International Institute for
Applied Systems Analysis
IIASA www.iiasa.ac.at

Advanced Decision-Oriented Software for the Management of Hazardous Substances: Part II: A Demonstration Prototype System

Fedra, K.

IIASA Collaborative Paper
February 1986

ADVANCED DECISION-ORIENTED SOFTWARE FOR THE MANAGEMENT OF HAZARDOUS SUBSTANCES:

Part II: A Demonstration Prototype System

Kurt Fedra

February 1986
CP-86-10

INTERNATIONAL INSTITUTE FOR APPLIED SYSTEMS ANALYSIS
2361 Laxenburg, Austria

ACKNOWLEDGEMENTS

The research described in this report is sponsored by the Commission of the European Communities' (CEC) Joint Research Centre (JRC), Ispra Establishment, under Study Contracts No.2524-84-11 ED ISP A and No. 2748-85-07 ED ISP A. It was carried out by IIASA's Advanced Computer Applications (ACA) project, within the framework of the CEC/JRC Industrial Risk Programme, and in cooperation with the Centre's activities on the Management of Industrial Risk.

The author is indebted to Drs. G. Volta, B. Versino, and H. Otway for their continuing support and constructive criticism.

Special thanks are due to the ACA project team, Stefan Bosnjakovic, Anna John, Elisabeth Weigkricht, and Lothar Winkelbauer, and the visiting or guest scholars J. Bartnicki, H.-J. Diersch, M. Grauer, S. Kaden, D.P. Loucks, M. Posch, G. Römer, T. Rhys, C. Schönowitz, M. Skocz, K. Strzepek, R. Vetschera, M. Zebrowski and W. Ziembla who contributed substantially to this project.

The opinions expressed in the report are those of the author and do not necessarily reflect those of IIASA or of IIASA's National Member Organizations. Neither the Commission of the European Communities, the Joint Research Centre, Ispra Establishment, nor any person acting on behalf of the above is responsible for the use which might be made of the information in this report.

TABLE OF CONTENTS

1. The Structure of the Demonstration Prototype	1
1.1 Information Management and Decision Support	2
1.2 Model Integration and User Interface	3
1.3 System Implementation	4
1.4 The Scope of the Demonstration Prototype System	5
2. Components of the Demonstration Prototype	8
2.1 Master Menu Level	7
2.1.01 Chemical Substances Data Bases	10
2.1.02 Industrial Accident Reports	19
2.1.03 Legislation and Regulations	21
2.1.04 Regional and Geographical Data Bases	23
2.1.05 Chemical Industry Data Bases	26
2.1.06 Chemical Industry Analysis	28
2.1.07 Chemical Process Plant Analysis	37
2.1.07.1 The Symbolic Process Simulator	37
2.1.07.2 Chemical Process Optimization	40
2.1.08 Waste: Treatment and Disposal	42
2.1.09 Waste: Industrial Waste Streams	44
2.1.10 Transportation System Analysis	49
2.1.11 Environmental Impact Assessment	55
2.1.11.1 Environmental Models Master Menu	55
2.1.11.2 TOXSCREEN Multi-Media Screening Level Model	55
2.1.11.3 FEFLOW 2D Groundwater Pollution	57

2.1.11.4 LRAT Long-Range Atmospheric Transport 66
2.1.12 Multi-Criteria Data Evaluation and Optimization 71
2.1.13 Explain Current Options 84

2.2 Implementation Structure: File Systems 86

2.3 Software Tools and Libraries 87

3. Status and Outlook 88
3.1 Components of the Demonstration Prototype: Operational Modules 88
3.1.01 Graphical User Interface, Menus 88
3.1.02 Data Bases 88
3.1.03 Simulation/Optimization Models 89

4. References and Selected Bibliography 91

APPENDIX:

A1. Summary Project Description 96
A1.1 Background 96
A1.2 Project Objectives 96

ADVANCED DECISION-ORIENTED SOFTWARE FOR THE MANAGEMENT OF HAZARDOUS SUBSTANCES:

Part II: A Demonstration Prototype System

Kurt Fedra

1. THE STRUCTURE OF THE DEMONSTRATION PROTOTYPE

This report describes the implementation of a first demonstration prototype of an integrated, interactive, computer-based decision support and information system for the management of hazardous substances. Design guidelines and the overall structure of the system have been described in Fedra (1985).

Recognizing the potentially enormous development effort required and the open-ended nature of such a project, we have opted for a strategy that takes advantage of the large volume of scientific software already available. A modular design philosophy enables us to develop individual building blocks, which are valuable products in their own right, in the various phases of the project. This also makes it possible to interface and integrate the modules in a framework which, above all, has to be flexible and easily modifiable with growing experience of use.

The demonstration prototype can be constructed at relatively low cost and only incremental effort, by using an open architecture concept for this framework, with a functional and problem-oriented, rather than a structural and methodological design

The system design combines several methods of applied systems analysis and operations research, planning and policy sciences, and artificial intelligence into one fully integrated software system. The basic objective is to provide a broad group of users direct and easy access to these largely formal and complex methods (see Appendix A1 for a summary description of the overall project).

1.1 Information Management and Decision Support

The sheer complexity of the management of hazardous substances and related risk assessment problems calls for the use of modern information processing technology. However, most problems that go beyond the immediate technical design and operational management level involve as much politics and psychology as science.

The demonstration prototype system described here is based on ***information management*** and ***model-based decision support***. It envisions experts as its users, as well as decision and policy makers, and in fact, the computer is seen as a mediator and translator between expert and decision maker, between science and policy. The computer is thus not only a vehicle for analysis, but even more importantly, a vehicle for communication, learning, and experimentation.

The three basic, though inseparably interwoven elements, are

- to supply ***factual information, based on existing data***, statistics, and scientific evidence,
- to assist in ***designing alternatives*** and to assess the likely consequences of such new plans or policy options, and
- to assist in a systematic ***multi-criteria evaluation and comparison*** of the alternatives generated and studied.

The framework foresees the selection of criteria for assessment by the user, and the assessment of scenarios or alternative plans in terms of these criteria. The evaluation and ranking is again done partly by the user, where the machine only assists through the compilation and presentation of the required information, and partly by the system, on the basis of user-supplied criteria for screening and selection.

The selected approach for the design of this software system is eclectic as well as pragmatic. We use proven or promising building blocks, and we use available modules where we can find them. We also exercise methodological pluralism: any "model", whether it is a simulation model, a computer language, or a knowledge representation paradigm, is by necessity incomplete. It is only valid within a small and often very specialized domain. No single method can cope with the full spectrum of phenomena, or rather points of view, called for by an interdisciplinary and applied science.

The direct involvement of experts and decision makers shifts the emphasis from a production-oriented "off line" system to an explanatory, learning-oriented style of use. The decision support and expert system is as much a tool for the expert as it is a testing ground for the decision maker's options and ideas.

In fact, it is the ***invention*** and definition of options that is at least as important as the estimation of their consequences and evaluation. For planning, policy and decision making, the generation of new species of ideas is as important as the mechanisms for their selection. It is such an evolutionary understanding of planning that this software system is designed to support. Consequently, ease of use and the necessary ***flexibility and expressive power*** of the software system are the central focus of development.

1.2 Model Integration and User Interface

The basic elements of a model-based decision support and information system as outlined above are the following.

From a user perspective, the system must first and foremost be able to assist in its own use, i.e., explain what it can do, and how it can be done. The basic conceptual components of this system are the following:

- *the interactive user interface* that handles the dialog between the users(s) and the machine; this is largely menu driven, that is, at any given point the user is offered several possible actions which he can select from a menu of options provided by the system;
- *a task scheduler or control program*, that interprets the user request – and, in fact, helps to formulate and structure it – and coordinates the necessary tasks (program executions) to be performed; this program contains the "knowledge" about the individual component software modules and their interdependencies;
  the control program can translate a user request into either:
  - a data/knowledge base query;
  - a request for "scenario analysis"

  the latter will be transferred to
- *a problem generator*, that assists in defining scenarios for simulation and/or optimization; its main task is to elicit a consistent and complete set of specifications from the user, by iteratively resorting to a data base and/or knowledge base to build up the *information context or frame* of the scenario. A scenario is defined by a delimitation in space and time, a set of (possibly recursively linked) processes, a set of control variables, and a set of criteria to describe results. It is represented by
- *a set of process-oriented models*, that can be used in either simulation or optimization mode. The results of creating a scenario and either simulating or optimizing it are passed back to the problem generator level through a
- *evaluation and comparison module*, that attempts to evaluate a scenario according to the list of criteria specified, and assists in organizing the results from several scenarios. For this comparison and the presentation of results, the system uses a
- *graphical display and report generator*, which permits selection from a variety of display styles and formats, and, in particular, facilitates viewing the results of the scenario analysis in graphical form. Finally, although not directly realized by the user, the system employs a
- *systems administration module*, which is largely responsible for housekeeping and learning: it attempts to incorporate information gained during a particular session into the permanent data/knowledge bases and thus allows the system to "learn" and improve its information background from one session to the next.

These conceptual elements cannot directly be mapped into corresponding software modules; most of the above described functions are embedded in several program elements. Also, it must be pointed out that most of these elements are linked recursively. For example, a scenario analysis will usually imply several data/knowledge base queries in order to make the frame and necessary parameters transparent. Within each functional level, several iterations are possible, and at

any decision breakpoint that the system cannot resolve from its current goal structure, the user can specify alternative branches to be followed.

It is also important to note that none of the complexities of system integration are directly obvious to the user: irrespective of the task specified, the style of the user interface and interactions with the system are always the same at the user end.

Individual modules of the system appear as self-contained, autonomous entities. It must be possible to select and run any of them in random sequence. However, much of the system's usefulness derives from the **integration** of its component elements. The complexity of this integration, the multitude of logical and practical problems resulting from the combinatorial explosion of possible options and their prerequisites and dependencies, as well as the heterogeneity of the component elements make it necessary to completely automate the dynamical interdependency of options.

In the prototype, several examples and techniques of module integration are implemented. As a simple example, data bases that provide input to simulation modules can also be accessed through special interactive ***browsing*** programs to examine their contents in detail (compare 2.1.01 and 2.1.07).

Simulation models that require input which may be predetermined by other models, or specifications resulting from the browsing of data bases (defining a **current problem context** or frame) use pass files to obtain such startup information. If a relevant pass file exists, prepared by any previous action of the system, the model will use it and automatically determine its startup state as far as that is possible from the frame information. Any remaining control or input options, not defined in the frame, either default to the models' default set of startup conditions, or are obtained by querying the user (compare 2.1.11.2, 2.1.11.3. and 2.1.11.4).

Switching from one module to another, e.g., using the next module as a post-processor of the current module's output data (compare 2.1.2) is accomplished by just selecting the corresponding option in the current module's menu. Since some of these options depend on certain preparatory actions, as is obvious in the case of data post processors, these menu options can only be activated under certain conditions. If these conditions are not given, a short message will invariably follow an attempt to invoke them.

Alternatively, modules can use ***dynamic*** menus, that change with the program's status and offer only those options that are currently meaningful. While this may reduce frustration for the novice, it is more difficult to use in the long run. Fixed menus can be used with the help of ***eye and muscle memory*** in a fairly mechanical manner, reducing the necessary re-reading of the current options by the operator.

1.3 System Implementation

The demonstration prototype software system described here is implemented on a SUN Microsystem's SUN-2/160*) color graphics workstation. The workstation is based on a 32 bit microprocessor (MC68010), supporting virtual memory

*)The software system can be be upgraded to run under the new SUN 3/160C hardware (MC68020) and corresponding software release by a few modifications of the corresponding configuration control files (*make* files) and libraries.

management, thus freeing the programmer from the onerous task of storage optimization for large engineering applications. An auxiliary floating point processor unit supports fast floating point operations, to make the interactive use of larger engineering programs feasible. The workstation offers sufficient and fast Winchester-based mass storage for large data bases and their interactive management.

The user interface is based on a high resolution (1152x900, i.e., 1 Mega-pixel) bit-mapped color screen (256 simultaneous colors or up to eight individual drawing planes).

The software system, based on UNIX (Berkeley release 4.2 bsd) supports several languages to allow the integration of already existing software. This also makes it possible to select the most efficient language for a given task. In the prototype described in this report, C, FORTRAN 77 and Pascal are used. Applications coded in LISP (Franz Lisp, Common Lisp) and PROLOG are under development.

1.4 The Scope of the Demonstration Prototype

When developing a complex software system, like the one outlined in this report, rapid prototyping is very important. Therefore, this first implementation is on a **demonstration prototype system** level. Given the time and resource limitation of the project, higher efficiency of the code had to be traded off for speed of development and ease of implementation. When dealing with larger systems, several of the prototype modules will have to be streamlined for higher performance.

The main purpose of the demonstration prototype is to implement several working examples of methods and approaches proposed and discussed in the structure and design report (Fedra, 1985), and thus provide a practical starting point for prospective users to work with. Only by being exposed to an operational prototype will users and co-developers be able to specify in greater detail the features they want supported by the system.

From the entire range of applications, a small, but sufficiently realistic and interesting subset has therefore to be chosen for this implementation. For the industrial origin of hazardous substances, the sector or group of substances chosen is **the chlorination of phenols**. Here many toxic compounds are involved, including the ill-famed 2,3,7,8-tetrachlorodibenzo-*p*-dioxin (2,3,7,8 TCDD), a reaction by-product in the production of 2,4,5-trichlorophenol (2,4,5 T).

At the same time, a realistic first prototype implementation can only involve a certain **small subset of simulation models** from the set discussed in the structure and design report (Fedra, 1985), that would ultimately be integrated in a real production system.

Further, the data and knowledge bases implemented are **not** extended to the full size and the level of detail necessary for a real production system. Data collection and verification is a major task in itself, undoubtedly beyond the scope of this study. The prototype implementation will use collections of fictional and/or readily available data from various sources and at various levels of aggregation. However, the prototype examples include a rich collection of major geographical features that need to be represented in any fully configured system. The data used are taken from or based upon historical data from various existing regions, rescaled wherever necessary. The prototype implementation spans the entire scale

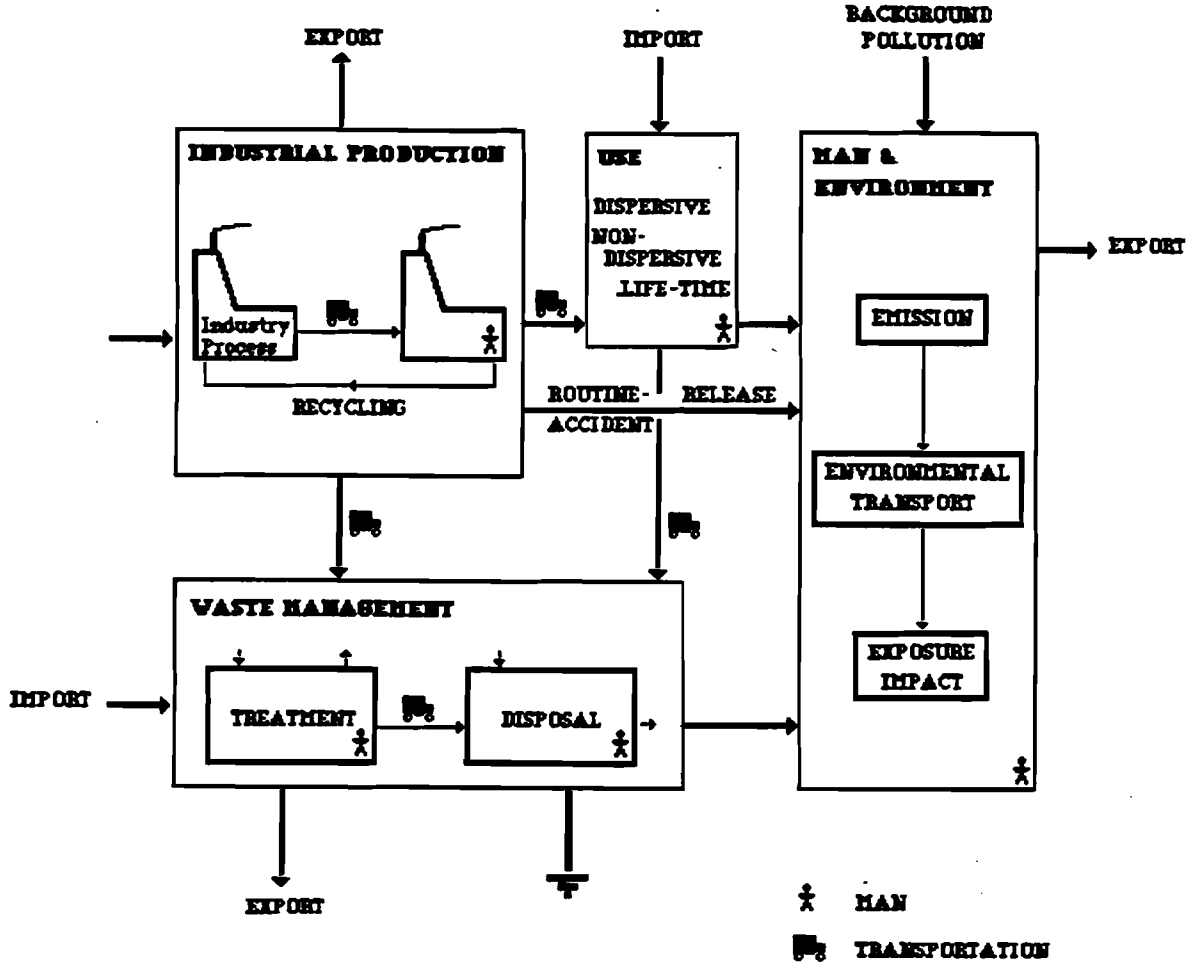


Figure 1.1: Elements of the simulation system.

from local to regional and up to national and European.

The production system and information bases of the prototype implementation is reduced to a **minimum set of functional elements** that still allow the description of the entire coupled system as outlined in Figure 1.1. The structure and framework, the style of the user interface, and the basic principles of the system's operation, are those of a fully configured production system. The development of a coherent family of such fully configured systems, implemented in several regional to national versions as well as a set of derived systems focused on individual problem areas, but integrated into a compatible European superset and framework version, is the ultimate long-term goal of the project.

We certainly do not believe in the usefulness or feasibility of one all-embracing super system, but see this as just one, albeit extremely powerful and effective, tool out of the array of tools required to deal with the complex problem area addressed here. But we also believe that only a rich and well-integrated set of **useful and usable tools**, which are readily available or can at least be reconfigured for a new task at low cost in a short period, can expect to make a meaningful contribution to real-world problem solving.

Any particular problem that can possibly be structured, analyzed and prepared for decision making by the tools in this system, will have strong case-specific peculiarities, involving individuals' idiosyncrasies as well as institutional cultures and regional or national features. These elements will have to be considered if this system is to be broadly accepted by potential users. Only a well-organized and truly modular-base system will be able to cope with these requirements.

The purpose of the demonstration prototype as the **nucleus** of this base system is therefore fourfold:

- first, it is to serve as a demonstration system that allows prospective users to gain hands-on experience with tools that might be able to help them solve their specific problems;
- second, it is a set of tools in its own right, that can be used to study the set of problems for which it is implemented;
- third, it is an operational reference collection of standardized building blocks, modules, and procedures and concepts that can be used as a basis from which similar tools for specific applications may be built;
- and finally, it is raw material, toolbox, and workbench at the same time, for further development of the base system and a number of possible spinoffs.

Due to its modularity, heterarchical organization, and the flexibility resulting from the standardized input/output structures used for the linkage of modules, the system can grow without getting complicated and intractable. The degree of resolution and detail required for any particular problem and application is largely determined by the data bases supplied with the system, and does not directly affect the basic software elements.

The current prototype implementation is designed as an open-ended system. No limitations on number, size, and connectivity of the modules are built into the design. The system is also open in the sense that further development can go in any of several directions with various degrees of effort and emphasis. By gaining experience with the potential of the approach in hands-on experiments with the prototype, the user community should increasingly be involved in defining future development options and goals.

2. COMPONENTS OF THE DEMONSTRATION PROTOTYPE

2.1 Master Menu Level

The Master Menu Level provides the top level of entry points to the system's functions. All components are directly or indirectly accessible from here, and all major functional units can return to this level. The structure of the menu program is flexible, so that different items can very easily be incorporated into the menu and the corresponding control programs. Modules accessible are either linked to the body of the main (**RUN**) program, or, alternatively, they are implemented functionally as stand-alone modules, invoked through systems level calls. In the latter case, communication with the stand-alone modules is either through pass-files, or via a limited set of command line arguments that can be passed with a systems call.

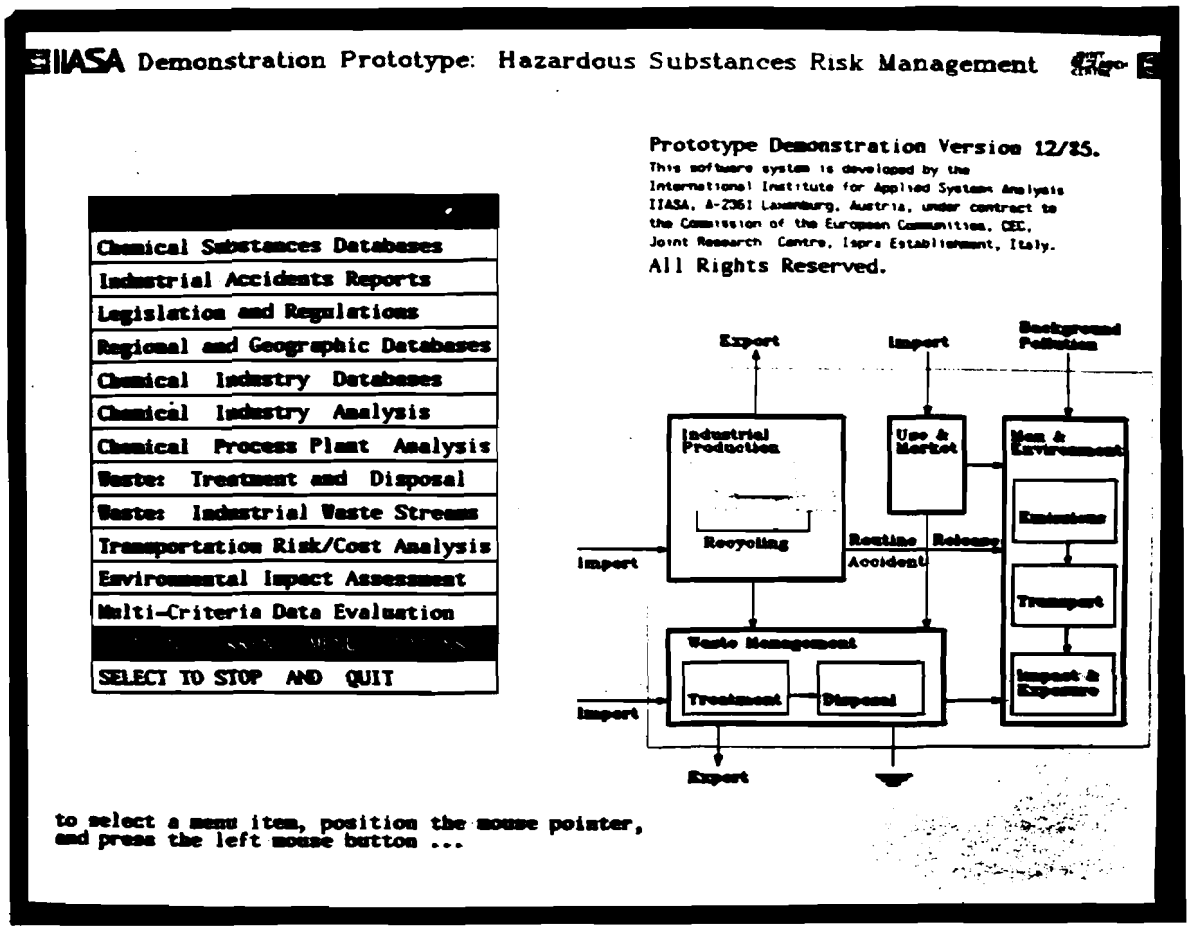


Figure 2.1: Top level system master menu.

Access to the system's functions and options is always through a system of menus. To ease the task of learning for the novice, and to provide an efficient working environment for the daily user, a consistent style of menus (Figure 2.1) is used wherever possible.

To select options from the menu, the mouse pointer is positioned in the appropriate section of the menu display, which will invert text/background colors to acknowledge proper positioning. Actual selection is performed by pressing the left mouse button. Instructions such as this (pressing the left mouse button), and information on the current status of the selection process are displayed in one or two status lines at the bottom of the screen, usually in red whenever user action is expected.

Each menu contains at least:

- a **QUIT AND RETURN** option, that terminates the current level and returns to the previous, higher level;
- an **EXPLAIN CURRENT OPTIONS** option, that gives possibly hierarchically structured information on the currently active menu options;
- one or more **OPTIONS**, as described by the EXPLAIN function.

Options are described with a short explanatory title. If an option is selected that is not currently active because it may require other options as prerequisites, a visible "bell" (short inversion of the current background color to black) and appropriate diagnostic and explanatory message, where necessary, are displayed.

In general, illegal user input (e.g., pressing the mouse buttons over non-selectable parts of the display) triggers the visible bell, or is silently ignored. A diagnostic message is supplied only if necessary.

As a special case, whenever the system waits for expected user input for more than a certain time interval (currently about 10 secs), the visible bell (inversion of background color) and a short message: "waiting for user input" is displayed on the status line (bottom line of screen) for about a second. This will be repeated every 10 secs, until some input event occurs.

Data Structures:

Control information for menus is stored as static structures in the programs using them. The structures include the position and size specifications as well as the text written into the menu's slots. Several routines (creating, hiding, restoring, and deleting menus) access this information through the structure pointer passed with the call.

Possible Extensions:

The menus could be improved by providing a broader range of visual and/or audible feedback to the user. For example, the slot (menu option) currently picked should be highlighted, e.g., by changing the text and/or background color. This is particularly useful whenever the slot is used to toggle (switch between enable and disable by repeated picking) a certain option (see, for example, criteria selection in sections 2.1.06 or 2.1.12).

In the long run, inclusion of more extensive sound and voice generators as well as voice understanding input should be considered to broaden the bandwidth of man-machine interaction and ease the task of using the system.

2.1.01 Chemical Substances Data Bases

The chemical substances data bases are built around a subset (about 500 substances and substance classes) of ECDIN, EC regulations, and various national and international listings of hazardous substances (see Fedra et al., 1986b).

The chemical substances data base currently has the following possible entry points:

- a list of the substances available in the data base with the option of scrolling up and down to pick one specific substance for further information
- a list of basic class names with the possibility to go down the tree of groups and subgroups:
  - o chemical heterarchy
  - o "black list" (EEC list C 176/4)
  - o "grey list" (EEC list C 167/7)
  - o industrial use
  - o danger at storage
  - o transportation risks and regulations
  - o hazard groups
  - o possible reactions
  - o toxicity
- the input of a substance name for some further information
- production process
- waste streams
- product and users

A special questionnaire has been developed for some additional properties of interest (Fedra et al., 1986b). Additional information covers physical-chemical properties as well as specific data for different models that have been implemented and can be linked to the data base. This questionnaire can continuously be updated for the different substances of interest. Auxiliary software is used to generate a random access data base from the individual (one per substance included) sequential files generated by the questionnaire manager.

Data Structures:

The minimum data for every chemical stored is loaded at the beginning of the data base program:

- a flag of ten characters, giving the following information:
  1) if the chemical is on the EEC list C 176/4
  2) if the chemical is on the EEC list C 167/7
  3) if the chemical is on the EEC list L 230/11
  4) if the chemical is highly toxic
  5) if the chemical is explosive

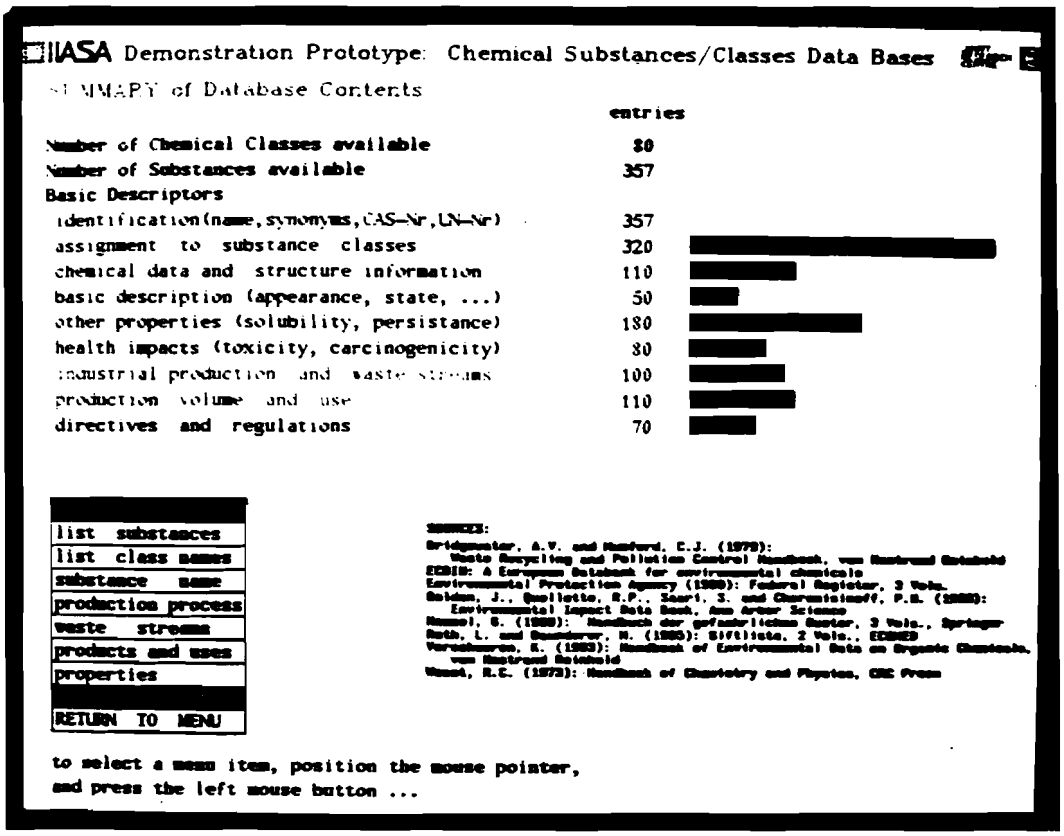


Figure 2.2: Top level menu for the chemical substances data base

6) if the chemical is a water pollutant
7) if the chemical is flammable
8) if the chemical is corrosive
9) if the chemical is radioactive
10) if there is more data on this chemical stored

- an identification number
- the name of the chemical.

Additional information is stored in separate files. These files can be edited directly and therefore be transformed and updated easily.

To make the access to these data fast and efficient, and to keep the run-time storage at a minimum, an intermediate program that converts separate, sequentially-structured chemical data files into a random access file was developed. Every time updates are made or new chemicals are added, this transformation program has to be used.

