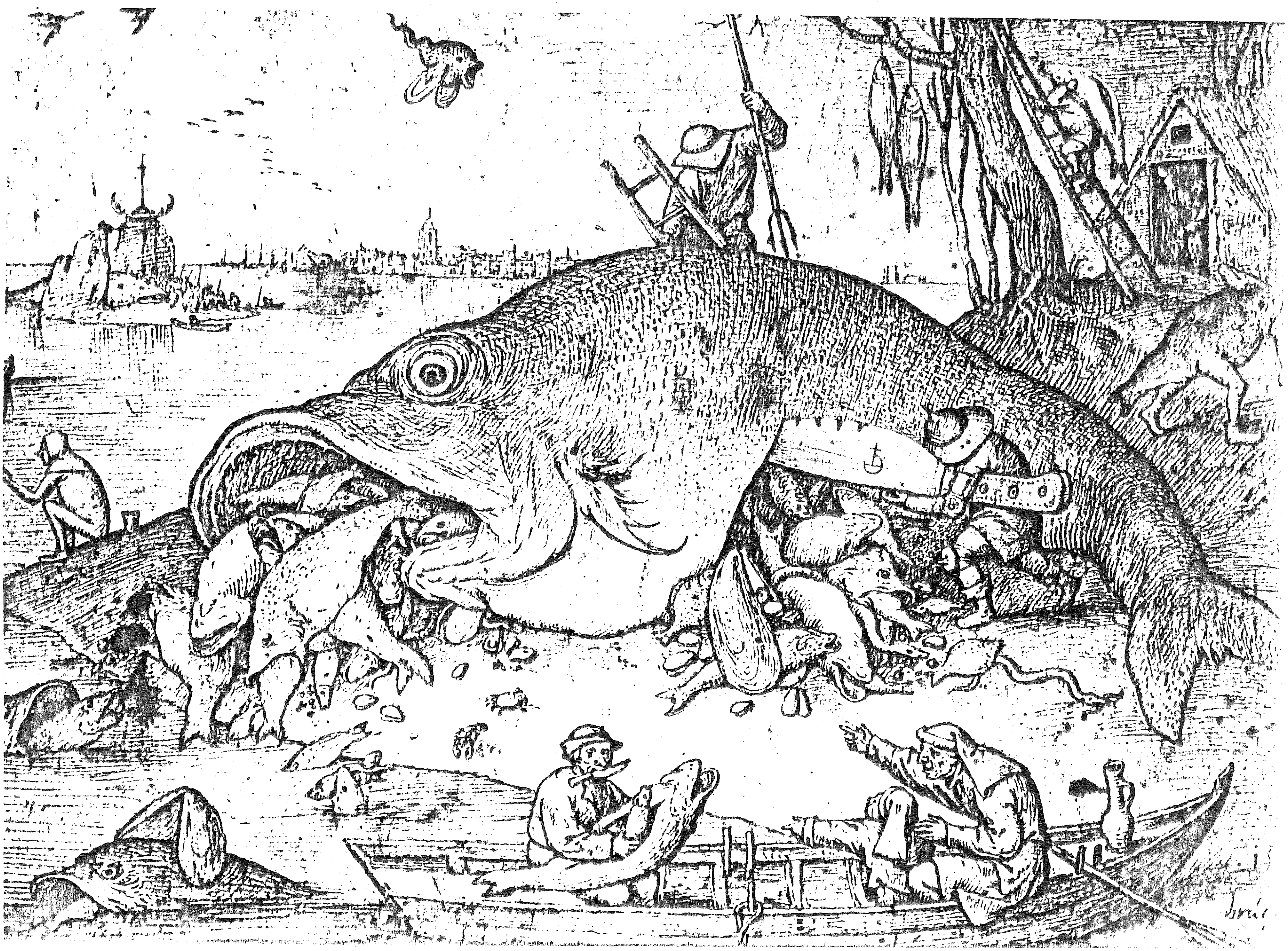
FILE COPY

Wood's Hole Laboratory Reference Document No. 82-07

GEORGE: An Interactive Ecosystem Model of Georges Bank

by

Wendell Hahm



1.0 Introduction

1.1 Background

GEORGE is an ecosystem model of the fish and fisheries of Georges Bank.

The model's intended audience include three distinct users

i. Policy makers

ii. Fisheries researchers

iii. Simulation modelers and analysts

Each user brings unique skills to bear on the research and management questions pertinent to Georges Bank. Policy makers can provide relevance to the overall modeling exercise by asking the applied questions that scientist can often overlook. In addition to asking the critical questions about the operation and validity of the proposed ecosystem model, staff researchers can provide their considerable experience and intuition about the key biological processes affecting the ecosystem's behavior. The modeler can answer the questions posed by the management and research staff by designing appropriate models given the constaints of time and available data.

To be successful, all three groups should work together while studying a particular problem. The simulations can be made more realistic with the cooperation of researchers of management, fisheries biology and ecosystem simulation modeling backgrounds. By looking over each others shoulders, a greater understanding of the ecosystem and one's fellow reseacher's desciplines can be nurtured.

To encourage this important sort of cooperation, the simulation model GEORGE was intentionally made "user friendly." These interactive programs allow the users to stipulate a scenario's initial conditions; to run simulations and to analyze the results. Because the programs are designed to be accessible to people with little or no computer programming experience, the modeling task is opened to a wider circle of potential users.

The purpose of this user manual is to first explain the processes and assumptions that are woven into GEORGE. Given this understanding of the model's conceptual underpinnings, the manual then explains how to design, run and analyze simulation studies.

i. Designing a simulation: The initial conditions and control parameters are explained. An interactive program, SETUP allows format free input of data and run parameters. The program's interactive dialogue is designed to help computer neophytes use the programs.

ii. Running the simulator: The simulator is named GEORGE. The simulator takes the instructions generated by SETUP and runs the simulation study. The simulation interactively allows the user to select and display time series graphs and tables, and phase plane plots of populations of interest. These graphs are intended to aid in the first visual analysis of the simulation.

iii. Flow Analysis: A more detailed analysis of the ecosystem model's internal flows is made available via a Flow Analysis program. Section 2.4 briefly describes the theory and applications of Flow Analysis.

The analytical package allows the fisheries biology and the modeling staff to double check the plausibility of the simulation results It also helps describe and quantify the subtle changes resulting from parameter changes.

1.2 The Intent of the Model

Researchers who have expended considerable time and effort studying an ecosystem like Georges Bank often have an intuitive grasp of the quantitative and qualitative dynamics of the system. This ecosystem model attempts to organize this background information in simulation studies, for those interested parties less familiar with the system. The utility of this model is its ability to familiarize those unfamiliar with the ecosystem with the subtleties of behavior and the sensitivity of ecosystems to certain changes. Besides testing the staff researchers preconceptions of ecosystem processes, the model can also serve to communicate the researcher's understanding of complex ecosystems to a wider circle of interested users.

The credibility and utility of an ecosystem model is related to its simplicity and focus on a specific issue. The nebulous generality of the large scale International Biological Program (IBP) Biome models often hindered attempts to simulate and predict specific phenomena (K.E.F. Watt, 1975). To those intent on using models as predictors, the "jack-of-all-trades was master of none."

Ecosystem modeling's greatest problem isn't a dearth of creative equations or a shortage of "critical mechanisms." For sometime, modeling's weakest aspect has been the ability to analyze and test the consequences of a model's hypotheses and initial conditions.

This model is specifically designed to examine the consequences of predator-prey interactions of a marine ecosystem. To do this, the model places an equal emphasis on the interactive simulation <u>and</u> the analysis of the model.

Ecosystem modelers have long recognized that the dynamics of natural systems are not just the summation of the component processes. V. Watson and O. Loucks (1979) suggest that some of the complexities stem from the recycling within and among system components. Recognizing the relevance of input/output modeling to the analysis of simulation model results, the simulator GEORGE is linked to a flow analysis package developed by J. Finn (1977).

Models can be classified in terms of their purposes: to forecast or to provide insight (H. Caswell, 1976; E.C. Pielou, 1981). GEORGE is a research and teaching tool. GEORGE is not a forecaster. The simulator GEORGE is designed to suggest the directions and relative magnitudes of system changes resulting from the assumptions programmed into the model and initial conditions specified by the user. In doing so, it should assist in familiarizing the users with the dynamic behavior of the ecosystem.

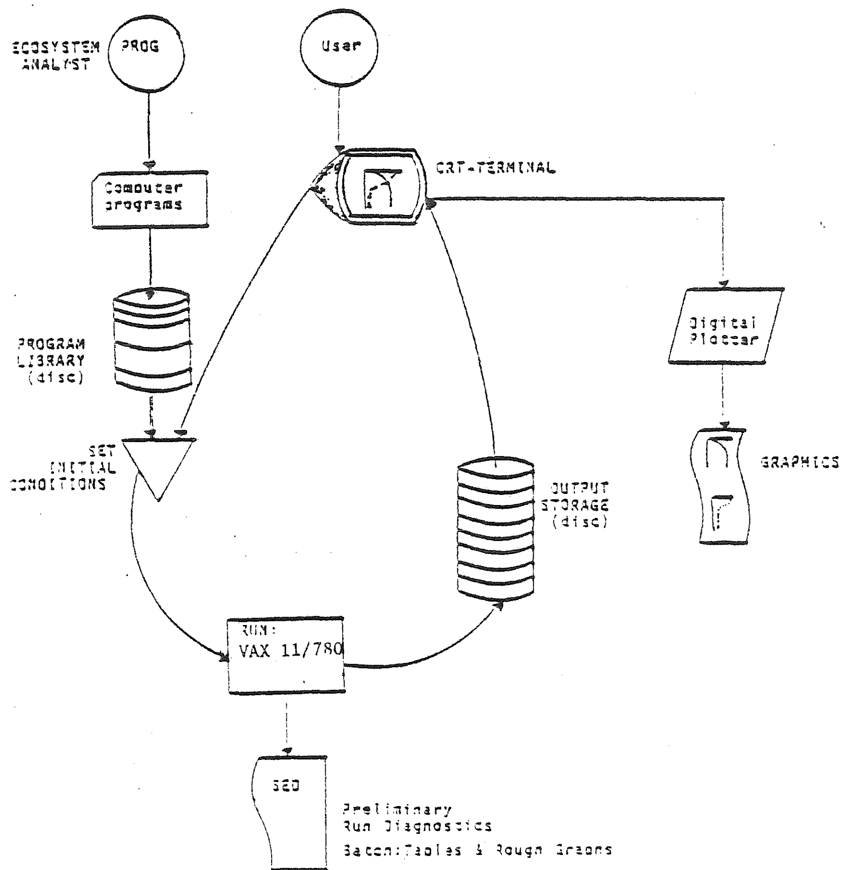


Figure 1

2.0 Modeling: The Ecosystem

2.1 The System Design

By itself the integration of a system of differential equations is insufficient to investigate a system's dynamic behavior. There are in fact three distinct facets to a modeling system. Figure 1 illustrates the relative roles of:

i. the simulator

ii. the differential equation based model

iii. the analytical package

The model GEORGE in reality is a library of linked computer programs. The looped program structure allows the user to iteratively simulate scenarios, display results (tables and graphs), and analyze results. By updating initial conditions and control parameters, the user can re-enter the simulator to carry out further simulations and analyses. Because all data and time series files are stored on computer disc files, the users can economically model the program over a series of days without having to reconstruct scenarios each time they log onto the computer.

2.2 The Simulation System

The simulator is the interactive portion of the modeling system that allows the user to manage the biological model and the analytical programs available. The simulator consists of two programs: SETUP and GEORGE. The first program (SETUP) allows the user to interactively specify the initial abundances of the benthos and fish. The user is also allowed to specify the biological attributes of these populations. This includes specifying the number of age classes and the per capita weights for these organisms. In addition, the user can tailor the feeding and reproductive processes through species and size class specific control parameters.

Once these parameters are set, the user is ready to run the model using these initial conditions. The program GEORGE allows the user to specify the number of years to be simulated. The program also allows the user to name for future reference, the input and output files to be used and created.

The program then allows the user to print on-line labelled time series tables of the changes in abundances, per captia weights, yields and biomasses for compartments of interest. The simulator also allows the user the option of plotting selected time series on x-y line graphs. Besides time series plots, the user can generate phase-plane plots [$(X_{1,t}$ -vs- $X_{2,t})$], to investigate interdependencies among the populations (R. May et al., 1979).

Besides setting up runs and displaying the results, the simulator also allows the user to analyze the inter- and intra-compartmental flow dynamics of the system over specificed days. The Flow Analysis option generates a flow matrix F(i,j) which traces the transfer of matter from compartment j to compartment i. To place this activity in a simpler perspective, a normalized flow matrix N(i,j) displays the proportionate partioning of predator_i's diet

(i.e., N(i,j) = the fraction prey_j contributes to predator_i's diet).

At this point it is possible for the user to terminate the simulation session to examine these tables and graphs, since all events of the simulation are stored in the computer disc memory banks, for future reference.

