELL THEN

LOC_KNOWN := TRUE;

PREV_LOC := TEMP_LOC;

TEMP_LOC := TEMP_LOC².NEXT;

END (* WHILE *);
                                                                 IF LODGE LOC <> A LODGE BEST CELL THEN BEGIN
TEMP_LOC :+ A LODGE LOCS KNOWN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          TEMP_REL := A_LODGE *.AFFINE;
WHILE TEMP_REL <> NIL DO BEGIN
CHECK_RELAT (A_LODGE, TEMP_REL *.FAMILY);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               TEMP_REL := A_LODGE<sup>2</sup>.KIN;
WHILE TEMP_REL <> NIL DO BEGIN
CHECK_RELAT (A_LODGE, TEMP_REL <sup>2</sup>.FAMILY);
TEMP_REL := TEMP_REL <sup>2</sup>.NEXT;
END (* WHILE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         := A_LODGE 2. CURRENT_LOC;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           IF NOT LOC KNOWN THEN BEGIN
    NEW(PREV_LOC<sup>2</sup>.NEXT);
    TEMP_LOC := PREV_LOC<sup>2</sup>.NEXT;
    TEMP_LOC<sup>2</sup>.NEXT := NIL;
    TEMP_LOC<sup>2</sup>.LOC := A_LODGE<sup>2</sup>.(
END (* IF *);
                                                                                                                                                                                                                                               live in the new location.]
                                                                                                                                                                                                                                                                                                                   LOC KNOWN := FALSE;
```

```
Obtain a list
                                                                                                                                     [Find new "best" location cells for all the lodges.
                                                                                                   Z
                                                                                                   0
                                                                                                   Н
                                                                                                   Ø
                                                                                                   r
o
TEMP REL := TEMP_REL '.NEXT;
END (* WHILE *);
                                                                                                   3
                                                                                                   ഠ
                                                             END (* MOVE A LODGE *);
                                                                                                    z
                                                                                                   Ω
                                                                                                    Z
                                     END (* IF);
```

PROCEDURE FIND\_NEW\_LOCATIONS;

best cell for the lodge.

of potential location cells. Put them in order of the season resource.

Delete existence of the lodge from its current location. Find the

Move the lodge to its new location.]

VAR

A\_LODGE : LODGE\_PTR; CELLS\_DIST : REAL; CELL\_COUNT : INTEGER; NUM\_OF\_CELLS : INTEGER; POT\_CELLS : NEW\_HOMES; TRY\_MORE : BOOLEAN;

BEGIN

A\_LODGE := LODGE\_HEAD; WHILE A\_LODGE <> NIL DO BEGIN [ Put all cells closer than SEARCH\_RADIUS into the array POT\_CELLS.

```
[ TRY_MORE indicates that if no suitable location cell is found,
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      TRY MORE := TRUE;
SELECT_BEST_CELL (POT_CELLS, A_LODGE, NUM_OF_CELLS, TRY_MORE);
                                                                                                                                                                                                                                                                                                                          IF CURRENT SEASON = SPRING THEN

RESOURCE := LOC TABLE[CELL COUNT].SPR SCORE

ELSE IF CURRENT SEASON = SUMMER THEN

RESOURCE := LOC TABLE [CELL COUNT.SUM SCORE

ELSE IF (CURENT SEASON = FALL) AND

A LODGE GILL STATUS THEN

RESOURCE := LOC TABLE[CELL COUNT].GILL SCORE

ELSE IF CURRENT SEASON = FALL THEN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  RESOURCE := LOC_TABLE[CELL_COUNT].FALL_SCORE ELSE WRITELN ('ERROR IN FINDING LOCATIONS');
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   SORT_POTENTIAL_CELLS (POT_CELLS, NUM_OF_CELLS);
NUM_OF_CELLS := 0;

FOR CELL COUNT := 1 TO MAX_CELLS DO BEGIN
CELLS DIST := DISTANCE BETWEEN CELLS
(CELL COUNT, A LODGE . CURRENT LOC);
IF CELLS DIST <= SEARCH RADIUS THEN BEGIN
NUM_OF_CELLS := NUM_OF_CELLS + 1;
WITH POT_CELLS [NUM_OF_CELLS + 1;
CELL NUM := CELL COUNT;
DIST := CELL COUNT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         then try the rest of the location cells
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    DELETE_FROM_PREV_CELL (A_LODGE);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                END (* WITH *)
END (* IF *);
END (* FOR *);
```

MOVE\_A\_LODGE (A\_LODGE);

```
Check if the new lodge has any relatives,
                                                                                                                                                                affines, so that their information can be given to the
                                                                                                                                                                                                           PROCEDURE CHECK_FOR_RELS (VAR A_LODGE : LODGE_PTR);
                                                                                                                                                                                                                                                                                                                                                                                                                                         CELL
                                                                                                         ഗ
                                                                                                        Н
                                                                                                        ы
                                                                                                                                                                                                                                                                                                                TEMP_REL := A_LODGE *.KIN *.FAMILY;
CHECK_RELAT(A_LODGE, TEMP_REL);
TEMP_REL := A_LODGE *.AFFINE *.FAMILY;
CHECK_RELAT(A_LODGE, TEMP_REL);
                                                                                                                                                                                                                                                                                                                                                                                                                                          U
                                                                                                         ĸ
                                                                                                                                                                                                                                                                                                                                                                                                                                         0
                                                                                                                                                  [A new lodge has been created.
                                                                                                         0
                                                                                                         Ŀı
                                                                                                                                                                                                                                                                                                                                                                                                                                         ø
                                                                                                       ×
                                                          (* FIND NEW LOCATIONS
:= A_LODGE 2.NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                         ഠ
                                                                                                       U
                                                                                                                                                                                                                                                      TEMP REL : INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                                                                         0
R
D
                                                                                                        ഠ
                                                                                                                                                                                                                                                                                                                                                                                            (* CHECK FOR RELS
                                                                                                        二
                                                                                                       ບ
                            END (* WHILE *);
                                                                                                                                                                i.e. kin or
                                                                                                                                                                              new lodge.
A LODGE
                                                                                                                                                                                                                                                                                                                                                                                            END
                                                          END
                                                                                                                                                                                                                                        VAR
```

[All location cells are ordered on the basis of descending

population cell in the array CELLS using bubble sort.

```
: PRDERED_CELLS);
                                                                                                                                                                                                                                                                                                                       IF CELLS NUM[INDEX2] > MAX NUM THEN BEGIN
MAX_NUM := CELLS NUM[INDEX2]:
MAX_INDEX := INDEX?
END (* IF *).
                                                                                                                                                                                                                                 := CELLS_NUM[INDEX1]
:= TEMP_OCCS *.NEXT;
                                                                                                                                                                                                                   <> NIL DO BEGIN
                                                                                                                                                                                                                                                                                                    FOR INDEX1 := 1 TO MAX CELLS DO BEGIN MAX_NUM := CELLS_NUM[1];
   (VAR CELLS
                                                                                                                                                                       := CURRENT OCCS;
DEX1] := 0;
                                                                                                                                      INDEX1 := 1 TO MAX CELLS DO
WITH LOC TABLE[INDEX1] DO BEGIN
TEMP OCCS
CELLS NUM[INDEX1] := 0;
                                                                      : ORDERED_CELLS;
                                        MAX_INDEX, MAX_NUM : INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                                                                      := MAX_INDEX;
                                                          : INTEGER;
                                                                                                                                                                                                               WHILE TEMP_OCCS <> NI
CELLS_NUM[INDEX1]
TEMP_OCCS
END (* WHILE *);
                                                                                     : OCC_PTR;
PROCEDURE ORDER_LOC_CELLS
                                                                                                                                                                                                                                                                                                                                                                                                                                                    CELLS_NUM[MAX_INDEX]
                                                                                                                                                                                                                                                                         END (* WITH *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                               · ( *
                                                         INDEX1, INDEX2
                                                                                                                                                                                                                                                                                                                                                                                                                                    CELLS [ INDEX1 ]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                             END (* FOR
                                                                                                                                                                                                                                                                                                                                                              FOR INDEX2
                                                                     CELLS NUM
                                                                                                                                                                                                                                                                                                                     MAX_NUM
MAX_INDEX
                                                                                                                                           FOR INDEX1
                                                                                                                BEGIN
                            VAR
```

```
[ The variable NEW_LODGES represents the number of new lodges to be created. A listing of all the location cells is made on the basis of decreasing population. New lodges are created, initialized, and put into the next location cell of the created listing, starting
                                                                   ഗ
                                                                    U
                                                                   о
О
                                                                    囟
                                                                   H
                                                                    ø
                                                                   回
                                                                   ĸ
(* ORDER LOC CELLS
                                                                   U
END
```

PROCEDURE CREAT\_LODGES (NEW\_LODGES : INTEGER);

at the top.

JAR

HEAVY CELLS : ORDERED CELLS; LODGE INDEX : INTEGER; A\_LODGE,PREV : LODGE PTR; REL\_TY : RELATIVE\_TYPS;

EGIN

ORDER\_LOC\_CELLS (HEAVY\_CELLS);
A\_LODGE := LODGE\_HEAD;
PREV := NIL;

WHILE A LODGE <> NIL DO BEGIN PREV := A LODGE; A LODGE := A LODGE \*.NEXT; END (\* WHILE \*);

A\_LODGE := PREV;