When the user runs the data base program and wants some information on a chemical, just that part is loaded from the random access file: this procedure saves time and space. Input of the original data, however, can be done by the user

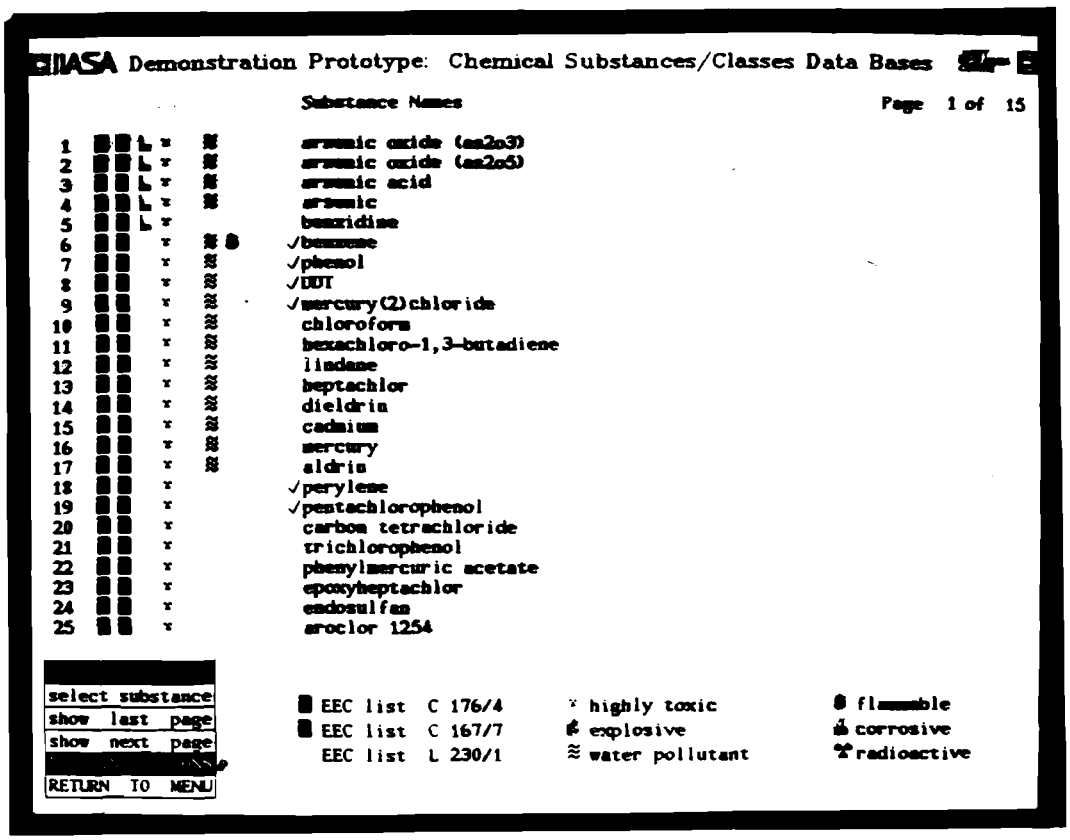


Figure 2.3: Listing of basic substances

by straightforward editing of the corresponding raw data files.

The information is stored in two kinds of records: *chem*, which contains the basic information for every chemical (fixed structure, fixed length), and *proc*, with the information on related industrial processes (variable length, depending on the number of processes involved).

chem

- o *name*
- o *synonyms*
- o *cas*: Chemical Abstracts Number
- o *un*: UN Number
- o *formula*: chemical formula
- o *diamond*: four flags for hazard ratings
- o *mw*: molecular weight
- o *mp*: melting point
- o *bp*: boiling point

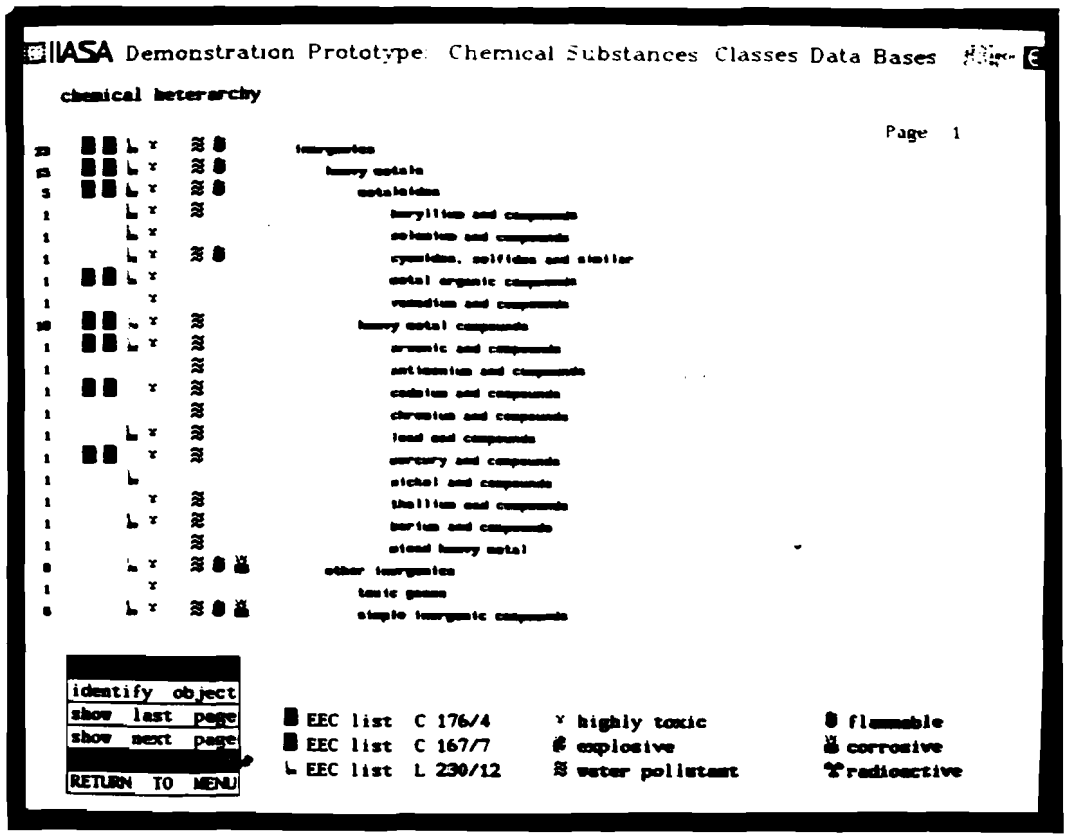


Figure 2.4: Listing of substance classes

- o *fp*: flash point
- o *vp*: vapor pressure
- o *vd*: vapor density
- o *sg*: specific gravity
- o *app*: appearance of the chemical
- o *odour*
- o *state*: solid, liquid, gas
- o *roe*: routes of entry
- o *symptoms*: symptoms in case of intoxication
- o *carc*: health impacts e.g., carcinogenicity
- o *irri*: health impacts e.g., irritation of skin or eyes
- o *sol*: water solubility of the chemical
- o *pers*: persistence of the chemical
- o *mak*: Maximale Arbeitsplatz Konzentration (TLV)
- o *apf*: air pollution factor

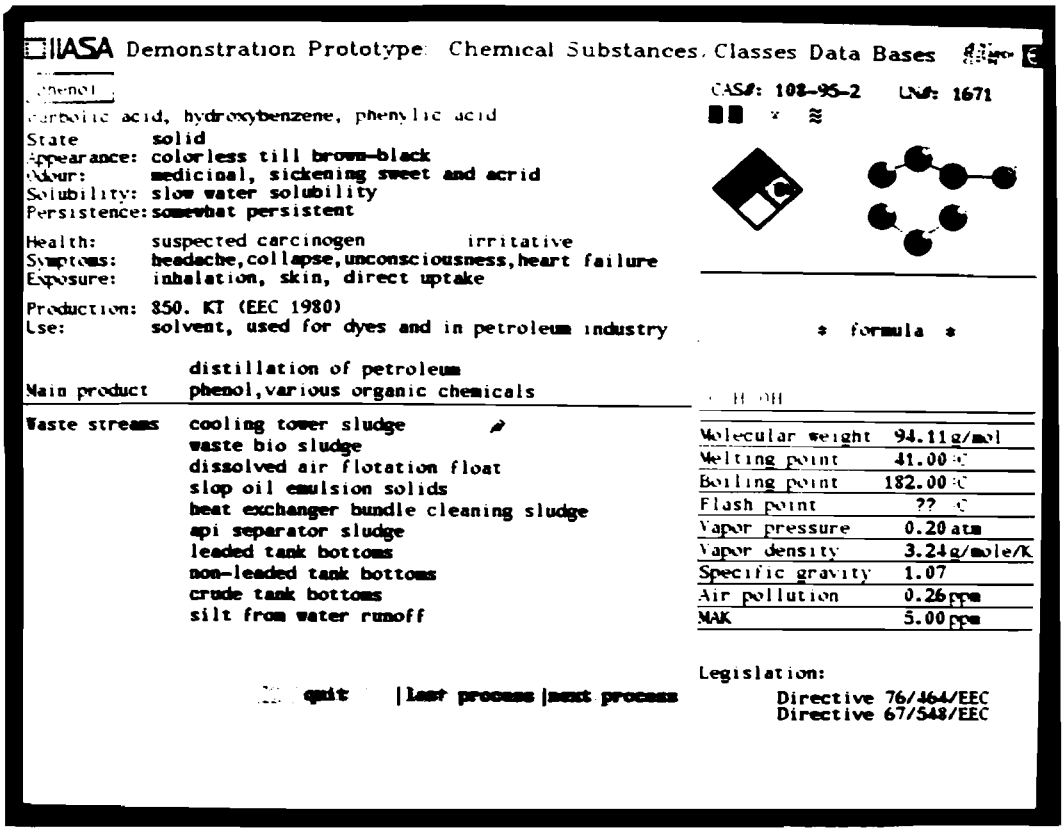


Figure 2.5: Summary page description for a basic substance

o *pw*: amount of production

o *use*: use of the chemical

o *drnr*: number of directives and regulations related to this chemical

o *dr*: directives and regulations concerned (*drnr* times)

o *ip*: number of industrial production processes concerned

proc

o *inf:* up to 16 groups of records covering the following topics:
process, feedstock, main products, by-products, number of waste streams, reference for each waste stream to the waste stream data base

o *process*: text lines with information described above.

Possible Extensions:

Another perspective of information on the different chemicals, a *heterarchical* version of the Chemical Data Base, is currently under design (Weigkricht and Winkelbauer, 1986).

Each entry point for the user will be internally represented by a class of chemicals. Each class consists of several subclasses, which are other entry points to the data base, or of a number of chemical substances. These classes form a heterarchical tree, i.e., each substance or class of chemicals can belong to several superclasses.

This enables the system to support many entry points to the data base and still keep short search paths to detailed information. Moreover the system then will be able to provide general information about classes of chemicals as well, which will be stored together with the description of a class and therefore immediately available to the user when he enters the data base.

This general information can be input from an "expert" or a synthesis of the specific information stored for the corresponding subclasses or substances in the form of a range, a statistical distribution or a verbal descriptor for the possible value. If the data for a specific substance or a class is insufficient or is missed at all, it should be possible to go up the path in the heterarchical tree and use the more general information available there.

To combine the benefits of an object-oriented approach with those of condition-action pairs, a heterarchical frame structure for the chemical data and knowledge bases is being developed in Common Lisp, i.e., an object can be a member of *several* classes and each class can belong to several superclasses, and by adding "rule abilities" to a special slot called *actions*, i.e., this slot does not store information but performs *procedural tasks* which are defined as condition-action pairs. A detailed description of the heterarchical frame-structure is given below.

Our approach foresees the use of a basic list of about 500 *substances* (or molecular substances, i.e., entities that do not have any sub-elements), constructed as a superset of EC and USEPA lists of hazardous substances. In parallel we construct a set of *substance classes* which must have at least one element in them. Every *substance* has a list of properties or attributes; it also has at least one *parent substance class* in which it is a member. Every member of a group inherits all the properties of this group. In a similar structure, all the groups are members of various other parent groups (but only the immediate upper level is specified at each level), ultimately all subgroups belong to the top group *hazardous substances*.

While attributes of individual substances are, by and large, numbers (e.g., a flash point or an LD_{50}), the corresponding attribute at a class level will be a range (flash point: 18-30°C) or a symbolic, linguistic label (e.g., toxicity: very high).

The structure outlined below also takes care of unknowns at various levels within this classification scheme. Whenever a certain property is not known at any level, the value from the immediate parent_class (or the composition of more than one value from more than one immediate parent_class) will be substituted. The structure is also extremely flexible in describing any degree of partial overlap and missing levels in a hierarchical scheme.

Frame Syntax:

Each class-frame consists of the following six slots:

- **Explanation:** verbal information about the current frame, concerning the substance class which is represented by the frame, its attributes, the default values and/or indirect references and the position of the frame in the heterarchical structure; the main purpose of this information is for updating and editing the frame by a knowledge engineer
- **Superclasses:** references to the classes to which the current frame belongs
- **Description:** attributes with values and/or procedural attachments (i.e., procedures which calculate the values or refer to them, or both) which describe the substance class represented by the current frame

  Subclasses: references to the classes which belong to the current frame
- **Instances:** references to the instances (i.e., substances) of the substance class represented by the current frame
- **Actions:** condition-action pairs, where the actions of an action part are carried out if the frame receives a message which matches the corresponding condition pattern.

The formal description of a frame is as follows:

```
(Class      classname
            (Explanation      (<Verbal Information>))
            (Superclasses     (<List of Classnames>))
            (Description      (
                              (<Slotname>-2 <Filler>-2)
                              ...
                              (<Slotname>-n <Filler>-n)))
            (Subclasses       (<List of Classnames>))
            (Instances        (<List of Substances>))
            (Actions          (If <Condition Pattern>
                               Then <Action Part>)))
```

<Verbal Information> = InfoText represented by a list of words

<List of Classnames> = classname | classname <List of Classnames>

<List of Substances> = substance | substance <List of Substances>

<Slotname> = attribute of the represented class

```
<Filler> =    (Class classname) |
              (Value value) |
              (default (Lisp s-expression)) |
              ($if-needed (Lisp s-expression)) |
              ($if-added (Lisp s-expression)) |
              ($if-changed (Lisp s-expression)) |
              ($if-deleted (Lisp s-expression))
```

<Condition Pattern> = (<Pattern>-1 <Pattern>-2 ... <Pattern>-m)

<Pattern> = constant | ?variable | #

<Action Part> = (Lisp s-expression)

As an example, two class-frames from the heterarchical knowledge base structure for phenols are given below:

(Class **aromatics**

(Superclasses	(Object))
(Description	(attribute-1 .....)
	(attribute-2 .....)
	...
	(attribute-n .....))
(Actions	(If (List your members)
	Then (prog (ask self subclasses) (ask self instances))))
(Subclasses	(aromatic_hydrocarbons aromatic_heterocyclics))
(Instances	NIL))

(Class **mixed_hydrocarbons_substituted_with_two_chlorines**

(Superclasses	(aromatic_hydrocarbons_double_substituted
	chlorinated_phenol
	mixed_chlorinated_aromatic_hydrocarbons))
(Descriptions	(attribute-o+1 .....)
	(attribute-o+2 .....)
	...
	(attribute-p .....))
(Subclasses	NIL)
(Instances	(2,4-Dichlorophenol 2,6-Dichlorophenol)))

To retrieve information the user may directly enter the name of a substance or of a substance class, or he may specify value ranges (numerical and/or symbolic) for one or more substance (class) attributes. The Information System transforms this specifications into messages for the top level classes (also called viewpoints). By receiving these messages the frames which represent the viewpoints are activated.

The frames then check if they are selected by the user's specification, and if they are, then proceed to create messages for their subframes which again perform their matching operations and create messages, and so on. This recursive procedure does not need to search through the whole structure because it is directed by the rules in the *Actions* slots and the references in the *$if-* slots, supported by the inherited information.

This procedure results in a substructure of valid substance classes which represents the systems view of the user's level of expertise. This substructure is from then on used to guide the user to the more detailed information, if he agrees to proceed with the interaction.

Then this recursive message sending and receiving can be applied again, starting from the current top level(s) of the substructure using the additional information provided by the user (based on the displayed status of the attributes of the classes level reached), until either the user is satisfied by the given information about the current substructure's attributes or until the level of instances (a single substance) terminates the user's attempts to get further information.

Updating is quite similar to the retrieval of information. First, the substructure which will be affected is localized by an interframe message sending/receiving sequence. Then the updates of the attribute values are entered and checked if they are consistent within the selected substructure by using the ***$if-added*** and the ***$if-needed*** slot fillers together with the rules of the *Actions* slot which deal with consistency tests.

After the consistency of the substructure has been proved, the same procedure is used for the next higher aggregation levels until the whole structure has been proved to be consistent with the new and/or changed attribute values.

2.1.02 Industrial Accident Reports

The industrial accident reports data base contains narrative accounts of major industrial accidents involving hazardous substances. A summary report, following the report format of the Seveso Directive, precedes each narrative account.

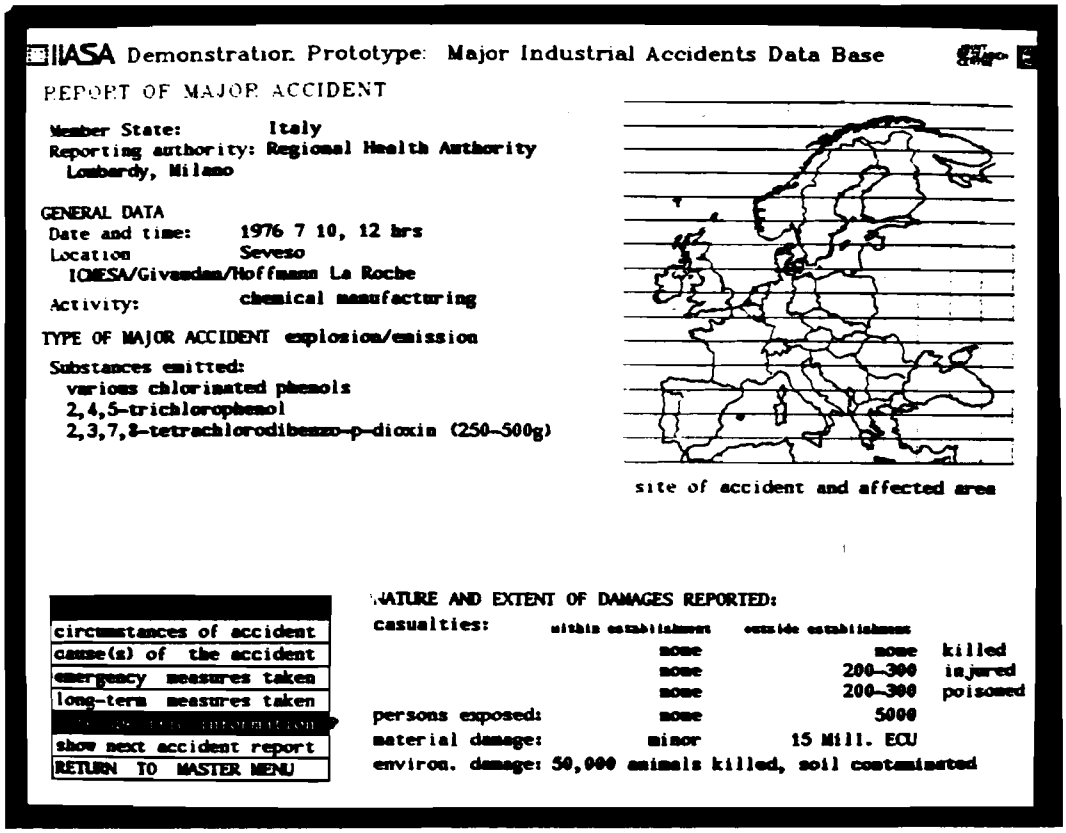


Figure 2.6: Example page from the industrial accidents data base (partially fictitious data).

Access to the data base is currently from a menu list of reports, where a header line is shown for each of the reports in the system. Selecting an appropriate header, the user will then read one or several screens of information, composed of text and graphics. Within a given report, forward and backward paging is supported.

The basic components of any accident description*) include:

- Place, date and time of accident

*) Based on Official Journal of the European Communities, No. L 230/17, 5.8.82, Annex VI.

- Type of accident (explosion, fire, emission of toxic substance: substances emitted);
- Description of circumstances
- Emergency measures taken (accident management)
- Causes of the accident
- Nature and extent of damage:
  - Casualties
  - Persons exposed to the accident
  - Material damage
  - Current status of danger
- Medium- and long-term measures (accident prevention).

Data Structures:

Reports are currently stored as individual sequential text files, access is by file name, which includes the report-ID, e.g., rep.0121.

Possible Extensions:

As an extension, selection of subsets by keywords and operations on identifiers (e.g., *show reports on accidents in western Europe since 1980 involving chlorinated phenols and causing damages of more than 1 Mill. US $*) is under development. This data base management is based on a straightforward implementation of a relational model.

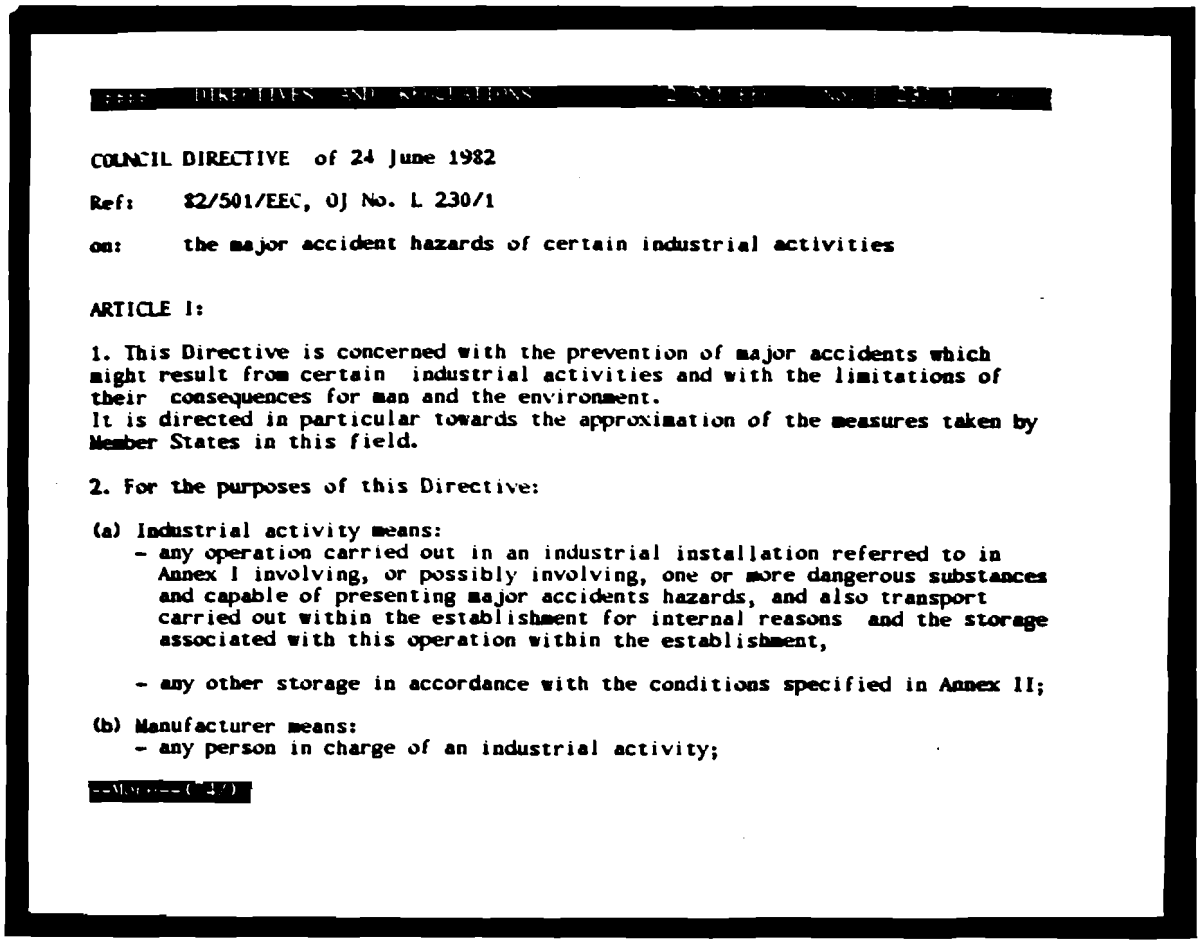
In a further step of development, for the interactive querying of primarily text-oriented data bases, we are developing a new concept, ***hypertext***, which is an interactive visual representation of a list and frame-oriented knowledge representation. Most items on display (e.g., a page composed of text, numerical data in the form of tables, and graphics) are like menu items, that is, if you pick them with the mouse pointer, they will expand into an entire new page of information (overlaying the current one at least in part) which again contains entries which are headlines for further entries and so on).

For example, by pointing to the names of *substances emitted*, the corresponding page from the chemical substances data base can be called; or, by pointing at the factory's name, the corresponding entry in the chemical industry locations data base will be displayed. The information accessible in the hypertext concept is thus a mixture of numerical, textual, symbolic/graphical and dynamic entries, related to each other by a heterarchical organization. First elements of this approach are built into the current chemical substances and industrial waste stream data bases.

With the introduction of optical disk technology, one could also combine high-resolution images (e.g., detailed maps, photographs, etc.,) with the basic text-oriented information, extending the information base of the hypertext concepts by orders of magnitude.

2.1.03 Legislation and Regulations

Similar to the above accident reports, the text files accessible through this module cover selected EC directives and other relevant European legislation. As an example, part of the Seveso directive (82/501/EEC, L 230/1) is included.



COUNCIL DIRECTIVE of 24 June 1982

Ref: 82/501/EEC, OJ No. L 230/1

on: the major accident hazards of certain industrial activities

ARTICLE 1:

1. This Directive is concerned with the prevention of major accidents which might result from certain industrial activities and with the limitations of their consequences for man and the environment.
It is directed in particular towards the approximation of the measures taken by Member States in this field.

2. For the purposes of this Directive:

(a) Industrial activity means:
- any operation carried out in an industrial installation referred to in Annex I involving, or possibly involving, one or more dangerous substances and capable of presenting major accidents hazards, and also transport carried out within the establishment for internal reasons and the storage associated with this operation within the establishment,

- any other storage in accordance with the conditions specified in Annex II;

(b) Manufacturer means:
- any person in charge of an industrial activity;

Figure 2.7: A sample page from the Seveso Directive

Data Structures:

The text files are stored as simple sequential ASCII files.

Possible Extensions:

Similar to the Accident Data Base described above, an extension to the *hypertext* concept would be possible for a convenient and flexible access to related information.

Also, to connect relevant directives to other modules (other than through direct reference, as, for example, in the case of the chemical substances data base, see section 2.1.01), each directive can be augmented with a list of context-oriented keywords, which can provide linkages to individual directives or parts of directives. For example, when defining a scenario for simulation (e.g., with the transportation risk/cost analysis model), a menu option **Relevant Legislation** could be activated if a trans-boundary shipment is considered, and Council Directive 84/631/EEC, L 326/31 applies. Selecting this option will then allow the user to read part or all of the relevant regulation which may or may not affect the scenario options.

2.1.04 Regional and Geographical Data Bases

The regional and geographical data base options provides a map-oriented browsing feature for the spatially organized data.

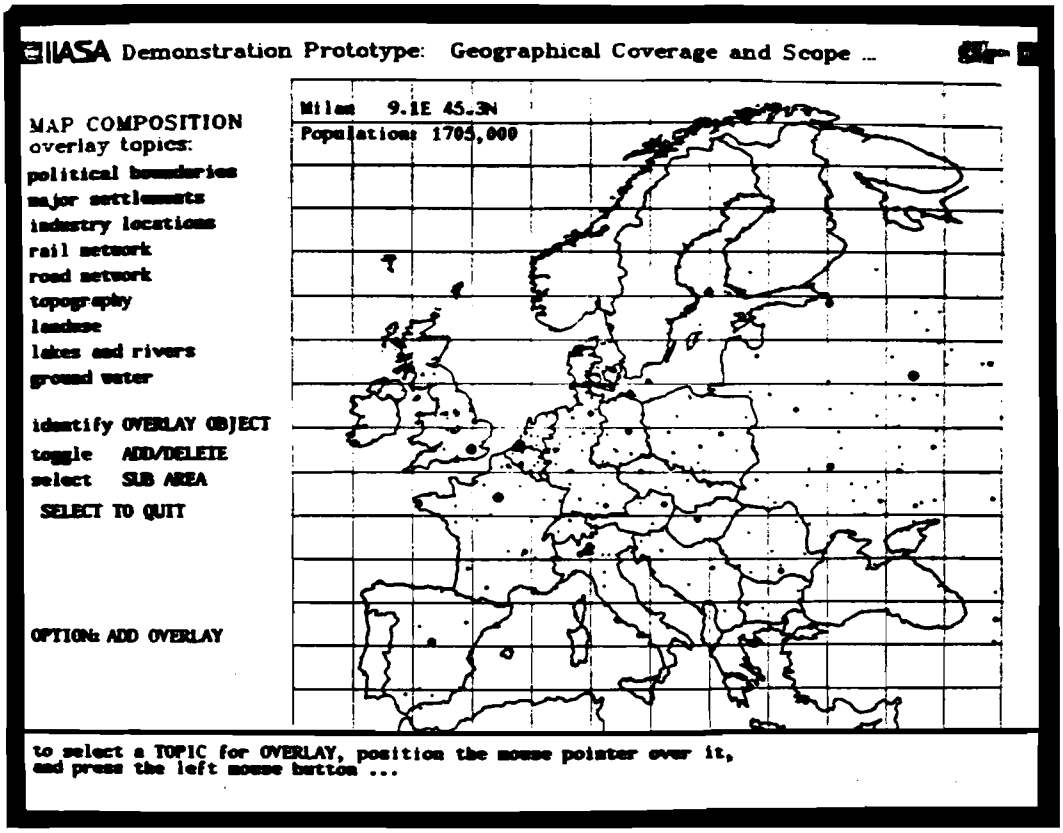


Figure 2.8: Map of Europe with major settlements displayed.

Based on a basic map of Europe at the highest level of aggregation, the module supports the interactive composition of topical overlays (see Figure 2.8) on the map.

Overlay topics include:

- national boundaries;
- major settlements;
- major industrial production sites;
- chemicals storage facilities;
- European highways and national roads network;
- major surface water systems (rivers, lakes).