After analyzing the results of a scenario, the user can set up new initial conditions through SETUP and redesign the scenario. The simulator's looped structure emphasizes the experimental role of a simulator. The interactive looped program allows the users to repeatedly simulate and examine the consequences of their system manipulations.

2.3 The Ecosystem Model: Processes and Assumptions

GEORGE focuses on the predator-prey interactions operating within the fish stocks on Georges Bank. To do this imagine that there exists a system of n-compartments, with potentially n^2-interconnecting pipes. The dynamics of GEORGE depends on whether the valves are open between these n-compartments, and the rates of flow between these compartments. The flows in GEORGE are dependent on three processes:

i. daily ration demands

ii. prey selection to meet these demands

iii. stochastic recruitment processes

The daily rations are based on the literature and laboratory studies by Durbin et al. (1980). The prey selection processes are based on feeding functions originally developed by Andersen and Ursin (1977). Prey selection parameters are based on the

Food Habits datasets (NEFC, Woods Hole) and the analytical work by Hahm and Langton (1983). The recruitment process is based on recruitment records of Hennemuth et al. (1980), and Patil and Boswell (1981).

There are presently six major compartments in the model representing the benthos, yellowtail flounders, cod, haddock, silver hake, and herring. With the exception of the benthos (which has only four size classes), each of the compartments are subdivided into the five size classes. This results in a total of twenty-nine compartments. For the fish, the five size groups represent the pre-recruits, the first year recruits, the two year olds, the three year olds, and the adult fish that are four or more years old.

2.3.1 Daily Ration Demands

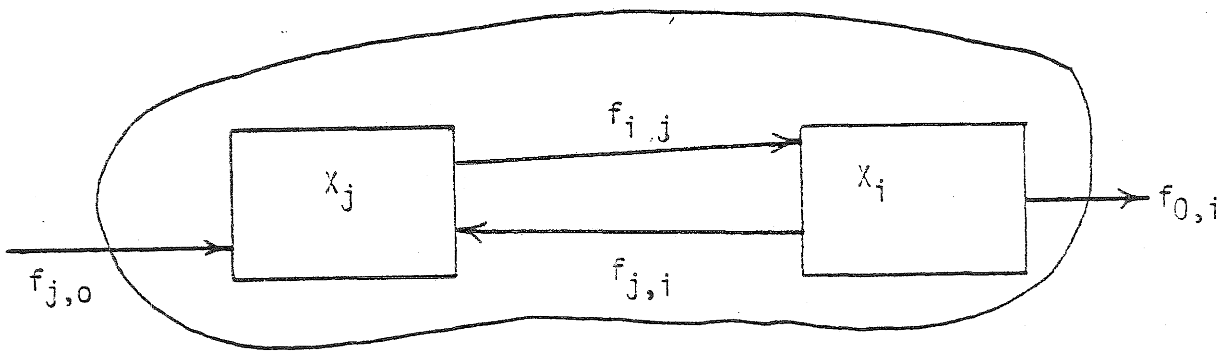


Figure 2

Compartmental Modeling Notations

Figure 2 illustrates the labelling convention of flow diagrams. Flow between compartments (x_i and x_j) are labeled by $f_{i,j}$, where the first subscript is the recipient and the second subscript is the donor (Patten, 1971). The subscript 0 signifies all that is external to the system.

For the ith compartment denoted as (x_i), $f_{i,j}$ represents the contribution to daily ration by the jth prey compartment (x_j) to meet maintenance, growth, respiration and reproductive activity. For simplicity, this demand is estimated to be a fraction, $a(i,day)$ of the consumer's weight, ranging between 2% and 0.5% per day (Durbin et al., 1980). It then follows that the feeding demand, (d_i) of the i^{th} predator x_i population is:

$$d_i = a(i,day) \cdot (n_i \cdot w_i) \qquad \text{eqn. 1a}$$

where: d_i = food demand of predator x_i (unit = kg wet weight/day)

n_i = predator abundance

w_i = predator weight (wet weight)

$a(i,day)$ = fraction of its own weight that a predator consumes. $a(i,day)$ can vary over the season.

The fraction is:

$$a(i,day) = \frac{[Amax(i)+Amin(i)]}{2} - \frac{[Amax(i)-Amin(i)]}{2}.\cos[2\Pi.(\frac{day+Lag}{CL})]$$

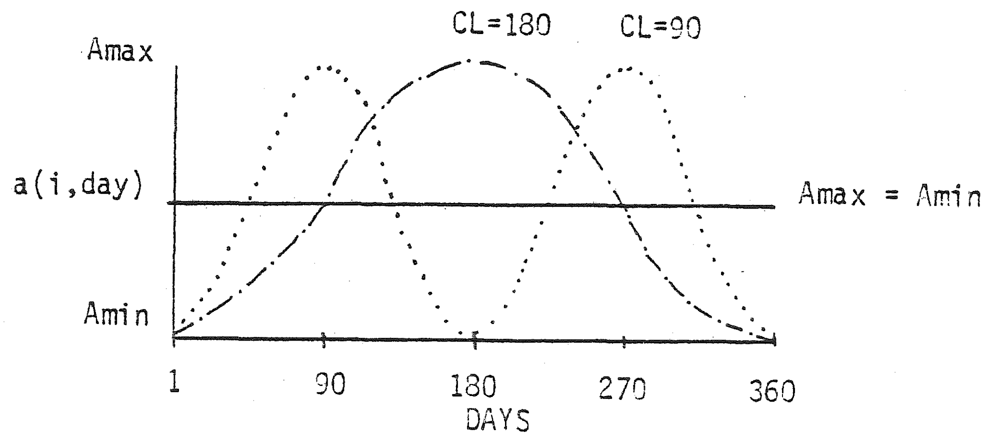
eqn. 1b

where: $Amax(i)$ = maximum $a(i,day)$ (i.e., approximately 2%)

$Amin(i)$ = minimum $a(i,day)$ (i.e., approximately 0.5%)

CL = days per cycle (days/radian)

Lag = phase in days.



a(i,day) as a function of time.

Figure 3

Examination of eqn. 1b reveals that feeding demands can be modified by varying the maximum and minimum daily rations. Figure 3 shows that the variation of feeding activity is determined not only by the maximum and minimum values of a(i,day), but also the number of seasonal feeding maxima. If (CL = 180), days/radians, one maximum occurs per year. If (CL = 90), two maxima are simulated. A two peak feeding pattern is suggested if feeding is elevated during the fall and spring. For the special case where Amax = Amin, there is no variation of feeding demand over the course of the year.

This flexibility is incorporated into the computation of a(i,day) since some fish may be more voracious than others (i.e., silver hake vs. flounder). It is then possible to consider the consequences of different types of feeding behavior on species coexistence.

2.3.2 Prey Selection

The choice of flows between compartments is not based on a fixed and preconceived diet, as in the Parrish (1975) model. The trophic linkages of GEORGE are dependent on prey vulnerability and prey abundance over time. The linkage equations allow for switching behavior and to some extent variable diet preferences.

The feeding selectivity is based upon an empirical analysis of stomach contents. Hahm and Langton (1983) found that there seems to be a central tendency in prey selection based upon the relative predator to prey weight ratios. The distributions of these ratios ($\ln({}^{w}i/w_j)$), where ${}^{w}i$ = predator weight and ${}^{w}j$ = prey weight, suggest that predators select prey whose weight is within a certain fractional range of their own. The mean, ETA, of the weight frequency distributions of prey found in their stomachs (Figure 4 ($\ln{}^{w}i/w_j$) -vs- frequency) indicates the relative weight of the average victim. The central tendencies of the distribution can be explained biologically. Predators shun very small prey as a waste of energy and effort. Conversely, predators avoid large prey candidates as potential threats. The standard deviations, SIGMA, of the frequency distributions indicate how selective a predator is. The larger the standard deviation, the greater the size variability of prey are consumed by the predator. The inverse is also true.

By plotting the mean (ETA) and standard deviations(SIGMA) of these frequency distributions, the predators fall naturally into groups with similar feeding patterns which are in agreement with field observations (Fig. 5; Hahm and Langton, 1983). Simulation of this prey selection process is based on the North Sea modeling work of Andersen and Ursin (1977).

In GEORGE, the feeding of fish is broken down into a two step process. In the first step, the vulnerability of prey to a predator is based on the previously mentioned feeding preference distributions (Figure 4). The measure of the relative vulnerability of the i^{th} prey to the j^{th} predator is g(i,j). This vulnerability measure is derived from the mean Eta(j) and standard deviation Sigma(j) of the j^{th} predator's feeding preference frequency distribution. In the second step, the actual feeding is modified by prey availability and selectivity e(i,j) of the predator.