```
WITH A_LODGE<sup>2</sup> DO BEGIN
CURRENT_LOC := HEAVY_CELLS[((LODGE_INDEX - 1) MOD MAX_CELLS)
+ 1)];
                                                                                                                                                                                                                                                                                                                    := CURRENT POPUL + PEOPLE;
FOR LODGE INDEX := 1 TO NEW_LODGES DO BEGIN NEW (A_LODGE .NEXT);
A_LODGE := A_LODGE .NEXT;
A_LODGE .NEXT := NIL;
                                                                                                                                                                                  := CURRENT_LOC;
                                                                                                                                 TOTAL_LODGES := TOTAL_LODGES
NAME := TOTAL_LODGES;
                                                                                                                                                                                                                                                                                                      := GET LODGE POPUL;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          := 0;
                                                                                                                                                                                                  := NIL;
                                                                                                                                                                                                                   := FALSE;
                                                                                                                                                                                                                                                                                                                                                                                                       WITH NO OF TRIES DO
                                                                                                                                                                                                                                                                                                                                                                                                                                      SEASON SCORE
CYCLE SCORE
PERIOD SCORE
TOTAL SCORE
(* WITH *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       WITH NO OF SUCC DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          SEASON_SCORE
                                                                                                                                                                NEW (LOCS KNOWN);
LOCS KNOWN<sup>2</sup>.LOC:
LOCS KNOWN<sup>2</sup>.LOC:
GILL STATUS :=
SEASON DIST :=
CYCLE DIST :=
                                                                                                                                                                                                                                                                                                                                                     AFFINE := NIL;
                                                                                                                                                                                                                                                                                                                   CURRENT POPUL
                                                                                                                                                                                                                                                                                TOTAL DIST
                                                                                                                                                                                                                                                                                                                                                                      := NIL;
                                                                                                                                                                                                                                                                                                                                                                                                                       BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        BEGIN
```

```
[ Growth and failure rates are determined on the basis of the gill percentage of lodges. New lodges are created if a chosen
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  random number is greater than the existing failure rate.
                                                                                                                                                                                                                                                                                                                                                                                                         3
                                                                                                                                                                                                                                                                                                                                                                                                        0
                                                                                                                                                                                                                                                                                                                                                                                                         OPULATION
                                                                                                                            REL_TY := KINS;

GET_RELATIVE (REL_TY,A_LODGE);

REL_TY := AFFINES;

GET_RELATIVE (REL_TY,A_LODGE);

INSERT_LOC_IN_CELL_(A_LODGE);

CHECK_FOR_RELS (A_LODGE);
CYCLE SCORE
PERIOD SCORE
TOTAL SCORE
(* WITH *);
                                                                                     END (* WITH A LODHE
                                                                                                                                                                                                                                                                                                                                 (* CREATE_LODGES *);
                                                                                                                                                                                                                                                                                                                                                                                                        CHECK
                                                                                                                                                                                                                                                                                        END (* FOR *);
                                                                                                                                                                                                                                                                                                                                    END
```

PROCEDURE CHECK\_POPULATION\_GROWTH;

VAR

NEW LODGES : INTEGER; FAIL, GROWTH : REAL;

BEGIN

IF MEAN\_GILLS > GILL\_PERC THEN BEGIN

```
[Order all of the location cells on basis of minimum distance
                                                                                                                                                                                                                                                                                            *
                                                                                                                                                                                                                                                                                                                                                                       : ORDERED CELLS; INTEEGER);
                                                                                                                                                          (ROUND (CURRENT_POPUL * GROWTH) /4);
                                                                                                                                                                                                                                                                                                                                                                   PROCEDURE ORDER_POT_GILL_CELLS (VAR GILL_CELLS : THE_CELL :
                                                                                                                                                                                                                                                                                い
回
                                                                                                                                                                                                                                                                                Ч
                                                                                                                                                                                                                                                                                                                                        from the specified cell, THE_CELL]
                                                                                                                                                                                                                                                                               GIL
                                                                                                                                                                                     NEW_LODGES := 1;
CREATE_LODGES (NEW_LODGES);
END (* IF **);
FAIL := GILL GROWTH RATE;
EAIL := GILL FAIL PDF;
END (* IF *)
ELSE BEGIN
                                                                                                                                                                                                                                (* CHECK POPULATION GROWTH
                                                                                                                                         IF GET_RANDOM > FAIL THEN BEGIN NEW_LODGES := ROUND (ROUN
                                                                                                                                                                                                                                                                               P O T
                                                                                                                                                                     = 0 THEN := 1;
                                                                              H := GROWTH RATE;
:= FAILURE_PDF;
                                                                                                                                                                                                                                                                              ORDE
                                                                                                           *
                                                                                                                                                                    IF NEW LODGES
                                                                                                             (* ELSE
                                                                                GROWTH
                                                                                              FAIL
                                                                                                              END
                                                                                                                                                                                                                                  END
                                                                                                                                                                                                                                                                                                                                                                                                                  VAR
```

DIST\_FROM\_CELL : ARRAY [1..MAX\_CELLS] OF REAL;
INDEX1,INDEX2,MIN\_INDEX : INTEGER;
MINM\_DIST,TEMP\_DIST : REAL;
TEMP\_INDEX : INTEGER;

BEGIN

FOR INDEX1 := 1 TO MAX\_CELLS DO BEGIN

The second secon

```
[If the gill net has to be introduced, a list of all of the potential location cells is made, on the basis of decreasing distance from the originally specified cell. The first occupied cell in the list
                                                                                                                                                                                                                                                                                                                                                                                                                                          := DIST_FROM_CELL[MIN_INDEX];
:= TEMP_DIST;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 := GILL_CELLS[MIN_INDEX];
:= TEMP_INDEX;
                                                                                                                                                                                                                   FOR INDEX2 := (INDEX1 + 1) TO MAX_CELLS DO
IF DIST_FROM_CELL[INDEX2] < MINMM_DIST_THEN BEGIN
MIN_INDEX := INDEX2;
MINM_DIST := DIST_FROM_CELL[INDEX2];
                                                                                                                                                                                                                                                                                        := DIST_FROM_CELL[INDEX2];
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  INTRO
                                                                                                                        - 1) DO BEGIN
                                                                                                                     INDEX1 := 1 TO (MAX_CELLS - 1) DO B]
MINM_DIST := DIST_FROM_CELL[INDEX1];
MIN_INDEX := INDEX1);
                                                                                                                                                                                                                                                                                                                                                              TEMP_DIST := DIST_FROM_CELL[INDEX1];
TEMP_INDEX := GILL_CELLS[INDEX1];
DIST_FROM_CELL[INDEX1] :=
DIST_BETWEEN_CELLS(THE_CELL,INDEX1);
GILL_CELLS[INDEX1] :=INDEX1;
(* FOR *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                GILL
                                                                                                                                                                                                                                                                                                                                                                                                                                     DIST_FROM_CELL[INDEX1]
DIST_FROM_CELL[MIN_INDEX]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     END (* ORDER POT GILL CELLS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      GILL_CELLS[MIN_INDEX]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             HECK
                                                                                                                                                                                                                                                                                                              END (* IF *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                GILL CELLS [INDEX1]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       · (*
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       END (* FOR
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               ပ
                                                                                                                                          MINM DIST
MIN INDEX
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             *
                                                                                                                      FOR
```

```
:= LOC_TABLE[GILL_CELLS[GILL_INDEX]].CURRENT_OCCS;
                                                                                                                                                                                                                                                                                                                                                                          WHILE LOC_TABLE[GILL_CELLS[GILL_INDEX]].CURRENT_OCCS = NIL
                                                                                                                                                                                                                                                                  IF CYCLE_COUNTER = CYCLE_NUM THEN BEGIN
THIS_CELL := CELL_NUM;
ORDER_POT_GILL_CELLS(GILL_CELLS, THIS_CELL);
GILL_INDEX := 1;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 WHILE TEMP OCC <> NIL DO BEGIN
GET_LODGE (TEMP OCC'.LODGE,A LODGE);
A_LODGE'.GILL_STATUS := TRUE;
is the place where the gill net is introduced.
                                                                                                             LODGE_PTR;
ORDERED_CELLS;
OCC_PTR;
                                                                                                                                                                                                                                                                                                                                                                                                                GILL_INDEX := GILL_INDEX + 1;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             := TEMP_OCC2.NEXT;
                                                                                              INTEGER;
                                                                                                                                                                                                                                  INDEX := 1 TO NUM OF GILL INVENS DO
WITH GILL LIST[INDEX] DO
                                                                                          GILL INDEX, INDEX, THIS_CELL A LODGE
                                                                                                                                                                                                                                                                                                                                                                                                                                  (* WHILE *);
                                   CHECK_GILL_INTRO;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             (* CHECK GILL INTRO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             TEMP OCC
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            (* WHILE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      END (* IF *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                          TEMP_OCC
                                                                                                                                                                                                                                                                                                                                                                                             DO BEGIN
                                                                                                                               GILL_CELLS
TEMP_OCC
                                                                                                                                                                                                                                                                                                                                                                                                                                    END
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 END
                                    PROCEDURE
                                                                                                                                                                                                                                 FOR INDEX
                                                                                                                                                                                           BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             END
```