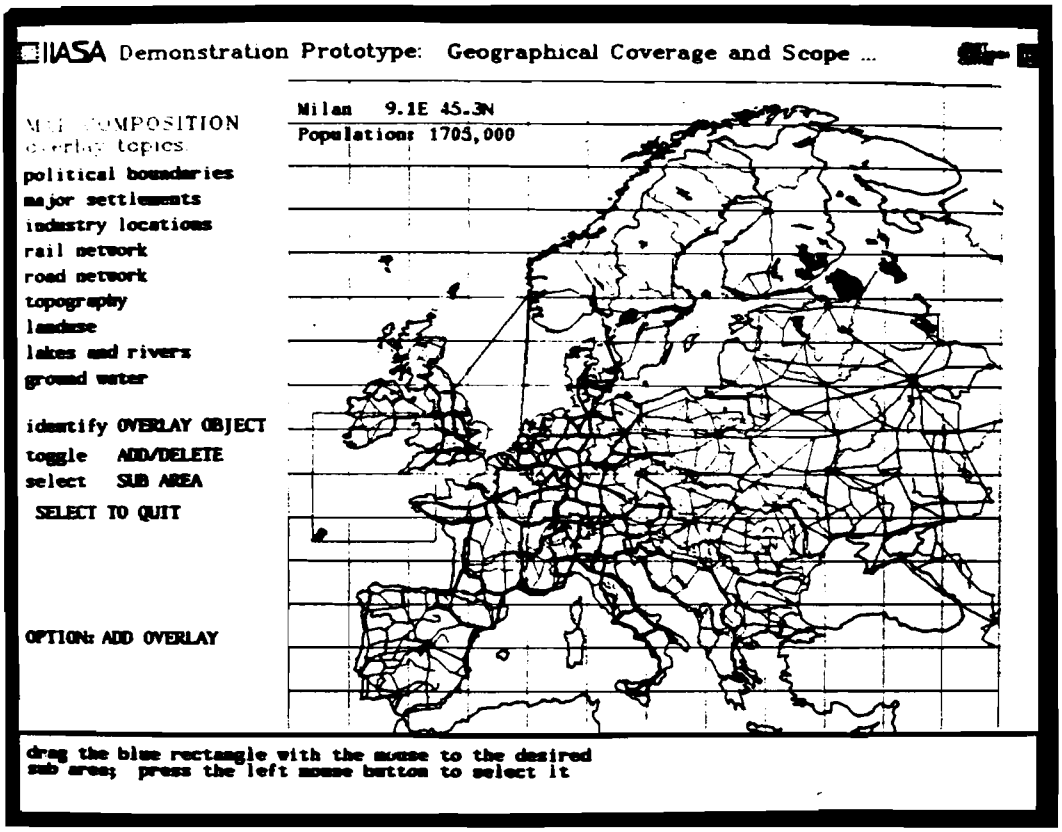


Figure 2.9: Major settlements with overlay of road network, rivers and lakes.

Data Structures:

The data accessible through the geographical data bases interface are of two types: they are either preprocessed graphical segments, i.e., pictures, or data files that contain spatially distributed information. For example, the display of major settlements is achieved by loading a preprocessed graphical segment. Identifying a settlement, and consequently displaying its name, location, and number of inhabitants, is based on a corresponding file (which, as a matter of fact, is also used to generate the segment in the first place), where the relevant information is found by matching the coordinates of the object identified on the map with the city's coordinates in the data base. This redundant procedure is installed for performance reasons. Raw data files for cities, roads (divided in arcs between two settlements), rivers, or lakes always consist of basic information (e.g., the name of the settlement and its size, or the name of a river, its length, average flow, etc.) followed by one or more sets of coordinates.

Possible Extensions:

Functionally, the most important extension of the current data base would be an arbitrary zooming facility, which, at all but the highest level of resolution, would allow the definition of a sub-area in the current display which would subsequently be magnified to the full display size available.

From an implementation point of view, the inclusion of raster-oriented image data (e.g., as produced by aerial photography or satellite-based imagery) and subsequent image-processing functions would be attractive and extremely powerful options. The use of satellite generated data can solve the major problem of large-scale geographical data bases, i.e., data acquisition and updating for several categories of information (e.g., physiography and land use).

The use of dense raster images, however, would require substantial mass storage facilities. With the emerging optical disk technology, this problem could be solved in a very attractive and cost-efficient way.

2.1.05 Chemical Industry Data Bases

Two major data bases are currently provided: they cover major European production sites (concentrating on phenol, chlorine, and related products) as well as major storage facilities.

The material is accessible, in part, through the regional data base, or through its own display module.

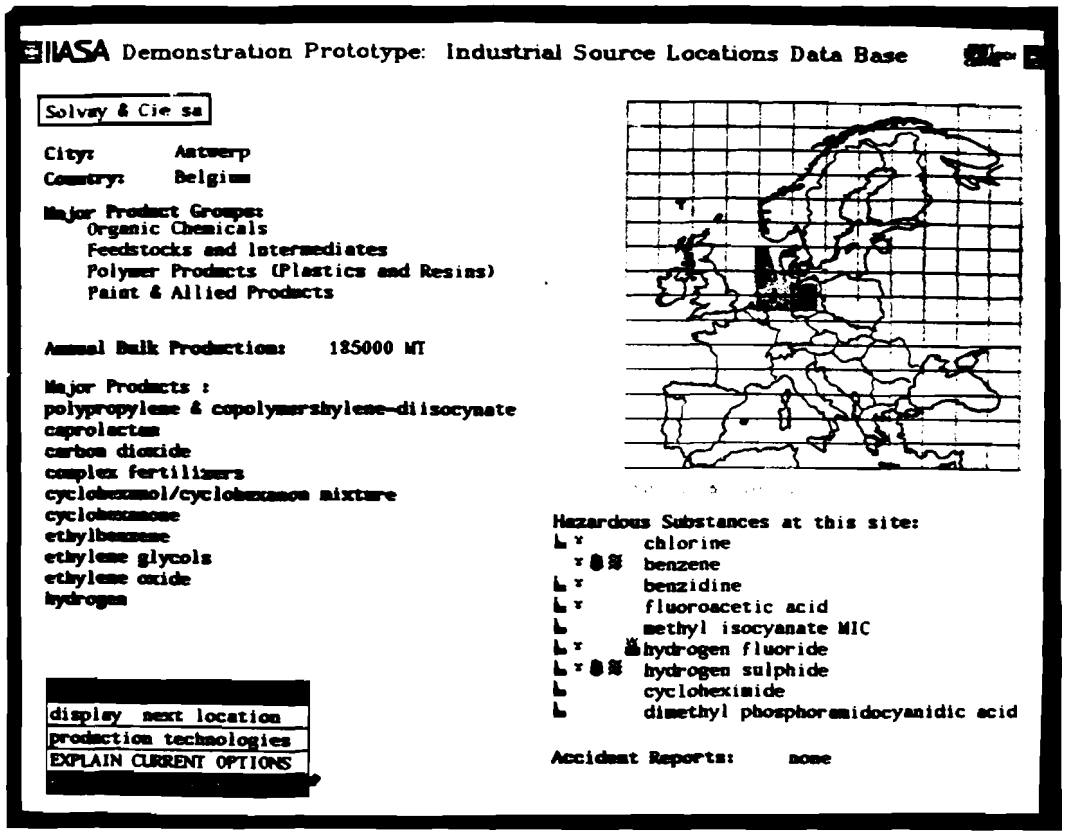


Figure 2.10: Sample display from the industrial producers data base (data partly fictitious).

The information provided includes:

- name and location of the producer;
- major products or product groups;
- information on the size of the installation (production volume, financial turn-over, number of employees);
- reference to the production technologies data base (see Chemical Process Plant Analysis);
- reference to the accidents data base, if appropriate.

Data Structures:

The industrial producers/sources data base contains information about major industrial producers and chemical production plants.

The data elements of the data base structure for each firm/location are the following:

- **name**: name of the firm
- **city, country**: where the firm is located
- **lat, lng**: word coordinates of the firm's location
- **employees**: number of employees
- **ref1**: reference from where above data are taken
- **tonnage**: annual production of chlorine and/or phenol
- **unit**: measure for chlorine and/or phenol production
- **cl_ph**: if either chlorine or phenol is produced
- **ref2**: reference from where above data are taken
- **finance**: capital (in most cases; in special cases turnover)
- **currency**: currency unit
- **ex_year**: basis year for exchange rate
- **ref3**: reference from where above data are taken
- **ref4**: reference from where the data about the chemicals involved in the production process are taken
- **chemical**: chemicals involved in the production process.

Possible Extensions:

In addition to several possible linkages, e.g., to the substances DB, industrial waste streams DB, or accidents DB, the industrial producer/locations DB must be merged with a generalized version used for the spatially extended PDA (Industrial Structure Optimization) models. A design study for such a generalized industry data base is underway at IIASA.

2.1.06 Chemical Industry Analysis

For the analysis of the chemical industry, simulation of its behavior and optimization of its structure, a linear optimization model was included in the system. PDA (Production-Distribution Area), is an interactive optimization code based on DIDASS and a linear problem solver, for chemical industry structures, configured for the pesticide industry of a hypothetical region. The model is described in detail in Zebrowski et al. (1985).

The basic model used is a form of input-output model, solved by means of a linear programming package (MINOS, Stanford 1981). As a pre- and post-processor for interactive problem definition using the reference point approach, a version of the IIASA package MM (a package of the DIDASS family) is used (Kreglewski and Lewandowski, 1983).

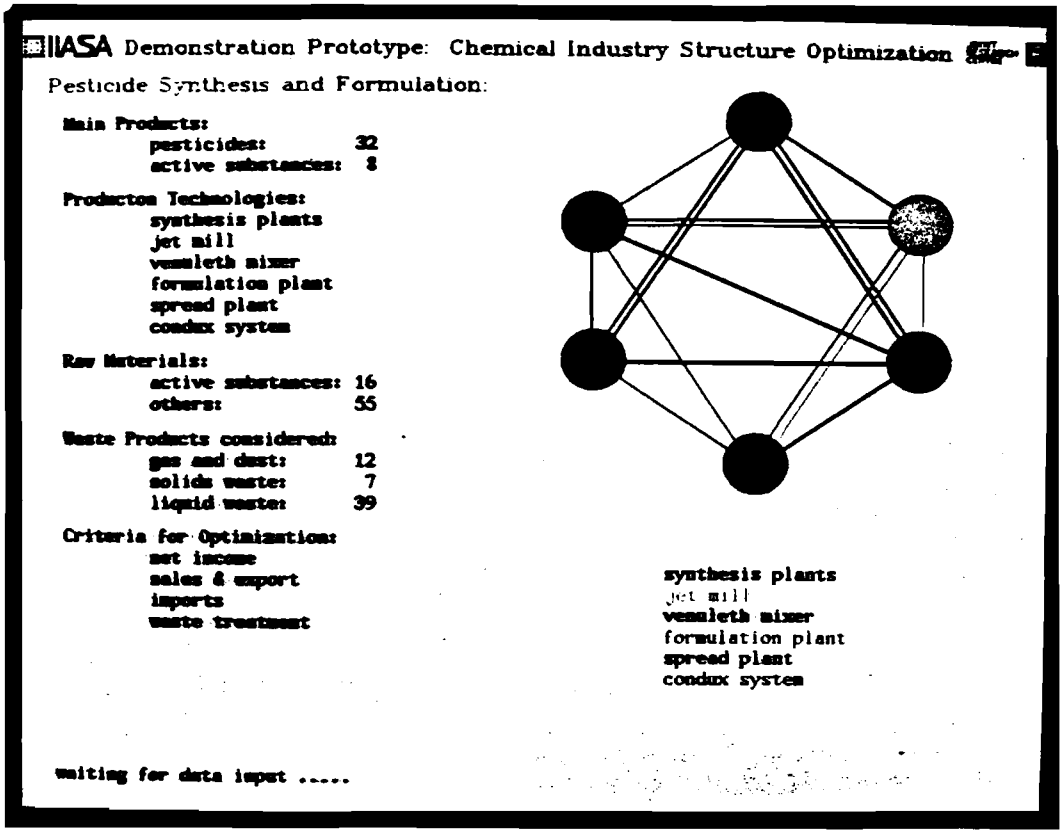


Figure 2.11: PDA start-up information

The chemical industry analysis includes a simplified behavioral model (cost effectiveness) of selected sectors of the chemical industry, with special emphasis on hazardous substances and the description of (hazardous) waste generation.

For the demonstration prototype, a Pesticide PDA was selected. The PDA was assembled from the set of processes used by several factories. The corresponding technological network includes some synthesis and formulation processes that are carried out on the basis of active substances from domestic and foreign production. The pesticide PDA comprises the following installations of chemical syntheses:

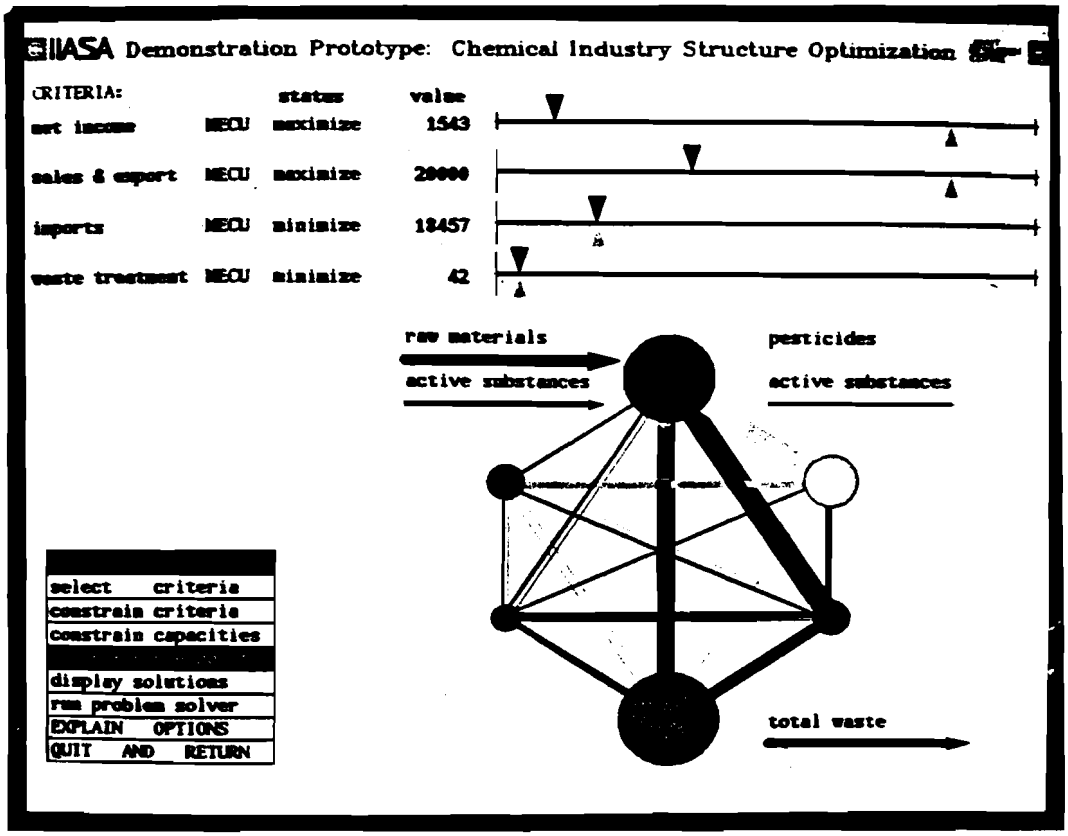


Figure 2.12: PDA Summary of results, selection of criteria, definition of reference point

- methoxychlor,
- akaritox (tetradifon),
- chlorofenwinfos,
- chlorofos (trichlorofon; dipterex),
- malathion,
- sodium trichloreacetate,
- copper oxychloride,

out of which only the last two are pesticides which can be used directly as final products. The products of other syntheses provide semiproducts for the formulation process. The PDA includes the following formulation plants:

- jet mill,
- Venuleth's mixer,
- active substances spread installation,
- formulation of liquid pesticides,
- condux system.

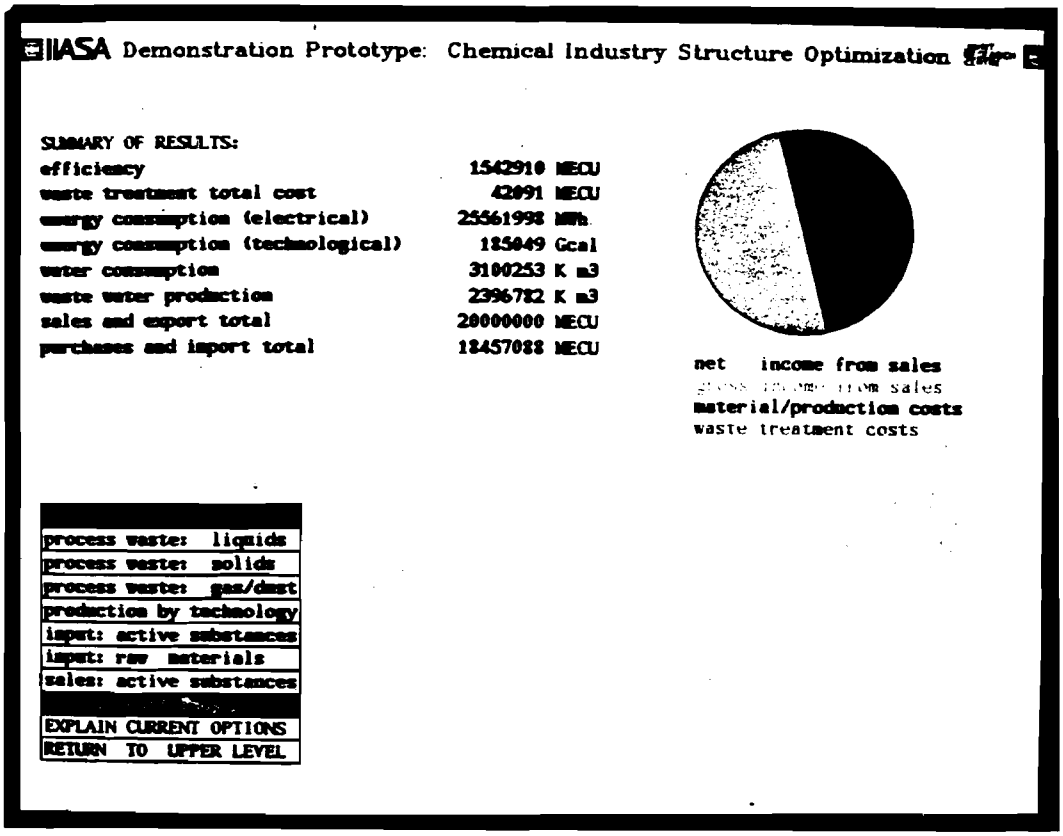


Figure 2.13: Summary of results and result display menu

The version of the model presented describes all possible modes of production, including alternative ranges of products made at a given installation, recycling of semiproducts and coupled production of a number of chemicals at each plant considered.

The model, based on the mass balance principle has been extended in order to include a description of processing and treatment of hazardous substances. It must, therefore, take into account:

- the processing and flows of chemicals with specification of hazardous substances and hazardous processes within the PDA,
- the flows of chemicals (with specification of hazardous substances) into and out of other areas and industries representing the marketing or business activity of the PDA,
- means of treatment of wastes and hazardous substances within the PDA,
- the flow of investment, revenue and other resources such as energy, manpower, etc.

The main assumptions of the basic model are that:

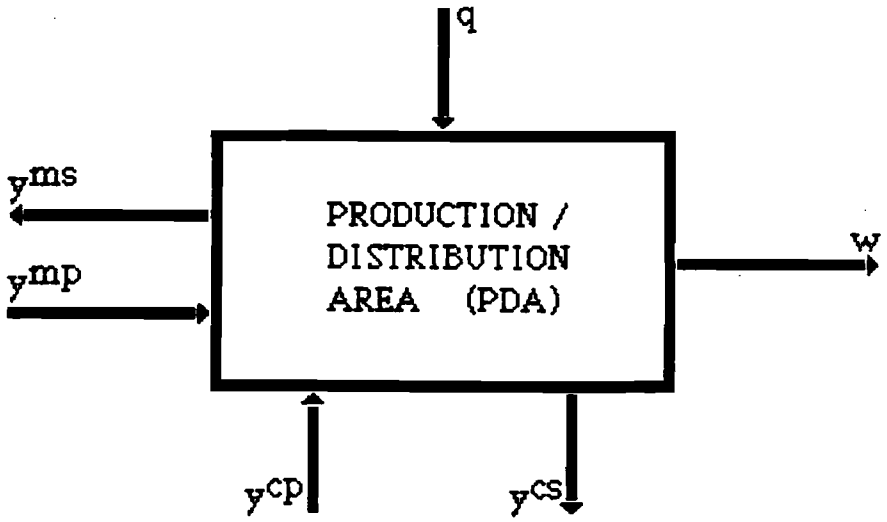


Figure 2.14: Basic input/output structure of the PDA model from Zebrowski et al.

1. it represents the equilibrium state of the PDA,
2. it includes only easily quantifiable physical elements of the system (without taking into account important but not quantifiable social or political factors).

Before describing the network of the PDA, we define its links with the environment. The following equation describes the outflow of any chemical j:

$$y_j = y_j^{ms} - y_j^{mp} + y_j^{cs} - y_j^{cp}, \quad j \in J \quad , J \supset J_h \tag{1}$$

y_j^{ms} - market sale of chemical j,
y_j^{mp} - market purchase of chemical j,
y_j^{cs} - coordinated sale of chemical j,
y_j^{cp} - coordinated purchase of chemical j,
J - set of indices representing chemicals of the PDA,
J_h - set of indices of hazardous substances under consideration.

The variables in Figure 2.14 are defined as follows:

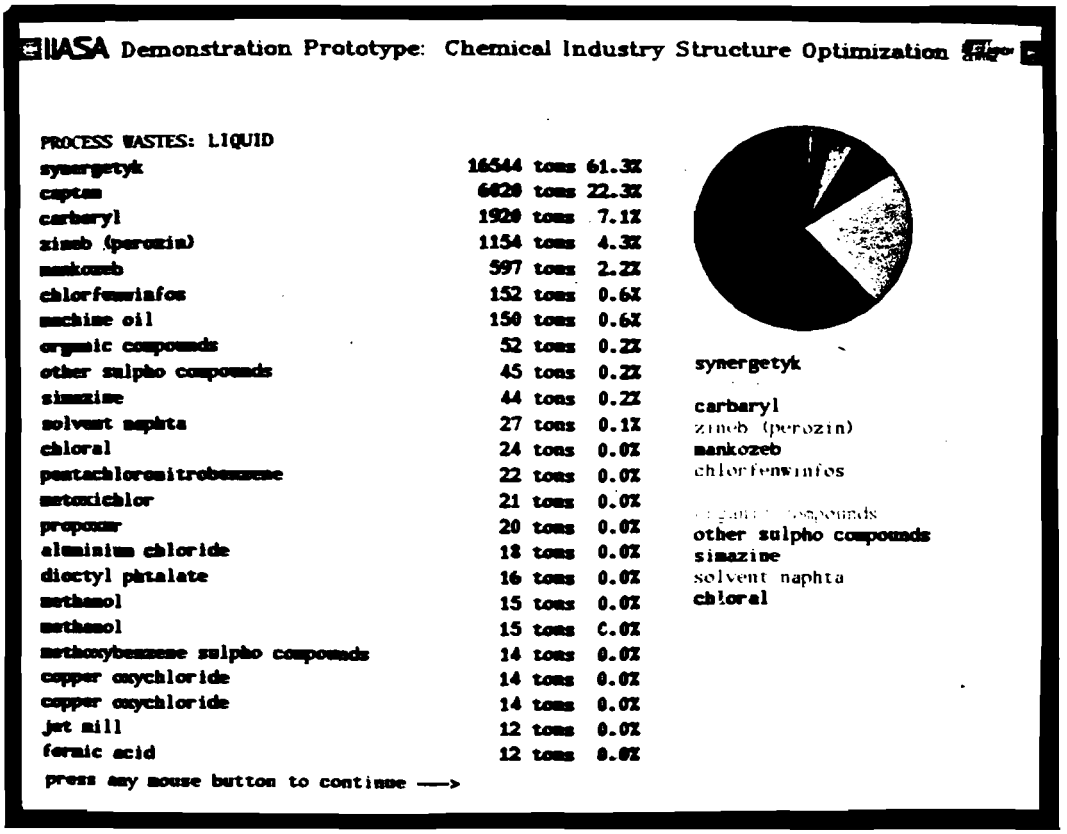


Figure 2.15: Display of results: liquid wastes.

z_k - production level of PE_k,

$\hat{z}_k$ - production capacity of PE_k,

$a_{jk}\ z_k$ - quantity of chemical j consumed by PE_k,

$b_{jk}\ z_k$ - quantity of chemical j produced by PE_k,

$d_{lk}\ z_k$ - quantity of waste l produced by PE_k,

$q_k(z_k)$ - necessary resources,

K_h - the set of indices of hazardous chemical processes.

Within K_h there might be distinguished subsets of indices corresponding to kinds of hazards or accidents which may possibly occur during a process (fire hazard, explosion etc).

For the balance nodes the following equations are satisfied:

$$y_j = \sum_{k \in K} b_{jk}\ z_k - \sum_{k \in K} a_{jk}\ z_k$$

$$w_l = \sum_{k \in K} d_{lk}\ z_k \quad , \ l \in L$$

By combining the above results with (1) we obtain:

$$y^{ms} - y^{mp} + y^{cs} - y^{cp} = (B - A)\, z \tag{2}$$

$$w = D\, z \tag{3}$$

To complete this description of the network we have to add the constraints imposed on production capacity:

$$z \leq \hat{z} \tag{4}$$

For multi-process installations, where processes run simultaneously, instead of the latter equation we use the following constraint:

$$\sum_{k \in K_i} z_k \leq \hat{z}_i \quad , i \in I \tag{4a}$$

where I, K_i denote correspondingly the sets of indices of installations and processes run on i-th installation. In addition, the model describes redevelopment of installations, i.e., substitution of an old process by a new one run on the same installation. For a given k-th process it is formulated as follows:

$$\frac{1}{\hat{z}_k^o} z_k^o + \frac{1}{\hat{z}_k^n} z_k^n \geq 1 \tag{5}$$

where:

$\hat{z}_k^o$, $\hat{z}_k^n$ denote capacities of an old and new k-th process,

z_k^o , z_k^n denote production levels of an old and new k-th process.

The idea of new technologies is fundamental to this approach, as it opens the way to technological restructuralization of the PDA.

It is obviously necessary to add some additional constraints on resource availability or waste production limits and a set of criteria which reflect the preference or goals of the decision maker.

First, it is assumed that for a fixed production goal only efficiency (or revenue) will be maximized. This leads to the problem:

$$Q_{rev} = \sum_{j \in J} c_j^s \, (y_j^{ms} + y_j^{cs}) - c_j^p \, (y_j^{mp} + y_j^{cp}) \;\rightarrow\; \max \tag{6}$$

with constraints given by market conditions and production capacities.

Following another decision strategy, resource consumption may be minimized, which results in the set of criteria:

$$Q_{ener} = \sum_{k \in K} e_k \, z_k \;\rightarrow\; \min \tag{7a}$$

$$Q_{inv} = \sum_{k \in K} n_k \, z_k \;\rightarrow\; \min \tag{7b}$$

$$Q_{labor} = \sum_{k \in K} l_k \, z_k \rightarrow \min \qquad (7c)$$

One of the four objectives implemented is minimal cost of waste treatment i.e.,:

$$Q_w = \sum_{l \in L} t_l \, w_l \rightarrow \min \qquad (8)$$

where t_l denotes unit cost of treatment of waste l.

From the above objective functions one may derive another useful optimization problem based on linear fractional functions (e.g., Q_{rev} / Q_{ener}) as well as various multiobjective problems. Of course, any objective (6) - (8) can be transposed onto a corresponding constraint.

Structural Correctness Control Module

This module is aimed at automatic analysis of the PDA network structure for the sake of formal correctness of the model. The module checks, for instance, whether a given constraint exists in the MPS file regardless of a value of the constraint, hence the analysis is only qualitative.

Menu-driven Simulation Experiments

The interactive and menu-driven interface module performs the following functions:

- it formulates a consistent and complete set of conditions of an experiment, i.e., defines a simulation scenario;
- initiates the optimization procedure (calls the solver module);
- displays the results of optimization in a hierarchically structured tabular and graphical format.

The user may communicate with the module through the Menu System which uses symbolic or linguistic variables instead of MPS codes. This makes the program system directly usable for non-specialists, and assures higher robustness through extensive input checking.

After initialization of the system, the present default scenario is displayed. A scenario description consists of an objective and global constraints of the model. The options on the top level include:

- modification of the problem,
- optimization experiment,
- survey of results obtained from the last experiment,
- exit from the module.

Problem modification includes:

- choice of criteria/objectives;
- constraint modification;
- definition of preferred target solution (reference point).

Constraints can be formulated as:

- constraints on the criteria (objective) values;

- general constraints (energy, manpower, water);
- constraints on amount of wastes;
- constraints on raw materials availability;
- constraints on production capacities.

Data Structures:

The data base of the PDA System consists of the following functional elements:

- data required for construction of the basic model of the PDA;
- data for construction of the extended (i.e., oriented toward management of hazardous substances) model of the PDA;
- data (links) for communication with other data bases of the expert system.

Note that all information concerning one PDA will be stored in one file of the data base.

The data structure and semantics are set out in Zebrowski et al. (1985).

Possible Extensions:

The main features of the extended PDA version (a corresponding IIASA local currency funds project has already been initiated) are the following:

- spatially distributed structure, the central concept is a site or a location, containing an industrial installation with one or several technologies installed;
- data base driven, flexible structure, that can configure (generate) an appropriate model for the LP problem solver based on a site data base (describing the locations (see 2.1.05) and a spatially distributed demand structure), and a chemical technology data base (see 2.1.07);
- extended list of criteria: the following criteria are required:
  - economic criteria
  - substance/waste criteria (for totals as well as for a few (3 to 5) specific substances or substance classes)
  - resource criteria (energy, water, labor, investment capital)
  - risk: substance related
    process related
    equipment related
  - transportation costs
  - transportation risk
  - cost of waste treatment and disposal.

Criteria apply globally, i.e., for the entire set of sites selected (which, however, could be a single site), constraints can be set individually.

The main modules of the extended system are:

1) A SITE DATA BASE:

The site data base contains, for every site considered, the following entries:

- ID and location (ID number, latitude, longitude, name of town, etc.);
- installed technologies and their design capacities;
- site-specific cost coefficients (e.g., unit costs for energy, labor, transportation cost array for all other nodes in the network etc.);
- site-specific risk coefficients (transportation);
- default set of constraints (on criteria as well as on technologies/capacities).

As a special case, or possibly an independent data base, demand nodes specifying an array of product substance demands and distances to the individual production nodes will be required.

2) A TECHNOLOGY DATA BASE:

The technology data base contains technology-oriented data that are assumed to be site independent. At least three alternative sets of technologies (standard, high efficiency, and pollution control) with correspondingly varying cost/waste/risk coefficients should be provided for every production technology considered.

In general, data base structures should be designed to accomodate a list- or frame-oriented extension, with a DBMS implemented in Lisp.

3) MODEL GENERATOR:

The model generator will generate an appropriate model for the problem solver, based on

1) the selection of an industrial sector (to be implemented as example test data sets: pesticides, fuels & feedstocks);
2) the selection of an arbitrary set of sites (representing, e.g., a single installation, a region, a multi-national firm, a national industry, etc.).