<u>Step 1: Estimating Vulnerability of Prey to a Predator</u>

$$g(i,j) = f(w_i, w_j, ETA, SIGMA) \quad \text{eqn. 2a}$$

$$g(i,j) = \exp[\ell n({}^{w}i/w_j) - ETA)^2/2(SIGMA^2)] \quad \text{eqn. 2b}$$

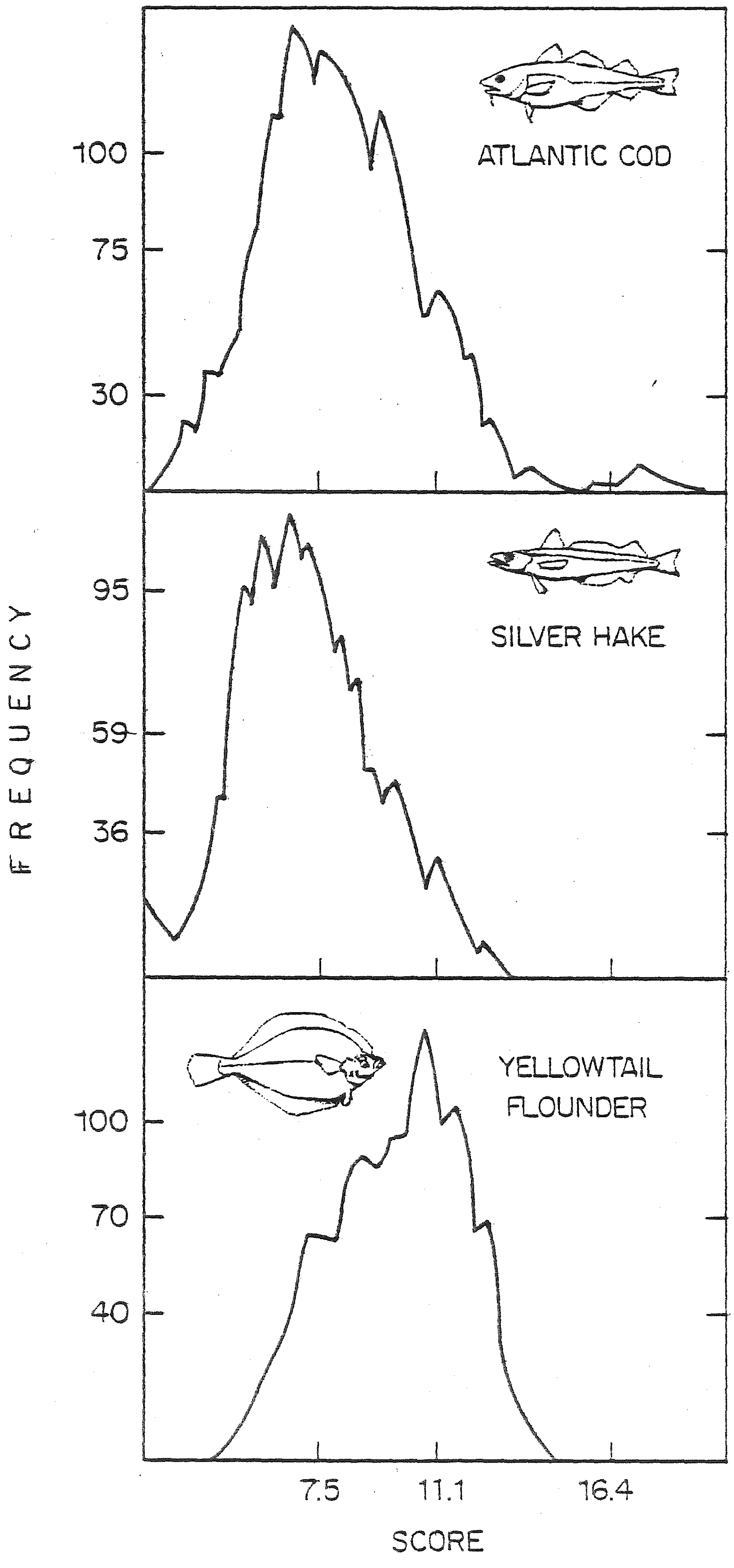
where: w_i = weight of predator

w_j = weight of prey

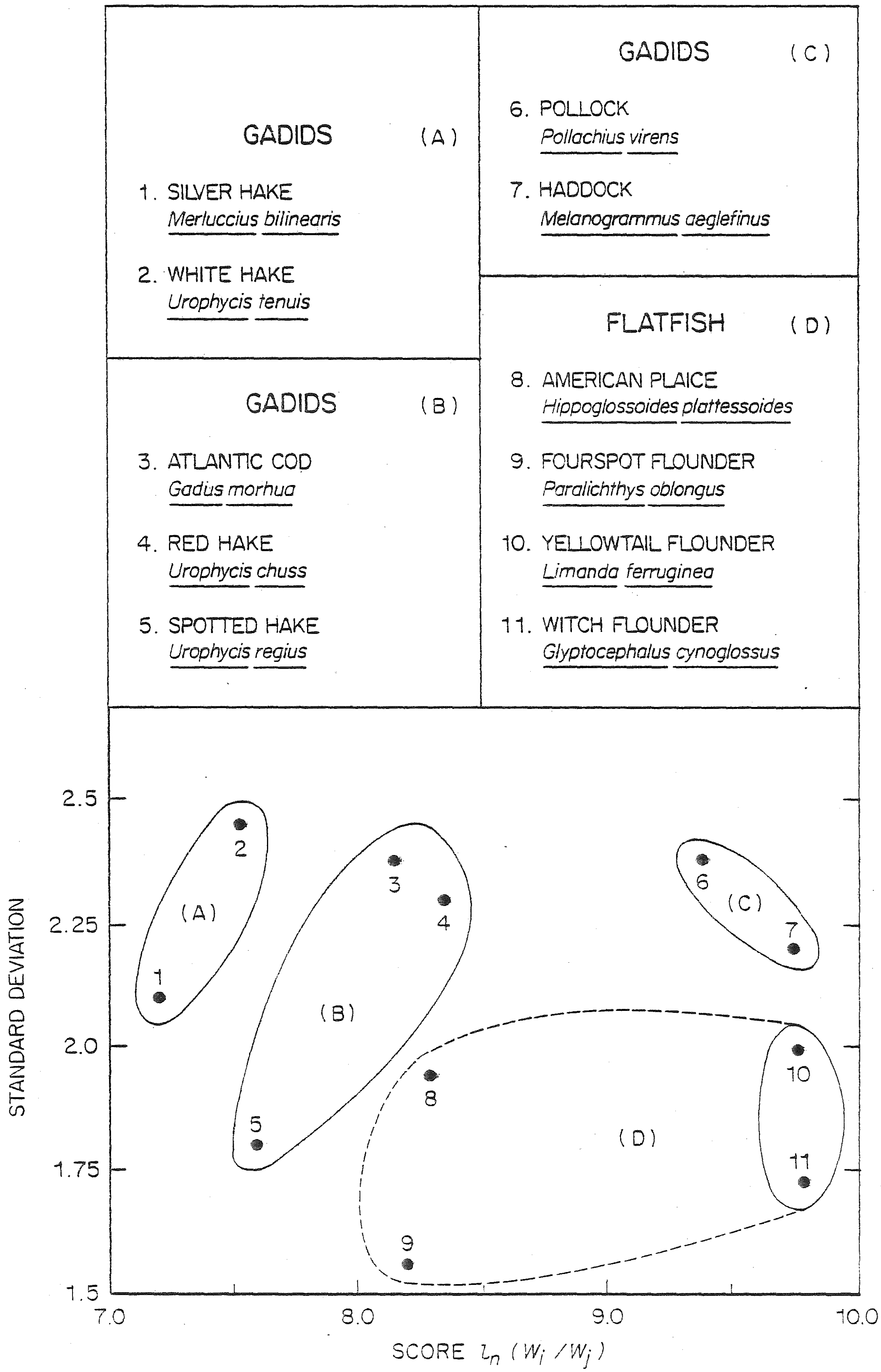
ETA = mean of feeding distribution (see Fig. 6)

SIGMA = standard deviation of feeding distribution (see Fig. 6)

Equation 2b computes g(i,j), where w_i is the predator's per capita weight and w_j is the prey's per capita weight. Figure 6



ATLANTIC COD
SILVER HAKE
YELLOWTAIL FLOUNDER
FREQUENCY
100
75
30
95
59
36
100
70
40
7.5
11.1
16.4
SCORE


GADIDS (A)
1. SILVER HAKE
Merluccius bilinearis
2. WHITE HAKE
Urophycis tenuis
GADIDS (B)
3. ATLANTIC COD
Gadus morhua
4. RED HAKE
Urophycis chuss
5. SPOTTED HAKE
Urophycis regius
GADIDS (C)
6. POLLOCK
Pollachius virens
7. HADDOCK
Melanogrammus aeglefinus
FLATFISH (D)
8. AMERICAN PLAICE
Hippoglossoides plattessoides
9. FOURSPOT FLOUNDER
Paralichthys oblongus
10. YELLOWTAIL FLOUNDER
Limanda ferruginea
11. WITCH FLOUNDER
Glyptocephalus cynoglossus
STANDARD DEVIATION
2.5
2.25
2.0
1.75
1.5
7.0
8.0
9.0
10.0
SCORE $l_n (W_i / W_j)$
(A)
(B)
(C)
(D)
1
2
3
4
5
6
7
8
9
10
11

is a plot of equation 2b over a range of $\ell n(^{w}i/w_j)$. The function is bell shaped, and reaches a maximum of g(i,j) = 1.0, at $[\ell n(^{w}i/w_j)]$ = ETA. The function drops to near zero as $\ell n(^{w}i/w_j)$ becomes much larger or smaller than ETA, (g(i,j) = 0.).

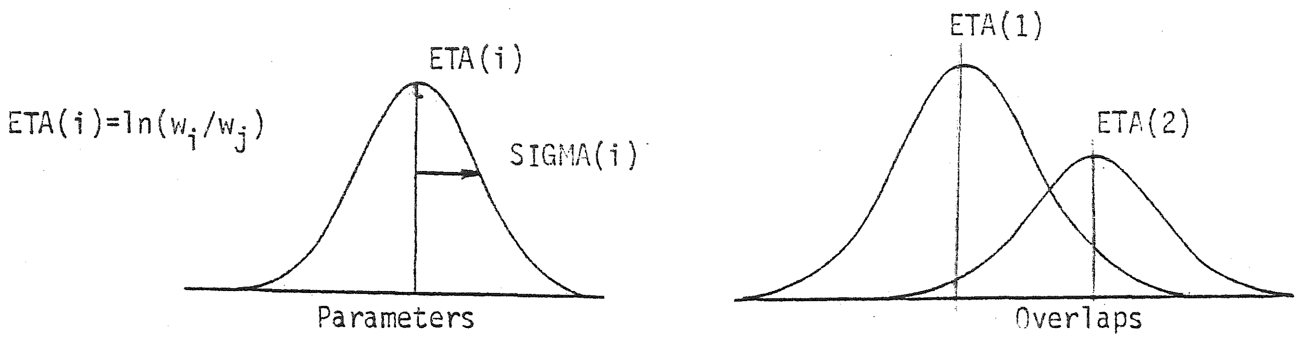


Figure 6. Feeding function parameters and an illustration of feeding overlaps.

For a predator of weight w_i, the prey of varying weight w_j have different vulnerabilities to capture. The value ranges from 1 = very vulnerable to 0 = unsuitable for consumption. By choosing ETA(i) and SIGMA(i) on the basis of the stomach prey distributions (Figure 3), it is possible to partition the resources among the predators.

Step 2: Determining Feeding and Mortality Rates

Let $\Phi_1(i,j)$ be the amount of biomass of prey(j) that is vulnerable to predator(i). Then:

$$\Phi_1(i,j) = g(i,j) \cdot e(i,j) \cdot n(j) \cdot w(j). \qquad \text{eqn. 3}$$

The product $n(j) \cdot w(j)$ represents the biomass of prey(j) available. The term, $g(i,j)$ represents the portion of $n(j) \cdot w(j)$ that is vulnerable to predator(i). The term $e(i,j)$ is a selectivity coefficient set between 0 and 1. It allows the user to modify the food habits of a predator. If food habit studies reveal that predator(i) never feeds on prey(j), then $e(i,j) = 0$. (NB: Even though prey(j) maybe the perfect sized morsel for predator(i), habitats do not always overlap.)

The total amount of prey available to predator(i) is the summation:

$$\Phi_2(i) = \sum_{j=1}^{nprey} \Phi_1(i,j) \qquad \text{eqn. 4}$$

The total amount of prey available to a predator should be greater than the daily ration d(i), (see eqn. 1a) required by the predator. With $\Phi_2(i) > d(i)$, the prey are removed in proportion to their electivity $(e(i,j))$ and abundance. The j^{th} species mortality due to predation is then:

$$M2(j) = \sum_{i=1}^{npred} \frac{\left[\dfrac{\Phi_1(i,j) . d(i)}{\Phi_2(i)}\right]}{n(j) . w(j)} \qquad \text{eqn. 5}$$

The numerator represents the amount of biomass from compartment(j) that is consumed by compartment(i). The mortality for a prey is the sum of all its donations to other predators, divided by the prey's biomass.

2.3.3 Recruitment

GEORGE is affected by recruitment fluctuations. The benthic production is made consistent with historic data (Cohen et al., 1980). Recruitment of the fish is based on the historical recruitment records of Hennemuth et al. (1980). The fish recruitment is a two step process. In the first step, the fish spawn a portion of their biomass.

$$P(i) = s(i).n(i).w(i), \qquad \text{eqn. 6}$$

P(i) is the pool of spawning products and s(i) is the portion of adult biomass spawned. The adult population biomass is reduced by P(i). The number of potential pre-recruits is then:
$n_o = \frac{P(i)}{w_o(i)}$, where w_o is the average weight of a pre-recruit. Stochastic methods are then used to set upper limits for the number of pre-recruits that actually reach commercial size and recruit.

$$R(i) = \text{NORAN}\ (\bar{x}, \text{s.d.}) \qquad \text{eqn. 7}$$

NORAN is a random number generator which produces output whose distribution is normally distributed with a mean = $\bar{x}$, and a standard deviation = s.d. One restriction is that the number of surviving recruits is less than or equal to the upper limit R(i). The surplus of pre-recruits that do not survive, enrich the benthos (BE = Benthic Enrichment).

$$BE = \sum_{i=1}^{n} [n_o(i) - R(i)]. \qquad \text{eqn. 8}$$

2.3.4 The System of Equations

GEORGE involves a system of four differential equations.

Abundance $$\frac{d\tilde{n}}{dt} = ADD(i) - [(M1+M2_i).n_i] - [f_i.n_i] \qquad \text{eqn. 9a}$$

Growth $$\frac{d\tilde{w}}{dt} = w_o(i) + [\frac{w_o(i+1) - w_o(i)}{360} * DAY] \qquad \text{eqn. 9b}$$

Yield $$\frac{d\tilde{y}}{dt} = f_i . n_i . w_i, \text{ where } f_i = \text{fishing mortality} \qquad \text{eqn. 9c}$$

Biomass $$\frac{d\tilde{B}}{dt} = n_i \frac{dw_i}{dt} . w_i \frac{dni}{dt} \qquad \text{eqn. 9d.}$$

The system of equations is integrated through time with a Euler integrator. The year is defined as having 360 days, with 30 day months. Day 1 represents the first day of fall. The desired time step is on the order of 6 days. Presently, per capita weights are interpolated over the course of the year. See eqn. 9b. By specifying the weight of each cohort, and by interpolatng growth, the model simulates an asymptotic growth curve for the fish. The fish abundances result from a balance between additions into an age group (ADD(i)) and mortality due to natural mortality and predation. The age subscript is i. The i^{th} comparment contains all fish between (i-1) and i years of age. When i=1, ADD(i) is the recruitment of pre-recruits to the second age class. When i>1, ADD(i) is the aging process shunting survivors of n(i-1) into age group n(i) at the end of a year.

2.4 The Analytical Section: Flow Analysis

The goal of the analysis loop is to decipher the voluminous results of the simulator. Experience has shown that many models have an uncanny ability to generate believable results through unbelievable means.

To evaluate the activity and validity of a simulation, an eye is kept on the intercompartmental flows. The flow matrix F(i,j,t) represents the n^2-flows of an n-compartment system at time t. A normalized version of this F-matrix, N(i,j,t) represents the normalized flow into a predator. That is,

$$N(i,j,t) = \frac{F(i,j,t)}{\sum_{j=1}^{n} F(i,j,t)}$$

Reading each row of the matrix, the analyst can determine the contribution of each prey type (column element) to the predator's (row element) diet. Initially, analysis is used to debug the computer model's alogrithms. The larger purpose is to interpret and illustrate this system's dynamic behavior.

Flow analysis is a descendent of the input/output economic modeling techniques of Leontief (1970). Given a time series of intercompartmental flows, it is possible to generate system coefficients that describe the patterns of flow, recycling and component interdependencies. It is possible to evaluate the merits of alternative simulated senarios through the examination of their respective system coefficients.