```
ĸ
                                                                                                                                                                                                                                                                                                                        ', CURRENT_POPUL: 4);
臼
Σ
Н
回
Ω
H
                                                                                                                                                 1 TO MAX COL DO
                                                                                                                                                                                                                                                                                                           WRITELN; WRITELN ('CURRENT POPULATION
                                                                                                                                                                                                                                                                     PROCEDURE PRINT_SYSTEM_VARS;
                                                                                                                                                                                                          END (* PRINT DELIMITER *);
                            PROCEDURE PRINT_DELIMITER;
ĸ
                                                                                    INDEX : INTEGER;
                                                                                                                                                                                                                                         24
                                                                                                                                              FOR INDEX := WRITE (''' WRITELN; WRITELN; WRITELN;
                                                        MAX_COL
                                                                                                                            WRITELN;
                                                                                                                                      WRITELN;
                                               CONST
                                                                                                        BEGIN
                                                                                                                                                                                                                                                                                         BEGIN
                                                                            VAR
```

```
[Calculate and print the means for distance moved, tries made, and gill percentages. Cycle statistics are recorded in the temporary file F1. Location cells not occupied are printed out.
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        DIST_SUM := DIST_SUM + SEASON_DIST;
TRIES_SUM := TRIES_SUM + NO_OF_TRIES>SEASON_SCORE;
SEASON_DIST := 0.0;
', TOTAL LODGES:4);
                                                                                                                        ß
                                                                                                                                                                                                                                                                                         PROCEDURE PRINT_MEANS (OUOTPUT_TYP : PRINT_TYPS);
                                                                                                                       A
N
                                                                                                                       臼
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              IF OUTPUT TYP = SEASON THEN BEGIN
                                                                                                                     PRIN
                                                                                                                                                                                                                                                                                                                                                                : INTEGER;
                                                                                                                                                                                                                                                                                                                                                       GILLS SUM, INDEX : INTEG DIST SUM : REA A LODGE : LODGE PTR; OUT INFO : FINAL INFO; LODGE DIVISOR : INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  WHILE A LODGE <> NIL DO BEGIN WITH A LODGE DO BEGIN
WRITELN ('NUMBER OF LODGES
                                                END (* PRINT SYSTEM VARS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   A_LODGE :- LODGE_HEAD;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 DIST_SUM
TRIES_SUM
GILLS_SUM
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 BEGIN
```

```
ELSE IF OUTPUT TYP = PERIOD THEN BEGIN

DIST SUM := DIST SUM + PERIOD DIST/3.0;

TRIES SUM := TRIES SUM + NO OF TRIES.PERIOD SCORE/3.0;

PERIOD DIST := 0.0;

NO OF TRIES.PERIOD SCORE := 0;

END (* ELSE *)
                                               ELSE IF OUTPUT TYP = CYCLE THEN BEGIN

DIST SUM := DIST SUM + CYCLE DIST/3.0;

TRIES SUM := TRIES SUM + NO OF TRIES.CYCLE SCORE/3.0;

CYCLE DIST :=0.0;

NO OF TRIES.CYCLE SCORE := 0;

END (* ELSE *)
                                                                                                                                                                                                                                                                                                                                                                               ELSE IF OUTPUT TYP = TOTAL THEN BEGIN

DIST SUM := DIST SUM + TOTAL DIST/3.0;

TRIES SUM := TRIES SUM + NO OF TRIES.TOTAL SCORE/3.0;

END (* ELSE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           [Update sums of lodges for the period and total average
    :0 =:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      IF OUTPUT_TYP = SEASON THEN
LODGE_DIVISOR := TOTAL_LODGES
ELSE IF OUTPUT_TYP = CYCLE_THEN_BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       IF GILL STATUS THEN
GILLS SUM := GILLS SUM +
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           := TOTAL_LODGES;
NO OF TRIES. SEASON SCORE (* IF *)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               A_LODGE := A_LODGE '.NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       statistics of the lodges.]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          END (* WITH *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     END (* WHILE *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         LODGE_DIVISOR
                          END
```

```
: ',DIST_SUM/LODGE_DIVISOR:6:2);
'TRIES_SUM/LODGE_DIVISOR:6:2);
',MEAN_GILLS * 100:6:2);
 := TOTAL_LODGE_SUM + TOTAL_LODGES;
                                                                                                                                                                                                                                                                                                                                                            (* RECORD OUTPUT FOR THIS CYCLE FOR THE FINAL OUTPUT*)
                                                                                                                                                                                                                                                                                                                                                                                                                                      : = CURRENT POPUL;

S := TOTAL LODGES;

IS := SEARCH RADIUS;

DIST := DIST SUM/TOTAL LODGES;

TRIES := TRIES SUM/TOTAL LODGES;

KNOWN := MEAN GILLS * 100;
                                                                                                                                                                                                 := GILLS_SUM / TOTAL_LODGES;
                                                           := PERIOD_LODGE_SUM;
                                    ELSE IF OUTPUT TYP = PERIOD THEN BEGIN
LODGE DIVISOR := PERIOD LODGE SUM
PERIOD LODGE SUM := 0;
END (* ELSE *)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             WRITELN ('LOCATION CELLS NOT OCCUPIED WRITE (' ':8);
                                                                                                                ELSE IF OUTPUT TYP = TOTAL THEN LODGE_DIVISOR := TOTAL_LODGE_SUM;
                                                                                                                                                                                                                                                                                                                                                                                                  IF OUTPUT TYP = CYCLE THEN BEGIN
WITH OUT INFO DO BEGIN
                                                                                                                                                                                                                                                           MEAN TRIES MADE

** OF LODGES WITH GILL
                                                                                                                                                                                                                                      (' MEAN DISTANCE MOVED
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      WRITE (F1, OUT_INFO);
(* IF *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    MEAN_DIST :=
MEAN_TRIES :
GILL_KNOWN
END (* WITH *);
TOTAL LODGE SUM
                (* ELSE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 RADIUS
                                                                                                                                                                                                                                                                                                                                                                                                                                                              LODGES
                                                                                                                                                                                                                                                                                                                                                                                                                                           POPUL_
                                                                                                                                                                                                 MEAN GILLS
                                                                                                                                                                                                                                                                                                 WWRITELN;
                                                                                                                                                                                                                   WRITELN;
                                                                                                                                                                                                                                                                             WRITELN
                                                                                                                                                                                                                                        WRITELN
                                                                                                                                                                                                                                                           WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          END
```

```
[Prints out all currently occupied location cells along with the lodges that occupy them. ]
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         U
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          z
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          Д
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          U
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         U
                                                                                                                                                                                   ELSE IF OUTPUT TYP = PERIOD THEN BEGIN
IF NOT PERIOD OCC THEN
WRITE (INDEX:3)
                                                                                               ELSE IF OUTPUT TYP = CYCLE THEN BEGIN
IF NOT CYCLE OCC THEN
WRITE (INDEX:3)
                                                                                                                                                                                                                                                                      ELSE IF OUTPUT TYP = TOTAL THEN BEGIN
IF NOT EVER OCC THEN
WRITE (INDEX:3);
                                IF OUTPUT TYP = SEASON THEN BEGIN IF CURRENT OCCS = NIL THEN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         ELL
                                                             WRITE (INDEX :3);
INDEX := 1 TO MAX CELLS DO
WITH LOC_TABLE[INDEX] DO
                                                                                                                                                                                                                                         ELSE PERIOD OCC:=FALSE; (* ELSE IF *)
                                                                                                                                                    ELSE CYCLE OCC:=FALSE;
END (* ELSE IF *)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         H
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         z
                                                                                                                                                                                                                                                                                                                                                                                                                                  END (* PRINT MEANS *)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         R
                                                                                                                                                                                                                                                           END
                                                                                                                                                                                                                                                                                                                                                              WRITELN;
                                                                                                                                                                                                                                                                                                                              END
                                                                                                                                                                                                                                                                                                                                                                                WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                WRITELN;
                                                                                   END
FOR
```

PROCEDURE PRINT\_CELL\_OCCUPANCY;

```
[Keep record of the occcupancy of each cell for later calculations.
                                                                                                                                                                                                                                            FOR INDEX := 1 TO MAX CELLS DO

IF LOC TABLE[INDEX].CURRENT_OCCS <> NIL THEN BEGIN

WRITE (' ':5,INDEX:2,' ':7,'-',' ':4);