A parallel pilot feasibility study for an extended version of CARBO (Industrial Structure Optimization for the feedstocks and chemical intermediates) see Fedra et al. (1985) has been initiated. The extended model system will include:

- spatially distributed representation of production facilities and corresponding model coefficients;
- non-linear extension of the current model;
- detailed treatment of (hazardous) waste and process hazards;
- extension to several sectors/sub sectors and groupings of sub sectors of the industry;
- linkage to transportation risk/cost analysis;
- linkage to OPTIMIZER for updating plant level data;
- inclusion of individual competing firms' economic behavior;
- inclusion of multi-site and transnational firms.

2.1.07 Chemical Process Plant Analysis

Chemical process plant analysis provides several related modules:

- a Chemical Technologies data base; here major manufacturing processes (e.g., chlorination of phenol, monochlorobenzene production of phenol) are described in terms of:
  - major feedstocks
  - major products
  - wastes and trace contaminants
  - individual process stream information (chemical processes, substances involved, pressure, temperature ranges, etc.)
  - hazard rating (overall and for individual process streams);
  - hardware configuration
- a simulation model (data set describing the chlorination of phenols), and
- an optimization model, which enables the cost minimizing process plant configurations and process parameters to be determined (example data set describing phenol production by monochlorobenzene method).

The simulation model is based on a symbolic description of the chemical processes involved (see Table in section 2.1.07.1) and is implemented in Lisp. Process representation as well as the process risk description is based on Goldfarb et al., (1981). A detailed description of the symbolic simulator and its implementation in Common Lisp is in preparation (Winkelbauer, 1985).

The optimization code, based on a generalized reduced gradient method (Grauer et al. 1980), is configured for the production of monochlorobenzene.

2.1.07.1 The Symbolic Process Simulator

Processes and substances involved in the simulator are summarized in the Table below.

In the simulation module the production process is represented by *Unit Activities* (Unit Processes [Herrick et al., 1979] and Chemical Processes) and *Units* (Zanelli et al., 1984), where the Unit Activities take place.

The combination of a Unit Activity and a Unit, which is necessary if the process is to occur, is called an *Operating Unit*.

In order to satisfy a special production goal the Operating Units are linked by their input/output streams (direct or indirect recursive included).

The production process starts as soon as input material is provided to the Operating Units which are connected to the external input streams. These Operating Units perform their Unit Activities depending on the input materials, the operating conditions of the Unit and the constituents of the Unit, and by this produce some output material, which they send (via the linked input/output streams) to other Operating Units, which are activated by getting input material. They too perform their Unit Activities and produce output, this activates other Operating Units and so on. After the production and the release of output material an Operating Unit is deactivated until it gets new input material.

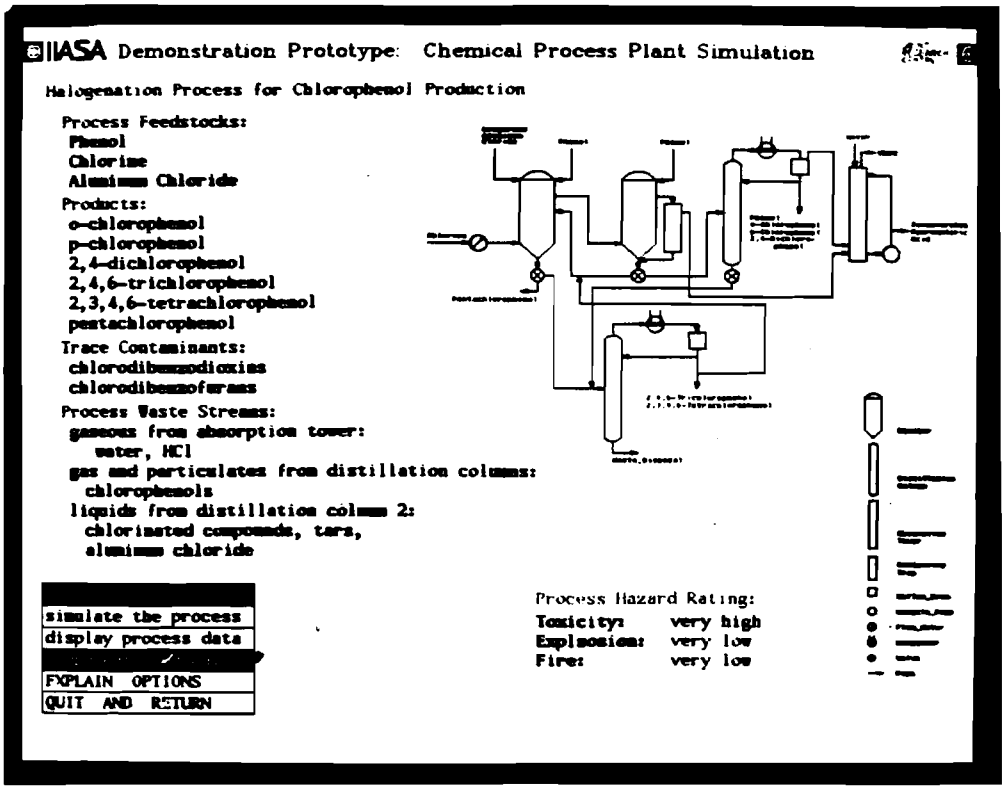


Figure 2.16: Summary of process data and plant layout

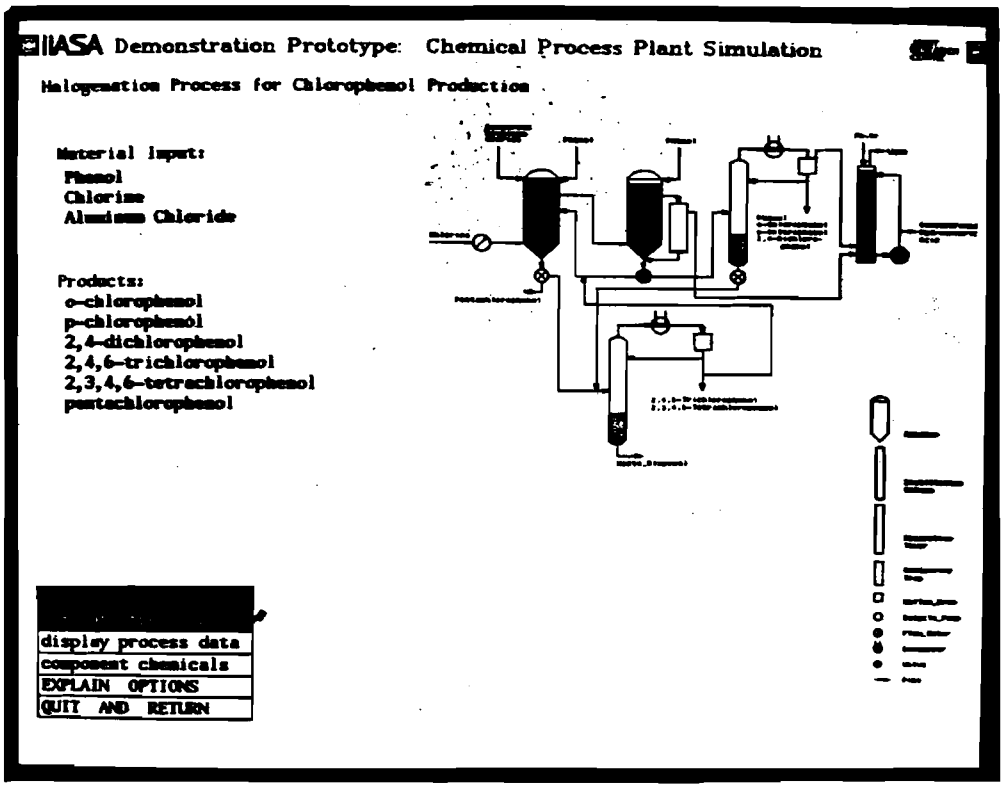


Figure 2.17: Dynamic simulation of the production process

Process:	Chlorination of Phenols
Sub-processes:	phenol + chlorine --> o-chlorophenol + HCl phenol + chlorine --> p-chlorophenol + HCl o-chlorophenol + chlorine --> 2,4-dichlorophenol + HCl o-chlorophenol + chlorine --> 2,6-dichlorophenol + HCl p-chlorophenol + chlorine --> 2,4-dichlorophenol + HCl 2,4-dichlorophenol + chlorine -(AlCl3)--> 2,4,6-trichlorophenol + HCl 2,6-dichlorophenol + chlorine -(AlCl3)--> 2,4,6-trichlorophenol + HCl 2,4,6-trichlorophenol + chlorine -(AlCl3)--> 2,3,4,6-tetrachlorophenol + HCl 2,3,4,6-tetrachlorophenol + chlorine -(AlCl3)--> pentachlorophenol
Substances:	phenol, chlorine o-chlorophenol p-chlorophenol 2,4-dichlorophenol 2,6-dichlorophenol 2,4,6-trichlorophenol 2,3,4,6-tetrachlorophenol pentachlorophenol anhydrous aluminium chloride hydrogen chloride, hydrochloric acid

This sequence of activation and deactivation of Operating Units by materials terminates when there is no more input material for any of the Operating Units, e.g., all external input has been transformed to the desired products, by-products and waste.

During the simulation of the production process the *Operating Hazards* of the Units and the hazards caused by the materials used and produced (e.g. input materials, interim products, end products, waste materials), the *Material Hazards*, are recorded and dynamically updated in the form of Hazard Ratings (NFPA, 1977; AICE, 1973; Sax, 1975).

Model Input:

- **production goal:** desired product (one of o-chlorophenol, p-chlorophenol, 2,4-dichlorophenol, 2,6-dichlorophenol, 2,4,6-trichlorophenol, 2,3,4,6-tetrachlorophenol, pentachlorophenol)
- **input materials:** the required input materials depend on the desired product and are listed by the simulation module automatically; the user only has to acknowledge that all materials can be assumed as being supplied

Model Output:

- **product mix:** a qualitative description of the desired product and by-products
- **waste,** a qualitative description of wastes, including Hazard Ratings
- **hazard description of the simulated process,** represented by the Hazard Ratings of all interim products with respect to the Operating Units.

The module is currently implemented in Franz Lisp on a VAX 11/750 under Berkeley UNIX 4.2. It will be converted to SUN Common Lisp and integrated with the graphical output and control modules as soon as an appropriate compiler is available.

2.1.07.2 Chemical Process Optimization

An optimization package specifically designed to assist in analyzing chemical engineering problems at the process and plant level is provided as the second major element in the plant level analysis. Within the framework of the management of hazardous substances and industrial risk, the system can help determine trade offs and interdependencies among and between design parameters of production plants and process technologies. The model implemented is provided with one specific example data set described below.

Phenol is one of the more important (interim) products of the chemical industry. The chlorobenzene method is one of the three industrial processes for its production. The specific cost of phenol produced in this way is high, so it is of the utmost importance to design an optimal cost plant for producing monochlorobenzene. The design of this plant is based on a general reactor-separator-system. The optimization problem solver is the software package OPTIMIZER based on the GRG (Generalized Reduced Gradient) method (Grauer et al., 1979).

The nonlinear, constrained optimization problem consists of designing a plant processing a given input stream of benzene (15 kmol/h) with maximal profit.

The corresponding objective function is then

$$G(x)=E(x)-K(x) \longrightarrow \max$$

with K(x) costs, as a sum of the costs of the input product benzene, for heating and cooling agents and the costs for equipment

E(x) proceeds from the sale of the output product obtained; the proceeds from the by-products (dichlorobenzene and hydrogen chloride) are used for the reimbursement of the cost of chlorine.

G(x) profit, where x is a vector of the decision variables (structure variables, apparatus dimensions, process parameters).

The vector of the decision variables is made up of the following components:

(1) Structure variables x_1 and x_2; by the variation of these parameters all feasible reactor combinations may be obtained

(2) Structure variable x_3; This variable identifies how much of the recycled raw material stream will be used for regenerative preheating

(3) Mean residence time t_1 and t_2 for the reactors R1 and R2

(4) Reflux ratio for distillation columns C1 and C2

(5) Temperature difference in the heat-exchanger E3

Given a realistic set of constraints, the model can estimate optimal profit and corresponding decision variables (design parameters).

Clearly, by extending and generalizing the existing demonstration example to

- accept arbitrary problem definitions from a process technology data base;
- include waste streams;
- include process risks,

the package could form a valuable and extremely powerful building block in the overall system. From the point of view of waste/risk minimization, or the definition of constraints on waste, emissions, or process risks as a regulatory agency might have to formulate them, the system offers the possibility of determining:

- what is technically feasible for a given technology;
- what is economically feasible with a given technology;
- what is the form of the trade-off relationship between economics and waste/risk aspects of the technology.

Since several additional criteria (such as resource consumption, capital requirement and life-time, labor costs etc. can be included in the optimization framework, a broad assessment of trade-offs is supported.

In terms of the overall system, OPTIMIZER can be used not only to study critical production processes in detail, but also to provide aggregate design variables in the form of yield or waste coefficients for the industry structure analysis PDA (Production Distribution Area), Dobrowolski et al., 1982, 1984. These coefficients can than be selected to represent various levels of waste/risk reduction versus economic efficiency trade offs.

Also, plant designs optimized for various criteria can be subjected to a more detailed risk assessment, i.e., fault-tree and event-tree analysis (e.g., by the package SAFETI [Technica, 1984] or by means of automatic fault-tree generator/analysis systems).

OPTIMIZER is currently implemented only at IIASA as a stand-alone program with its original user interface. As soon as the necessary licences are available, it can be integrated into the overall system

Data Structures:

The Chemical Technology Data Base is currently structured as a set of sequential files; the same random access conversions as described above can be used with increasing volume of the data.

A generalized data base for OPTIMIZER is currently under construction; it will supersede the current version of the Chemical Technologies data base and generalized to be used by several related program systems, including the PDA Industrial Structure Optimization package.

Possible Extensions:

In the next step of development, the symbolic simulator as well as OPTIMIZER will be integrated with the graphics based user interface to the Chemical Technologies Data Base. The extended system will contain:

- a generalized process technology data base (see section 2.1.07);
- an automatic input data preprocessor and model generator;
- a generalized problem solver.

As another extension, information required by fault tree analysis programs such as SAFETI could be included in this data base, such that process risks or overall plant risk can be estimated on a more detailed technical level. Clearly, the coupling of the set of programs covering the entire range from detailed physical simulation of individual physical/chemical processes (e.g., runaway reactions as simulated by SAFIRE) to the overall industrial structure and location/capacity distribution (spatially extended PDA) driven by a generalized chemical technologies DB and the industrial locations DB would provide a powerful package for industrial risk assessment and management.

2.1.08 Waste: Treatment and Disposal

The Waste Treatment and Disposal Option is based on EPA's RCRA Risk/Cost Analysis Model. A simple access to the waste treatment technology data base is under development, including a sideways-chaining mini-expert system to identify appropriate treatment technologies.

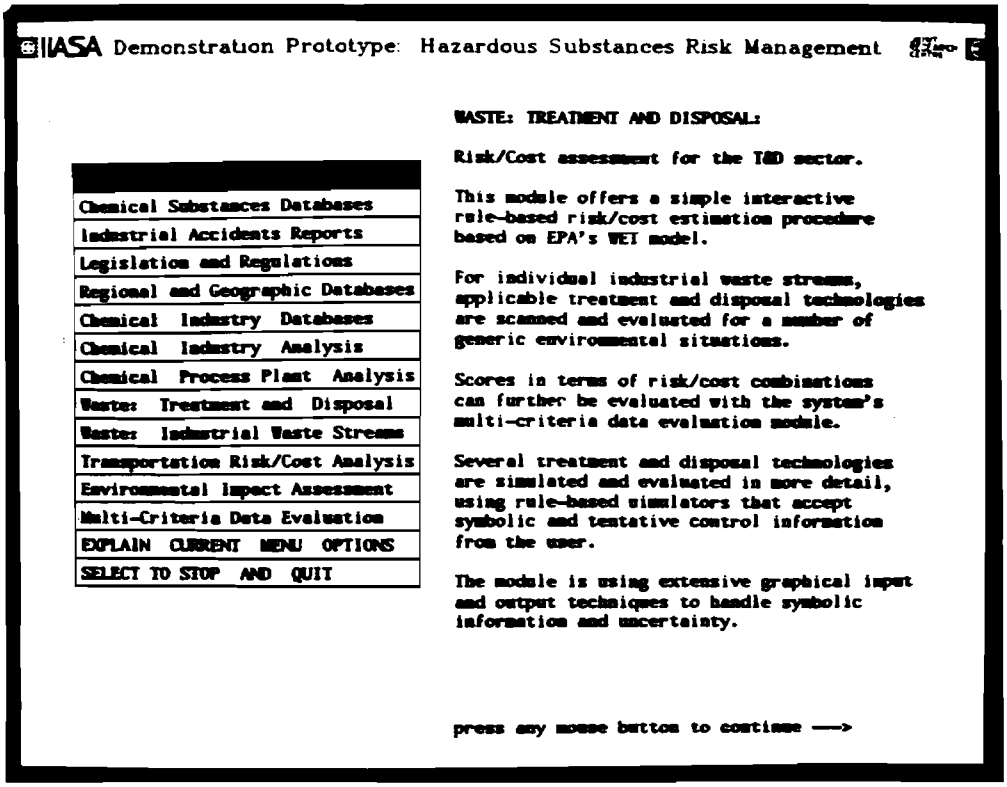


Figure 2.18 EXPLAIN page for the waste T&D option.

Sideways chaining is based on the Bayesian approach to inferencing and allows approved strategies for knowledge application and user interaction.

At first the system tries to identify the substances in the waste stream by asking the user about several attributes of the waste stream, in order to get the basis for estimating what kinds of substances the user wants to dispose. The dialog is structured by *rule values*, i.e., the system always asks the "most decisive" questions that would contribute most to resolving the diagnostic problem.

The user may not only answer "Yes" or "No". He can express uncertainty about the evidence in question with a certain set of qualifiers. As an example, a numerical range can be used (e.g., from +5 = "Yes" to -5 = "No", where 0 represents "Don't know at all"). More conveniently, values within this range can be elicited on the basis of linguistic hedges, or graphically, by pointing to an appropriate point within an interval representing the range from *Definitely YES* to *Definitely NOT*.

On the basis of these user inputs and estimations the system deduces the set of feasible treatment technologies and technology combinations which are appropriate for the identified substances. This listing of technologies also contains information about the Risk/Cost factors of the different technologies to enable the user to decide according to his preference.

Model Input:

- **Waste stream-specific information** (e.g., source, heating value, biodegradation rate)
- **Constituent-specific information** (e.g., concentration, mass fraction, molecular weight, vapor pressure)

The input values necessary to identify the waste stream and select appropriate treatment technologies originate from three major sources, namely:

- from the systems waste stream/treatment technologies data base;
- elicited from the user in the interactive dialog;
- inferred from the system on the basis of its waste stream knowledge base and the diagnostic user input.

Model Output:

- **Appropriate treatment technologies** depending on the outcome of the substance identification process
- **Risk/Cost factors** for the listed treatment technologies.

In the prototype version, this module is based on the USEPA RCRA Risk/Cost Analysis Model (WET, Waste-Environment-Technology). Only a small subset of the entire waste stream and treatment technologies data base is used. The module, however, could serve as the starting point for an improved European version of the model system.

Currently, only the waste stream identification part (see section 2.1.09) is implemented as an integrated part of the Industrial Waste Streams Data Base.

2.1.09 Waste: Industrial Waste Streams

Industrial waste streams (based on a subset of the RCRA data base) are described in a similarly structured data base. Access to individual waste stream descriptions is either from a list of waste streams, through industrial origin, waste stream names (interpreted by an intelligent parser), or waste stream properties.

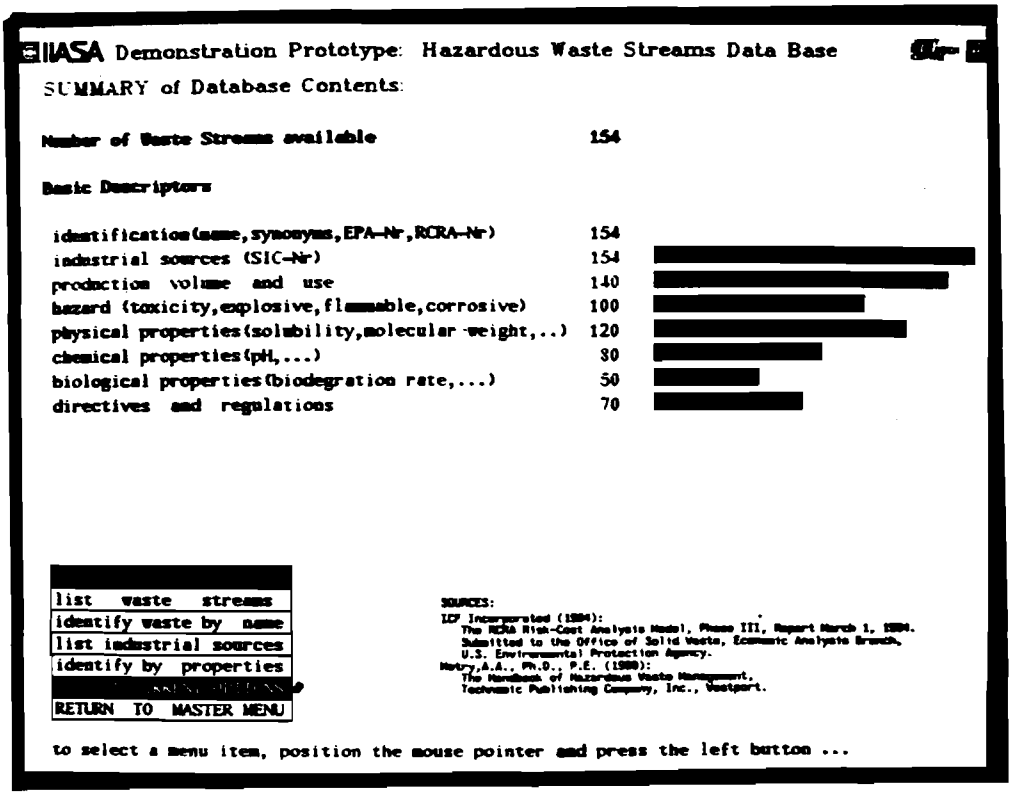


Figure 2.19: Top level of the Industrial waste streams data base

The industrial Waste Stream Data Base is organized like the Chemical substances data base (section 2.1.01). Based on the EPA W-E-T or RCRA waste stream data base, compiled by ICF (1984) for the US EPA, the data base provides information on currently 154 industrial waste streams.

Data Structure:

The Waste Stream Data Base consists of 154 industrial waste streams. Each waste stream is described by a data structure which contains specific information about the waste stream and a data structure with substance-specific information for each of the constituents of the waste stream which pose the greatest risk in the waste stream.

The data elements containing the waste stream-specific information are the following:

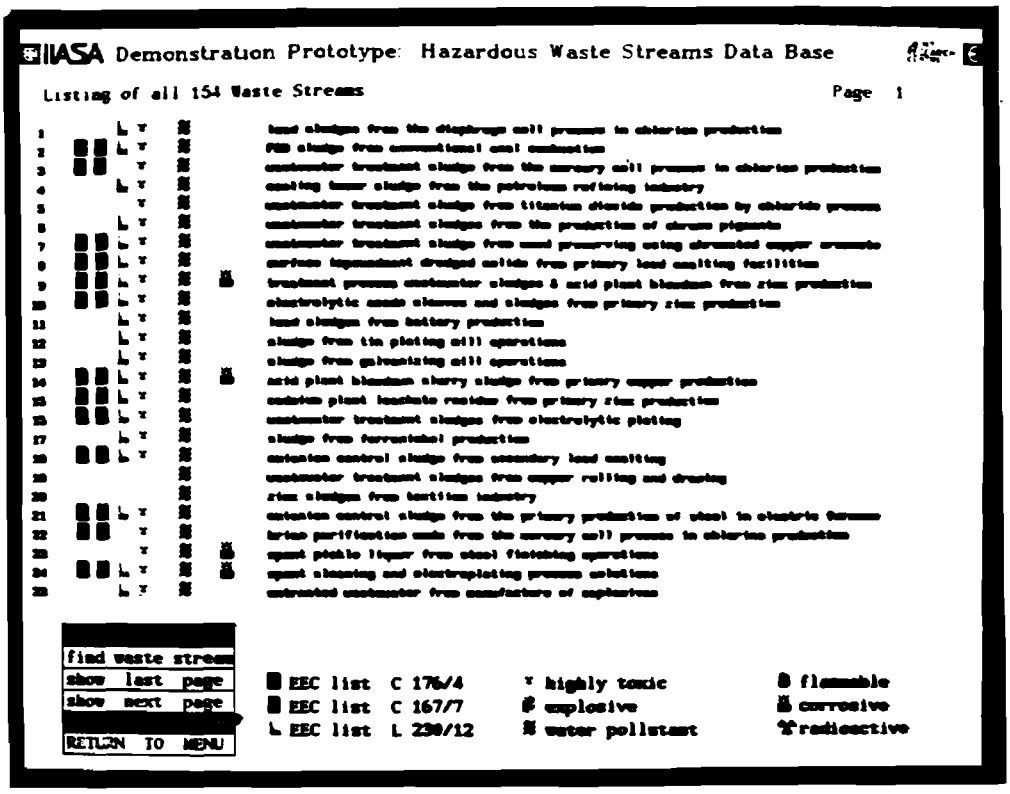


Figure 2.20: Listing the waste streams

- **id**: identification number for internal references
- **name**: describes the waste stream name by waste type, process type and production chemical for process wastes; for non-process wastes the waste type and source are indicated
- **rcra, epa, sic**: identification number from the RCRA-Report, the number(s) assigned by EPA and the the standard industrial code
- **quantity_year**: the total annual quantity of the waste stream generated in thousands of metric tons
- **facilities**: the total number of U.S. facilities that generate the waste stream
- **quantity_fac**: the average waste generation per facility
- **uncertainty**: an index used to denote the reliability of the information (1 = data available, 2 = values calculated from partial data, 3 = crude estimations obtained using minimal data)
- **sources**: number of source references
- **source**: list of sources from which the data is taken
- **fract_nonwater**: the mass fraction of the waste that is not water
- **fract_suspended**: the mass fraction of the waste that is or is not suspended in the liquid phase
- **solids_sg**: the specific gravity of the solids in the waste stream

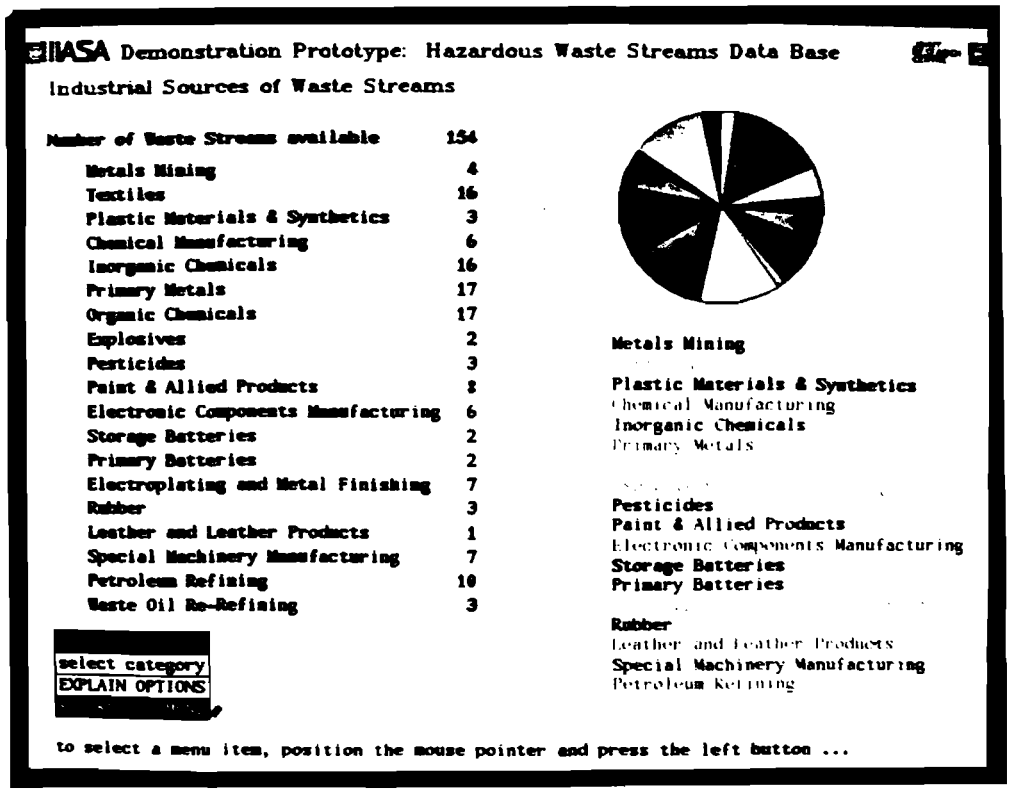


Figure 2.21: Selection by industrial origin

- **avg_sg**: the average specific gravity of the waste stream, calculated as the weighted mean of the liquid and solid fractions
- **heating_val**: the heating value of incinerable wastes, calculated as the weighted mean of the heating values of the waste constituents
- **fract_cl**: the organic chlorine content as a mass fraction of the waste stream (only for incinerable waste streams)
- **fract_ash**: the ash content as a mass fraction of the waste stream
- **ph**: the pH of the waste stream
- **biodegradation**: the first-order biodegradation rate per day for the waste stream
- **bod**: the ultimate biological oxygen demand of the waste stream, in milligrams per liter
- **constituents**: the number of waste constituents

The data elements containing the constituent-specific information are the following:

- **name**: name of the substance
- **concentration**: concentration in the waste stream, in parts per million
- **uncertainty**: an index used to denote the reliability of the information (1 = data available, 2 = values calculated from partial data, 3 = crude estimations obtained using minimal data)

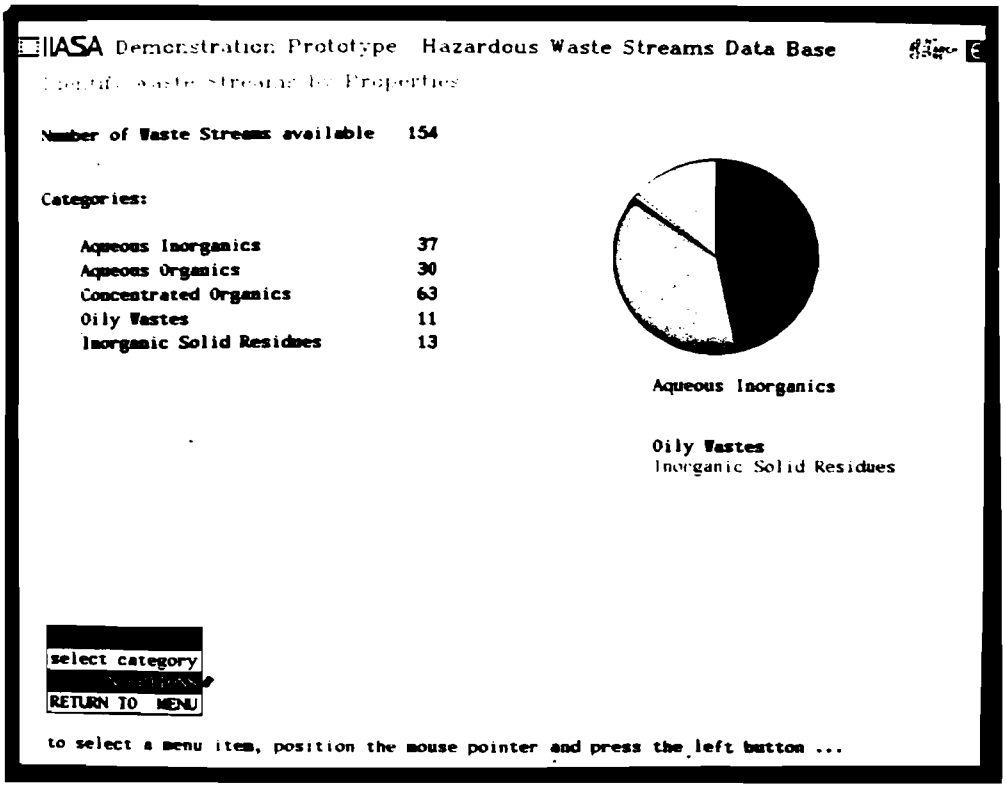


Figure 2.22: Selection by waste stream properties

- **sources**: number of source references
- **source**: list of sources from which the data is taken
- **mass_fract**: mass fraction of the constituent in the waste stream on a wet basis
- **fract_dissolved**: the mass fraction of the constituent in solution or in the liquid phase
- **mol_weight**: molecular weight of the constituent
- **vapor_pressure**: the vapor pressure of the pure constituent in millimeters of mercury at 25°C
- **solubility**: the solubility of the constituent at 25°C and pH 7.0
- **biodegradation**: the first-order biodegradation rate per day for the constituent of concern
- **bod**: the ultimate biological oxygen demand for the constituent of concern

Basically all information about hazardous waste streams which is available in the Waste Stream Data Base is entered and updated in a specific directory of the file system. Each file in this so called *Edit-Directory* contains all waste stream-specific and constituent-specific information for one waste stream. These text files contain a lot of redundant information and empty space (blanks) to keep the editing of the information easy.