The flows between the ecosystem's compartments are described by the general equation:

$$d\underset{\sim}{X}/dt = A\underset{\sim}{X} \qquad \text{(eqn. 1)}$$

where dX/dt is the change of nutrient level in $\underset{\sim}{X}$. That is:

$$dX_i/dt = \sum_{j=1}^{n} (a_{i,j} * X_j) \qquad \text{(eqn. 2)}$$

There are n^2 linear combinations possible in this n-compartment system. Flow analysis examines the network of flows

$$f_{i,j} = a_{i,j} * X_j \qquad \text{(eqn. 3)}$$

between compartments. By definition, the throughflow of a compartment is the combination of incoming flows from other compartments ($f_{i,j}$), the environment ($Z_{i,0}$) and internal storage ($\dot{X}_i^-$). Throughflow, using the symbols of Figure 1 is:

$$T_i = [\sum_{j=1}^{n} f_{i,j}]-[\dot{X}_i^- + Z_i] \qquad \text{(eqn. 4)}$$

The flows ($f_{i,j}$) can then be expressed in terms of fractions ($q_{i,j}$) of donor throughflow

$$q_{i,j} = f_{i,j}/T_i \qquad \text{(eqn. 5)}$$

Thus equation 4 in matrix form is

$$\underset{\sim}{T} =(Q * \underset{\sim}{T}) + (\dot{X}^- + \underset{\sim}{Z}) \qquad \text{(eqn. 6)}$$

Solving algebraically for the throughflow vector we find:

$$\underset{\sim}{T} = (I-Q)^{-1} * (\dot{\underset{\sim}{X}}^- + \underset{\sim}{Z}) \qquad \text{(eqn. 7)}$$

Intuitively, equation 7 solves for the dynamics of Figure 1. The throughflow (T_i) is a measure of the activity of the i^{th} compartment. The inverse matrix $(I-Q)^{-1}$ is called the normalized inverse matrix (N^{**}). In it, the activity of the i^{th} compartment is apportioned among the inputs to the (n-1) other compartments and itself. It is shown that the activity of the i^{th} compartment is a function of $(\dot{X}^- + \underset{\sim}{Z})$, i.e. $T_i = f(\dot{X}_i^- + Z_i)$.

$$T_i = \sum_{j=1}^{n} [n^{**}_{i,j} * (\dot{X}^{-}_i + Z_i)] \qquad \text{(eqn. 8)}$$

The elements of N^{**}, $n^{**}_{i,j}$ represent the apportioned throughflow caused by a unit input into the i^{th} compartment. The compartmental recycling index is derived from this normalized inverse matrix:

$$RE_i = (n^{**}_{ii} - 1)/n^{**}_{ii} \qquad \text{(eqn. 9)}$$

The cycling index in turn is,

$$CI = (\sum_{j=1}^{n} RE_i * T_i)/TST \qquad \text{(eqn. 10)}$$

The total system throughflow (TST) is the system equivalent of the T_i and is the sum of all the system's compartmental throughflows.

$$TST = \sum_{i=1}^{n} T_i \qquad \text{(eqn. 11)}$$

3.0 Using the Model: A "Cookbook"

3.1 Background

The simulation model GEORGE was originally developed on Woods Hole Oceanographic Institution's Honeywell SIGMA-7 system. It has since been converted to run on VAX 11/780 and PRIME 750 virtual memory systems. The programs are written in ANSI-77 FORTRAN.

To model a problem, the user has the choice of four programs. SETUP allows the user to change the initial conditions and control parameters discussed in the preceding Section 2.3. GEORGE allows the user to run a simulation based on the SETUP controlled coefficients. FLOWF sets up the model's output for Flow Analysis. Finally, FLOWAN is used to run the model output through the Flow Analysis program.

3.2 Technical Details: Running GEORGE on the VAX 11/780 System

The interactions you'll have with the VAX 11/780 can be grouped under the following headings.

- 3.2.1. Copying the model into your own account
- 3.2.2. Dealing with an Operating System
  - a. Setting up the terminal/CRT
  - b. File naming conventions
  - c. Running Programs/Correcting Errors/Hard Copies

3.2.3. Dealing with GEORGE

a. The Dialogue

b. The Occasional Pauses

Before you can do anything, you'll need to copy versions of GEORGE into your own computer account. Once you have a copy of the model you are free to run or edit it to your heart's content. To use this model, however, you'll need to deal with the:

1. VAX operating system and
2. GEORGE.

When you log onto the computer, you'll first notice that every line begins with a dollar sign ($). This is called a system prompt. This prompt signifies that the VAX operating system is ready to accept and process your commands. Only the most rudimentary VAX operating system commands are given in this manual. For a more complete coverage of the subject, refer to the VAX/VMS Command Language User's Guide.

Now for a comforting note. It is all but impossible to destroy a main frame computer from the vantage point of your keyboard. It is, however, possible after only a cursory reading of this manual to get over your head in the software of this modeling system. If you're confused while the terminal churns out nonsense, push the "reset button." This turns all programs off, while not "breaking anything." On the VAX, the "reset button" involves tapping the CONTROL key and the C key simultaneously.

3.2.1. Copying the model into your own account

To use GEORGE, you'll need a National Marine Fisheries Service computer account. The following program files should be copied for your use. The names and purposes of each file are as follows:

File Library

File	Purpose
GEORGE.EXE	Simulator Program
SETUP.EXE	Program that sets up initial conditions for GEORGE
FLOWF.EXE	Program that sets up data file for Flow Analysis
FLOWAN.EXE	Flow Analysis Program
ICØ1.DAT	Standard initial conditions for the model

To copy these files into your account, the following commands are necessary.

```
$COPY  [713.WKH1] GEORGE.EXE  GEORGE.EXE
$COPY  [713.WKH1] SETUP.EXE  SETUP.EXE
$COPY  [713.WKH1] FLOWF.EXE  FLOWF.EXE
$COPY  [713.WKH1] WHFLOW29.EXE  WHFLOW29.EXE
$COPY  [713.WKH1] ICØ1.DAT  ICØ1.DAT
```

3.2.2. Dealing with the VAX Operating System

The VAX Operating System is essentially a highly sophisticated program developed by the Digital Equipment Corpration (DEC) that allows a user to control the VAX computer. The operating system allows the user to operate selected programs. It also allows the user to feed data files to these programs and to store the results of these programs. The principle activities are as follows.

a. Setting up the Terminal

You have the choice of using either a CRT such as the VT100 or a matrix head printer such as the LA-120 DEC-writer (Figure 7). Both terminals allow you up to 132 columns of printed characters. To set the wide format enter the command:

$PLOTTER

The more efficient terminal is the CRT. It is faster and it avoids the accummulation of printed output. All results are stored on disc and, as shall be shown, are easily recalled from memory disc files when necessary.

b. File Naming Conventions

The programs you are about to run require and generate data. Table 1 lists the files that are processed and produced by each program. The interactive design of the programs allows the user to select and name the input and output data files. There is a convention for naming these files.

File Name Format

Name	Purpose
ICnn.DAT:	Initial conditions for biological model (e.g. IC15.DAT)
Rnnn.DAT:	Results of the simulation: A data file (e.g. R251.DAT)
Gnnn.DAT:	Graphics file: Plots and Tables stored on disc (e.g. G252.DAT)
FIJn.DAT:	Flow Matrix file (e.g. FIJ5.DAT)

The user is free to label each file by inserting appropriate identification (nnn, nn, n) suffixes to the files. The suffix characters may be digits (0-9) or letters (A-Z) of the alphabet. The number of characters per label are limited to the number of n's in the file name conventions shown above.

c. Running Programs/Correcting Errors/Hard Copies

The simulator is a library of four (4) computer programs. The names, function, and files required for input and output purposes are given in Table 1.

The programs SETUP, GEORGE, FLOWF, and FLOWAN are stored as compiled load modues. To run these programs one need only type:

$RUN (program name):

e.g.:

$RUN SETUP

$RUN GEORGE

$RUN FLOWF

$RUN.FLOWAN

note: If you make mistakes while using the CRT, just hit the DELETE key and retype your response. If you are using the DECWRITER, it may be simpler to hit CONTROL X and just retype the line.

The four programs are as the term goes, "user friendly." This means that the programs are designed with an interactive dialogue. The program alternately informs you of your choices before asking for your response. Your responses are format free. You needn't worry about the number of spaces or decimal places you use. The program also tries to weed out accidental errors. If an error that could "crash" the program is detected, your answer is rejected and the question is repeated.

The programs also allow you to route all output (tables and graphs) to your disc storage file (e.g. Gnnn.DAT), when you want copies of a large quantity of output. It is faster and more economical to route all your graphics to these disc files and have them printed on the high speed line printer at the WHOI computer center. You can first preview the file with the on-line

EDITOR (see EDT Editor Manual) with the command $EDIT <u>filename</u>. Another option is to enter the command $TYPE <u>filename</u>. This just displays the file on your terminal. To have the results printed on the high speed line printer use the command $PRINT <u>filename</u>.

3.2.3. Dealing with GEORGE

a. The Dialogue

The dialogues in SETUP and FLOWF are simple since they only deal with reading in run condition parameters and initial conditions. The interaction between the user and the program is much more flexible in GEORGE.

There are three dialogue modes in GEORGE. In the first the user is simply providing names of the data files and storage files that will be used by GEORGE. In the second mode the user is allowed to run the model specified in the previous step and display the results in tables and graphs. In the third mode the user can analyze the results using either phase plane plotting methods or Flow Analysis.

In the following sections describing GEORGE, note that it is possible to hop between these three modes. To tackle a problem it is necessary to jump between

i. deciding/designing run parameters
ii. displaying results
iii. analyzing results

The program allows the user to make these transitions. It also allows the user to "back up" the model and rerun portions of the year under different conditions.

b. The Occasional Pauses

The VAX 11/780 is configured as a time-sharing system. When the system is very busy the turn around time slows down. The best time to run large runs is early in the morning or later in the afternoon.

After completing the simulation there is a pause, that results from the time it takes to empty the system data arrays onto the disc storage files, (Rnnn.DAT, Gnnn.DAT, FIJn.DAT). Do not turn the terminal off until the message $FORTRAN STOP appears on the screen. If you log-off before this, you've aborted the job. See Section 3.4 for details.

3.3 SETUP: The Input of Run Coefficients

INPUT FILE: ICij.DAT: Standard Initial Conditions

OUTPUT FILE: ICkl:DAT: Revised Initial Conditions

The program SETUP allows the user to design the simulation run. IC01.DAT contains the standard run coefficients. When run, the program SETUP displays the standard run coefficients, and allows the user to interactively change any value to fit the desired scenario. All responses are read by the computer in free format. Therefore, the user needn't worry about placing the numbers in the proper format. Remember to hit return after entering your response. The program can also detect improper responses and reacts by flagging an error and asking the question again.

Here's an example:

```
==> HOW MANY SIZE CLASSES ARE THERE IN EACH FORM?
SEPARATE NUMBERS WITH COMMAS (MAX=29)
4,5,5,5,5,8
===>YOU HAVE EXCEEDED THE MAXIMUM' NUMBER OF COMPARTMENTS(29):TRY AGAIN
```

So relax...you can't break the machine.

The SETUP session is initiated by logging onto account 713. To run SETUP, enter the command:

```
$RUN SETUP
```

The system responds:

```
PROGRAM:WHSETUP
THIS PROGRAM DISPLAYS AND CHANGES INPUT SPECS
```

The first set of questions ask you to specify names for the file to be altered, and the name of the new file to be created.

```
----------------------------------------------------------------
    INFILE IS THE INPUT DATA FILE TO BE MODIFIED
    OUTFILE IS THE NEW DATA FILE
----------------------------------------------------------------
    FILE SPECIFICATIONS
----------------------------------------------------------------
    1.SPECIFY AN 8 CHARACTER INFILE NAME
     CONVENTION: IC__.DAT
IC01.DAT
     2.SPECIFY AN 8 CHARACTER OUTFILE NAME
     CONVENTION: IC__.DAT
IC05.DAT
================================================================
```

In this example the input file is IC01.DAT. The new file created is IC05.Dat.

The second question concerns the number of life forms there are in the model. The third question asks how many age classes there are in each animal type. Note that questions are preceded with arrow prompts (==>). In this example, the user responds that there are six (6) major animals (the benthos, yellowtail flounder, cod, haddock, silver hake, and herring).