TEMP_OCC := LOC_TABLE[INDEX].CURRENT_OCCS;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                              := CELL_OCCUP[INDEX] + 1;
                                                                                                                                            WRITELN(' CURRENTLY OCCUPIED LOCATION CELLS ARE :');
                                                                                                                                                                                                                                                                                                                                                 WHILE TEMP_OCC <> NIL DO BEGIN WRITE (TEMP_OCC<sup>2</sup>.LODGE:4);
TEMP_OCC := TEMP_OCC<sup>2</sup>.NEXT;
                                                                                                                                                                                     LODGES');
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  Þ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                回
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               ഗ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      (* PRINT CELL OCCUPANCY
                                                                                                                                                                                                                                                                                                                                                                                                                                                                           CELL_OCCUP[INDEX]
END (* WHILE *);
                                                                                                                                                                                   LOCATION CELL
                                                                                                                                                                                                           !
: INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              PRODU
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              END (* IF
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     WRITELN;
                    TEMP_OCC
INDEX
                                                                                                   WRITELN;
                                                                                                                                                                                                                            WRITELN;
                                                                                                                        WRITELN;
                                                                                                                                                                WRITELN;
                                                                                                                                                                                 WRITELN
                                                                                                                                                                                                      WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     END
```

```
= SPRING THEN
                                                                                                                                                                                                                                     = SUMMER THEN
PROCEDURE PRODUCE SEASON OUTPUT;
                                                                                                                                               WRITE ('THE SEASON IS : ';
TF CURRENT SEASON + FALL THEN
                                                 : PRINT_TYPS;
                                                                                                                                                                                                    ELSE IF CURRENT SEASON WRITE ('SPRING')
                                                                                                                                                                                                                                   ELSE IF CURRENT SEASON WRITE ('SUMMER');
                                                                                                                                                                                                                                                                                                                                                    PRINT_MEANS(OUT-TYP);
PRINT_CELL_OCCUPANCY;
                                                                                                                                                                                                                                                                                                                       PRINT SYSTEM VARS;
OUT TYP := SEASON;
                                                                                                                                                                   IF CURRENT SEASON WRITE ('FALL')
                                                                                                                 PRINT DELIMITER;
                                                OUT TYP
                                                                                                                                                                                                                                                                                      WRITELN;
                                                                                                                                                                                                                                                                                                       WRITELN;
                                                                                                                                   WRITELN;
                                                                                 BEGIN
```

and the number of occupants in each season ofr cycle, period and total output. ] [This procedure prints out all the locations that were occupied

ပ ပ

ഗ

z o

ഗ

回

ALL

回

Ö

PRODU

END (\* PRODUCE SEASOPN OUTPUT \*);

```
SUM := SPR OCCS.CYCLE_SCORE + SUM_OCCS.CYCLE_SCORE +
FALL_OCCS.CYCLE_SCORE;
WRITELN (' ':5,INDEX:3,' ':8,SPR_OCCS.CYCLE_SCORE:4,' ':5,
SUM_OCCS.CYCLE_SCORE:4,' ':5,FALL_OCCS.CYCLE_SCORE:
4,' ' ':7,SUM:4);
       : PRINT_TYPS);
                                                                                                                                                                                                                                                                                                                                                                                                    SCORE (> 0)
                                                                                                                                                                                                                                                                 TOTAL');
                                                                                                                                                                                                                                                                                                                                                                                IF OUTPUT TYPS = CYCLE THEN BEGIN

IF (SPR_OCCS.CYCLE_SCORE <> 0) OR (SUM_OCCS.CYCLE_ORCS.CYCLE_SCORE ,. )) THEN BEGIN
      (OUTPUT_TYPS
                                                                                                                                                                                                                                                                NUMBER OF LODGES
                                                                                                                                                                                                                                                                                                                   FALL');
                                                                                                                                                                                            CELLS:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                0 =:
PROCEDURE PRODUCE_ALL_SEASONS_OCCUP
                                                                                                                                                                                                                                                                                                                                                FOR INDEX := 1 TO MAX CELLS DO WITH LOC_TABLE[INDEX] DO BEGIN
                                                                                                                                                                                                                                                                                                                 SUMMER
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            SPR_OCCS.CYCLE_SCORE
SUM_OCCS.CYCLE_SCORE
FALL_OCCS.CYCLE_SCORE
                                                                                                                                                                                           LOCATION
                                                                                                                                                                                                                                                                                                                ;17, SPRING
                                                                                                                                                                                                                                                            LOCATION CELL
                                                  : INTEGER;
                                                                   INTEGER;
                                                                                                                                                                                         ('OCCUPIED
                                                  INDEX
                                                                                                                                                                                                                                                                                                               WRITELN (
                                                                                                                                     WRITELN; WRITELN;
                                                                   SUM
                                                                                                                                                                                       WRITELN
                                                                                                                                                                                                                                                                                             WRITELN;
                                                                                                                                                                                                                                                                                                                                WRITELN;
                                                                                                                                                                                                                                                           WRITELN
                                                                                                                                                                      WRITELN;
                                                                                                                                                                                                          WRITELN
                                                                                                                                                                                                                                                                              WRITELN
                                                                                                                                                                                                                         WRITELN;
                                                                                                                                                                                                                                          WRITELN;
                                                                                                    BEGIN
```

```
IF OUTPUT TYPS = TOTAL THEN BEGIN

IF (SPR_OCCS.TOTAL_SCORE <> 0) OR (SUM_OCCS.TOTAL_SCORE <> 0)

OR_FALL_OCCS.TOTAL_SCORE <> 0) THEN BEGIN

SUM := SPR_OCCS.TOTAL_SCORE + SUM_OCCS.TOTAL_SCORE +

FALL_OCCS.TOTAL_SCORE;

WRITELN (' ':5,INDEX:3,' ':8,SPR_OCCS.TOTAL_SCORE:4,' ':
                                                                                                                                                                                                                                                                                                                                                                                                       ":8,SPR_OCCS.TOTAL_SCORE:4,"
4,' ':5,FALL_OCCS.TOTAL_SCORE
                                                             0
                                                IF (SPR OCCS.PERIOD SCORE <> 0) OR (SUM OCCS.PERIOD SCORE <> OF (SUM OCCS.PERIOD SCORE <> 0) THEN BEGIN SUM :=SPR OCCS.PERIOD SCORE = SUM OCCS.PERIOD SCORE + FALL OCCS.PERIOD SCORE; WRITELN (' ':5,INDEX:3,' ':8,SPR OCCS.PERIOD SCORE:4,' SUM OCCS.PERIOD SCORE:4,' ':5,FALL OCCS.PERIOD SCORE:5,
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             ('ERROR IN PROCEDURE : PRINTING OCCUPANTS');
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          H
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          Д
                                                                                                                                                                                                                                                                                                                                                                                                                          SUM OCCS. TOTAL SCORE: 4, 4, '-':7, SUM: 4);
                                                                                                                                                           SUM OCCS.PERIOD_SCORE:4,'
4,'-':7,SUM:4);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          H
                                       = PERIOD THEN BEGIN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          D
                                                                                                                                                                                                  0 =:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         0
                                                                                                                                                                                                                                  FALL OCCCS. PERIOD SCORE END (* IF *)
                                                                                                                                                                                              SPR_OCCS.PERIOD_SCORE SUM_OCCS.PERIOD_SCORE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         臼
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        Ы
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          (* PRINT ALL SEASON OCCUP *);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        ပ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        K
C
                                                                                                                                                                                                                                                                              (* ELSE IF *)
                                    ELSE IF OUTPUT TYPS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                   END
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        团
*
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     (* ELSE IF
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        Ö
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        D
Q
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        м
0
                 END (* IF
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 HLIM *)
                                                                                                                                                                                                                                                                                                 ELSE
END
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       END
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 END
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      *
```

```
[After every PERIOD_TIME (a user supplied value) cycles, a periodic output of all the statistics is produced. ]
[A cycle output produces the statistics for that cycle.
                                                                                                                                                                                                                                                    ', SEARCH_RADIUS:5:2);
                                                                                                                                                                                                                                                                                                                                                                                                                          D
                                                                                                                                                                                                                                                                                                                                                                                                                         D
0
                                                                                                                                                                      : ', CYCLE_COUNTER);
                                                                                                                                                                                                                                                                                                                                                                                                                          U
                                                                                                                                                                                                                                                                                                                                                                                                                         H
Q
                                                                                                                                                                                                                                                                                                                                                                                                                          0
                                                                                                                                                                                                                                                                                                          OUT_TYP := CYCLE;
PRINT_MEANS(OUT_TYP);
PRODUCE_ALL_SEASONS_OCCUP (OUT_TYP);
                                                                                                                                                                                                                                                                                                                                                                                                                          Н
                                                                                                                                                                                                                                 IF NOT CONSTANT RADIUS THEN BEGIN WRITELN ('SEARCH RADIUS
                                                                                                                                                                                                                                                                                                                                                                                                                          æ
                                                                                                                                                                                                                                                                                                                                                                         (* PRODUCE CYCLE OUTPUT *);
                             PROCEDURE PRODUCE_CYCLE_OUTPUT;
                                                                                                                                                                                                                                                                                                                                                                                                                         臼
                                                                                                                                                                                                                                                                                                                                                                                                                          Д
                                                                           : PRINT TYPS;
                                                                                                                                                                  WRITELN ('THE CYCLE IS
                                                                                                                                                                                                                                                                                                                                                                                                                         ပ
                                                                                                                                                                                                                                                                                                                                                                                                                         臼
                                                                                                                                                                                                  PRINT_SYSTEMM_VARS;
                                                                                                                                                                                                                                                                                                                                                                                                                         H
                                                                                                                                      PRINT DELIMITER;
                                                                                                                                                                                                                                                                                                                                                                                                                         O
                                                                                                                                                                                                                                                                               · ( *
                                                                                                                                                                                                                                                              WRITELN;
                                                                         OUT TYP
                                                                                                                                                                                                                                                                              (* IF
                                                                                                                                                                                  WRITELN;
                                                                                                                                                    WRITELN;
                                                                                                         BEGIN
                                                                                                                                                                                                                                                                                                                                                                         END
                                                                                                                                                                                                                                                                               END
                                                           VAR
```

PROCEDURE CHECK\_PERIODIC\_OUTPUT;