To reach a satisfying data base access speed these edit files are compiled to a binary file by the binary data base generator, which not not only removes all the redundant edit information and strips out the empty spaces to improve the storage

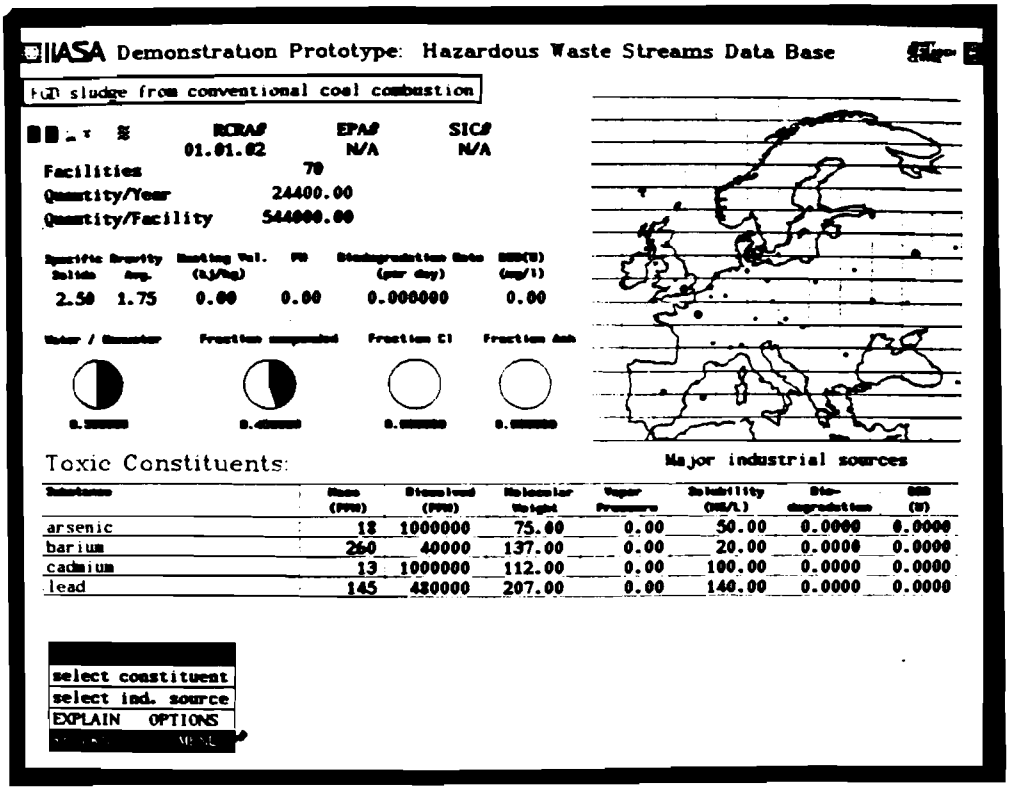


Figure 2.23: Individual waste stream description page (partly fictitious data).

efficiency, but also allows the binary file to be accessed like a random access file.

This is done by appending an array of offsets at the bottom of the binary file. These offsets are the starting positions of the information blocks which represent the information for each waste stream. When the data base is to be accessed, the binary file is opened and the offsets are read. Then the information about the specified waste stream is read from the position indicated by the offset which corresponds to the specified waste stream to the position indicated by the following offset (i.e., the starting position of the next waste stream information block) into a prepared buffer.

This reduces access time twice: once by avoiding the sequential reading of the file every time information about a specific waste stream is required, and once by transferring the whole information about one waste stream in one block into the buffer instead of reading every single data element.

2.1.10 Transportation System Analysis

The core part of the transportation system analysis model is currently under development by the Ludwig Boltzmann Institute, Vienna, under contract to the CEC/JRC (Study Contract No. 2684-85-04 ED ISP A). The model is described in detail in Kleindorfer and Vetschera, 1986.

The user interface supports the interactive selection of starting and end points for a transportation problem on the map. Substances can be defined by class or name as well as in qualitative terms using the HAZCHEM code for toxicity, fire hazard, and reactivity.

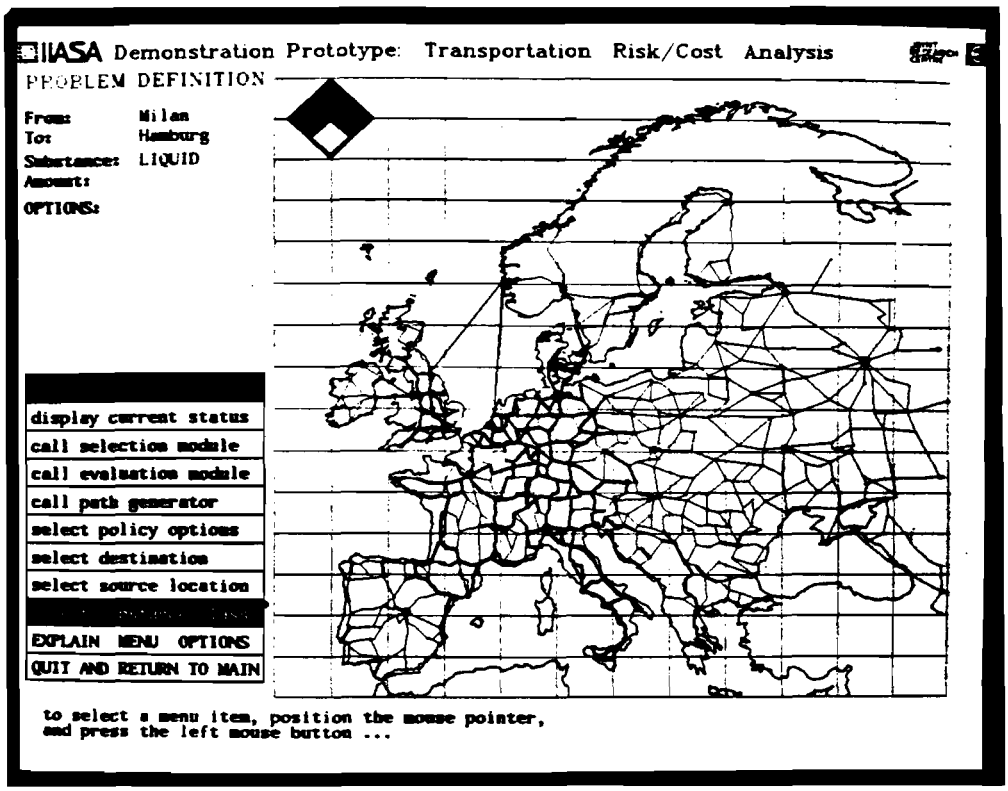


Figure 2.24: Transportation systems analysis: problem definition

The model generates all feasible routing alternatives under a number of heuristic constraints, evaluates in terms of risk/cost estimates, and calls on the multi-criteria data post-processor and optimizer to identify non-dominated (pareto-optimal) alternatives and finally to interactively select a preferred alternative.

Policy options include traffic restrictions and technical specifications (e.g., packaging, vehicles, accident management) as well as insurance and liability regulations.

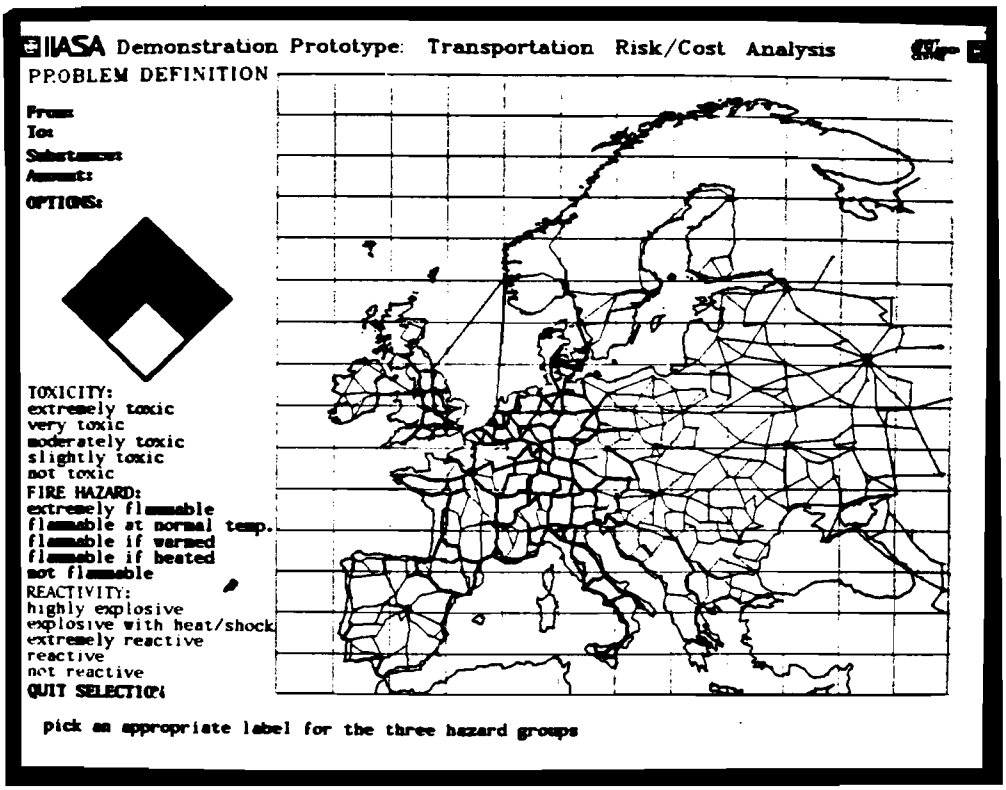


Figure 2.25: Transportation systems analysis: substance description by HAZCHEM selecting codes

The model is based on

- a geographical representation of a given region (e.g., of Europe) which specifies supply and demand points together with various routes connecting these points,
- on regulatory policies such as risk minimization and
- on economic policies such as cost minimization.

The function of this model is to enable the user to solve the problem of choosing the "best" route and mode for the transportation of hazardous substances from a certain supply point to a certain demand point, and in defining policies that ensure the selection of these mode/route alternatives.

Overall Structure of the Model

The model is designed as a policy-oriented tool. Its structure therefore, has to closely follow the structure of decision variables open to regulators. In general we can distinguish two different levels at which regulations might operate:

- a ***micro level***, dealing with individual transport activities or connections,
- an ***aggregated level*** aiming at global regulations that can be applied to a large class of shipments.

The model currently implemented in the framework system concentrates on the **micro level decision problem,** e.g., individual shipments of hazardous substances.

For analysis at the micro level the model will generate and evaluate possible transportation alternatives for a given transport objective. A transport objective is defined by the amount and type of hazardous substance to be transported and the points between which the goods are to be transported.

A transport alternative in the model is represented by a geographical route along which the transport is to occur and the choice of a transport mode, both associated with risk-cost criteria. The possibility of mode changes along the route is also considered in the model.

A detailed cost and risk analysis for all the alternatives generated is then performed and the results of this evaluation are presented to the decision maker for his final choice among the alternatives using the Interactive Data Post Processor.

From the perspective of software engineering the implementation of the model consists of three main modules:

- The first module generates candidate paths and consequently generates different route/mode combinations. To limit the amount of alternatives to reasonable numbers, the search area is restricted.
- The second module performs a risk-cost evaluation of the paths generated in the first phase. The outcome of the second phase is a list of criteria values of all the alternatives for further evaluation.
- The third module selects the "best" transportation alternative with respect to the criteria specified by the decision maker and the preferences expressed.

In most cases the number of alternatives is large and the selection of a preferred alternative from the set of feasible alternatives generated will require computer-assisted information management and decision support.

Based on the data structure described below, all possible paths (within a heuristically defined "window") are generated for each vehicle under consideration from the specified support point to the specified demand point.

For these paths risk and cost are estimated, and finally they are compared and evaluated.

Evaluation of Alternatives

Alternatives are evaluated in terms of cost and risk. The criteria of cost and risk are incommensurable; for instance, the cost of transportation is measured in monetary value and the risk of transportation is measured in the number of fatalities in the event of an accident.

Sometimes cost and risk are contradictory, for example the shortest – and thus usually the most cost-effective – connection is a highway that passes close to densely populated areas, with a higher risk potential than more remote, and therefore more expensive, routes.

In this model cost evaluation is based on freight rate sampled from commercial transport firms. The cost function is simply described by the following formula.*)

*) This cost function is only a very crude first approximation and strictly speaking only valid when the volume to be shipped is very large in relation to the capacity of any vehicle to be used. Also, the linear distance dependency only holds for relatively large distances.

$$C = c_f + (c_0 + c_s * X) * L$$

where

c_f: fixed costs
c_0: initial part of the variable costs function
c_s: slope of the variable costs function
X: amount of substance to be shipped
L: length of the path.

The risk analysis in the model covers both losses in the form of property damage and losses in the form of injuries and fatalities. Considering the stochastic nature of these losses expected values and the variance of losses are taken as decision criteria.

A simplified lognormal distribution risk analysis submodel is employed to evaluate the alternatives. As outcomes of risk analysis, the criteria of alternatives are described in terms of expected losses and variance of losses to a given group along a route in the network. Further on, the groups of objects that can be affected by accidents (population, property values etc.) will be represented by g.

The formulations of these criteria are as follows. The expected loss $E[R_g]$ of group g along route $(r_1, r_2, \ldots, r_l)$ is :

$$E[R_g] = \sum_{i=1}^{l} \prod_{k<i} (1 - p_a(r_k)) \cdot p_a(r_i) \cdot \sum_{n=1}^{N} q_n \cdot e^{\mu_n + \frac{1}{2}\sigma_n^2}$$

where

$p_a(r_k)$: the probability of an accident on arc k
q_n: the probability of an accident, which happens for type n land usage on arc r
μ_n, σ_n^2: parameters of lognormal distribution of conditional density function for type n.

The variance of losses to a given group g along route $(r_1, r_2, \ldots, r_l)$ is :

$$var(R_g) = E[R_g^2] - E[R]^2$$

where

$$E[R^2] = \sum_{n=1}^{N} q_n \cdot e^{2.(\mu_n + \sigma_n^2)}$$

and

$$E[R_g^2] = \sum_{i=1}^{l} \prod_{k<i} (1 - p_a(r_k)) \cdot p_a(r_i) \cdot \sum_{n=1}^{N} q_n \cdot e^{2.(\mu_n + \sigma_n^2)}$$

Both the expected value and the variance of losses to several groups are characteristic of a route/mode combination that will be used in evaluating the different alternatives. For three risk groups (property damage, fatal and non-fatal injuries) six risk-related objectives can be considered in the evaluation.

Combining these six objectives with cost, we can get a well-defined multiobjective decision problem with seven criteria.

Model Output

The output of the transportation model consists of a list of criteria for all the alternatives:

The *risk indicators* are represented as follows:

- risk groups (e.g., damages, injuries, deaths);
- possibilities of accidents (*a priori*);
- consequences of an accident, depending on the substance involved, land usage class and risk group.

The *cost factors* are described by the following variables:

- transport costs, fixed and variable;
- insurance costs, depending on the type of arc and the transportation medium used.

Data Structures:

The data structure of the Risk-Cost Analysis Model for the Transportation of Hazardous substances consists of four main parts:

- a description of the transportation network, i.e., the cities and the links between them,
- risk indicators,
- cost factors,
- general information about the model.

The *general information* about the model is represented by the following elements:

- substances to be transported, described by their specific gravity,
- a description of the *descriptors* of the arcs,
- a list of risk groups: damages, injuries and deaths,
- a list of land usage classes: urban, suburban and agricultural,
- the vehicles (i.e., trucks, cars, trains, etc.), described by capacities.

The *transportation network* is described as follows:

The *nodes* describe the cities by their relative coordinates.

The *arcs* describe the links between the cities, e.g., the road or rail system by their

- length,
- mode (e.g., road, railroad, etc.)

- descriptors (e.g., tunnel, bridge, etc.)
- type (e.g., highway, minor road, etc.)
- shares of land usage class, i.e., the kind of environment (e.g., urban, suburban, agricultural) the road or rail passes through.

2.1.11 Environmental Impact Assessment

Environmental impact assessment offers a choice among several environmental transport and impact models. Again, a direct connection of some of these models to other modules (e.g., an atmospheric dispersion model connected to the industrial process/accident simulation modules) is foreseen.

In the current prototype version the following models are implemented:

- TOXSCREEN Multi-Media Screening Level Model (river sub-model)
- FEFLOW 2D Groundwater Pollution
- LRAT Long-Range Atmospheric Transport.

2.1.11.1 Environmental Models Master Menu

As an intermediate level, an environmental models master level allows selection among the models listed above. Two modes of selection are supported: the user can either select a symbolic problem description from a collection of icons, or he can compose a problem description along the following categories:

- **Source spatial characteristics**: point, multiple point, area;
- **Source time characteristics**: instantaneous, intermittent, continuous;
- **Receiving system**: atmosphere, soil, food chain, man, groundwater, surface water (river, lake/reservoir, estuary, coastal marine);
- **Spatial scale**: local, regional, European;
- **Time scale**: short term, long term;
- **Required resolution**: screening level, high resolution;

From the composition of indicators selected by the user through simply pointing at them, the system then selects the appropriate model based on a set of rules and mappings. If no appropriate model is currently available in the system, a corresponding diagnostic message is printed and the user is asked to modify his problem description.

2.1.11.2 TOXSCREEN Multi-media Screening Level Model

A generic multi-media framework for substance evaluation is provided by TOXSCREEN (Hetrick and McDowell-Boyer, 1979, 1984). TOXSCREEN, developed at Oak Ridge National Laboratory, is designed to assess the potential environmental fate of toxic chemicals released to air, water, or soil. It evaluates the potential of chemicals to accumulate in environmental media and is intended for use as a screening device.

The model makes a number of simplifying assumptions and originally operates on a monthly time step. Assumptions include a generic (worst case) positioning of surface water bodies relative to atmospheric pollutant sources and contaminated land areas. The data used are typical of large geographic regions rather than site-specific. This multi-media screening tool will therefore be augmented by a second layer of more detailed and site-specific models for the individual environmental media. This results in a hierarchically organized system of models of various degrees of resolution in time and space as well as in the complexity of the model equations.

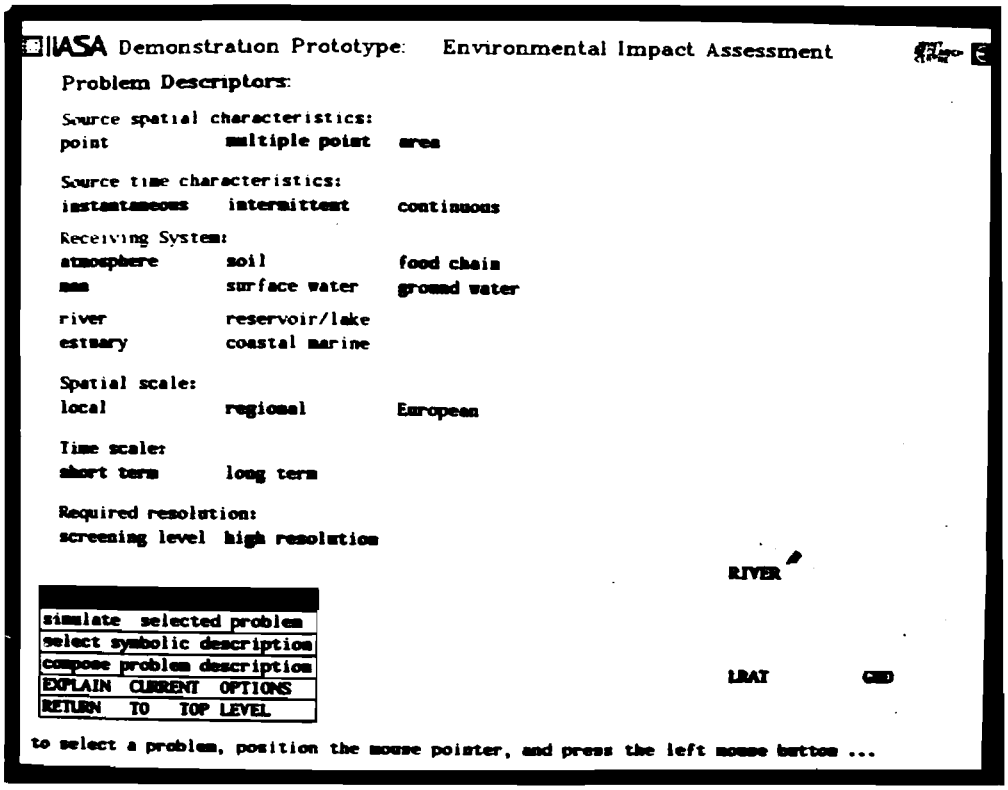


Figure 2.26: Environmental master menu

In TOXSCREEN, the physical/chemical processes which transport chemicals across air-water, air-soil, and soil-water interfaces are simulated explicitly. Deposition velocities, transfer rate coefficients, and mass loading parameters are used. Monthly pollutant concentrations in air, surface waters, and soil reflect both direct input to any or all of the media from a specified source or sources, and subsequent interaction via processes such as volatilization, atmospheric deposition, and surface runoff. Methods for estimating bioaccumulation in the food chain are also included.

As one concrete example for this generic style, a river model based in part on TOXSCREEN's river module, has been incorporated into the demonstration prototype. To simulate dispersion in a river or part of a river, the river is divided into a number of geometrically equivalent reaches all of which have the same flow rate. An equation similar to the one used in EXAMS (Smith et al., 1977; Burns et al., 1981) is used to estimate the pollutant mass in each timestep in each reach. Instantaneous mixing of pollutants upon entry into each reach is assumed; pollutant concentrations are computed for dissolved neutral, dissolved ionic, and adsorbed forms, according to chemical equilibria. Adsorption on sediment is also described. A number of first-order rate constants (e.g., biodegradation, hydrolysis, volatilization) are used to simulate decay phenomena. A more detailed description of the river model, the models it is based on and various alternative models of higher resolution is given in Fedra et al., 1986c.

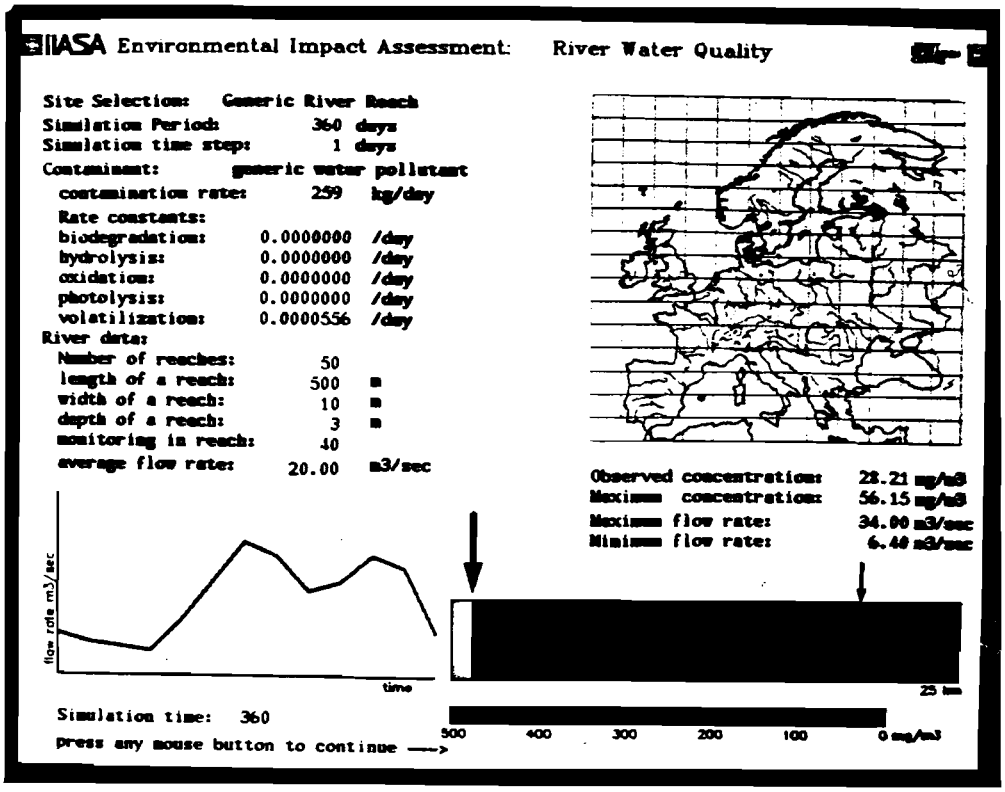


Figure 2.27: TOXSCREEN: the river pollution module

Data Structure:

All default data for the river module are stored in the model code itself. They only serve as reference points for the interactive definitions by the user.

Possible Extensions:

In addition to the direct coupling to the chemical substances data base in order to obtain model-specific properties for selected substances (rather than defining these properties in terms of rate constants directly), the module should be exchanged or augmented by a more sophisticated version, allowing for variable river geometry and flows (e.g., TOXIWASP).

2.1.11.3 FEFLOW 2D Groundwater Pollution

FEFLOW is a sophisticated two-dimensional finite element model for the simulation of contaminant transport in porous media (Diersch, 1980; Diersch and Kaden, 1984). It has been used successfully in several case studies.

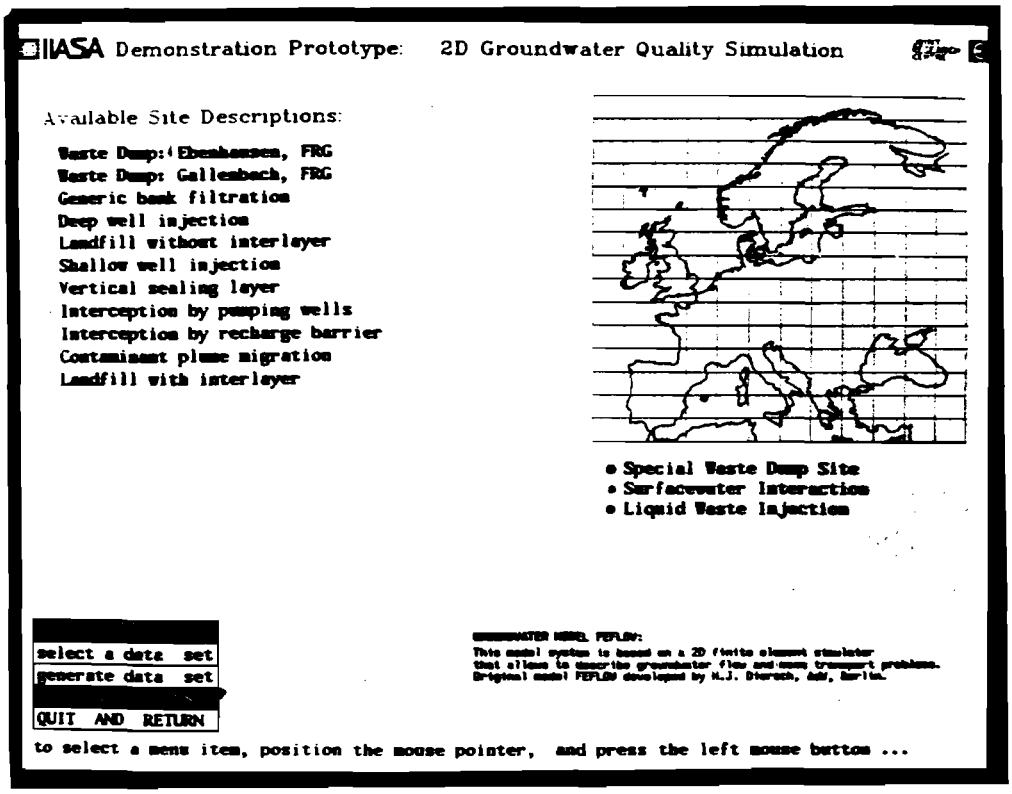


Figure 2.28: FEFLOW Master Menu, problem/site selection.