The response to the third question signifies the number of size classes there are for each respective compartment. There are 29 compartments among six animal types. There are four size classes of benthos, and five age classes apiece for the flounders, cod, haddock, silver hake, and herring. You are also asked how many compartments are non-fish. This is because non-fish compartments are treated differently by the model.

```
==>FOR STARTERS,HOW MANY LIFE FORMS ARE THERE?
6
==> HOW MANY SIZE CLASSES ARE THERE IN EACH FORM?
SEPARATE NUMBERS WITH COMMAS (MAX=29)
4,5,5,5,5,5
 ==>HOW MANY OF THE LIFE FORMS ARE NOT FISH?
1
```

The program then sets the values M1 (the natural mortality rate of fish). To change this value, enter either 1 or 3 along with the new value. The program just moves along if 3,0 is entered.

```
M1=  0.5000000E-01 PI=  0.2000000
==>WHICH VARIATE DO YOU WISH TO CHANGE?
1=M1 2=PI 3=NONE OF THE ABOVE
==>ENTER RESPONSE CODE AND  NEW VALUE ;SEPARATE BY COMMA
3,0
```

The user is then given a series of coefficients that control the system equations of the 29 compartments. If no changes are desired, enter a zero.

```
------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE MAXIMUM DAILY RATION AS A % OF BODY WEIGHT(PER YEAR)
   ENTER 1=YES ELSE ENTER A ZERO,(0)
0

------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE MINIMUM DAILY RATION AS A % OF BODY WEIGHT(PER YEAR)
   ENTER 1=YES ELSE ENTER A ZERO,(0)
0

------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE FISHING MORTALITY (PER YEAR)
   ENTER 1=YES ELSE ENTER A ZERO,(0)
0

------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE ABUNDANCE
   ENTER 1=YES ELSE ENTER A ZERO,(0)
0

------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE PER CAPITA WEIGHT IN KILOGRAMS
   ENTER 1=YES ELSE ENTER A ZERO,(0)
0

------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE PREY SELECTION COEFFICIENT (1/2*(SIGMA**2))
   ENTER 1=YES ELSE ENTER A ZERO,(0)
0

------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE PREY SELECTION COEFFICIENT (MEAN(LN(W(I)/W(J))))
   ENTER 1=YES ELSE ENTER A ZERO,(0)
0
```

The size specific coefficients are:

i. A1(i) = maximum daily ration *360. Therefore if MDR=2%, A1(i)=7.2/yr. This is Amax(i) in eqn. 1b.

ii. A2(i) = minimum daily ration *360. Therefore if MDR=.5%, A2(i)=1.8/yr. This is Amin(i) in eqn. 1b.

iii. F1(i) = Fishing mortality on a yearly basis. See eqn. 9a and eqn. 9c.

iv. N(i) = The number of individual organisms in each compartment that would be expected on the first fall day of the simulation.

v. W(i) = The individual weight in kilograms.

vi. GAMMA(i) = $(2*SIGMA^2)^{-1}$ where sigma is the standard deviation in the feeding function as in Figure 4.

vii. ETA(i) = The mean of the feeding function as in Figure 4.

For example, to change the prey selection coefficient the user responds with a "1" to the sixth question:

```
------------------------------------------------------------------
=>WOULD YOU LIKE TO EDIT THE PREY SELECTION COEFFICIENT (1/2*(SIGMA**2))
  ENTER 1=YES ELSE ENTER A ZERO,(0)
1
```

The SETUP program then displays the standard values for these size specific values.

```
-----------------------------------------------------------------------
VARIATE: PREY SELECTION COEFFICIENT (1/2*(SIGMA**2))

 COMPARTMENT   SIZE CLASS    COEFFICIENT,
       1            1         0.1000E+00
       1            2         0.1000E+00
       1            3         0.1000E+00
       1            4         0.1000E+00
 ==>ENTER SIZE CLASS AND NEW COEF:(ENTER 0  IF NO CHANGES ARE NEEDED)
0
,0
       2            1         0.1280E+00
       2            2         0.1280E+00
       2            3         0.1280E+00
       2            4         0.1280E+00
       2            5         0.1280E+00
 ==>ENTER SIZE CLASS AND NEW COEF:(ENTER 0  IF NO CHANGES ARE NEEDED)
0,0
       3            1         0.8900E-01
       3            2         0.8900E-01
       3            3         0.8900E-01
       3            4         0.8900E-01
       3            5         0.8900E-01
 ==>ENTER SIZE CLASS AND NEW COEF:(ENTER 0  IF NO CHANGES ARE NEEDED)
0,0
       4            1         0.1030E+00
       4            2         0.1030E+00
       4            3         0.1030E+00
       4            4         0.1030E+00
       4            5         0.1030E+00
 ==>ENTER SIZE CLASS AND NEW COEF:(ENTER 0  IF NO CHANGES ARE NEEDED)
0,0
       5            1         0.1122E+00
       5            2         0.1122E+00
       5            3         0.1122E+00
       5            4         0.1122E+00
       5            5         0.1250E+00
 ==>ENTER SIZE CLASS AND NEW COEF:(ENTER 0  IF NO CHANGES ARE NEEDED)
0,0
       6            1         0.1280E+00
       6            2         0.1280E+00
       6            3         0.1280E+00
       6            4         0.1280E+00
       6            5         0.1280E+00
 ==>ENTER SIZE CLASS AND NEW COEF:(ENTER 0  IF NO CHANGES ARE NEEDED)
0,0
```

Again, if there are any changes to be made just enter the size class and new value desired (i.e., ? size class, value).

The next set of variables to be examined are the recruitment and spawning coefficients.

```
==>DO YOU WISH TO REVIEW THE
   RECRUITMENT COEFFICIENTS? 1=YES 0=NO
1
```

The program prints out at the terminal the compartment number (1-29), the mean recruitment level observed and the standard deviation of recruitment records. Also specified are the season of spawning (1=fall, 0=spring) and a maturity index indicating whether or not a given age class spawns (1=yes, 0=no).

The last column indicates the percentage of female biomass that would be spawned as eggs.

RECRUITMENT AND SPAWNING COEFFICIENTS

	MEAN RECRUIT	STD.DEV.	SEASON	MATURITY	% INTO EGGS
1	0.5000E+10	0.0000E+00	0.0000E+00	0.1000E+01	0.1000E-03
2	0.5000E+10	0.0000E+00	0.0000E+00	0.1000E+01	0.1000E-03
3	0.5000E+10	0.0000E+00	0.0000E+00	0.1000E+01	0.1000E-03
4	0.5000E+10	0.0000E+00	0.0000E+00	0.1000E+01	0.1000E-03
5	0.0000E+00	0.0000E+00	0.1000E+01	0.0000E+00	0.0000E+00
6	0.6435E+08	0.1517E+08	0.1000E+01	0.1000E+01	0.1000E-02
7	0.6435E+08	0.1517E+08	0.1000E+01	0.1000E+01	0.1000E-02
8	0.6435E+08	0.1517E+08	0.1000E+01	0.1000E+01	0.1000E-02
9	0.6435E+08	0.1517E+08	0.1000E+01	0.1000E+01	0.1000E-02
10	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
11	0.2933E+08	0.7653E+07	0.0000E+00	0.1000E+01	0.1000E-02
12	0.2933E+08	0.7653E+07	0.0000E+00	0.1000E+01	0.1000E-02
13	0.2933E+08	0.7653E+07	0.0000E+00	0.1000E+01	0.1000E-02
14	0.2933E+08	0.7653E+07	0.0000E+00	0.1000E+01	0.1000E-02
15	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
16	0.4956E+08	0.1010E+09	0.0000E+00	0.1000E+01	0.1000E-02
17	0.4956E+08	0.1010E+09	0.0000E+00	0.1000E+01	0.1000E-02
18	0.4956E+08	0.1010E+09	0.0000E+00	0.1000E+01	0.1000E-02
19	0.4956E+08	0.1010E+09	0.0000E+00	0.1000E+01	0.1000E-02
20	0.0000E+00	0.0000E+00	0.1000E+01	0.0000E+00	0.0000E+00
21	0.8623E+09	0.6204E+09	0.1000E+01	0.1000E+01	0.1000E-02
22	0.8623E+09	0.6204E+09	0.1000E+01	0.1000E+01	0.1000E-02
23	0.8623E+09	0.6204E+09	0.1000E+01	0.1000E+01	0.1000E-02
24	0.8623E+09	0.6204E+09	0.1000E+01	0.1000E+01	0.1000E-02
25	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
26	0.1833E+10	0.1039E+10	0.0000E+00	0.1000E+01	0.1000E-02
27	0.1833E+10	0.1039E+10	0.0000E+00	0.1000E+01	0.1000E-02
28	0.1833E+10	0.1039E+10	0.0000E+00	0.1000E+01	0.1000E-02
29	0.1833E+10	0.1039E+10	0.0000E+00	0.1000E+01	0.1000E-02

```
 ==>SPECIFY ROW AND COL TO CHANGE AND NEW COEF
SPECIFY 0,0,0 IF NO CHANGES ARE DESIRED
0,0,0
 ==>ARE THERE ANY MORE CHANGES?1=YES 0=NO
0
```

The final piece of information required before a simulation can be run is the selectivity matrix (see equation 3).

For an n-compartment system there are n^2 possible interactions. Obviously not all of these occur. The <u>E</u>-matrix establishes whether or not a linkage exists between two compartments (1=linkage, 0=no linkage). SETUP displays the <u>E</u>-matrix. To change an element: enter the row, column, and $e_{i,j}$ element. As always hit return when no further changes are desired.

```
THE E(I,J)MATRIX:FLOW FORM J TO I
NOTE THAT THE VALUES ARE ROUNDED TO F3.1
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0 1.0
==>CHANGING THE E(I,J)MATRIX:
  ENTER I,J,VALUE  :LET I=0,0,0 IF ALL IS WELL
0,0,0
```

Once the E-matrix is complete, the program writes the new scenario's input data to a datafile labelled ICk 1.DAT. The program then terminates.

```
==>YOUR RUN PARAMETERS ARE FILED.
FORTRAN STOP
$
```

3.4 The Simulator: GEORGE

INPUT FILE: ICk 1.DAT: Initial Conditions

OUTPUT FILES: Ri j k.DAT: Simulation Results

Rl m n.DAT: Review File Option

Gi j k.DAT: Graphs and Tables

FIJn.DAT : Flow Matrices

Run command file GEORGE.713: RUN GEORGE

The programs are accessed with the command $RUN GEORGE. The system responds:

```
=============================================================
GEORGE : A MULTISPECIES ECOSYSTEM
=============================================================
```

The execute command file requires four files. ICk 1.DAT is the data input file generated by SETUP. The state variable time series storage file Ri j k.DAT is a storage disc file. You can also specify a previously run Rl m n.DAT file to review it. FIJ_.DAT is a flow matrix file generated by this simulator. You specify these names.

```
-------------------------------------------------------------
FILE SPECIFICATIONS:
-----------------------------------------------------------------
    1.SPECIFY A UNIQUE FILE NAME TO STORE THE RESULTS
       OF THE NEW SIMULATION. USE A 8 CHARACTER
     NAME IN FORM OF: R___.DAT


R051.DAT

    2.SPECIFY A UNIQUE FILE NAME FOR GRAPHIC OUTPUT
      8 CHARACTERS IN FORMAT OF: G___.DAT

G051.DAT

     3.IF YOU ARE REVIEWING THE RESULTS OF A PREVIOUS
    SIMULATION RUN...SPECIFY THE 8 CHARACTER NAME
     OF THE ARCHIVED RUN FILE: R___.DAT

R050.DAT

     4.SPECIFY THE INITIAL CONDITION FILE NAME
     8 CHARACTERS: IC__.DAT

IC01.DAT

     5.SPECIFY THE OUTPUT FILE FOR THE
      FLOW ANALYSIS F(I,J) MATRIX: FORM=FIJ_.DAT

FIJ1.DAT
```

The simulator takes the initial conditions and control coefficients specified in IC0 1.DAT and runs them through the model. Upon completion, the simulator then displays the results of these simulation specifications. The results of the simulation are stored on the storage disc file (in this case, R0 5 1.DAT). To make a permanent record of each simulation, define a unique file for each simulation, before accessing GEORGE.

The first decision to be made is whether to run the model based on the new input file (IC0 1.DAT) or to review the results of the simulation stored on the disc.

```
==============================================================
GEORGE : A MULTISPECIES ECOSYSTEM
==============================================================
==>DO YOU WANT TO REVIEW THE RESULTS OF THE
    PREVIOUS SESSION? YES=1  NO=0

0
```

The next set of questions follow.

```
==>ENTER THE NUMBER OF DAYS PER TIMESTEP(INTEGER)

6
==>HOW MANY YEARS DO YOU WISH TO SIMULATE?
==>WITH A TIMESTEP OF     6 DAYS, YOUR  MAX IS   16 YEARS

5
THE SIMULATOR IS RUNNING......
```

Once the integration time-step is set, the simulator calculates on the maximum number of years that can be stored on the output storage file. The user then specifies the number of years to be simulated.

If the user opted to review the previous session, the program immediately sets up the table and graph generators.

IF THE USER OPTED TO SIMULATE A NEW RUN, THE COMPUTER WILL PAUSE FOR A FEW MINUTES. BE PATIENT. ON BUSY DAYS, TIME-SHARING COMPUTER SYSTEMS CAN BE SLOW.