```
, CYCLE_COUNTER DIV PERIOD_TIME:5);
                                                                                                                                                                                                                                                                                                                                                                                                                 S
                                                                                                                                                                                                                                                                                                                                                                                                                  臼
                                                                                                                                                                                                            AVG_RADIUS _:= PERIOD_RADIUS/PERIOD_TIME;
WRITELN ('SEARCH RADIUS : ',AVG_RADIUS:5:2);
                                                                              IF (CYCLE_COUNTER MOD PERIOD_TIME) = 0 THEN BEGIN
    PRINT_DELIMITER;
    WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                 ø
                                                                                                                                                                                                                                                                                                                                                                                                                E
L
                                                                                                                                                                                                                                                                                                                                                                                                                  24
                                                                                                                                                                                                                                                                                                                             (OUT_TYP);
                                                                                                                                                                                                                                                                                                                                                                                                                ALL
                                                                                                                                                                                             IF NOT CONSTANT RADIUS THEN BEGIN
                                                                                                                                                                                                                                                                                                                                                                     (* CHECK PERIODIC OUTPUT
                                                                                                                                                                                                                                                                                                                                                                                                                H
                                                                                                                         WRITELN ('THE PERIOD IS
          : PRINT TYPS; : REAL;
                                                                                                                                                                                                                                                                                                                                                                                                                RIN
                                                                                                                                                                                                                                                                                            OUT_TYP := PERIOD;
PRINT_MEANS (OUT_TYP);
PRODUCE_ALL_SEASONS_OCCUP
                                                                                                                                                                                                                                                        :0 =:
                                                                                                                                                                   PRINT_SYSTEM_VARS;
                                                                                                                                                                                                                                                     PERIOD_RADIUS (* IF *);
                                                                                                                                                                                                                                                                                                                                                      (* IF *);
           OUT_TYP :
                                                                                                                                                                                                                                      WRITELN;
                                                                                                                                        WRITELN;
                                                     BEGIN
                                                                                                                                                                                                                                                                                                                                                       END
END
                                                                                                                                                                                                                                                                     END
VAR
```

[Print the kin and affines of all the lodges.

```
(' A LISTING OF ALL THE RELATIVES :');
                                                                                                                                                                                                                                                                                                                                                     WHILE TEMP_REL <> NIL DO BEGIN
WRITE (TEMP_REL *.FAMILY:3);
TEMP_REL := TEMP_REL *.NEXT;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      <> NIL DO BEGIN
                                                                                                                                                                                                                                                                                                             WRITE (' ':4, NAME:2,' ':5,"KIN
PROCEDURE PRINT_ALL-RELATIVES;
                                                                                                                                                                                                                                                                    WHILE A LODGE <> NIL DO BEGIN WITH A LODGE DO BEGIN WRITELN;
                                        : LODGE PTR; REL_PTR;
                                                                                                                                                                                                                                                                                                                                                                                                                                          WRITE (' ':11,"AFFINE
                                                                                                                                                                                                             ('LODGE NO.');
('----');
= LODGE_HEAD;
                                                                                                                                                                                                                                                                                                                                                                                                                                                          := AFFINE;
                                                                                                                                                                                                                                                                                                                                                                                                *
                                                                                                                                                                                                                                                                                                                          TEMP_REL := KIN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     WHILE TEMP REL
                                                                                                                                                                                                                                                                                                                                                                                               END (* WHILE
                                       A_LODGE
TEMP_REL
                                                                                                                                                                                                                                                                                                                                                                                                                                                          TEMP_REL
                                                                                                                                                                                                                                                                                                                                                                                                                              WRITELN;
                                                                                                                                                                    WRITELN
                                                                                                                                                                                                             WRITELN
                                                                                                                                                                                 WRITELN;
                                                                                                             WRITELN;
                                                                                                                           WRITELN;
                                                                                                                                       WRITELN;
                                                                                                                                                     WRITELN;
                                                                                                                                                                                               WRITELN;
                                                                                                                                                                                                                           WRITELN
                                                                                                                                                                                                                                        A_LODGE
                                                                                  BEGIN
                          VAR
```

```
[Prints out the statistics for each cycle from the temporary file F1.
                                                                                                                                                                                                                                                                                                                                                                                                                                                    GILL 8')
                                                                                                                                                                                                                                                                                                                                                                                                                                       RADIUS
                                                                                                                                                                                                                                                                                                                                                                                                                                                                       1 1 1 1 1
                                                                                                                                                                        0
                                                                                                                                                                                                                                                                                                                                                                                                                                       LODGES
                                                                                                                                                                         回
                                                                                                                                                                                                                                                                                                                                                                                                                                                                       TRIES
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   1 1 1 1 1
                                                                                                                                                                                                                                                                                                                                                                                                                                                      MEAN
WRITE (TEMP_REL *.FAMILY:3);
TEMP_REL := TEMP_REL *.NEXT;
END (* WHILE *);
                                                                                                                                                                                                                                             PROCEDURE PRODUCE_CYCLES_OUTPUT;
                                                                                                                                                                                                                                                                                                                                                                                                                                       POPULATION
                                                                                                                                                                         回
                                                                                                                                                                        U
                                                                                                                            END (* PRINT ALL RELATIVES
                                                                                                                                                                                                                                                                                                      OUT_INFO : FINAL_INFO;
                                                                                  A_LODGE := A_LODGE *.NEXT;
END (* WHILE *);
                                                                                                                                                                        DOO
                                                                                                                                                                                                                                                                                                                                                                                                                                                     ('MEAN DISTANCE
                                                                                                                                                                                                                                                                                      INDEX : INTEGER;
                                                       END (* WITH *);
                                                                                                                                                                                                                                                                                                                                                                                                                                    WRITE ('CYCLE 3
                                                                                                                                                                                                                                                                                                                                                                                                                                                                   1 1 1
                                                                                                                                                                                                                                                                                                                                                                             INDEX := 0;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                               WRITELN (WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                        WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                     WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                        WRITELN;
                                                                                                                                                                                                                                                                                                                                                              RESET (
                                                                                                                                                                                                                                                                                                                                                                                                                                                                 WRITE
                                                                                                                                                                                                                                                                                                                                  BEGIN
                                                                                                                                                                                                                                                                          VAR
```

```
affines for each lodge, and a chi-square analysis of the random numbers
                                                                                                                                                                                                                                                 [Prooduce total output which includes the final averages, occupancy
                                                            9, LODGES:4, ':7);
:13, MEAN_TRIES:5:2,
                                                                                                                                                                                                                                                                of all the location cells, statistics for each sysle, all kin and
                                                                                                                                                                                                   H
                                                                                                                                                                                                   P U
                                                                                                                                                                                                   T O
                                                                         MEAN_DIST:5:2,
                                                           :10, POPUL:4,
                                                                                                                                                                                                   OTAL
                                                                          WRITELN (RADIUS:5:2,' ':7 ' ':7 ' ':9, GILL_KNOWN:5:2);
                                                                                                                                                                                                                                                                                                                             PROCEDURE PRODUCE TOTAL OUTPUT;
                                                                                                                                                                                                   ы
ы
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    : REAL;
                                                                                                                                                     END PRODUCE CYCLES OUTPUT *);
                                                        WRITE ('' ':2, INDEX:3,
                                                                                                                                                                                                                                                                                                                                                                          : PRINT TYPS;
             READ ( F1, OUT INFO);
WITH OUT INFO DO BEGIN
WRITELN;
                                                                                                                                                                                                   O D O
                                                                                                                                                                                                                                                                                                                                                                                                                                                     : REAL;
INDEX + 1;
                                                                                                                                                                                                                                                                                                                                                                                                                        : REAL;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                      REAL;
                                                                                                                                                                                                                                                                                                                                                                                      : INTEGER;
                                                                                                                                                                                                                                                                                                                                                                                                                                      : REAL;
                                                                                                                                                                                                    24
                                                                                                       END (* WITH *);
(* WHILE *);
                                                                                                                                                                                                                                                                                                                                                                                                        : REAL;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                VAR MEAN RATIO
                                                                                                                                                                                                    Д
                                                                                                                                                                                                                                                                                                                                                                                                                                                                     SUM OCCUP
                                                                                                                                                                                                                                                                                                generated.
                                                                                                                                                                                                                                                                                                                                                                                                                                 SUM_VAR
OCCUP_VAR
                                                                                                                                                                                                                                                                                                                                                                                                                        AVG RADIUS
                                                                                                                                                                                                                                                                                                                                                                         OUT TYP
                                                                                                                                                                                                                                                                                                                                                                                          COUNT
INDEX
                                                                                                                        END
                                                                                                                                                                                                                                                                                                                                                            VAR
```

WHILE NOT EOF (F1) DO BEGIN