In the current implementation, FEFLOW can be used for a number of either specific or generic problem situations. The demonstration data bases include:

- landfill with observation wells and pumping gallery (horizontal);
- bank filtration with pumping well gallery (horizontal);
- landfill with deep and shallow well and interlayer (vertical);
- deep well injection with interlayer (vertical);
- landfill with recharge barrier (horizontal);
- landfill with pumping barrier (horizontal);

There are numerous ways of describing or specifying problems that can be addressed with the system described here. Control variables that can possibly be modified by the user include:

- selection of a specific site;
- selection of a generic problem type (e.g., contamination from a landfill, bank filtration, deep-well injection);
- selection of basic hydrogeological conditions (affecting groundwater recharge and flow regimes);

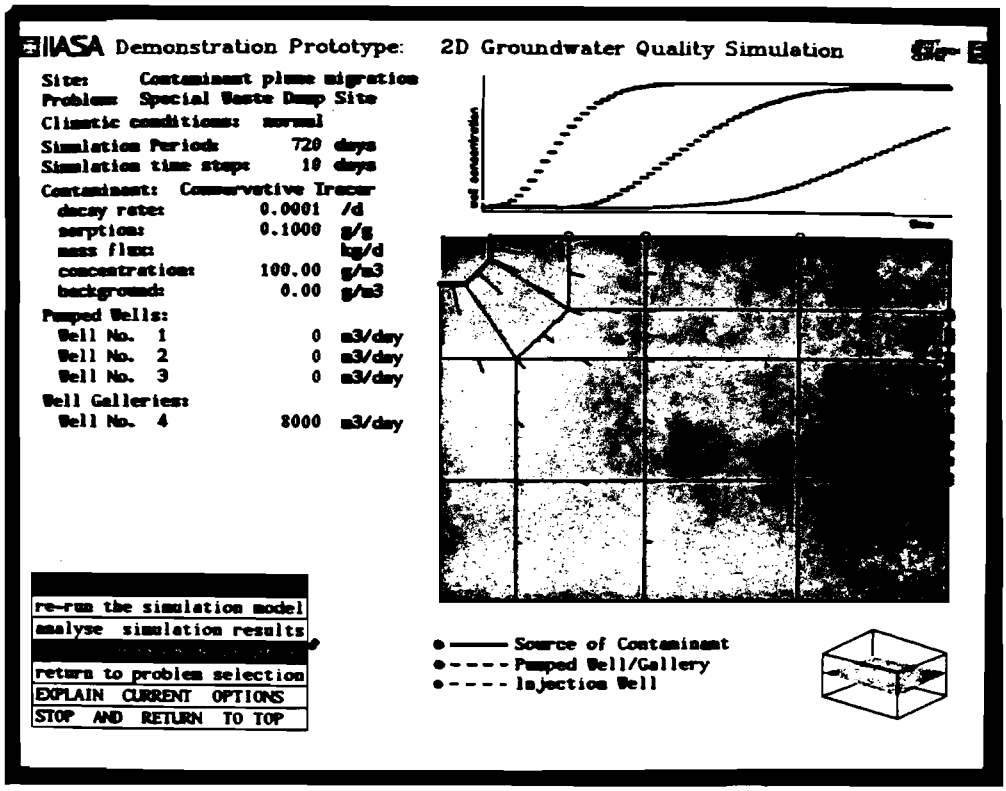


Figure 2.29: Generic Problem: contaminant plume migration.

- selection of substance and contamination source characteristics (e.g., type of substance, decay rate, sorption rate, mass flux or concentration at the source);
- background concentration of contaminant;
- pumping rates for wells and well galleries.

In addition, for the simulation of operational decisions, the user can either set the maximum simulation time and then run the simulation like a movie, or he can single-step through the dynamic part, and modify source characteristics as well as pumping rates, at any time step.

Data Bases and Pre-processors

Once a certain method has been selected on the basis of the user's problem specification, the system must then prepare the necessary input data. Every method has its own set of data requirements, and since the component modules are by and large based on existing software products, they also have their specific formats and conventions.

Background data for the model system are compiled for a number of fictitious "specific" test sites as well as for a number of generic problem descriptions. For all sites or problem descriptions, this information includes:

- finite element mesh with initial and boundary conditions, hydrogeological data, and operational (well) data;
- control variables for numerical problem solution.

The Simulation Model

From the user's point of view, the 2D subsurface transport simulation model FEFLOW incorporated into the system generates predictions of groundwater contamination consequences in terms of concentrations in time and space. The system has to provide estimates to answer questions such as:

- how does a contaminated plume migrate from its source (e.g., waste dump, waste-well injection, landfill) through a flowing groundwater field under given hydrogeological conditions and, which is of particular interest here, under the influence of technologies for water utilization (e.g., drinking water supply wells)?
- in a given time, what extent of contaminant distribution can be expected? What is the duration of the contaminant movement?
- will wells or galleries be influenced by waste water migration, and if so, at what level of contamination?
- how can the contamination be stopped or decreased?
- which remediation strategies are recommended and how effective are they for the decontamination of the aquifer?

To support the experimental nature of the system, each of the control variables determining a problem situation can be modified independently. For example, once a certain problem is defined, the user can run it for several different amounts of substances, or different substances. Pumping rates can be changed, a hydraulic barrier can be introduced, or the dump site can be sealed off. In addition to the simulation of contaminant mass flow, the system also keeps track of the costs (and benefits for water sold) of these control options.

For the basic simulation model, three different types of input data sets or problem descriptions exist:

1) specific sites in existence;
2) generic problem descriptions;
3) user-generated problem descriptions.

In the current version of the software system, only types 1 and 2 are fully supported, while type 3 is limited to problem modifications that do not require changes in the geometry of the finite element idealization. For user-generated problems, the system currently only supports modifications on the basis of any of the generic problem descriptions, but not from scratch.

For existing specific sites, the user can choose any location either from a list of available locations or from a map with these locations, the problem type they represent depicted by the color of the marker symbol on the map. The completely specified problem description is then loaded from an input file, one for each specific site. The control variables available to the user depend on the specific

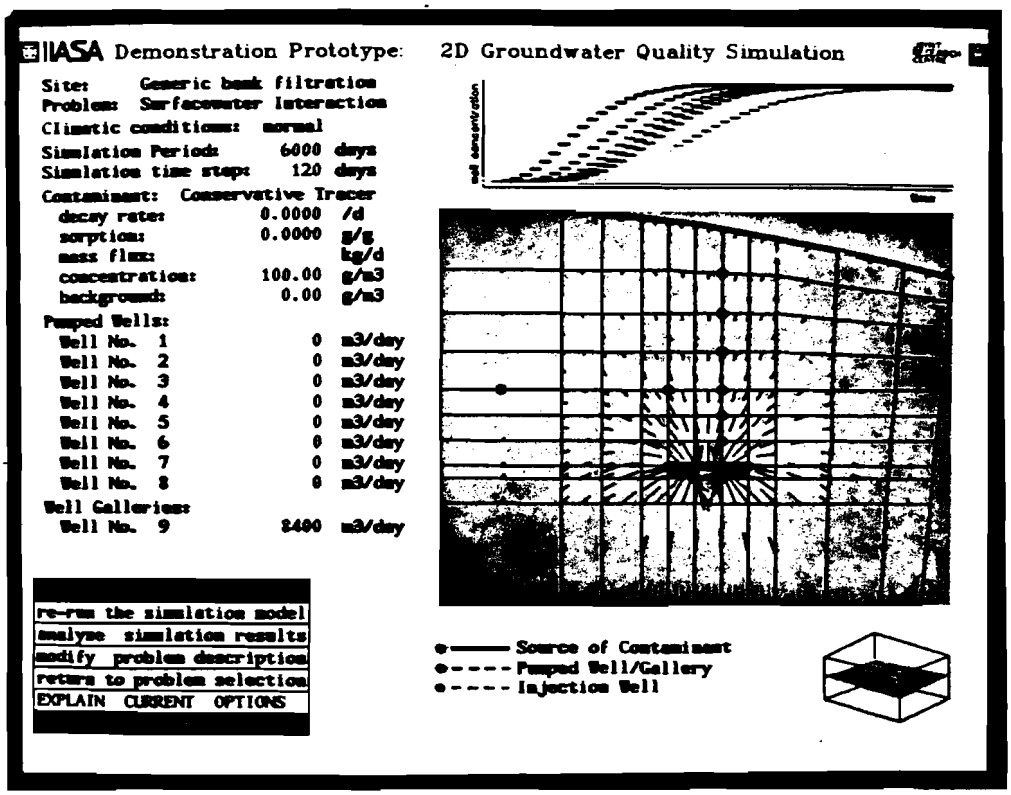


Figure 2.30: Generic Problem: bank filtration from polluted river.

case, but always include operational decisions (modification of the contaminant flow, activating/deactivating wells, modification of *uncontrollable* assumptions (hydrometeorology, substance properties). These variables can be modified within pre-defined limits, which are also part of the site-specific input file.

For generic problem descriptions, the user can choose from a list of cases supported. Again, his control variables are the same as above. However, in the generic cases many options are built into the problem descriptions. For example, most cases include a large number of wells at different locations, the majority of which, however, are not active. By activating/deactivating them through prescribing appropriate pumping rates, changes in well locations can be performed without having to modify the geometry of the problem.

Finally, the user-defined problems are either based on

- modifications and reconfiguration of existing descriptions, starting from any of the generic or specific problem descriptions, or
- completely generated from scratch.

In the latter case, the user starts by defining the mesh and the elements of his problem, specifying boundary conditions by identifying their type from a menu of available types and then picking the nodes to which they should apply, picking appropriate elements such as wells and well galleries and positioning them in the mesh, specifying hydrogeological parameters for a set of nodes/elements by just

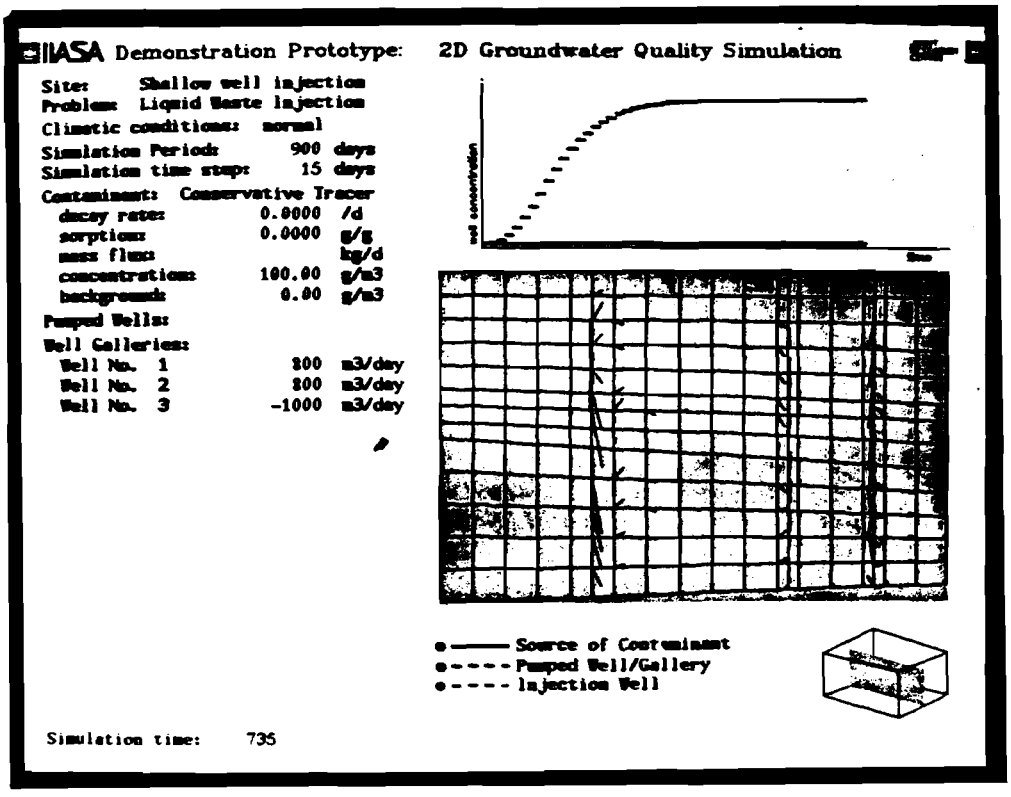


Figure 2.31: Generic Problem: shallow-well injection.

pointing at them or circumscribing them with the pointing device, etc. Design considerations for these options are currently underway.

The Simulation Model:

Basic Equations, Parameters, Initial and Boundary Conditions

Groundwater Flow

The flow processes in a two-dimensional (horizontal) aquifer are described by the following differential equation (for more details see also Diersch and Kaden 1984, Fedra et al., 1986c).

$$S \frac{\partial h}{\partial t} - \frac{\partial}{\partial x_i} (T_{ij} \frac{\partial h}{\partial x_j}) = Q_h \qquad (1)$$

for solving of $h = h(x_i,t)$ with the parameters to be prescribed as aquifer storage coefficient

$$S = n_0 + S_s M \qquad (2)$$

transmissivity tensor

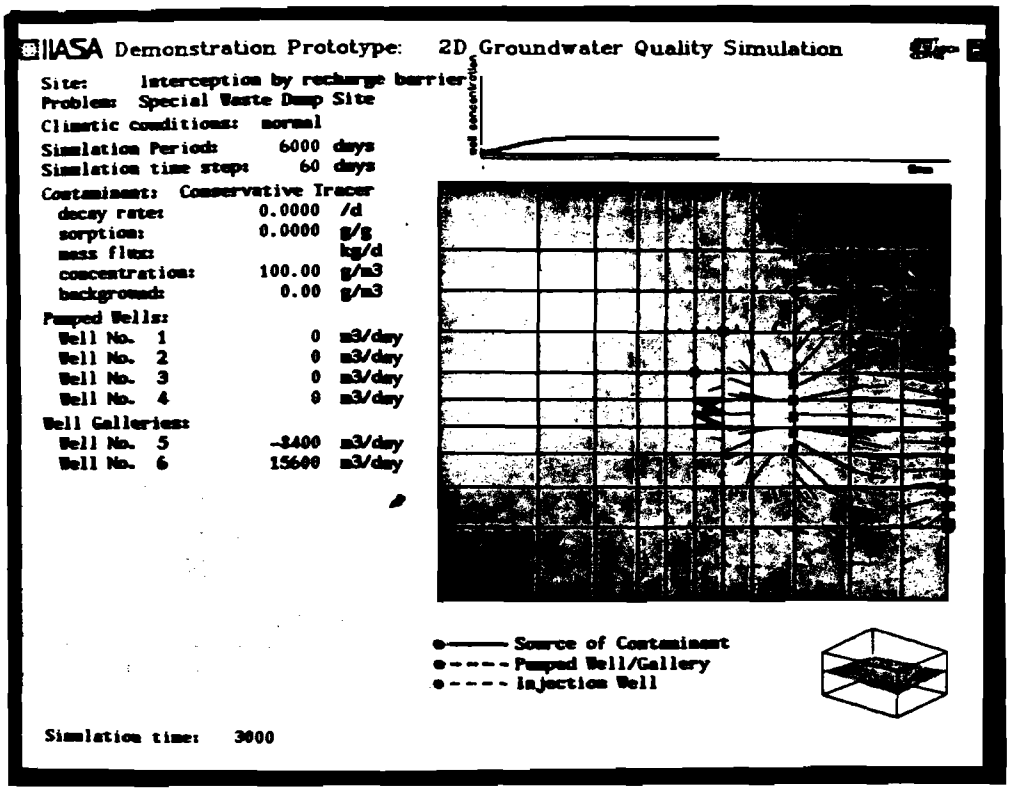


Figure 2.32: Generic Problem: recharge barrier.

$$T_{ij} = K_{ij}\, M \tag{3a}$$

or for isotropic conditions which are used practically

$$T_{ij} = T\ \delta_{ij} = K\ M\ \delta_{ij} \quad (L^2 T^{-1}) \tag{3b}$$

where

h = hydraulic head (L);

n_o = drainable or fillable porosity (L^0);

$S_s = i$ specific storage coefficient (L^{-1});

M = aquifer thickness (L);

Q_h = known source/sink function ($L\ T^{-1}$);

$T(T_{ij})$ = transmissivity ($L^2\ T^{-1}$);

$K(K_{ij})$ = hydraulic conductivity ($L\ T^{-1}$);

δ_{ij} = $\begin{cases} 1 & \text{for } i=j \\ 0 & \text{for } i \neq j \end{cases}$ Kronecker tensor;

x_i = spatial coordinates;

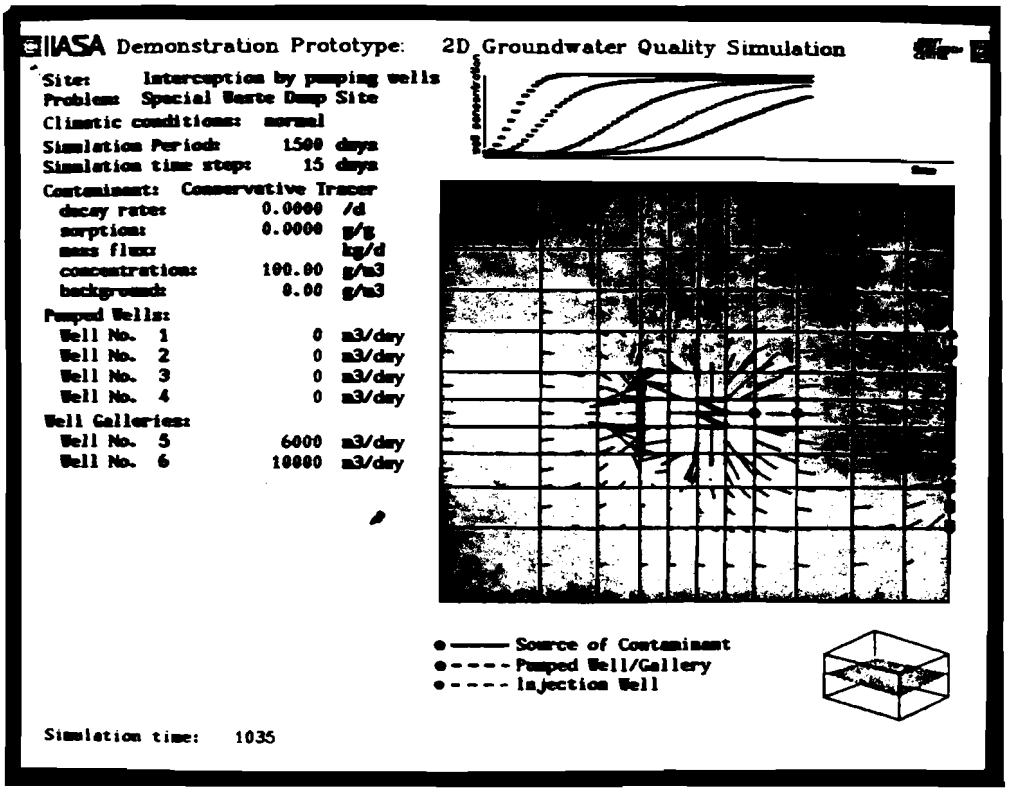


Figure 2.33: Generic Problem: pumping barrier.

$i, j = 1,2$ (summation indices).

Following initial (*I.C.*) and boundary (*B.C.*) conditions are given

$$\begin{aligned} &\textit{I.C.} \quad h(x_i,0) = h_o(x_i,0) \\ &\textit{B.C.} \quad h(x_i,t) = h_{1R} && \text{on } G_1 \text{ (first kind)} \\ &\qquad q_h = q_{hR} = -T_{ij}\frac{\partial h}{\partial x_j} l_i && \text{on } G_2 \text{ (second kind)} \qquad (4) \\ &\qquad q_h = L_h(h_{2R} - h) && \text{on } G_3 \text{ (third kind)} \end{aligned}$$

where

h_o = prescribed initial distribution of hydraulic head (L);

h_{1R} = prescribed Dirichlet-type boundary hydraulic head (L);

q_h = prescribed Darcy volumetric flux ($L^2 T^{-1}$);

q_{hR} = prescribed Neumann-type boundary volumetric flux ($L^2 T^{-1}$);

l_i = directional cosines (L^0)

h_{2R} = prescribed Cauchy-type boundary hydraulic head (L);

L_h = so-called colmation (transfer) coefficient describing the aquifer-surface waters interactions ($L\ T^{-1}$).

The flux q_h is positive if the fluid flux is outward and negative if the fluid flux is inward directed on the section G_2 of the entire boundary G. The parameter L_h is so introduced that it is always positive if along the boundary G_3 an inflow for $(h_{2R} - h) > 0$ is to be simulated (h_{2R} is a reference boundary hydraulic head, e.g., the water table of an adjoining river or lake).

--> Summary:
Model equation (1)
for solving hydraulic head h
necessitates the knowledge of
* four parameters/functions T, S, Q_h, L_h and
* initial (h_o) and boundary (h_{1R}, q_{hR}, h_{2R}) conditions.

Mass Transport

The governing field equations for describing the movement of dissolved chemical species in flowing groundwater is given by (see also Diersch and Kaden 1984)

$$M\,R\,\frac{\partial C}{\partial t} + \frac{\partial}{\partial x_i}(q_i\,C) - \frac{\partial}{\partial x_i}(D_{ij}\,\frac{\partial C}{\partial x_j}) + M\,R\,\lambda\,C = M\,n\,Q_c \tag{5}$$

for solving of $C = C(x_i, t)$ with the specific Darcy flux

$$q_i = -T_{ij}\frac{\partial h}{\partial x_j} \tag{6}$$

known from the solution of flow equation (1), and the parameters to be prescribed as retardation factor (assuming a linear isotherm equilibrium adsorption)

$$R = n + (1 - n)\,\kappa \tag{7}$$

the dispersion tensor written for isotropic conditions

$$D_{ij} = (n\,D_d\,M + \beta_T\,V_q)\,\delta_{ij} + (\beta_L - \beta_T)\,\frac{q_i\,q_j}{V_q} \tag{8}$$

where further

C = concentration of chemical species ($M\,L^{-3}$);

n = kinematic porosity (L^0);

κ = sorption coefficient (1);

D_d =molecular diffusion coefficient (L^2T^{-1});

β_L, β_T=coefficients of longitudinal and transverse dispersivity, respectively (L);

$V_q = (q_i\,q_i)^{1/2}$ absolute specific flux ($L^2\,T^{-1}$);

λ= J concentration decay rate (T^{-1});

Q_c = known source/sink function ($ML^{-3}T^{-1}$).

The following initial and boundary conditions hold:

$$\text{I.C.} \quad C(x_i, 0) = C_o(x_i, 0)$$

$$\text{B.C.} \quad C(x_i, t) = C_{1R} \qquad \text{on } G_4 \text{ (first kind)}$$

$$q_c = q_c R = -D_{ij}\frac{\partial C}{\partial x_j}\,l_i \qquad \text{on } G_5 \text{ (second kind)} \tag{9}$$

$$q_c = -L_c (C_{2R} - C) \qquad \text{on } G_6 \text{ (third kind)}$$

where

C_o = prescribed initial distribution of species concentration (ML^{-3});

C_{1R} = prescribed Dirichlet-type boundary concentration (ML^{-3});

q_c = prescribed normal concentration flux ($ML^{-1}T^{-1}$);

q_{cR} = prescribed Neumann-type boundary concentration flux ($ML^{-1}T^{-1}$);

C_{2R} = prescribed Cauchy-type boundary concentration (ML^{-3});

L_c = concentration transfer coefficient (L^2T^{-1}).
The signs of the normal fluxes and transfer coefficients have the same meaning as the fluid fluxes discussed in.

--> Summary:
Model equation (5)
for solving species concentration C
necessitates the knowledge of
* nine parameters/functions M, n, κ, D_d, β_L, β_T, λ, Q_c, L_c and
* initial (C_o) and boundary (C_{1R}, q_{cR}, C_{2R}) conditions.

Possible Extensions:

An automatic mesh generator that will allow the interactive (CAD) construction of a complete problem description including arbitrary specifications of wells, boundary conditions, mesh geometry and geomorphological properties of the aquifer, is under construction.

Also, a second layer of a simplified model that can be subjected to optimization procedures (e.g., to minimize pumping losses in a pumping barrier while keeping in-well concentrations below a specified standard) is under construction.

Finally, the model describing the sources of contaminations (i.e., rivers, see section 2.1.11.2) and landfill operations should be linked with the groundwater modules. In particular, models for transport through the unsaturated (soil) zone are of importance (e.g., SESOIL).

2.1.11.4 LRAT Long-Range Atmospheric Transport

The Long-Range Atmospheric Transport Model describes the pathways and deposition patterns of clouds of toxic material, emitted in a major industrial accident, on a European scale, covering a window from 15°W 35°N to 45°E 72.5°N. Simulations of individual trajectories extend from one to a maximum of six days or until the cloud moves outside the above window.

The model used is a Lagrangian transport model, based on Eliasson(1978). It is driven by 6-hourly synoptic wind-field data for Europe (EMAP data), specified on a 150 x 150 km grid, interpolated for the current simulation time and period. These data are used as the seed for a Markov-type synthetic time-series generator, that uses the user-specified weather conditions (storm/normal/calm) for scaling.

Similarly, precipitation data are generated based on overall probabilities (seasonal dependency) and the user choice of a weather situation (only affecting the weather at and around the point of release). The model considers the extension of the vertical mixing layer (depending on season and time of the day), and a time variable dispersion of the cloud. It simulates wet and dry deposition (as a

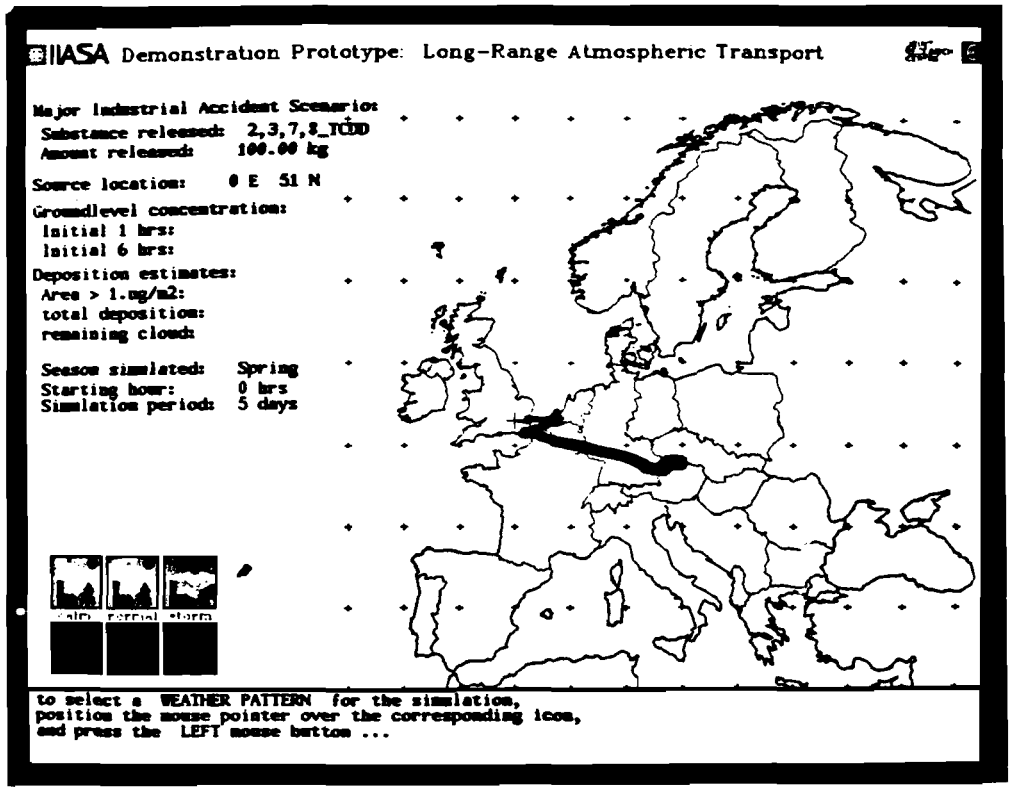


Figure 2.34: Long-Range Atmospheric Transport: selecting a weather pattern by selecting its symbolic icon representation

function of the specific gravity and, in the case of solids (dust) the average particle size of the substance emitted. A detailed description of the model and its extension is under preparation (Fedra and Bartnicki, 1986d).

The user specifies the following control variables:

- type and amount of substance emitted;
- location of the accident;
- season of the year and time of the day;
- weather pattern (sun/rain, and calm/normal/storm);
- duration of simulation.

The model calculates and displays the following information:

- initial concentration and concentration after 6 hrs;
- area contaminated with more than 1 microgram per square meter;
- mass of toxic substance still remaining in the cloud at the end of the simulation run;
- travel path of the cloud and its spatial extension are indicated on the map.

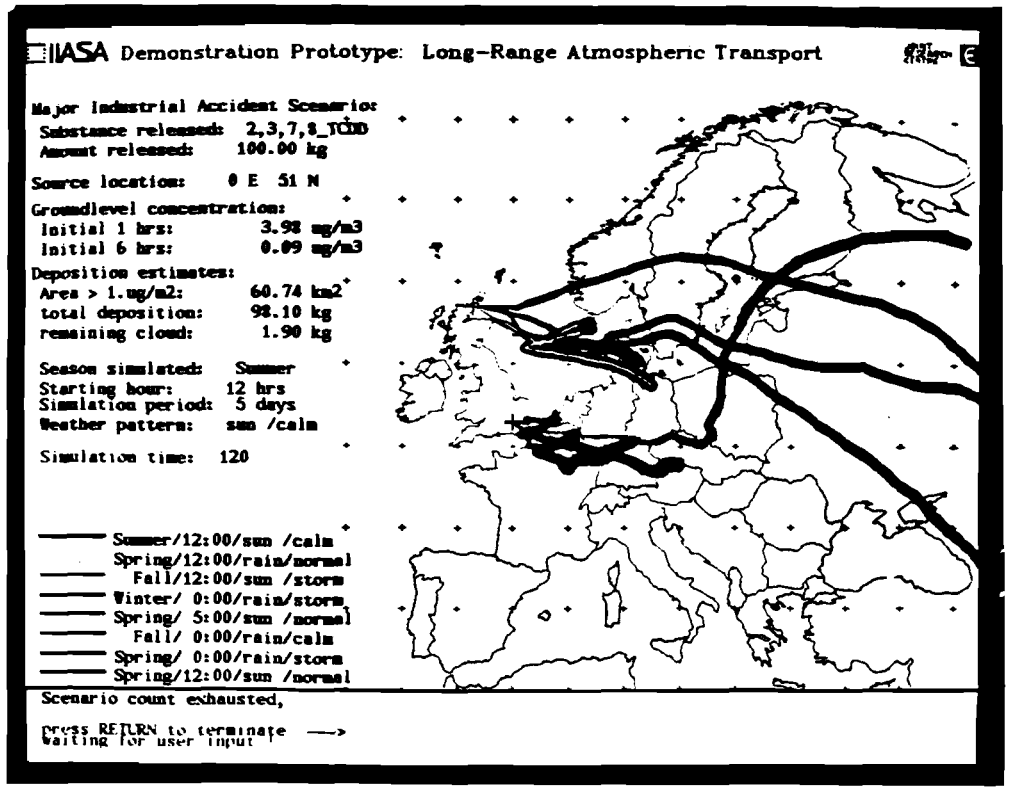


Figure 2.35 Long-Range Atmospheric Transport: run comparison

Possible Extensions:

An important extension of the model would be the development of a Monte-Carlo simulation framework, that can, given an accident description (i.e., some location, substance, amount) generate risk contours by automatically solving the model repeatedly for certain aggregate season/weather scenarios.

The resulting probabilistic information, together with the analysis of individual technologies (from this probabilistic set) would greatly increase the usefulness of the model. The probabilistic version, run for alternative sites, can then be directly examined with the post-processor/optimizer for e.g., siting alternative selection.

A basic prerequisite for the probabilistic multi-run version would be a largely extended wind field data base or the inclusion of a spatially distributed synthetic time-series generator.

A complex atmospheric modeling system, integrating several models ranging from simple analytical techniques to various Eulerian models and the Lagrangian approach described above, is under design. This system will feature a Problem Generator for specifying problems that can be addressed with the system under design.

Entry points are

- type of emission to be considered:
    - accidental release (instantaneous)
    - routine release (continuous)
- the source type:
    - single point source
    - multiple point source
    - area source
- the type of substance emitted:
    - dust
    - vapor
    - gas, heavy gas;
- the type of facility emitting (type of plant and technologies used)
- the location of the facility (existing or planned).
- the type of analysis:
    - simulation of a single event
    - integration over all possible events (risk contours).