The next phase of the program involves the display of the results in the form of

i. Time series tables and plots,

ii. phase plane plots, and

iii. tables of intercompartmental flows.

i. Time Series Plots and Tables

The first decision to be made is which time series is to be displayed. There are n-time series per category in an n-compartment system.

```
==================================================================
OUTPUT SECTION
=================================================================
==>SPECIFY OUTPUT OPTIONS:
   1=ABUNDANCE
   2=INDIVIDUAL  WEIGHT
   3=YIELD
   4=BIOMASS
   5=NONE  OF THE ABOVE

4
```

In this instance the biomass time series are to be displayed. The user is then directed to specify the table format. The table format specifications include:

i. The first and last day of the time series. Since there are 360/days/year in this model, 5 years = 1800 days. The user is free to plot Days $\leq$ (n years *360).

```
==>TIMESPAN: SPECIFY THE FIRST AND LAST DAY

1,3600
```

ii. The interval between records. On some occasions (debugging sessions) it may be of interest to see the changes one time step at a time. On others, brief ouputs are adequate.

```
==>INTERVAL : SPECIFY #DAYS BETWEEN OUTPUT:
  MAKE IT EQUAL TO OR A MULTIPLE OF THE TIMESTEP

90
```

iii. Since the number of compartments are variable, the user can print up to 12 compartments at a time. If, as in this case, there are more than 12 compartments simply ouput several tables with 12 compartments apiece, and a short table with the remaining compartments.

```
==>HOW MANY COMPARTMENTS PER LINE? MAX=12

12
```

iv. Specify compartments to be displayed.

```
==>ENTER THE SPECIFIC COMPARTMENTS
    SEPARATE EACH ENTRY WITH A COMMA. IE. (1,2,3...N)

1,2,3,4,5,6,7,8,9,10,11,12
```

The first decision is whether the tables should be viewed immediately, or routed to a remote printer for faster printing.

```
==>OUTPUT OPTION:
  ENTER 6 FOR TERMINAL : ENTER 20 FOR THE LINE PRINTER

6
```

When the table is ready for output, the simulator prompts:

```
==>OUTPUT READY : ENTER A ZERO (0)<==

0
```

This allows the user to position the print head of the computer terminal at the top of the page for a neat copy.

==========......BIOMASS.......==========

DAY	1	2	3	4	5	6	7	8	9	10	11	12
IC	.300E+10	.300E+10	.300E+10	.300E+10	.392E+04	.257E+07	.892E+07	.132E+08	.145E+08	.113E+05	.557E+07	.165E+08
6	.300E+10	.298E+10	.298E+10	.299E+10	.469E+04	.260E+07	.899E+07	.133E+08	.144E+08	.412E+07	.108E+08	.252E+08
96	.295E+10	.273E+10	.278E+10	.290E+10	.674E+05	.307E+07	.101E+08	.148E+08	.124E+08	.375E+07	.145E+08	.305E+08
186	.291E+10	.258E+10	.266E+10	.283E+10	.924E+06	.360E+07	.114E+08	.165E+08	.107E+08	.946E+05	.225E+07	.191E+08
276	.288E+10	.250E+10	.257E+10	.278E+10	.136E+08	.397E+07	.126E+08	.182E+08	.919E+07	.496E+05	.306E+07	.233E+08
366	.257E+10	.248E+10	.255E+10	.277E+10	.274E+05	.136E+07	.440E+07	.139E+08	.278E+08	.529E+05	.407E+07	.282E+08
456	.270E+10	.245E+10	.251E+10	.273E+10	.492E+06	.145E+07	.488E+07	.154E+08	.238E+08	.100E+05	.539E+07	.340E+08
546	.276E+10	.244E+10	.249E+10	.272E+10	.725E+07	.152E+07	.533E+07	.169E+08	.204E+08	.121E+06	.175E+04	.699E+07
636	.263E+10	.246E+10	.250E+10	.271E+10	.114E+09	.154E+07	.574E+07	.183E+08	.174E+08	.610E+05	.235E+04	.852E+07
726	.111E+10	.252E+10	.257E+10	.275E+10	.299E+05	.218E+07	.168E+07	.622E+07	.348E+08	.173E+06	.310E+04	.103E+08
816	.211E+10	.251E+10	.254E+10	.272E+10	.558E+06	.218E+07	.182E+07	.681E+07	.296E+08	.320E+05	.406E+04	.123E+08
906	.245E+10	.251E+10	.253E+10	.271E+10	.862E+07	.214E+07	.195E+07	.739E+07	.252E+08	.143E+06	.638E+04	.521E+04
996	.244E+10	.253E+10	.255E+10	.272E+10	.139E+09	.209E+07	.206E+07	.795E+07	.214E+08	.102E+06	.847E+04	.632E+04
1086	.114E+10	.259E+10	.262E+10	.276E+10	.248E+05	.334E+07	.223E+07	.221E+07	.268E+08	.340E+06	.111E+05	.760E+04
1176	.212E+10	.259E+10	.261E+10	.275E+10	.492E+06	.339E+07	.240E+07	.241E+07	.227E+08	.199E+06	.145E+05	.911E+04
1266	.246E+10	.260E+10	.261E+10	.275E+10	.813E+07	.344E+07	.257E+07	.261E+07	.193E+08	.136E+06	.144E+06	.186E+05
1356	.244E+10	.262E+10	.264E+10	.276E+10	.136E+09	.360E+07	.276E+07	.283E+07	.164E+08	.290E+06	.193E+06	.226E+05
1446	.117E+10	.267E+10	.269E+10	.279E+10	.183E+05	.293E+07	.405E+07	.301E+07	.171E+08	.159E+07	.255E+06	.272E+05
1536	.213E+10	.266E+10	.268E+10	.279E+10	.362E+06	.338E+07	.451E+07	.333E+07	.147E+08	.340E+07	.337E+06	.328E+05
1626	.246E+10	.267E+10	.268E+10	.280E+10	.595E+07	.393E+07	.501E+07	.367E+07	.126E+08	.943E+05	.612E+07	.439E+06
1716	.250E+10	.268E+10	.269E+10	.280E+10	.984E+08	.465E+07	.558E+07	.406E+07	.108E+08	.396E+06	.832E+07	.537E+06

After printing out the table the user can then plot out any of these time series plots. To do this, four questions must be answered.

```
=================================================================
TIME-SERIES PLOTS
==>SPECIFY COMPARTMENT(1-29)TO PLOT.
   IF NONE ARE NEEDED ENTER A ZERO



==>WHICH UNIT PLOTS DATA?
 ENTER 6(TERMINAL PRINTER) OR ENTER 20(LINE PRINTER



 ==>THERE ARE  300 RECORDS ON FILE.
    ENTER THE FIRST AND LAST RECORD TO PLOT(1000=MAX)



  LOG TRANSFORM OPTION:
     IF THE DATA VARIES OVER ORDERS OF MAGNITUDE
     LOG IT!  LOG TRAN=1    NO TRANSFORM=0

 ENTER A TITLE FOR THIS PLOT
```

The user is given the choice of using the line printer to display the graph for immediate inspection. It is advisable to use the computer center's high speed line printer to shorten turn around time. The user can pick the portion of the time series to be plotted to view subtle details, by selecting the first and last record to be included on the graph. For convenience, a log-transform can be done on the data before plotting. For the record, a label is attached to every graph. The responses on the previous page would route a plot of the first compartment (300 records) to the line printer.

The following response would route the plot to the user's computer terminal. This is convenient but time-consuming.

```
==================================================================
TIME-SERIES PLOTS
==>SPECIFY COMPARTMENT(1-29)TO PLOT.
   IF NONE ARE NEEDED ENTER A ZERO

1

==>WHICH UNIT PLOTS DATA?
 ENTER 6(TERMINAL PRINTER) OR ENTER 20(LINE PRINTER

6
==>THERE ARE  300 RECORDS ON FILE.
   ENTER THE FIRST AND LAST RECORD TO PLOT(1000=MAX)

1,300

 LOG TRANSFORM OPTION:
    IF THE DATA VARIES OVER ORDERS OF MAGNITUDE
    LOG IT:  LOG TRAN=1   NO TRANSFORM=0

0

ENTER A TITLE FOR THIS PLOT

COMPARTMENT 1: DEMONSTRATION PLOT
```

The user can generate as many plots as desired. When finished, leave the loop by entering a zero.

```
=======================================================
TIME-SERIES PLOTS
==>SPECIFY COMPARTMENT(1-29)TO PLOT.
   IF NONE ARE NEEDED ENTER A ZERO
    ?

0
```

Leaving the graphing loop, the program returns to the output section allowing the user to generate time series tables and plots of other categories. Enter a zero when you have all the time series data you need. Before exiting the time series for good, the system asks if you'd like a time series of other data types. If you're finished with all time series plots, enter a 5.

```
================================================================
OUTPUT SECTION
================================================================
==>SPECIFY OUTPUT OPTIONS:
   1=ABUNDANCE
   2=INDIVIDUAL  WEIGHT
   3=YIELD
   4=BIOMASS
   5=NONE  OF THE ABOVE

5
```

ii. Phase-Plane Plots

Upon leaving the time series plots, the user can examine species relationships. To operate this section the user responses are very similar to the time series section's. The single difference is that it is necessary to indicate the compartments to be plotted on the x- and y-axes. The z-time axis is perpendicular to the plane of the (x-y) plane.

```
=================================================================
PHASE PLANE PLOTS:CHOOSE DATA SET OF INTEREST

==================================================================
OUTPUT SECTION
=================================================================
==>SPECIFY OUTPUT OPTIONS:
   1=ABUNDANCE
   2=INDIVIDUAL  WEIGHT
   3=YIELD
   4=BIOMASS
   5=NONE  OF THE ABOVE

4

=========================
PHASE PLANE PLOT ROUTINE
========================
==>ENTER THE X-AXIS AND Y-AXIS     ID#S
==>OR ENTER A ZERO IF NO PLOTS ARE DESIRED

1,4

==>THERE ARE  300 RECORDS ON FILE.
   ENTER THE FIRST AND LAST RECORD TO PLOT(1000=MAX)

1,300

==>WHICH UNIT PLOTS DATA?
 ENTER 6(TERMINAL PRINTER) OR ENTER 20(LINE PRINTER

6

==>OUTPUT READY : ENTER A ZERO (0)<==

0
```

iii. Flow Analysis:

Upon exiting the phase plane plot section, the program enters its final phase.

```
=================================================================
FLOW MATRIX OPTION:
    YES=1  NO=0
=================================================================
1
```

The user then specifies the day that is to be analyzed for intercompartmental flows. Because the raw and normalized tables of intercompartmental flows cover considerable paper the user is allowed the option of having these tables printed down the street, or onto a disc file.

```
SPECIFY THE DAY TO BE FLOW ANALYZED

90

==>ENTER AN 80 CHARACTER COMMENT FOR FILE LABEL

DAY 90 : DEMONSTRATION RUN

PRINT OPTIONS:
COMPLETE FLOW OUTPUT:ENTER 1
SUPPRESS ALL PRINTOUT:ENTER 0
PRINT JUST A SUMMARY:ENTER 2

2

SELECT OUTPUT MEDIA:TERMINAL=6 LP=20

6
==>OUTPUT READY: ENTER A ZERO (0)

0
FILE LABEL:DAY 90 : DEMONSTRATION RUN

FLOW MATRIX FOR DAY   90 IS STORED'  ON THE FIJ_.DAT DISC FILE
```

In the tables below, for example, F(i,j) reports the flow in kilograms from compartment j to compartment i. N(i,j) is the normalized flow from j to i. N(i,j) is then the percentage contribution of the j^{th} compartment to the i^{th} predator.

```
--COMPARTMENTAL FLOWS---
F(14, 1)=0.837E+02 F(14, 2)=0.290E+05 F(14, 3)=0.138E+06 F(14, 4)=0.262E+06
F(14, 5)=0.423E+02 F(14, 6)=0.798E+04 F(14, 7)=0.994E+04 F(14, 8)=0.781E+04
F(14, 9)=0.432E+04 F(14,10)=0.244E+05 F(14,11)=0.585E+04 F(14,12)=0.424E+04
F(14,13)=0.181E+04 F(14,14)=0.000E+00 F(14,15)=0.392E+03 F(14,16)=0.853E+04
F(14,17)=0.712E+04 F(14,18)=0.545E+04 F(14,19)=0.726E+04 F(14,20)=0.419E+03
F(14,21)=0.122E+06 F(14,22)=0.109E+06 F(14,23)=0.746E+05 F(14,24)=0.501E+05
F(14,25)=0.200E+04 F(14,26)=0.431E+05 F(14,27)=0.272E+06 F(14,28)=0.231E+06
F(14,29)=0.377E+06 F(
```

```
---NORMALIZED FLOWS

N(14, 1)=0.464E-04 N(14, 2)=0.161E-01 N(14, 3)=0.766E-01 N(14, 4)=0.145E+00
N(14, 5)=0.234E-04 N(14, 6)=0.442E-02 N(14, 7)=0.550E-02 N(14, 8)=0.432E-02
N(14, 9)=0.239E-02 N(14,10)=0.135E-01 N(14,11)=0.324E-02 N(14,12)=0.235E-02
N(14,13)=0.100E-02 N(14,14)=0.000E+00 N(14,15)=0.217E-03 N(14,16)=0.472E-02
N(14,17)=0.394E-02 N(14,18)=0.302E-02 N(14,19)=0.402E-02 N(14,20)=0.232E-03
N(14,21)=0.674E-01 N(14,22)=0.602E-01 N(14,23)=0.413E-01 N(14,24)=0.277E-01
N(14,25)=0.111E-02 N(14,26)=0.239E-01 N(14,27)=0.151E+00 N(14,28)=0.128E+00
N(14,29)=0.209E+00 N(
```