```
IF NOT CONSTANT_RADIUS THEN BEGIN
   AVG_RADIUS := TOTAL_RADIUS/TOTAL_CYCLES;
   WRITELN ('SEARCH_RADIUS :: ', AVG_RADIUS:5:2);
                                                                                                                                                                                                                              OUT_TYP := TOTAL;
PRINT_MEANS (OUT_TYP);
PRODUCE_ALL_SEASONS_OCCUP (OUT_TYP);
PRODUCE_CYCLES_OUTPUT;
PRINT_ALL_RELATIVES;
                                                                                                                                                                                                                                                                                                                                                                                                                                     := 1 TO MAX_INTERVALS DO
                                                                                                                                                                                                                                                                                                                                                            ANALYSIS')
                              THE FINAL OUTPUT IS
                                                  1 1 1
                                                                                                                                                                                                                                                                                                                                                                                                                                                   (COUNT : 10);
                                                                                                                                                                                                                                                                                                                                                          CHI-SQUARE
                                                 11111
                                                                                                        PRINT_SYSTEM_VARS;
                                                                                                                                                                                                                                                                                                                                                                                                                     WRITE ('INTERVALS
PRINT DELIMITER;
                                                                                                                                                                                                 · ( *
                                                                                                                                                                                  WRITELN;
                                                                                                                                                                                                 (* IF
                                                                                                                                                                                                                                                                                                                                                                                                                                                    WRITE
                                                                                                                                                                                                                                                                                                                                                                                                                                       FOR COUNT
                                            WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                   WRITELN;
              WRITELN;
                             WRITELN
                                                            WRITELN;
                                                                           WRITELN;
                                                                                                                                                                                                                                                                                                                          WRITELN;
                                                                                                                                                                                                                                                                                                                                         WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                       WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                       WRITELN;
                                                                                                                                                                                                                                                                                                                                                        WRITELN
                                                                                                                                                                                                                                                                                                                                                                        WRITELN
                                                                                                                                                                                                 END
```

BEGIN

```
U := U + SQR (RANDOM_LIST[COUNT] - TOTAL_RANDOMS/MAX_INTERVALS);
                                                                                                                                                                                                                                                                                                                                                                           [Calculate and print out the variance to mean ratio for the occupancy of the cells.
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    SUM_VAR := SUM_VAR + SQR(MEAN_OCCUP - CELL OCCUP[COUNT]);
                                                                                                                        , TOTAL_RANDOMS);
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      :', MEAN OCCUP:6:3);
:', OCCUP VAR:6:3);
:'VAR_MEAN_RATIO:6:3);
                                                                                                                                                                                                                                                                                                   WRITELN (' CHI-SQUARE STATISTIC U = ',U:5:3);
                                                                                                                WRITELN ('TOTAL NUMBER OF RANDOM NUMBERS :
                                                                                                                                                                                                                                                                                                                                                                                                                                               SUM OCCUP := SUM OCCUP + CELL OCCUP[COUNT];
MEAN OCCUP := SUM OCCUP/MAX CELLS;
SUM VAR := 0.0;
FOR COUNT := 1 TO MAX CELLS DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       OCCUP_VAR := SUM_VAR/MAX_CELLS - 1);
VAR_MEAN_RATIO := OCCUP_VAR/MEAN_OCCUP;
                                                                                                                                                                                                                                                        U := U * MAX_INTERVALS / TOTAL_RANDOMS;
    EQUENCY: ');
:= 1 TO MAX INTERVALS DO
(RANDOM_LIST[COUNT]: 10);
                                                                                                                                                                                     FOR COUNT := 1 TO MAX_INTERVALS DO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            'CELL OCCUPANCY VARIANCE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  'VARIANCE TO MEAN RATIO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       'CELL OCCUPANCY MEAN
WRITE ('FREQUENCY:
                        FOR COUNT
                                              WRITE
                                                                                           WRITELN;
                                                                                                                                                                                                                                                                               WRITELN;
                                                                    WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           WRITELN:
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 WRITELN;
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    WRITELN
                                                                                                                                                                  \Omega := 0
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      WRITELN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               WRITELN
```

```
*
                                                                                                                                                                                                                                                := SPRING TO FALL DO
                                                                                                                                                    := 1 TO TOTAL_CYCLES DO
                                                                                                                                                                                                                 := GET_LODGE_RADIUS;
                                             Z
                                                                                                                                                                                                                                                                          IF CURRENT_SEASON = FALL THEN
   CHECK GILL INTRO;
FIND_NEW_LOCATIONS;
                                             Н
                                              Ø
                                                                                                                                                                                                                                                                                                                                        IF SEASON OUTPUT THEN PRODUCE SEASON OUTPUT;
                                                                                                                                                                                              IF NOT CONSTANT RADIUS THEN SEARCH RADIUS := GET_LO
                                                                                                                                                                                                                                                                                                                                                                                                                 PRODUCE CYCLE OUTPUT;

IF PERIOD OUTPUT THEN

CHECK PERIOD OUTPUT;