Selection of combinations of these descriptors is again menu based, i.e., the user identifies textual and/or symbolic representations of the case descriptors that fit his problem best. For most descriptors, successive refinement down to detailed numerical values or by calling upon other modules of the system (e.g., the chemical substances data bases) is an option.

Data Bases and Pre-processors

Once a certain method has been selected on the basis of the user's problem specification, the system must then prepare the necessary input data. Every method has its own set of data requirements, and since the component modules are by and large based on existing software products, they also have their specific formats and conventions.

Required background data for the model system include:

- Climatic data, i.e., spatially and seasonally distributed wind speeds and directions, and precipitation data;
- Geographical data, that include population, land use, and topographical information;
- Chemical substances data (Fedra et al., 1986b), which include basic physical and chemical characteristics relevant for transport and impact modeling such as specific mass, solubility in water, biodegradation rates, toxicity etc.;
- Chemical technology information, that links production technologies and end products with an array of feedstocks, interim, by- and waste products;
- Chemical industry data, comprising process plant and storage facility sizes and locations, production technologies used at each location, etc.

In addition to these background data bases, each of the models the system supports has its own model-specific data and knowledge base, which contains the default values, ranges, or algorithms (rules) that allow the necessary input data

required by the model to be generated. Since most of these data are interdependent, rules specifying dependencies and how the selection of a given value influences others, are part of this information.

Depending on the information available, which describes the problem situation or context (provided by the user by means of the menu-driven problem generator or editor), the pre-processor has to then generate an appropriate input file for the method selected by the model-selection procedure.

Clearly, all these steps are not as orderly and sequential as the linear discussion suggests. Since backtracking and randomly sequenced specifications are supported, the consistency of the specifications has to be checked before any model can be used. All user-determined values however have default values that are used in lieu of an explicit specification.

Simulation Models

From the user's point of view, the simulation models incorporated into the system generate estimates of accident or routine release consequences in terms of ground-level concentrations or depositions in time and space. The system has to provide estimates to answer questions such as:

- for a given location, substance, and amount released, what will be the size of the area affected by depositions above a certain threshold ?
- what will be the spatial distribution of depositions for various weather situations ?
- what will be the (probabilistic) spatial distribution for the entire range of likely weather situations ?
- how far will a cloud of toxic material travel under worst-case assumptions ?
- how far, under the most likely weather conditions ?
- which maximum and average concentrations can be expected around the site of an accidental release ?
- for how long will these concentrations persist under various weather conditions ?
- what is the size of the population exposed to a certain concentration, or range of concentrations, for a given weather situation, or for all likely weather situations ?
- for a number of locations, how will the surrounding area and population be affected under various scenarios of weather conditions (worst case, most likely case, etc.) or substance/amount combinations ?

To support the experimental nature of the system, each of the control variables determining a problem situation can be modified independently. For example, once a certain problem is defined, the user can run it for several different amounts of substances, or different substances. Or, using the same worst-case definition, he can directly compare various locations. As a post-processor, the discrete optimizer (see 2.1.12) can be used.

2.1.12 Multi-Criteria Data Evaluation and Optimization

The multi-criteria data evaluation and optimization post-processor is available from the top level through this menu option. It can also be made accessible from several other options, notably the transportation risk/cost analysis module.

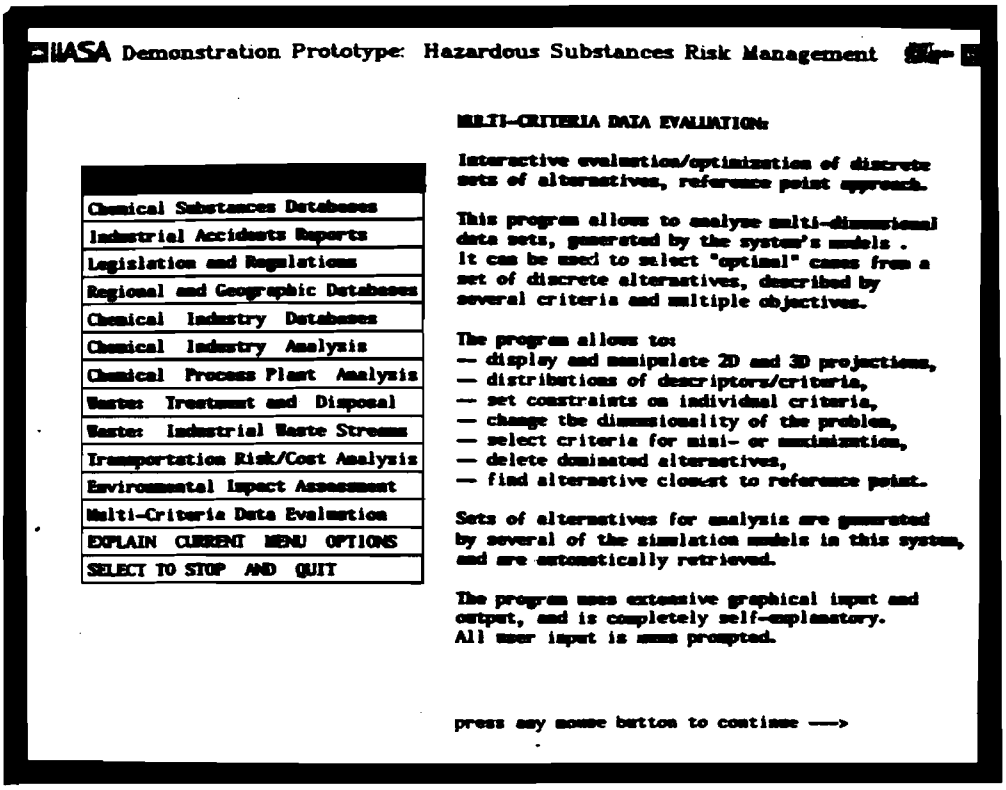


Figure 2.36: EXPLAIN page for the multi-criteria data evaluation option.

The module is based on features of several decision-support packages, and in particular on a discrete version of the reference-point method (Grauer et al., 1984, Majchrzak, 1985). This approach combines the analytical power of a 'hard' computer model with the qualitative assessments of the decision maker, moving from the classical OR approach closer to the human thinking process. As a decision-support system, the main task of the module is to solve multicriteria optimization problems with discrete criteria values, a finite number of alternatives and usually incommensurable criteria.

An overview of the state of the art of interactive decision-support packages and a description of the DSS module implemented here is given in Zhao et al. (1985). The report contains a detailed discussion of the underlying mathematics and a detailed treatment of an example session using the transportation risk/cost analysis model output.

The module supports the selection of criteria (minimize/maximize/ignore), the setting of constraints (maximum/minimum values of criteria to be considered), various display options (2D projections and frequency distributions), and finally the identification of non-dominated (pareto-optimal) alternatives. Extensions

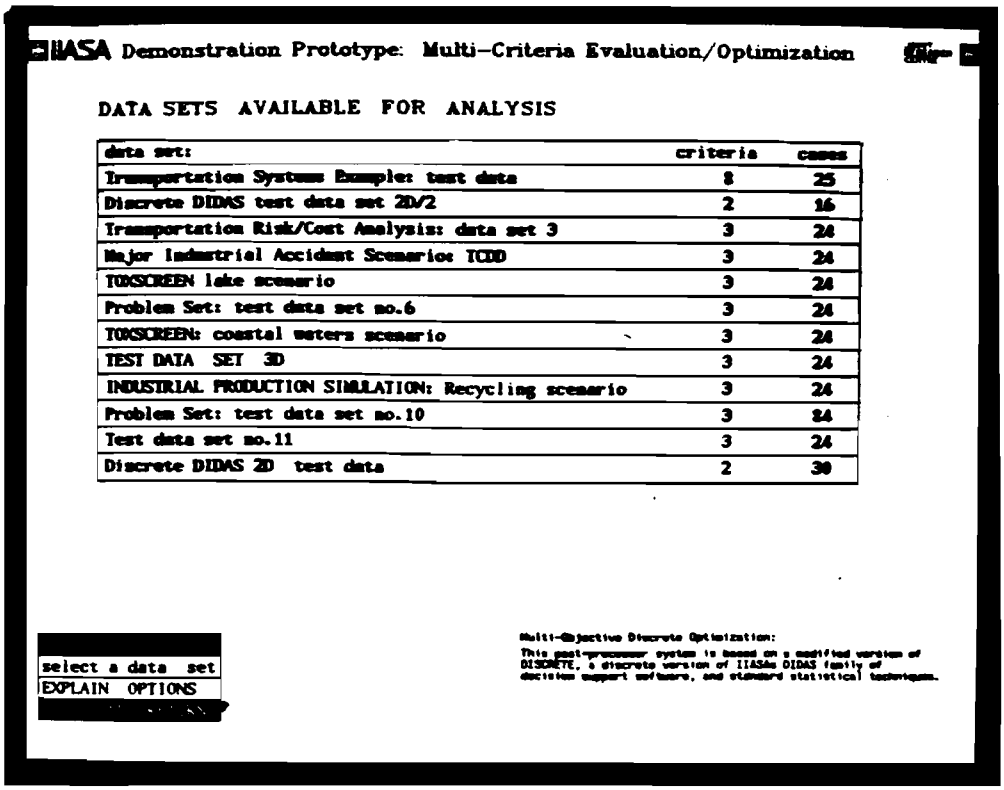


Figure 2.37: Multi-criteria data evaluation: selection of data set

include the definition of reference points and the display of complete alternative descriptions in terms of the original model generating them, e.g., transportation route alternatives displayed on the map.

The decision-support system can also be used as a post-processor for other multi-criteria optimization programs such as the plant and industry structure models. Here, a number of experiments involving the same number of criteria but various settings for constraints and reference points can generate data sets for post processing.

In general, the package is a powerful post-processor as well as optimization package for any discrete set of multi-criteria data, which is to say, any set of model outputs generated by repeated scenario analysis of a single model.

In a discrete, multiobjective decision problem all feasible alternatives are explicitly listed in the finite set $x^0=\{x_1,x_2,\ldots,x_n\}$, and the values of all criteria of each alternative are known and listed in the set $Q=\{f(x_1),f(x_2),\ldots,f(x_n)\}$. There are many tools which could be employed to solve this problem (e.g., Korhonen, 1985, Majchrzak, 1984). We have drawn on the method developed by Majchrzak (1985).

[9] This section is based on the Reference Point Approach developed by Wierzbicki (1979, 1980) and draws on the DISCRET package developed by Majchrzak (1984, 1985).

Usually, the procedure of problem solving is divided into two stages. The first stage is the selection of elements of a nondominated set from all the alternatives of set x^0. In the second stage, the "best" solution is identified as the decision maker's final solution to the problem under consideration, in accordance with his preferences, experience etc., as the basis for his decision.

In the discrete, multicriteria optimization module of the overall system, at the first stage of problem solving, the dominated approximation method is used to select the elements of the pareto set, because of its calculation efficiency and its ability to solve relatively large scale problems. For instance, this method can be used to solve a problem with 15-20 criteria and more than a thousand alternatives, which is sufficient for processing the data arising from scenario analysis in the framework system.

In the second stage, an interactive procedure based on the reference point theory is employed to help the user to find his final solution. This approach combines the analytical power of the "hard" computer model with the qualitative assessments of the decision maker in the decision process. It makes the decision process more reasonable and closer to the human thinking process. In the following, the methodology used in these two stages will be described briefly.

Selection of the Nondominated Set of Alternatives

Problem Formulation

We may describe the problem considered as a minimizing (or maximizing or mixed) problem of m criteria with discrete values of criteria and a finite number of alternatives n.

Let x^0 be the set of alternative admissible decisions. For each of the elements of x^0, all criteria under consideration have been evaluated. Let Q be the criteria values set for all feasible discrete alternatives in the space of criteria F. Let a mapping f: $x^0 \rightarrow Q$ be given.

Then the problem can be formulated as follows:

$$\min f(x) \qquad x \in x^0$$

$$x^0 = \{x_1, x_2, \ldots, x_n\} \subset R^s$$

$$f(x) = \{f^1(x), f^2(x), \ldots, f^m(x)\}$$

$$f: x^0 \rightarrow Q$$

$$Q = \{f(x_1), f(x_2), \ldots, f(x_n)\} \subset F = R^m$$

The partial pre-ordering relation in space Q is implied by the positive cone $\Lambda = R_+^m$:

$$f_1, f_2 \in Q \quad f_1 < f_2 \Longleftrightarrow f_1 \in f_2 - \Lambda$$

This means f_1 dominates f_2 in the sense of partial pre-ordering.

Element $f^* \in Q$ is nondominated in the set of feasible elements Q, if it is not dominated by any other feasible element. Let $N = N(Q) \subset Q$ denote the set of all nondominated elements in the criteria space and let $N_x = N(x^0) \subset x^0$ denote the set of the corresponding nondominated alternatives (decisions) in the decision space.

To solve this problem means to delete all the dominated alternatives – that is, alternatives for which a better one can be found in the sense of the natural partial ordering of the criteria – or to find the set N of nondominated elements and the corresponding set N_x of nondominated alternatives. Eventually, a final solution

should be found from the set of nondominated alternatives.

The Algorithm to Select the Nondominated Set of Alternatives

The algorithm to select the nondominated set of alternatives is quite simple. The method implemented in our system is of the explicit enumeration type. It is called the method of dominated approximations and is based on the following notion.

Def. 1: Set A is called a dominated approximation of N if, and only if

$$N \subset A - \Lambda$$

i.e., if for each $f_i \in N$ there exists $f_j \in A$ such that $f_i < f_j$ in the sense of partial pre-ordering induced by Λ.

Def. 2: The A_2 approximation dominates the A_1 approximation of the nondominated set N if, and only if

$$A_1 \subset A_2 + \Lambda$$

The method of dominated approximations generates a sequence of approximations A_k, k=0,1,2,...l such that

$$Q = A_0 \supset A_1 \supset \ldots \supset A_k \supset \ldots \supset A_l = N$$

given Q and Λ select $N = N(Q)$, and assuming that all criteria are to be minimized. Then the procedure of problem solving can be described as follows.

Step 0: let $A_0 = Q$, $N = \Phi$, $K = 0$

Step 1: If $A_k \setminus N = \Phi$ then stop,
else choose any index $i \in I=\{1,2, \ldots ,m\}$ and find $f^* \in Q$ such that

$$f^*_i = \min f^i$$

set $N = N \cup \{ f^* \}$ and go to step 2.

Step 2: Create the new approximation A_{k+1} by f^*

$$A_{k+1} = \{ A_{k+1} \setminus N \} \{ (f^* + \Lambda) \cap (A_k \setminus N) \} \cup N$$

set $K = K + 1$ and go to step 1.

As a result of the above procedure the nondominated set N of alternatives is found when the stopping condition $A_k \setminus N = \Phi$ is satisfied.

The Reference Point Approach

After the system eliminates, by the method mentioned above, all the dominated alternatives, the set of remaining nondominated alternatives is usually large and its elements are incomparable in the sense of natural partial ordering. To choose from among them, additional information must be obtained from the decision maker. The main problem of multicriteria optimization is how and in what form this additional information may be obtained, such that it satisfactorily reflects the decision maker's preferences, experience and other subjective factors.

There are many methods for obtaining that additional information and to then find the final or the "best" solution according to the decision maker's preference. The most common method is the weighting coefficients method, which plays a central role in the basic classical theory of multi-objective decision analysis. It represents a traditional method of multi-criteria optimization.

However, certain difficulties often arise when applying the weighting coefficients method to real-world decision processes: Decision makers usually do not know how to specify their preferences in terms of weighting coefficients. Before running a multiobjective model, some of them do not even have an idea about their weighting coefficients.

Most of them are not willing to take part in psychometric experiments in order to learn about their own preferences. Sometimes the decision maker has variable preferences as time, and the information available to him changes. The applicability of the weighting coefficients method to real world problems is severely restricted by these factors.

It is obvious that decision makers need an alternative approach for multicriteria optimization problems. Since 1980 many versions of software tools based on reference point theory have been developed at IIASA, such as DIDASS/N, DIDASS/L, MM, MZ, Micro DIDASS etc. These tools can deal with nonlinear problems, linear problems, dynamic trajectory problems, and committee decision problems. Recently many application experiments have been reported by numerous scientific papers and reports (e.g., Grauer, et al. 1982, Kaden, 1985,).

The reference point approach is based on the hypothesis that in everyday decisions individuals think rather in terms of goals and aspiration levels than in terms of weighting coefficients or maximizing utility. This hypothesis is quite close to the real-world decision-making process.

Using the reference point approach, the decision maker works with a computer interactively. There are two distinct phases in the approach:

In the first stage, the exploratory stage, the decision maker may acquire information about the range and the frequency distribution of the alternatives thus giving him an overview of the problem to be solved. The decision maker may also set some bounds for the criteria values of the alternatives set to focus his interests on a certain area.

In the second stage, the search stage, at first the decision maker is required to specify his preferences in terms of a reference point in the criteria space. The values of the criteria represented by the reference point in the criteria space are the values the decision maker wants to obtain, i.e., the goal of the decision maker, which reflects his experience and preferences.

Next, the system identifies an efficient point, which is one of the alternatives closest to the reference point. The efficient point is the "best" solution of the problem under the constraints of the model and with respect to the reference point specified by the decision maker.

If the decision maker is satisfied by this solution, he can take it as a basis for his final decision. If the decision maker is not satisfied by this solution, he may modify his goal, i.e., change the reference point or change the constraints, i.e., change the bounds he had set before, or both, or create some additional alternatives in order to obtain a new efficient point. In the case of continuous variables problems, i.e., the problems described by continuous models (linear or nonlinear programming models or dynamic control models), the reference point method is able to generate new alternatives by running the model again.

The Mathematical Description of the Approach

The approach currently implemented in the framework system is as follows: for the sake of computability, it is necessary to define an achievement scalarizing function which transforms the multiobjective optimization problem into a single objective optimization problem. After having specified his preferences in terms of a reference point, which need not be attainable, the decision maker obtains an efficient point which is the nondominated point nearest to the reference point in the sense of the scalarizing function.

In our data post-processor the Euclidean-norm scalarizing function is used. Let q be the reference point specified by the user. Then assuming that the optimization problem under consideration is a minimization problem for all criteria (for maximizing problems one may easily transform it into a minimizing problem by changing the sign of the related criteria), the following scalarizing function is minimized:

$$S(f-q) = - ||(f-q)||^2 + \rho \, ||(f-q)_+||^2$$

where $(f-q)_+$ denotes the vector with components max(0,f-q), ||.|| denotes the Euclidean norm and $\rho > 1$ is a penalty scalarizing coefficient.

The solution f^e for minimizing the scalarizing function S is an efficient point of the problem with respect to the specified reference point.

If necessary, this procedure can be repeated until the decision maker is satisfied by an efficient point.

Module Implementation

While the current package stresses multi-criteria optimization, a more statistically oriented extension or counterpart (featuring various data display options, regression and time-series analysis, clustering etc.) is proposed for the next stage of development (see below).

In the overall software system, the multi-criteria optimizer or post-processor is implemented as an independent module as well as an optional function of several other modules, notably the transportation risk-cost analysis model. The only difference is in terms of access – either from the system's master menu level, or from the appropriate level of other models. If used as a stand-alone module, the program first examines its data directory and lists all data sets by a one-line identification in a sequence depending on modification dates, i.e., the data set generated last is offered as the first choice. The user then simply points at the desired data set, which is then loaded for further analysis.

Wherever the multi-criteria optimization package is used as an integrated post-processor, this step is not necessary, as only one data set, namely the one generated with the current model, will be examined.

In case of the transportation risk-cost analysis model, this data set, one record for each feasible alternative generated, consists of:

- an alternative identification;
- an array of criteria for each feasible transportation alternative;
- additional model output for each alternative, e.g., the node-arc sequence of the path;

- an array of control and policy variables corresponding to each alternative.

All interaction with the system is menu-driven. At the top level, summary information on the set of alternatives loaded is provided (Figure 2.38).

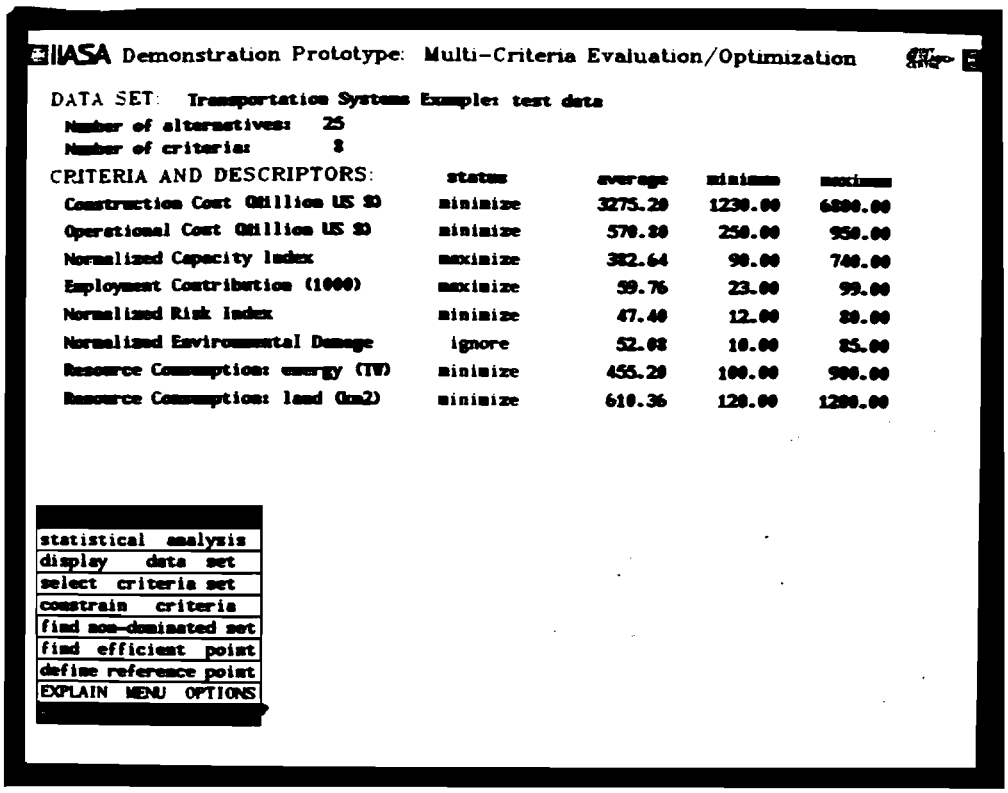


Figure 2.38: Summary of current data set

This information includes:

- the number of alternatives;
- the number of criteria considered;
- a listing of criteria, together with their status information (default settings for the three possible status indicators ***minimize***, ***maximize***, ***ignore***), and basic statistical information (average, minimum, maximum) for the individual criteria.

At that level, the menu offers the following choices:

- ***display data sets available for analysis:***
- ***select criteria:*** this allows the user to modify the status characterization, i.e., change the dimensionality of the problem by ignoring or including additional criteria from the list;

- *display data set:* this invokes the second level menu for the display options, discussed below;
- *constrain criteria:* here upper and lower bounds for the individual criteria can be defined, based on a graphical representation of the range and distribution of the criteria values (Figure 2.39); setting these constraints results in the reduction of the set of alternatives considered; the bounds are defined by dragging, with the mouse graphical input device, a vertical bar within the range of criteria values, and cutting off alternatives left or right of the bar. The system displays the current value of the constraint, and indicates how many alternatives will be deleted whenever the user sets a constraint. If the constraint setting is verified by the user, the alternatives excluded are deleted from the data set and new values for the descriptive statistics are computed.

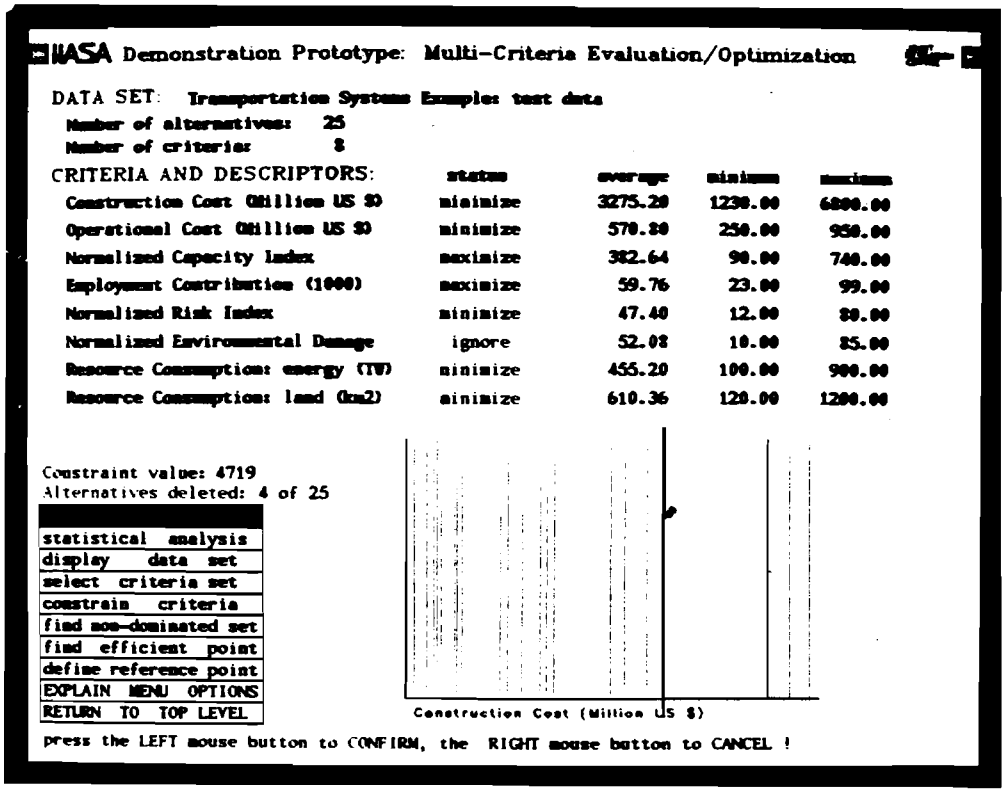


Figure 2.39: Setting constraints on criteria

- *find pareto set:* this option identifies the set of nondominated alternatives, and indicates how many nondominated alternatives have been identified;
- another feature at this, as well as any other, level in the system is an explain function that provides a more detailed explanation of the menu options currently available.

The option: *display data set* generates a new menu of options. The display options are:

- *default scattergrams:* the default scattergrams provide 2D projections of the data set, using pairwise combinations of the relevant criteria (Figure 2.40). The first three combinations are displayed in three graphics windows. If the set of nondominated alternatives has already been identified, the pareto-optimal points will be displayed in yellow and will be larger than the small, red, normal (dominated) alternatives;
- *default distributions:* this option displays the first three relevant criteria as discretized frequency distributions (Figure 2.41); again, three criteria distributions can be displayed simultaneously;
- *display selection, select x-axis, select y-axis:* these three options are used to display criteria combinations other than the default selections. Defining the x-axis only, by identifying one of the criteria lines by pointing at it, and then selecting one of the graphics windows for display, a frequency distribution will be displayed; if x and y axis are identified, a scattergram will be produced. Thus, any combination of distributions and scattergrams can be generated (Figure 2.42), allowing the user to gain some insight into the geometry and structure, e.g., dependencies of criteria, of the data set. Also, on the basis of the graphical display, it is much easier to define constraints (by returning to the previous level and invoking the appropriate menu option), if solutions are obviously clustered, i.e., distributions are multi-modal.