An alternative to the detailed output of flow and normalized matrices is the summary option. The summary output lists the mortality losses to the 29 compartments due to natural cause, (M1(i)), fishing (FM(i)), and predation (PR(i)).

```
----------------------------------------------------------------
NATURAL,FISHING AND PREDATION SUMMARIES

M1( 1)=0.246E+07 M1( 2)=0.229E+07 M1( 3)=0.233E+07 M1( 4)=0.242E+07
M1( 5)=0.471E+02 M1( 6)=0.253E+04 M1( 7)=0.838E+04 M1( 8)=0.123E+05
M1( 9)=0.104E+05 M1(10)=0.317E+04 M1(11)=0.119E+05 M1(12)=0.251E+05
M1(13)=0.415E+05 M1(14)=0.502E+05 M1(15)=0.362E+02 M1(16)=0.552E+04
M1(17)=0.184E+05 M1(18)=0.476E+05 M1(19)=0.888E+05 M1(20)=0.424E+03
M1(21)=0.447E+05 M1(22)=0.655E+05 M1(23)=0.735E+05 M1(24)=0.836E+05
M1(25)=0.349E+03 M1(26)=0.614E+04 M1(27)=0.119E+06 M1(28)=0.200E+06
M1(29)=0.385E+06 M1(

FM( 1)=0.000E+00 FM( 2)=0.000E+00 FM( 3)=0.000E+00 FM( 4)=0.000E+00
FM( 5)=0.000E+00 FM( 6)=0.000E+00 FM( 7)=0.000E+00 FM( 8)=0.000E+00
FM( 9)=0.104E+06 FM(10)=0.000E+00 FM(11)=0.000E+00 FM(12)=0.000E+00
FM(13)=0.000E+00 FM(14)=0.502E+06 FM(15)=0.000E+00 FM(16)=0.000E+00
FM(17)=0.000E+00 FM(18)=0.000E+00 FM(19)=0.888E+06 FM(20)=0.000E+00
FM(21)=0.000E+00 FM(22)=0.000E+00 FM(23)=0.000E+00 FM(24)=0.836E+06
FM(25)=0.000E+00 FM(26)=0.000E+00 FM(27)=0.000E+00 FM(28)=0.000E+00
FM(29)=0.000E+00 FM(

PR( 1)=0.133E+07 PR( 2)=0.190E+08 PR( 3)=0.157E+08 PR( 4)=0.653E+07
PR( 5)=0.614E+04 PR( 6)=0.314E+05 PR( 7)=0.270E+05 PR( 8)=0.180E+05
PR( 9)=0.881E+04 PR(10)=0.183E+06 PR(11)=0.121E+05 PR(12)=0.747E+04
PR(13)=0.181E+04 PR(14)=0.000E+00 PR(15)=0.599E+04 PR(16)=0.248E+05
PR(17)=0.144E+05 PR(18)=0.937E+04 PR(19)=0.121E+05 PR(20)=0.584E+05
PR(21)=0.443E+06 PR(22)=0.322E+06 PR(23)=0.185E+06 PR(24)=0.107E+06
PR(25)=0.105E+06 PR(26)=0.294E+06 PR(27)=0.925E+06 PR(28)=0.625E+06
PR(29)=0.930E+06 PR(
```

iv. Further Runs

By now, you the user, should be sitting on quite a pile of tables and graphs. The sensible things to do would be to sit back and analyze the results of the simulation. By invoking $RUN SETUP.713, the user can vary the initial conditions and do a comparative run by changing the disc file and executing GEORGE.713.

The alternative is to continue the simulation to extend the run further into the future.

```
============================================================
==>RUN SCHEDULE
============================================================
  ENTER THE DAY YOU WOULD LIKE TO START THE MODEL ON
  ENTER A ZERO IF YOU DONT WANT TO CONTINUE

0
```

If a non-zero Julian day were entered the initial state variables would be set for that day and the process begins anew.

Upon termination the system responds: ==>SIMULATION ENDED

Wait for the system to respond: $FORTRAN STOP

C A U T I O N $

IF YOU TURN OFF THE TERMINAL BEFORE THE $ PROMPT APPEARS, YOU HAVE INADVERTANTLY DELETED THE ARCHIVE FILES R_ _ _.DAT, G_ _ _.DAT, and FIJ_.DAT. THE PAUSE BEFORE THE $ PROMPT'S APPEARANCE IS THE TIME REQUIRED TO STORE ALL DATA ON THE VAX DISC. <u>ON BUSY DAYS, THIS CAN BE A FEW MINUTES</u>.

3.5 FLOWF: Flow Analysis Data Preparation

INPUT FILE: FIJn.DAT: Flow matrices to be analyzed

OUTPUT FILES: FLW1.DAT: Flow data to be analyzed with analysis parameters

MATRIX.DAT: Aggregated matrix and vectors

FLOWOUT.DAT: Flow Analysis Print Out

The program FLOWF plays an analogous role to SETUP. This program takes the intercompartmental flow matrix, FIJn.DAT (produced by the simulator) and prepares it for a flow analysis program to follow. To access the file, log into account 713 and enter: $RUN FLOWF

The system responds:

```
--------------------------------------------------------
     ENTER THE FILE TO BE ANALYZED:FORM= FIJ_.DAT

FIJ1.DAT

--------------------------------------------------------
```

The user must first label the run with a title and specify how many compartments are included in the model.

```
THIS PROGRAM HANDLES UP TO 29 COMPARTMENTS
==>ENTER A 68-CHARACTER TITLE : BE BRIEF

DAY 90 : DEMONSTRATION

==>ENTER THE NUMBER OF COMPARTMENTS

29
```

The user then specifies whether:

1. The initial conditions are required.

```
==>PRINT OUT OPTIONS
  1.IDIAG=1 INITIAL CONDITIONS PRINTED
  2.IDIAG=2 INITIAL CONDITIONS SUPPRESSED
  ==>ENTER IDIAG

1
```

2. Whether the normalized flow matrices of all compartments are desired.

```
  1.IFLOW=0 PRINT N*,N** FOR COMPARTMENTS WITH I/O
  2.IFLOW=1 PRINT N* AND N** FOR ALL COMPARTMENTS
  3.IFLOW=2 SUPPRESS PRINT
==>ENTER IFLOW

1
```

The program is then ready to read in the flow matrices. To enter the FIJ_.DAT matrix, enter "15". If you wish to enter a matrix by hand, enter "5".

```
F-MATRIX:SPECIFY INPUT MEDIA
ENTER 5:IF F(I,J) IS  ENTERED BY HAND.
ENTER 15 :IF FILE FIJ_.DAT IS TO BE ACCESSED

15
```

If there are inputs (z(i)) to the i^{th} compartments or outflows (y(i)) from the j^{th} compartments, these must be listed in the following i/o. In this case there are none, so all values are zero.

```
==>ENTER Z-VECTOR <MAX=10 VALUES PER LINE  >
==>ENTER IN FREE FORMAT(SEPARATE WITH COMMAS

0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
==>ENTER THE Y-VECTOR <MAX= 10 VALUES PER LINE  >
==>ENTER IN FREE FORMAT(SEPARATE WITH COMMAS

0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0,0
0,0,0,0,0,0,0,0,0
```

It is possible to aggregate the compartments and flows of the system to simplify the analysis task. The following section shows how to recombine the 29 compartment model into a 7 compartment system. A new normalized row and normalized flow matrix are displayed. These changes are stored on a disc file MATRIX.DAT for future reference.

---->DAY 90

THE AGGREGATED MATRIX F(I,J) J--->I IN KILOGRAMS

	1	2	3	4	5	6
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	0.1212E+07	0.1216E+02	0.7588E+01	0.8859E+00	0.1135E+03	0.8988E+02
3	0.1843E+07	0.6243E+05	0.8262E+05	0.5393E+05	0.7514E+06	0.1951E+07
4	0.5230E+07	0.1119E+05	0.3908E+05	0.5221E+04	0.1419E+06	0.3468E+06
5	0.8728E+07	0.1742E+05	0.8252E+05	0.7541E+04	0.2202E+06	0.5802E+06
6	0.2558E+08	0.2997E+03	0.6407E+02	0.9497E+01	0.2736E+04	0.1410E+04

THE NORMALIZED ROWS

	1	2	3	4	5	6
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	0.9998E+00	0.1003E-04	0.6261E-05	0.7309E-06	0.9364E-04	0.7416E-04
3	0.3885E+00	0.1316E-01	0.1742E-01	0.1137E-01	0.1584E+00	0.4112E+00
4	0.9058E+00	0.1938E-02	0.6769E-02	0.9043E-03	0.2458E-01	0.6006E-01
5	0.9058E+00	0.1808E-02	0.8564E-02	0.7826E-03	0.2285E-01	0.6021E-01
6	0.9998E+00	0.1172E-04	0.2505E-05	0.3712E-06	0.1069E-03	0.5511E-04

THE NORMALIZED MATRIX

	1	2	3	4	5	6
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	0.2581E-01	0.2590E-06	0.1616E-06	0.1887E-07	0.2417E-05	0.1914E-05
3	0.3925E-01	0.1330E-02	0.1760E-02	0.1149E-02	0.1600E-01	0.4155E-01
4	0.1114E+00	0.2383E-03	0.8325E-03	0.1112E-03	0.3023E-02	0.7386E-02
5	0.1859E+00	0.3711E-03	0.1758E-02	0.1606E-03	0.4691E-02	0.1236E-01
6	0.5448E+00	0.6384E-05	0.1365E-05	0.2023E-06	0.5827E-04	0.3003E-04

The program is then ready to read in another matrix to be analyzed. If another matrix is to be analyzed the program loops back to the beginning of this program. To terminate enter a zero.

```
THE FILE FLW1.DATA HAS BEEN CREATED
==>ENTER 1 IF THERE ARE MORE ANALYSES
   ELSE ENTER 0
0
```

The system responds:

```
THE AGGRAGATED MATRICES AND THE ASSOCIATED
 NORMALIZED ROWS AND MATRIX ARE STORED IN FILE:MATRIX.DAT
FORTRAN STOP
$
```

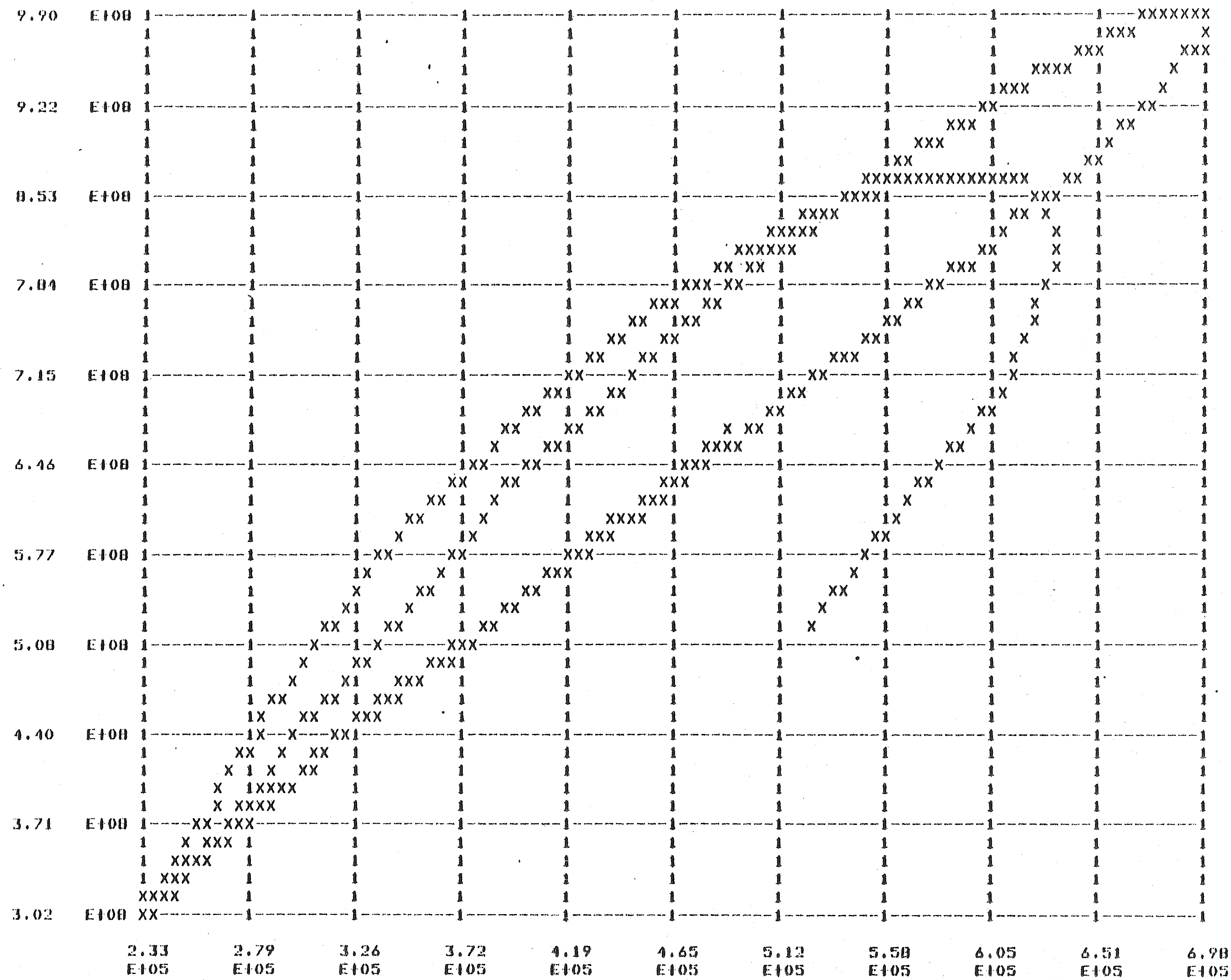
To execute the flow analysis, just say:

$RUN FLOWAN

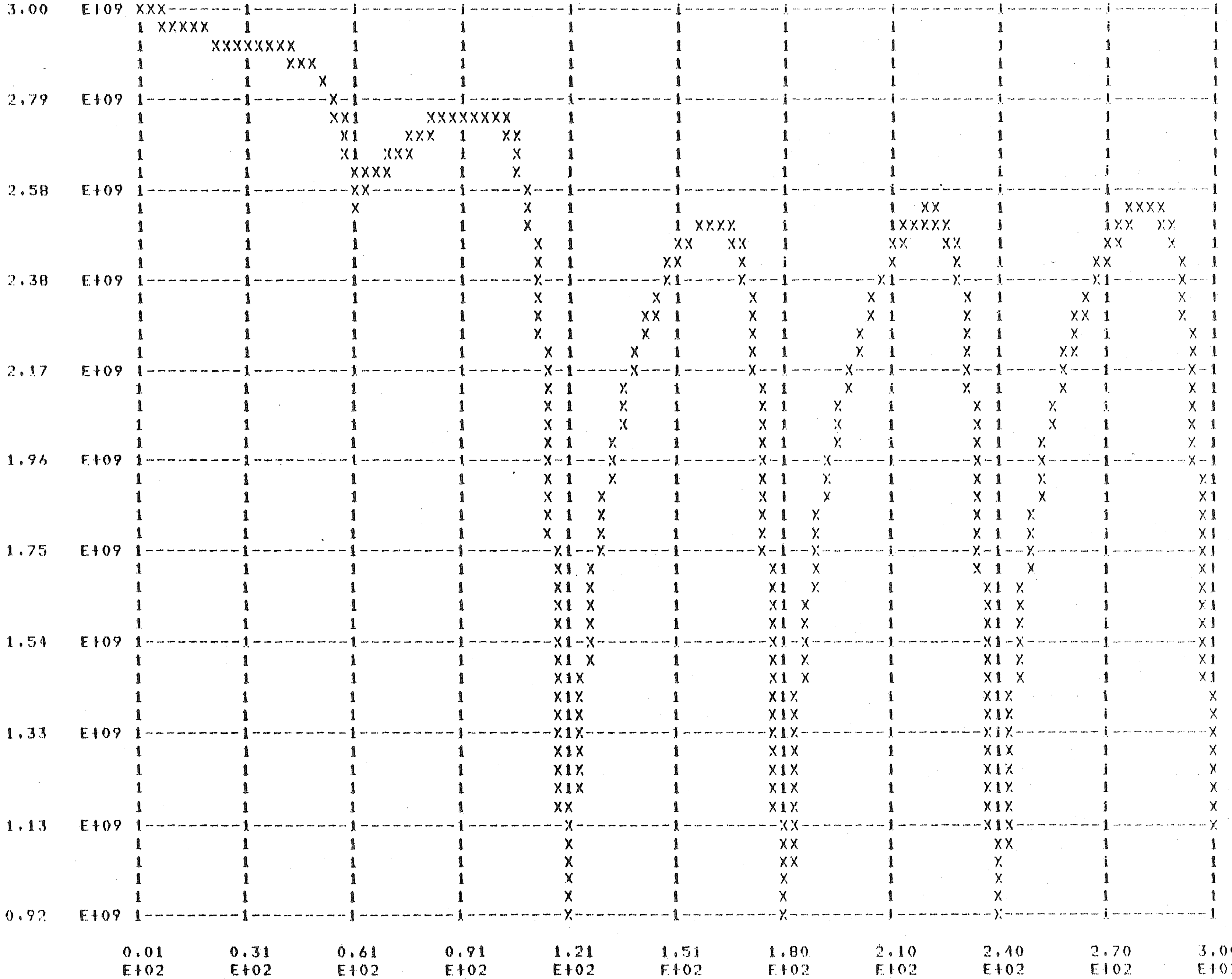
The flow analysis print out is obtained by running FLOWOUT, in account 713.

The system responds by assigning an ID number to the run. Log off... and think.

Illustrations



GENERAL PLOTSICHOOSE DATA SET



====>PLOT ID:COMPARTMENT 1: DEMONSTRATION PLOT <====

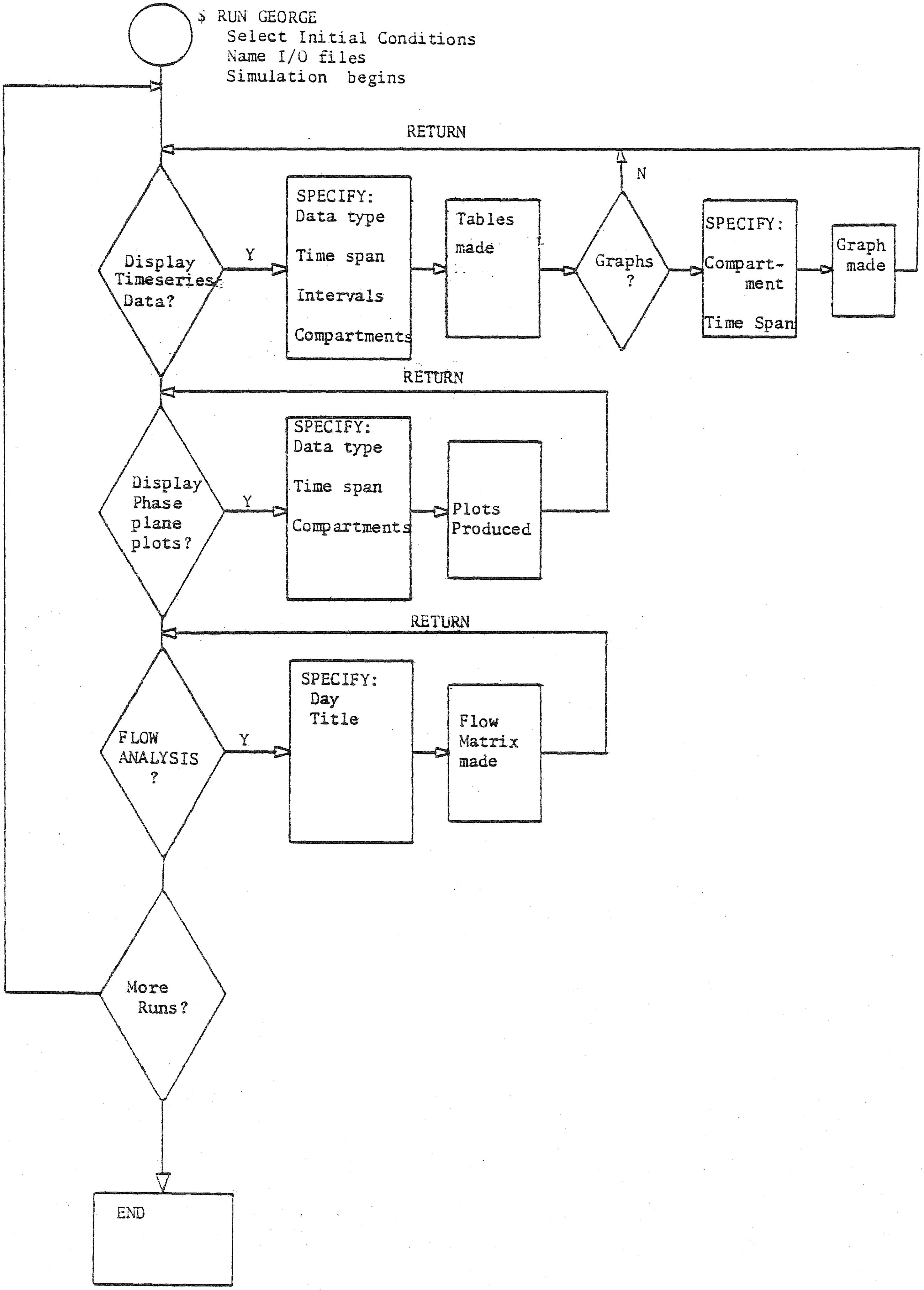

$ RUN GEORGE
Select Initial Conditions
Name I/O files
Simulation begins
RETURN
N
Display Timeseries Data?
Y
SPECIFY: Data type Time span Intervals Compartments
Tables made
Graphs ?
SPECIFY: Compart- ment Time Span
Graph made
RETURN
Display Phase plane plots?
Y
SPECIFY: Data type Time span Compartments
Plots Produced
RETURN
FLOW ANALYSIS ?
Y
SPECIFY: Day Title
Flow Matrix made
More Runs?
END

TABLE 1. Programs and Files Required to Run GEORGE

File Name	Purpose	Files Required	Files Generated	
LOGIN.COM	Command Files for VAX	None	None	
IC01.DAT	Standard Initial Conditions	None	None	
SETUP.EXE	Display and Revise Initial Conditions	IC01.DAT	ICnn.DAT	Initial Conditions
GEORGE.EXE	Run Simulation Model and Display Results	ICnn.DAT Rnnn.DAT	Rnnn.DAT Gnnn.DAT FIJn.DAT	Storage of Simulation Results Graphics Storage Flow Matrix Storage
FLOWF.EXE	Prepare Flow Matrix for Analysis	FIJn.DAT	MATRIX.DAT FLW1.DAT	Aggregated System Summary Input for FLOWAN
FLOWAN.EXE	Flow Analysis	FLW1.DAT	FLOWOUT.DAT	Flow Analysis Results

Bibliography

Caswell, H. (1976) The validation problem. In: B.C. Patten, System Analysis and Simulation in Ecology, Vol. IV. Academic Press, N.Y.

Cohen, E., M. Grosslein, M. Sissenwine, and F. Steimle (1979) An energy budget of Georges Bank. Presented at: Multispecies Approaches to Fisheries Management, St. John's Newfoundland, 26-30 November 1979.

Finn, J. T. (1977) Flow analysis: A method of tracing flows through ecosystem models. University of Georgia Ph.D. Thesis. University Microfilms International. Ann Arbor, Michigan.

Hahm, W. (1982) GEORGE: a simulation model and flow analysis system user manual. NMFS, Woods Hole Lab Ref. No. 82-07.

Hahm, W. and R. Langton (1983) Prey selection based on predator/prey weight ratios for some northwest Atlantic fish. Marine Ecology Progress Series (In Press).

Hennemuth, R. C., J. E. Palmer, and B. E. Brown (1980) A statistical description of recruitment in eighteen selected fish stocks. J. Northw. Atl. Fish. Soc., Vol. 1:101-111

Leontief, W. (1970) The Dynamic Inverse. In: A.P. Carter and A. Brody (eds.), Contribution to Input-Output Analysis. North Holland Publishing Company. Amsterdam, London.

Mauer, R. and R. Bowman (1975) Food chain investigations. NMFS, Woods Hole Lab. Ref. No. 75-03.

May, R., J. Beddington, C. W. Clark, S. J. Holt, R. M. Laws (1979) Management of multispecies fisheries. Science (205)4403: 267-275.

Pielou, E. C. (1981) The usefulness of ecological modeling: A stock taking. Quarterly Review of Biology 56(1): 17-31.

Watson, V. and O. Loucks (1979) An analysis of turnover times in a lake ecosystem and some implications for system properties. In: E. Halfon (ed.), Theoretical Systems Ecology. Academic Press, N.Y.

Watt, K. E. F. (1975) Critique and comparison of biome ecosystem modeling. In: B.C. Patten (ed.), System Analysis and Simulation in Ecology, Vol. III. Academic Press, N.Y.