CHECK POPULATION GROWTH;
(* PRODUCE TOTAL OUTPUT
                                                                                                                                                                                                                                                                                                                                                                                                    IF CYCLE OUTPUT THEN
                                                                                                                                                                                                                                             FOR CURRENT_SEASON
                                                                                                                    READ_AND_PRINT_INPUT;
INITIALIZE_GLOBALS;
FOR_CYCLE_COUNTER :=
                              ----*)
                                                                                                                                                                                                                                                                                                                                                                       END;
                                          *
                                                                                                                                                                  BEGIN
                                                                                        BEGIN
END
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                END
```

WRITELN; WRITELN;

PRODUCE\_TOTAL\_OUTPUT;

END (\* MAIN \*).

LIST OF REFERENCES

## LIST OF REFERENCES

Aldenderfer, Mark S.

1981 Creating assemblages by computer simulation; the development and uses of ABSIM. In Simulations in archaeology, edited by Jeremy A. Sabloff, pp. 67-118. University of New Mexico Press, Albuquerque.

Methods of cluster validation for archaeology. World Archaeology 14:61-72.

Bell, James A.

1981 Scientific method and the formulation of testable computer simulation models.

In Simulations in archaeology, edited by Jeremy A. Sabloff, pp. 51-64.

University of New Mexico Press, Albuquerque.

Bettarel, Robert and Sidney Harrison
1962 An Early ossuary in Michigan. The
Michigan Archaeologist 8:37-42.

Bigony, Beatrice A.

1968 An Archaeological survey along the south shore of Lake Michigan. Ms. on file, Museum of Anthropology, University of Michigan.

Binford, Lewis R. and George I. Quimby
1963 Indian sites and chipped stone materials
in the northern Lake Michigan area.
Fieldiana: Anthropology 36:277-307.

Blair, Emma H. (editor)

1911-1922

The Indian tribes of the Upper Mississippi Valley and regions of the Great Lakes, as described by Nicholas Perrot, French commandant in the Northwest; Bacqueville de la Potherie, French royal commissioner to Canada; Morrell Marston, American army officer; and Thomas Forsyth, United States agent at Fort Armstrong. 2 vols. Arthur H. Clark, Cleveland.

Brandt, Andres von

1972

Fish catching methods of the world. Fishing News, London.

Brose, David S.

1970

The Archaeology of Summer Island: changing settlement systems in northern Lake Michigan. Museum of Anthropology, University of Michigan, Anthropological Papers 41.

Buckmaster, Marla

1980

Scott Point: a preliminary report.
Paper presented at the 1980 Annual
Meeting of the Midwest Archaeological
Conference, Milwaukee.

Carver, Jonathan

1956

Travels through the interior parts of North America in the years 1766, 1767, 1768. Ross and Haines, Minneapolis.

Casselman, J.M., J.J. Collins, E.J. Crossman, P.E.

Ihssen, and G.R. Spangler

1981

Lake whitefish (<u>Coregonus clupeaformis</u>) stocks of the Ontario waters of Lake Huron.

<u>Canadian Journal of Fisheries and Aquatic Sciences</u> 38:1772-1789.

Chadwick, A.J.

1978

A Computer simulation of Mycenaean settlement. In <u>Simulation studies in archaeology</u>, edited by Ian Hodder, pp. 47-57. University Press, Cambridge.

Champlain, Samuel de

1907 <u>Voyages of Samuel de Champlain, 1604-1618</u>. Edited by William L. Grant.

Barnes and Noble, New York.

Charlevoix, Pierre F.X. de

1923

Journal of a voyage to North America.
Caxton Club, Chicago.

Christie, W.J.

Effects of artificial propagation and the weather on recruitment in the Lake Ontario whitefish fishery. Journal of the Fisheries Research Board of Canada 20:597-646.

The Salmonid community in Lake Ontario.

Journal of the Fisheries Research Board
of Canada 29:913-929.

Clarke, David L.

1977 Spatial information in archaeology. In Spatial archaeology, edited by David L. Clarke, pp.1-32. Academic Press, London.

Cleland, Charles E.

The Prehistoric animal ecology and ethnozoology of the Upper Great Lakes region. Museum of Anthropology, University of Michigan, Anthropological Papers 29.

1967 The Traverse Corridor: a study of prehistoric culture contact. Ms. on file, the Museum, Michigan State University.

1974 Northern Michigan canoe Indians: a model for prehistoric settlement in the Traverse Corridor of Michigan. Paper presented at the 1974 Midwest Archaeological Conference, Milwaukee.

The Inland shore fishery of the northern Great Lakes: its development and importance in prehistory. American Antiquity 47:761-784.

Collins, N.E.
1942 Two Delta County sites. The Totem Pole
10 (2).

Copway, George

1847

The Life, history, and travels of Kah-GeGa-Gah-Bowh (George Copway). James
Harmstead, Philadelphia.

Cordell, Linda S.

1982 The Wetherill Mesa simulation: a retrospective. In Simulations in Archaeology, edited by Jeremy A.
Sabloff, pp. 119-141. University of New Mexico Press, Albuquerque.

Cowgill, George
1975 Population pressure as a non-explanation.
In Population studies in archaeology and biological anthropology: a symposium, edited by Alan C. Swedlund. Memoirs of the Society for American Archaeology
30:127-131.

Cutting, Charles L.

1956

Fish saving: a history of fish processing from ancient to modern times.

Philosophical Library, New York.

Dice, Lee R.

1943

The Biotic provinces of North America.
University of Michigan Press, Ann Arbor.

Dixon, W.J. (editor)

1981 BMDP statistical software. University of California Press, Berkeley.

Doran, James E.

1970 Systems theory, computer simulations, and archaeology. World Archaeology 1:289-98.

1973 Computer models as tools for archaeological hypothesis formation. In Models in archaeology, edited by David L. Clarke, pp. 425-451. Methuen, London.

Dumont, William H. and Gustaf T. Sundstrom

1961 Commercial fishing gear of the United
States. <u>United States Fish and</u>
Wildlife Service Circular 109.

Dunning, R.W.

1959a Rules of residence and ecology among the northern Ojibwa. American Anthropologist 61;806-816.

1959b Social and economic change among the northern Ojibwa. University Press, Toronto.

Dyke, Bennett and Jean W. MacCluer (editors)

1974

Computer simulation of human population studies. Academic Press,

New York.

Eddy, Samuel and James C. Underhill
1974 Northern fishes. University of
Minnesota Press, Minneapolis.

Fitting, James E.

1968 The Prehistory of the Burnt Bluff.

Museum of Anthropology, University of Michigan, Anthropological Papers 34.

An Archaeological survey of Beaver Island, Charlevoix County, Michigan. Michigan

Department of State, Michigan History
Division, Archaeological Survey Reports 1.

1975 Aboriginal artifacts from the Schoolcraft House. Ms. on file, Michigan History Division, Michigan Department of State.

1976a Archaeological investigations at the Marquette Mission site, St. Ignace, Michigan, in 1972. The Michigan Archaeologist 22(2-3).

Patterns of acculturation at the Straits of Mackinac. In <u>Cultural change and continuity; essays in honor of James B. Griffin</u>, edited by Charles E. Cleland, pp. 321-334. Academic Press, New York.

The Gyftakis and MacGreggor sites: Middle Woodland occupations in St. Ignace, Michigan. Ms. on file, Michigan History Division, Michigan Department of State.

Middle Woodland cultural development in the Straits of Mackinac region: beyond the Hopewell frontier. In Hopewell archaeology, edited by David S. Brose and N'omi Greber, pp. 109-112. Kent State University Press, Kent, Ohio.

Fitting, James E. (editor)

1974 Contributions to the archaeology of the St. Ignace area. The Michigan Archaeologist 20(3-4).

Fitting, James E. and Charles E. Cleland
1969 Late prehistoric settlement patterns in
the Upper Great Lakes. Ethnohistory
16:289-302.

Forrester, Jay W.
1971 World dynamics. MIT Press, Cambridge.

Franzen, John
1975
An Archaeological survey of Chippewa County.

Michigan Department of State, Michigan

History Division, Archaeological Survey

Reports 5.

1979 <u>A Cultural resource survey of portions of Hiawatha National Forest, final report.</u>
Commonwealth Associates, Jackson, Michigan.

Goodyear, Carole D., Thomas A. Edsall, Diane M. Ormsby
Dempsey, G. David Moss, and Paul E. Polanski

1982

Atlas of the spawning and nursery areas
of Great Lakes fishes. 14 vols. United
States Fish and Wildlife Service,
Washington, D.C.

Greenman, Emerson

1927 Museum report: archaeological investigations in Emmet County, Michigan. July 22-30, 1927. Ms. on file, Museum of Anthropology, University of Michigan.

Griffin, James B. and H.R. Crane
1972 University of Michigan radiocarbon dates
XIV, B. Great Lakes. Radiocarbon
14: 162-4.

Hamilton, Nathan D., James B. Petersen, and Alan McPherron

1982

The Late Woodland perishable industries of the Juntunen site in the Upper Great Lakes:

a preliminary research report. Paper presented at the 47th Annual Meeting of the Society for American Archaeology, Minneapolis.

Hamond, F.W.

The Contribution of simulation to the study of archaeological processes. In <u>Simulation studies</u> in <u>archaeology</u>, edited by Ian Hodder, pp.1-9. University Press, Cambridge.

Hartman, W.L.

Lake Erie: effects of exploitation, environmental changes and new species on the fishery resources. <u>Journal of the Fisheries Research Board of Canada</u>
29:899-907.

Hearne, Samuel

1958 <u>A Journey from Prince of Wales fort in Hudson's Bay to the northern ocean, 1769-1772</u>. Macmillan, Toronto.

Henderson, B.A., J.J. Collins, and J.A. Reckahn

1983 Dynamics of an exploited population of lake whitefish (Coregonus clupeaformis) in Lake Huron. Canadian Journal of Fisheries and Aquatic Sciences 40:1556-1567.

Hennepin, Louis

1972 <u>A New discovery of a vast country in America</u>. Edited by Reuben G. Thwaites. Kraus Reprint, New York.

Henry, Alexander

1969 <u>Travels and adventures in Canada and the Indian territories between the years 1760 and 1776</u>. Edited by James Bain. Charles Tuttle, Rutland, Vermont.

Hinsdale, Wilbert B.

1931 Archaeological atlas of Michigan. Michigan Handbook Series 4.

Distribution of the aboriginal population of Michigan. Museum of Anthropology,
University of Michigan, Occasional
Contributions 2.

Hodder, Ian (editor)

1978 <u>Simulation studies in archaeology</u>. University Press, Cambridge.

Holman, Margaret B.

1978

The Settlement system of the Mackinac
Phase. Ph.D. dissertation, Michigan State
University. University Microfilms, Ann
Arbor.

JR

1896-1901 <u>The Jesuit Relations and allied documents.</u> Edited by Reuben G. Thwaites. 73 vols. Burrows Brothers, Cleveland.

Janzen, Donald R.

The Naomikong Point site and the dimensions of Laurel in the Lake Superior region.

Museum of Anthropology, University of Michigan, Anthropological Papers 36.

Jenness, Diamond

1935

The Ojibwa Indians of Parry Island, their social and religious life. National Museum of Canada Bulletin 78.

Keating, William H. (compiler)

1825

Narrative of an expedition to the source of the St. Peter's River. G.B. Whittaker, London.

Kellogg, Louise B. (editor)

1945

Early narratives of the northwest, 1634-1699. Barnes and Noble, New York.

Kohl, Johann

1956

Kitchi-Gami. Ross and Haines,
Minneapolis.

LaHontan, Louis Armand de Lom d'Arce

1905

New voyages to North America. Edited by Reuben G. Thwaites. McClurg, Chicago.

Landes, Ruth

1961

The Ojibwa of Canada. In <u>Cooperation and competition among primitive peoples</u>, edited by Margaret Mead, pp. 87-126. Beacon Press, New York.

Lanman, Charles

1856

Adventures in the wilds of the United States and British America. Moore, Philadelphia.

Lawler, G.H.

1965

Fluctuations in the success of year-classes of whitefish populations with special reference to Lake Erie. <u>Journal of the Fisheries Research Board of Canada</u>

22:1197-1227.

Loftus, K.H.

1958

Studies of river spawning lake trout.

Transactions of the American Fisheries
Society 87:259-277.

Lovis, William A.

1970 Wycamp Creek: a Middle and Late Woodland village in Emmet County, Michigan. Ms. on file, the Museum, Michigan State University.

1973

<u>Late Woodland cultural dynamics in the northern lower peninsula of Michigan.</u>

Ph.D. dissertation, Michigan State University. University Microfilms, Ann Arbor.

Lovis, William A. (editor)

1976 Archaeological investigations within Fisherman's Island State Park: 1976 season.

Michigan State University Museum, Archaeological Survey Report 10.

Lovis, William A. and Margaret B. Holman

1976

Subsistence strategies and population: a hypothetical model for the development of Late Woodland in the Mackinac Straits - Sault Ste. Marie area. Papers of the Michigan Academy of Science, Arts, and Letters 8:267-276.

Lovis, William A. and Robert Mainfort

1971 Prehistoric materials from Fort
Michilimackinac: 1967 to 1969. Ms. on file,
the Museum, Michigan State University.

Low, Gilbert W. 1981

Using system dynamics to simulate the past. In <u>Simulations in archaeology</u>, edited by Jeremy A. Sabloff, pp. 249-282. University of New Mexico Press, Albuquerque.

Luedtke, Barbara E.

1976

Lithic material distributions and interaction patterns during the Late Woodland period in Michigan. Ph.D. dissertation, University of Michigan. University Microfilms, Ann Arbor.

Margry, Pierre 1876-1886

Discoveries and settlements of the French in western and southern North America, 1614-1754. Translation of Decouvertes et etablissements des francais dans l'ouest et dans le sud de l'Amerique, 1614-1754.

6 vols. Jouaust, Paris. Ms. on file, Detroit Public Library, Burton Historical Collection.

Martin, Susan R.

1977 A Preliminary cultural resource mamagement study of the Hiawatha National Forest, Michigan. Michigan State University Museum, Survey Report 20.

Final report: phase II archaeological site examination of the Gros Cap cemetery area, Mackinac County, Michigan. Archaeology
Laboratory, Michigan Technological University, Cultural Resource Management Report

6.

Martin, Susan R. and Patrick E. Martin
1979 Preliminary archaeological site examination of the proposed expansion of US-2,
Mackinac County, Michigan. Archaeology
Laboratory, Michigan Technological University, Cultural Resource Management Report
3.

Martin, Terrance J.

1980a Animal remains from 20MK90 (Pointe Aux Chenes), Mackinac County, Michigan. Ms. on file, Archaeology Laboratory, Michigan Technological University.

1980b Animal remains from the Winter site, a
Middle Woodland occupation in Delta County,
Michigan. The Wisconsin Archeologist
61:91-99.

Animal remains from the Scott Point site, a stratified Late Woodland occupation in Mackinac County, Michigan. Paper presented at the 57th annual meeting of the Central States Anthropological Society, Cincinnati.

Mason, Philip P. (editor)

1958 <u>Schoolcraft's expedition to Lake Itasca.</u>
Michigan State University Press, East

Lansing.

Mason, Ronald J.

1966 Two stratified sites on the Door Peninsula

of Wisconsin. <u>Museum of Anthropology</u>, <u>University of Michigan</u>, Anthropological

Papers 26.

1981 Great Lakes archaeology. Academic Press,

New York.

Masson, Louis R.

1960 <u>Les Bourgeois de la Compagnie du</u>

Nord-Ouest. 2 vols. Antiquarian Press,

New York.

Maxwell, Moreau S.

1964 Indian artifacts at Fort Michilimackinac.

The Michigan Archaeologist 10:23-30.

McKenney, Thomas L.

1827

Sketches of a tour to the lakes.

F. Lucas, Baltimore.

McPherron, Alan

1967

The Juntunen site and the Late Woodland prehistory of the Upper Great Lakes area.

Museum of Anthropology, University of

Michigan, Anthropological Papers 30.

Milner, James W.

1874

Report on the fisheries of the Great Lakes;

the results of inquiries prosecuted in 1871 and 1872. Government Printing Office,

Washington.

Mosimann, J.E. and Paul S. Martin

1975 Simulating overkill by paleoIndians.

American Scientist 63:304-313.

Nie, Norman H., C.H. Hull, J.G. Jenkins, K. Steinbrenner, and D.H. Bent

SPSS: statistical package for the social sciences. McGraw-Hill, New York.

Noble, Vergil E.

1975

In dire Straits; subsistence patterns at Mackinac. The Michigan Archaeologist 29:29-48.

Orton, Clive

1980 <u>Mathematics in archaeology</u>. University Press, Cambridge.

O'Shea, John M.

1978 A Simulation of Pawnee site development. In Simulation studies in archaeology, edited by Ian Hodder, pp. 39-46. University Press, Cambridge.

Perrault, J.B.

Narrative of the travels and adventures of a merchant voyageur in the savage territories of northern America leaving Montreal the 28th of May 1783. Edited by J.S. Fox.

Michigan Pioneer and Historical Collections 37:508-619.

Peske, G.R. and B. Kent

1963 Survey of the northern shore of Lake
Michigan. Ms. on file, Museum of
Anthropology, University of Michigan.

Rau, Charles

1884

Prehistoric fishing in Europe and North America. <u>Smithsonian</u> <u>Contributions</u> <u>to Knowledge</u> 25.

Renfrew, Colin 1981

The Simulator as demiurge. In <u>Simulations</u>
in <u>archaeology</u>, edited by Jeremy A.
Sabloff, pp. 283-306. University of New
Mexico Press, Albuquerque.

Richner, Jeffrey J.

Depositional history and stone tool industries at the Winter site; a Lake Forest Middle Woodland cultural manifestation.
Unpublished M.A. thesis, Department of Anthropology, Western Michigan University.

Ryder, R.A.

Limnology and fishes of oligotrophic lakes in North America (about 1800 A.D.).

Journal of the Fisheries Research Board of Canada 29:617-628.

Sagard, Gabriel

1968 The Long journey to the country of the Hurons. Edited by George M. Wrong.

Greenwood Press, New York.

Schoolcraft, Henry R.

1851-1856 <u>Information respecting the history, condition and prospects of the Indian tribes of the United States</u>. Lippincott, Grambo, Philadelphia.

Narrative journals of travels from Detroit

northwest through the great chain of

American lakes to the sources of the

Mississippi River in the year 1820. Arno
and the New York Times, New York.

Shannon, R.E.

1975 <u>Systems simulation: the art and science.</u>
Prentice-Hall, Englewood Cliffs, New Jersey.

Skinner, Alanson

Notes on the eastern Cree and northern Saulteaux. American Museum of Natural History Anthropological Papers 11(1).

Smith, Beverley

1983 Faunal identifications for seven northern Michigan archaeological sites. Ms. on file, the Museum, Michigan State University.

Smith, James

1978 <u>Scoouwa: James Smith's Indian captivity</u> narrative. Ohio Historical Society,

Columbus.

Smith, S.H.

1972 Fac

Factors of ecologic succession in oligotrophic fish communities of the Laurentian Great Lakes. <u>Journal of the Fisheries</u> Research <u>Board of Canada</u> 29:717-730.

Sokal, Robert R. and F. James Rohlf

1973 <u>Introduction to biostatistics</u>. W.H.

Freeman, San Francisco.

Sommers, Lawrence M. (editor)

1977 <u>Atlas of Michigan</u>. Michigan State

University Press, East Lansing.

Stoltman, James B.

1973 The Laurel culture in Minnesota. Minnesota

Prehistoric Archaeology Series 8.

Stone, Lyle M.

1972

Archaeological investigation of the Marquette Mission site, St. Ignace, Michigan, 1971: a preliminary report.

Mackinac Island State Park Commission, Reports in Mackinac History and Archaeology

1.

1975 <u>Archaeological</u> <u>site</u> <u>survey</u> <u>in</u> <u>the</u> <u>Mackinac</u>

Straits. Mackinac Island State Park Com-

mission, Mackinac Island, Michigan.

Tanner, John

1956

A Narrative of the captivity and adventures

of John Tanner. Ross and Haines,

Minneapolis.

Thomas, David H.

1973 A Computer simulation model of Great Basin

Shoshonean subsistence and settlement patterns. In <u>Models in archaeology</u>, edited by David L. Clarke, pp. 671-704.

Methuen, London.

1976 <u>Figuring anthropology</u>. Holt, Rinehart and Winston, New York.

Veatch, J.O.

1959

Presettlement forest in Michigan.
Michigan State University Department of
Resource Development, East Lansing.

Weston, Donald E.

1978

Ethnography for archaeology: a functional interpretation of an Upper Great Lakes prehistoric fishing artifact. Unpublished M.A. thesis, Department of Anthropology, Western Michigan University.

Wobst, H. Martin
1974 Boundary conditions for paleolithic social
systems: a simulation approach. American
Antiquity 39:147-178.

Wyman, F.P.

1970

Simulation modelling: a guide to using SIMSCRIPT. Wiley, New York.

Zimmerman, Lawrence J.

1977 Prehistoric locational behavior: a computer simulation. Office of the Iowa State Archaeologist Report 10.

1978 Simulating prehistoric locational behavior.

In <u>Simulation</u> <u>studies</u> <u>in archaeology</u>,
edited by Ian Hodder, pp. 27-37. University
Press, Cambridge.

Zubrow, Ezra
1975

Prehistoric carrying capacity: a model.
Cummings, Menlo Park.