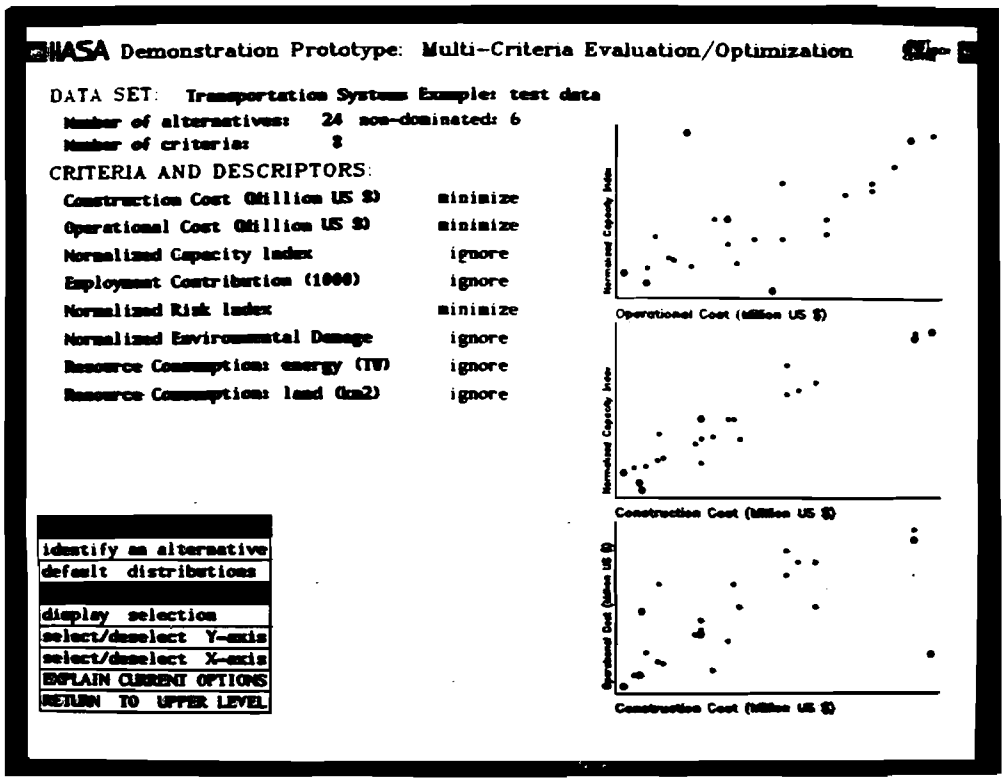


Figure 2.40: Data display: scattergrams

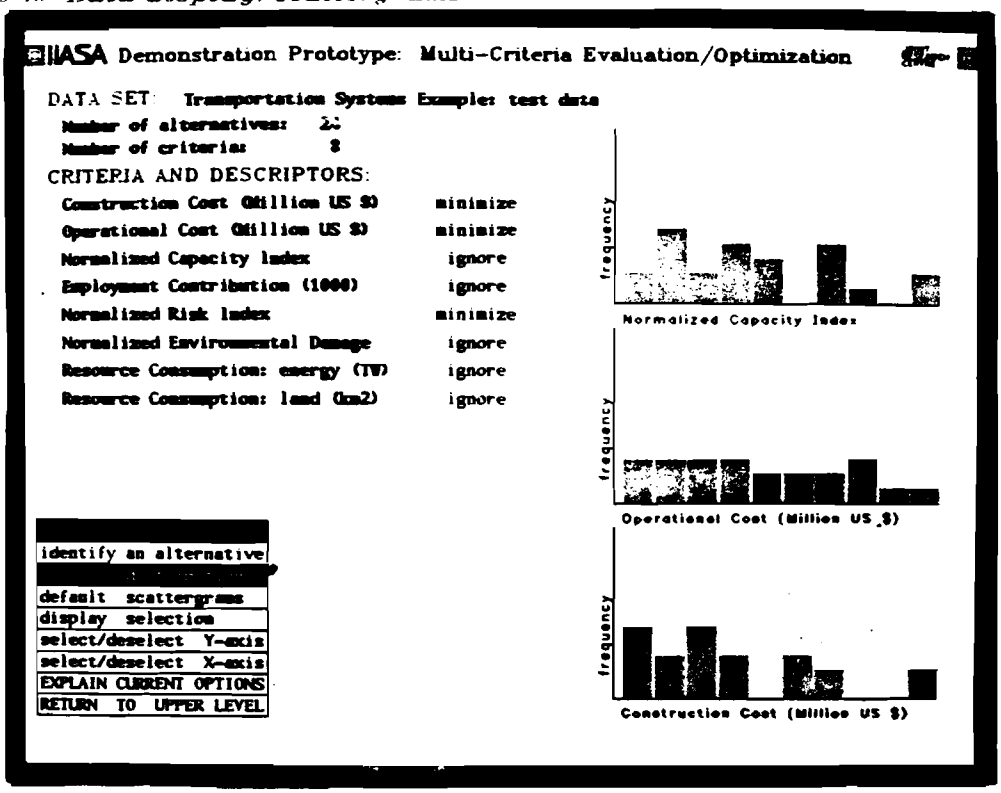


Figure 2.41: Data display: default distributions

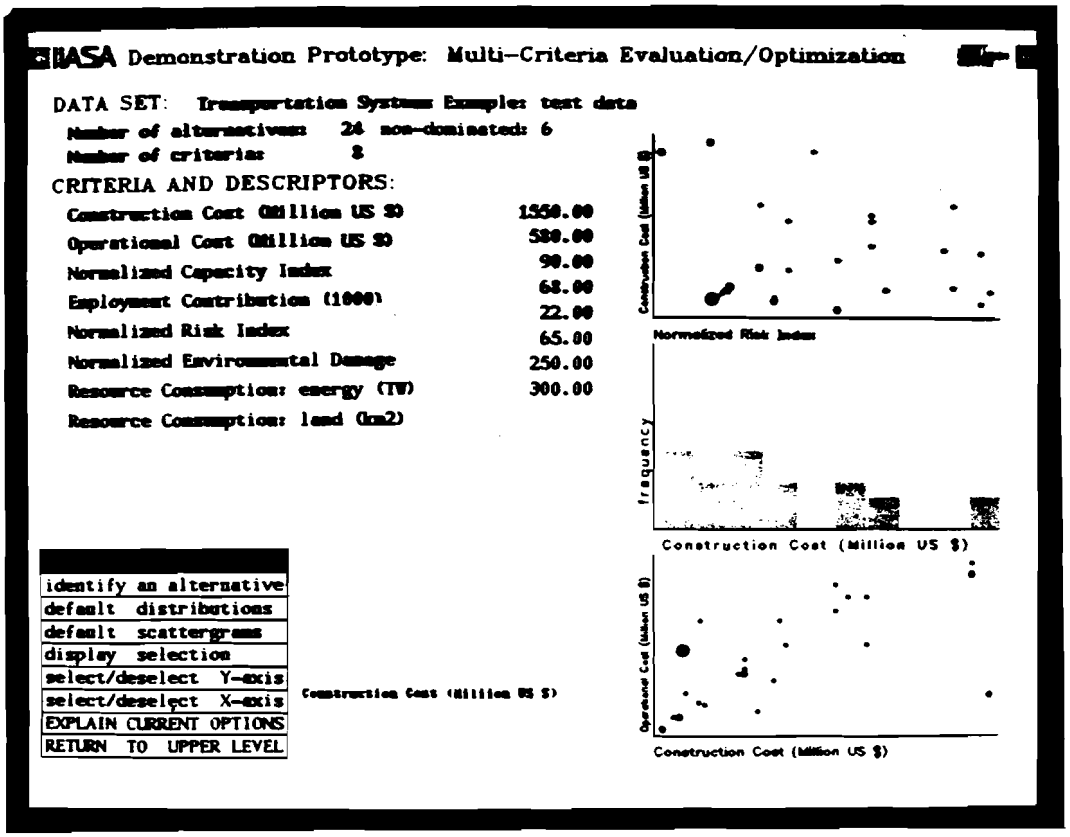


Figure 2.42: Data display: mixed scattergrams and frequency distributions with a cross-referenced alternative

- *identify alternative:* one individual alternative can be identified by pointing at one of the dots in either of the graphics windows. The dot will be marked by a large blue dot in all the scattergrams currently on display. Repeating this identification process several times, changes in the relative position of these identifiers along the individual axes support some intuitive impression on trade-offs among criteria. Parallel to marking the selected alternative on the scattergrams, numerical values for the individual criteria are displayed (Figure 2.42).

The most powerful option in this system, however, is the selection of a reference point and the resulting identification of an efficient point.

Depending on the level at which a reference point is defined, two techniques for its identification are supported, namely a numerically oriented one, considering all criteria simultaneously, and a graphically oriented one based on a sequence of pairwise trade-off specifications (Figure 2.43).

In the first case, the (extended) range for each of the criteria is displayed besides the listing of the criteria. Thus, while all criteria as well as the utopia points and the possible ranges for a reference point are in view, the user can specify the desired level (aspiration level) for each or a few of the criteria by selecting the respective criterion and then entering either a number or pointing at

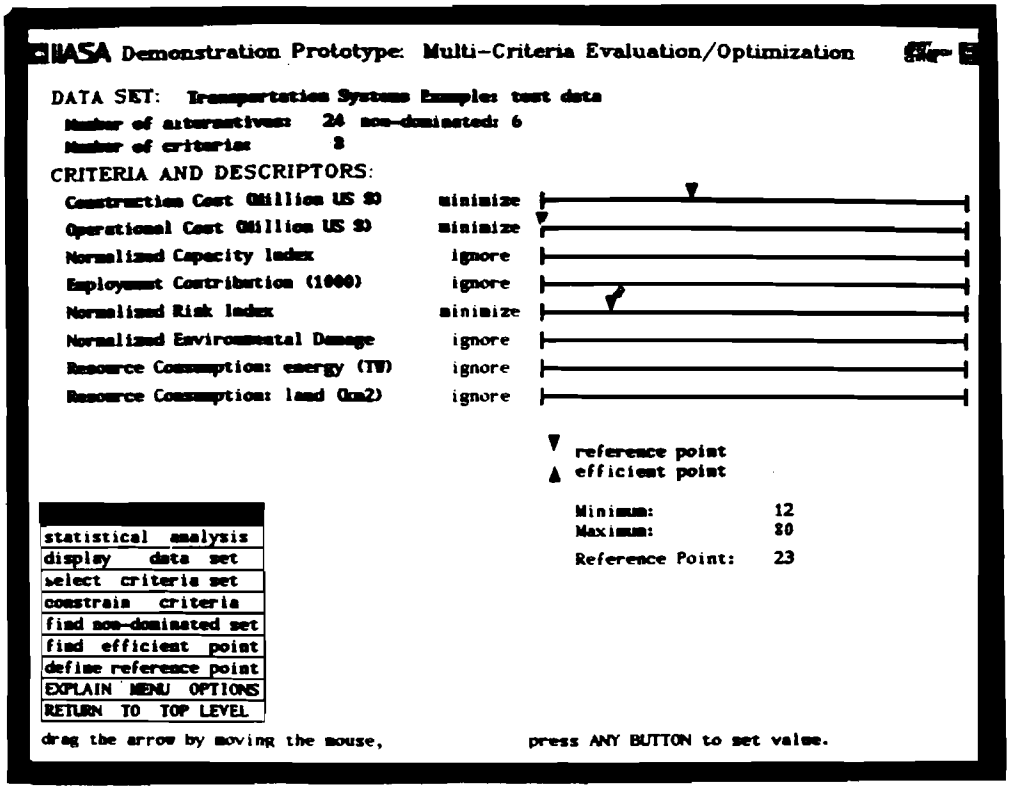


Figure 2.43: Defining a reference point with criteria list

an appropriate position within the interval displayed (Figure 2.43). For the dimensions (i.e., criteria) not explicitly specified by the user, the reference point value defaults to the utopia point's value.

In the second case, the user can have up to six scattergrams on the screen (covering up to a total of twelve criteria simultaneously). For each pairwise combination of criteria, he can then specify a reference point in this 2D projection, thus defining two dimensions at a time. Since the same dimension can be displayed in more than one scattergram, more than one value could be specified for any one dimension. Therefore, as soon as a value for a dimension that is represented more than once is set, a vertical or horizontal line, indicating this setting, is displayed in all other scattergrams with this dimension. This serves as a reminder to the user that this dimension was already defined. If the user sets another value for this dimension anyway, all previous settings are updated accordingly, since the last specification always supercedes any previous one.

Once all or a subset of the criteria dimensions deemed important by the user have been defined, this reference point will then be used to find an efficient point as the solution to the selection procedure. Several rounds of iteration, however, may be used to find a satisfying solution. With each efficient point, the user has the option of returning to the model that generated the alternative selected. There he can re-simulate the alternative, and thereby generate additional descriptive information on his choice. This may lead to yet another setting for the

reference point, another efficient point and so on.

In summary, the discrete optimizer or post-processor is a tightly coupled option of several simulation models used for scenario analysis and/or generating a larger set of alternatives to be evaluated. Providing a combination of analysis and display options, powerful decision support can be made available to a non-expert user in a very efficient and effective way. Due to the ease of use, the high degree of flexibility and responsiveness, and the immediate understanding of results based on symbolic and graphical display combined with numerical information, the system invites a more experimental style of use. Complex models, which usually produces a confounding amount of output, can thus be made available as a direct information basis for decision making.

Data Structures:

When called from the Master Menu Level, the data files used by this module are organized as simple sequential files, with a header block containing title and number of alternatives and criteria, names of criteria and default status (ignore/minimize/maximize) information, followed by one record for each alternative with the set of criteria values. These files should be generated by the respective simulation models.

Possible Extensions:

- *statistical analysis*: here statistical information on the data set other than the minima, maxima, and average values displayed by default can be generated and displayed. In particular, this includes standard deviations and median values as well as pairwise and multiple correlation coefficients, indicating relationships of indicators. Also, a cluster analysis option is foreseen, allowing a similarity ranking of alternatives and subsets of alternatives.
- *ranking by individual criteria:* here the alternatives are ranked according to the individual criteria, resulting in a table of color-coded relationships.
- *robustness and sensitivity* of the solution: Robustness can be tested at the DSS level: here the system successively increases a noise term added to the raw data, until the efficient point, as defined by the current reference point, switches to another alternative. The noise level, (in percentage) is then displayed to the user. The indication is that with an assumed error of the model output up to the level indicated, the solution would stay the same and not be affected, i.e., robust. The higher the noise level indicated, the more confidence may be placed in the selection of the alternative.

  Sensitivity analysis, on the other hand, could be performed by switching back to the original model and exploring the neighborhood of the preferred solution. Small changes in critical control variables and parameters should not result in drastically different model outcomes, that, if re-introduced into the DSS system, would be dominated and far from the efficient solution.

2.1.13 Explain Current Options

Explain Current Options provides explanations on the currently available options. Whenever these explanations are more extensive, a second level of choice can be provided, allowing selection of an individual menu item for further explanation.

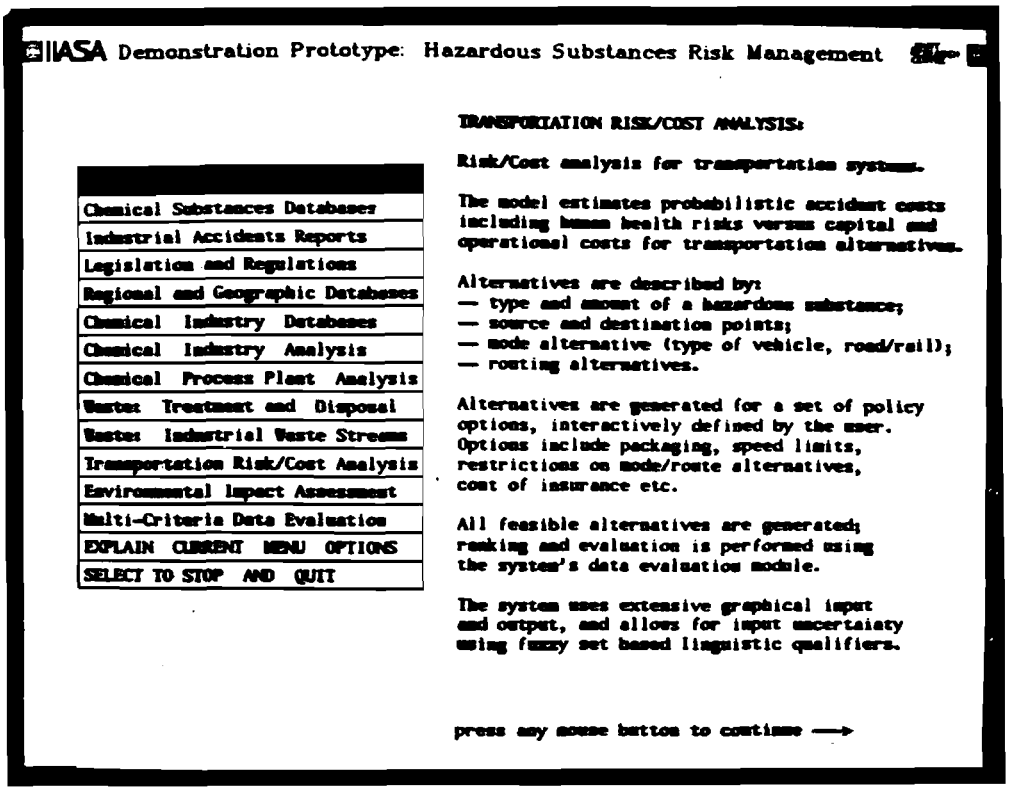


Figure 2.44: Explain page at the Master Menu Level.

Thus, all the necessary documentation can be made part of the software itself, so that cross-referencing with printed documentation is no longer necessary.

Data Structures:

The explanatory text is stored in simple sequential ASCII text files in the /data/explain directory and referenced by (option)ID-number. Storing the text in external files that can easily be modified, greatly simplifies the task of keeping relevant information current. Also, adaptations to different user groups or different languages are easily possible.

Possible Extensions:

While currently the responsibility for formatting the explanatory text is at the level of the input text file, this will be extended to include a minimal text processing system that allows for multi-page explain text, as weil as for multi-color and multi-font highlighting within the text. Processing the text according to appropriate control information in the input files will, however, slow down the

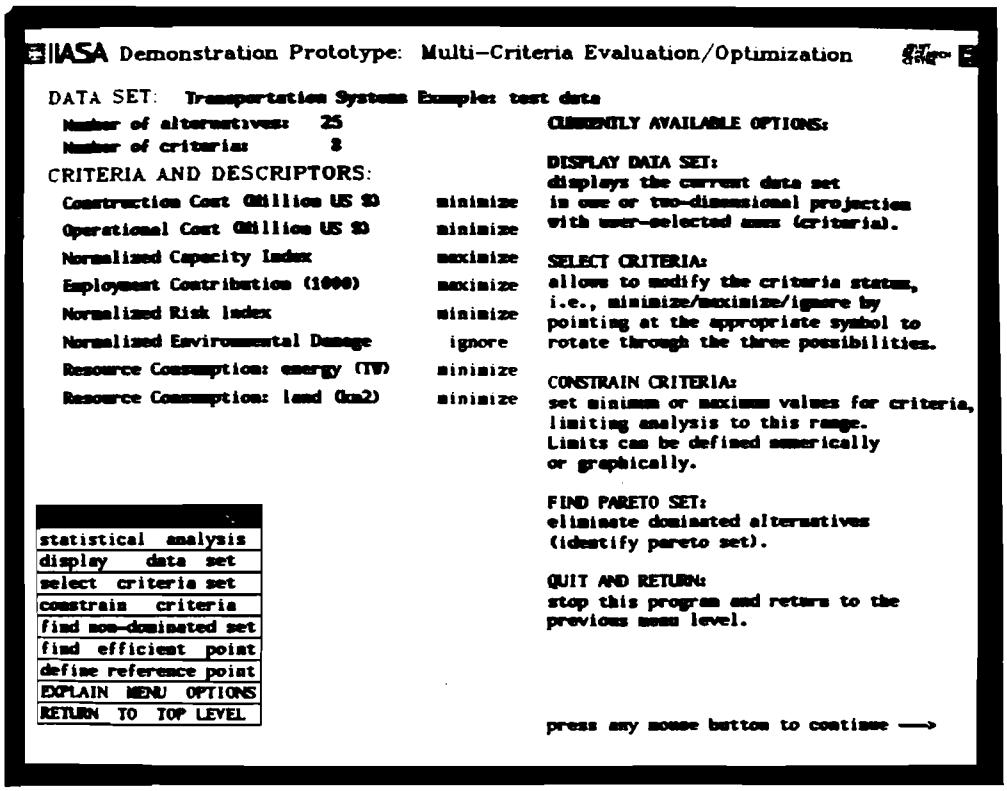


Figure 2.45: Explain page for the Decision Support Module.

speed of displaying the information. Alternatively, pre-processed text can be loaded as stored image segments. This however, will require considerably larger disk space than the current system uses.

At the next level of development, more detailed explain pages, combining text and pictures, can be used. This will require the introduction of high-capacity high-speed mass storage devices such as optical disks.

2.2 Implementation Structure: File Systems

The system is implemented on the */usr* partition of the root file system under UNIX 4.2 bsd. The top level directory contains the executable modules (currently RUN and ATM). The master modules RUN (that contains most of the currently available modules linked into one element) starts automatically when the user logs in as *demo*. RUN is started from demo's login file in /usr/IIASA. A few smaller, executable modules are resident in */usr/bin*. They are called via the shell command interpreter through the *system*() call from any C routine.

The top directory contains the following subdirectories:

- *data*, where all input data files are kept;
- *segments*, where the graphical segments to be loaded by core are located;
- *include*, where all include files required by the preprocessors of the various compilers used are found;
- *src*, where all the sources and related files are kept. *src* contains the makefile that can be used to compile, assemble, load and install the system, e.g., after changing any of the routines or include files.
  src contains one subdirectory for each functional module integrated; currently these are:
- *menu*: menu system and top level control programs;
- *maps*: data base managers and display facilities for the geographical data;
- *chem*: chemicals data base managers and display facilities;
- *acc*: accident data base manager and display programs,
- *loc*: industrial establishments and storage facilities data base manager and display programs,
- *prod*: technology and hardware data base, display programs for the chemical process level simulators/optimizers;
- *pda*: pesticide production system optimization programs;
- *wsdb*: waste streams data base manager and display programs,
- *env*: environmental master menu level,
- *gwd*: groundwater quality model,
- *river*: TOXSCREEN river water quality module,
- *atm*: LRAT, the long-range atmospheric transport model;
- *trans*: transportation risk-cost analysis model;
- *dss*: interactive decision support system, post-processor and multi-criteria optimizations routines;

2.3 Software Tools and Libraries

The major graphics software package used for this implementation is the CORE graphics package of 2D and 3D graphic primitive routines.

Within CORE, the pixrect layer is also used for more machine-dependent high-efficiency graphics operations.

Some of the modules yet to be implemented will be using the CGI (Computer Graphics Interface) package for high-efficiency dynamic graphics.

In addition to the software that is integrated into the demonstration prototype, various other programs had to be developed for e.g., data entry and visual control, data base generation; design and generation of graphical segments used by other models, etc.

A group of libraries is used, including:

- */usr/lib/libispra.a*, containing general interface routines;
- */usr/lib/libminos.a*, containing a general-purpose linear problem solver;
- */usr/lib/libdidas.a*, containing various pre- and post-processor routines for the DIDASS related elements of MM and DISCRETE, and DIDASS;
- */usr/lib/libopt.a*, containing the set of non-linear problem solvers used for OPTIMIZER.

3. PROJECT STATUS AND OUTLOOK

3.1 Components of the Demonstration Prototype: Operational Modules

For the demonstration prototype, we have chosen to include as many different modules and features as possible, at the expense, where necessary, of a complete integration and detailed finalization of individual modules. Some features of certain modules, although offered by several menus, are thus not yet supported.

At the time of publishing this report, the following elements of the system are operational and implemented on a SUN-2/160 workstation:

3.1.01 Graphical User Interface, Menus:

For the overall system, the hierarchical control program and a set of menus including an explain function (see 2.1) have been built. The current structure provides an open architecture hierarchical framework for all the layers and modules of the system as described in the design report. Individual modules are either linked with the main control program, or are implemented as stand-alone program units, activated through appropriate systems calls.

3.1.02 Data Bases:

Structures, test implementations, and integration with either interactive browsing programs, graphical display options, or operational simulation models have been completed for the following data bases:

- **Geographical data**: the basic background map of the demonstration prototype is a contour map of Europe; the contents of various data bases used in one or several of the simulation models can be viewed as interactively constructed map "overlays". They include:
  - political boundaries
  - major settlements (>100,000 inhabitants; the actual settlements data base is more than twice as large, including more than 1000 entries, which are used in the transportation network data base);
  - European highway and national roads network (the complete European highway network as well as selected national roads, connecting the above settlements and numerous auxiliary towns; this data base also provides direct input to the transportation risk/cost analysis module);
  - major industrial plant locations (concentrating on phenol and chlorine as major feedstocks or products);
  - chemical storage facilities (concentrating on phenol and chlorine);
  - major water bodies (rivers, lakes).
- **Chemical data bases**: the chemical substances data base includes a subset of the ECDIN data base. Its structure and contents are specifically geared towards the data requirements of the simulation models used. The preparation of a useful operational subset requires considerable input from the end user, in particular for compiling and processing the physical properties of selected substances, according to the substance description questionnaire developed. Detailed descriptions for a few individual substances as well as the allocation of substances to a few substance classes (organized largely by chemical

taxonomy), defined together with the end user, have been included in the data base (Fedra et al., 1986b).

An interactive query and display facility for the existing substance and substance classes structure and the subset of individual substances included is implemented and linked to the overall framework. It provides access to individual substance descriptions either from a listing of substances or from substance classes.

Several auxiliary programs to manipulate these data (e.g., transformation of a directly readable form, accessible with the systems editor, to an indexed random access form), and an interactive DBMS for data entry have been developed.

- **Industrial Waste Streams data base:** Industrial waste streams (based on a subset of the RCRA data base) are described in a similarly structured data base. Access to individual waste stream descriptions is either from a list of waste streams, through industrial origin, waste stream names (interpreted by an intelligent parser), or waste stream properties.
- **Chemical processes/unit equipment data base:** The structure of a chemical processes (unit processes, combined process technologies, unit equipment) data base has been developed. A test example describing phenol chlorination including plant hardware configuration has been completed. This example program is the graphical output component for the plant simulation modules described in section 2.1.06.
- **Industrial production sites:** A small subset of European producers (mainly related to phenol and/or chlorine production) has been compiled and integrated in a structure similar to the industrial waste stream DB.
- **Chemical storage facilities:** Chemical storage facilities provides a similar structure as the production facilities DB, concentrating, however, on major storage rather than production facilities.
- **Major industrial accidents:** A simple display program for text files structured according to Appendix VI of the Seveso directive has been developed. As an example, a short description of the Seveso accident is included.
- **Regulations and legislation:** Similar to the above accident reports, the text files accessible through this module cover selected EC directives and other relevant European legislation. As an example, the Seveso directive is included.

3.1.03 Simulation/Optimization Models:

Several simulation and/or optimization models have been prepared for integration into the demonstration prototype. They include:

- **PDA (Production-Distribution Area)**, an interactive optimization code based on DIDASS and a linear problem solver, for chemical industry structures, configured for the pesticide industry (12 processes, 13 major products) of a hypothetical region.
- **Industrial Process Simulation:** the industrial process simulation provides a simple database-driven dynamic extension of the corresponding data base. It features an animated display of the basic steps in the chemical manufacturing technology described in the data base.

As a more advanced and flexible companion product, a symbolic simulator implemented in Franz Lisp on a VAX 11/750 has been developed. It will be converted to SUN Common Lisp as soon as the required systems software is available with the necessary operating systems upgrade.

- **LRAT (Long-Range Atmospheric Transport)**, a Lagrangian trajectory model based on Eliasson (1978), using a subset of the EMAP European synoptic wind data has been completed and is implemented on a European scale with complete interactive problem definition, context driven auto-startup feature, and extended (animated) graphical display for the simulation (2.1.05.4).
- **RIVER**, extracted from the generic screening level EPA model system TOXSCREEN, simulates pollutant dispersion in an arbitrary river segment. The model features extensive interactive input modification based on predefined default values as well as animated graphical display.
- **FEFLOW**, a 2D finite element groundwater contamination model, configured for a set of 9 generic problem situations. Problem descriptions can be modified interactively; the model generates animated graphical output of flow fields and time-varying concentrations in the observation or pumped wells defined in a given problem.
- **HASTM (Hazardous Substances Transportation Model)**, a transportation risk/cost analysis (final operational version of the basic software elements to be supplied by the Ludwig Boltzmann Institute, Vienna). The graphics-based interface developed for the model allows interactive definition of a transportation problem. The model is to be used in conjunction with the discrete optimization system described below.
- **Multi-criteria Discrete Optimization** (Data-post processor, DIDASS based), which is available as a data post-processor accessible from the main menu level as well as from selected simulation models. The first stage of the module is completed with interactive problem definition and extended graphical display options.

4. REFERENCES AND SELECTED BIBLIOGRAPHY

AICE (1973) Dow's Safety and Loss Prevention Guide, Hazard Classification and Protection. American Institute of Chemical Engineers.

Barret, K. and Roberts, E. (1976) Survey of information sources with an annotated bibliography on organic emissions from the synthetic organic chemical manufacturing industry. MTR-7377, The MITRE Corporation.

Bonazuntas M. and Wagner, J. (1981) SESOIL:A Seasonal Soil Compartment Model. Draft Report, Arthur D. Little, Inc., Cambridge, Mass. Prepared for the USEPA, Office of Toxic Substances.

Brooks, N.H. (1983) Diffusion of Sewage Effluent in an Ocean Current. In proceedings of the First International Conference on Waste Disposal in the Marine Environment, Pergamon Press. pp.264-7.

Burns, L.A., Cline, D.M. and Lassiter, R.R. (1981) Exposure Analysis Modeling Systems (EXAMS): User Manual and Systems Documentation. USEPA Environmental Research Laboratory, Athens, GA. 460p.

CMA (1983) The CMA Hazardous Waste Survey for 1983. Final Report. Chemical Manufacturers Association, Washington, D.C. 35p.

Crowe, C.M., Hamielec, A.E., Hoffman, T.W., Johnson, A.I., Woods, D.R., and Shannon, P.T. (1971) Chemical Plant Simulation. Prentice Hall, Englewood Cliffs. 368p.

Diersch, H.-J. (1980) Finite-Element-Programmsystem FEFLOW. Program description. Institut für Mechanik der AdW der DDR, Berlin.

Diersch, H.-J. and Kaden, S. (1984) Contaminant Plume Migration in an Aquifer: Finite Element Modeling for the Analysis of Remediation Strategies: A Case Study. CP-84-11. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 88p.

Dillon, R.T., Lantz, R.B. and Pahwa, S.B. (1978) Risk Methodology for Geologic Disposal of Radioactive Waste: The Sandia Waste Isolation Flow and Transport (SWIFT) Model. SAND78-1267.

Dobrowolski, G., Kopytowski, J., Lewandowski, A. and Zebrowski, M. (1982) Generating Efficient Alternatives for Development in the Chemical Industry. CP-82-54. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 32p.

Dobrowolski, G., Kopytowski, J., Wojtania, J., and Zebrowski, M. (1984) Alternative Routes from Fossil Resources to Chemical Feedstocks. RR-84-19. International Institute for Applied Systems Analysis, IIASA, A-2361 Laxenburg, Austria. 53p.

Eliassen, A. (1978) The OECD study of long range transport of air pollutants: long range transport modelling. Atmos. Environ., 12, 479.

FAI (1984) System Analysis for Integrated Relief Evaluation (SAFIRE) User's Manual. FAI/84-5, Vol 1. Final Report. 133p.

Fedra, K. (1985) Advanced Decision-oriented Software for the Management of Hazardous Substances. Part I: Structure and Design. CP-85-18. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria.

Fedra, K., Schneider, J., and Ditterich, K. (1985) Advanced Decision-oriented Software for the Management of Hazardous Substances. Part II: Model

Inventory and Information System. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. Unpublished.

Fedra, K. and Otway, H. (1986a) Advanced Decision-oriented Software for the Management of Hazardous Substances. Part III. Decision Support and Expert Systems: The Whats and Wherefores. CP-86-xx. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria and EUR-xxx, Ispra. Forthcoming.

Fedra, K., Mutzbauer, G., Weigkricht, E. and Winkelbauer, L. (1986b) Advanced Decision-oriented Software for the Management of Hazardous Substances. Part III: Substances Data and Knowledge Base. CP-86-xx. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria and EUR-xxx, Ispra. Forthcoming.

Fedra, K., Diersch, H.-J. and Kaden, S. (1986c) Interactive Modeling of Groundwater Contamination Based on the Finite Element Simulator FEFLOW. CP-86-xx. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. Forthcoming.

Fedra, K. and Bartnicki, J. (1986d) Decision-oriented Transport Modeling with an Intelligent User Interface. CP-86-xx. International Institute for Applied Systems Analysis. A-2361 Laxenburg, Austria. Forthcoming.

Finley, N.C. and Reeves, M. (1981) SWIFT Self-Teaching Curriculum: Illustrative Problems to Supplement the User's Manual from the Sandia Waste-Isolation Flow and Transport Model (SWIFT). SAND81-0410, NUREG/CR-1968

Golden, J., Quellette, R.P., Saari, S. and Cheremisinoff, P.N. (1980) Environmental Impact Data Book. Ann Arbor Science, Ann Arbor, Michigan. 864p.

Goldfarb, A.S., Goldgraben, G.R., Herrick, E.C., Quellette, R.P. and Cheremisinoff, P.N. (1981) Organic Chemicals Manufacturing Hazards. Ann Arbor Science, Ann Arbor, MI. 430p.

Grauer, M., Gruhn, G. and Pollmer, L. (1979) Optimization of a Complex Plant by a GRG Algorithm. Computers and Chemical Engineering, Vol.3, pp.597-602. Pergamon Press Ltd., U.K.

Grauer, M. and Lewandowski, A. (1982) The Reference Point Approach - Methods of Efficient Implementation. WP-82-26. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 25p.

Grauer, M. and Kaden, S. (1984) A Non-linear Dynamic Interactive Decision Analysis and Support System (DIDASS/N). Users Guide. WP-84-23. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 55p.

Grauer, M., Lewandowski, A. and Wierzbicki, A. (1984) Multiple-Objective Decision Analysis Applied to Chemical Engineering. RR-84-15. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. 40p.

Hay, A. (1982) The Chemical Scythe. Lessons of 2,4,5-T and Dioxin. Plenum Press, New York. 264p.

Herrick, E.C., King, J.A., Quellette, R.P. and Cheremisinoff, P.N. (1979) Unit Process Guide to Organic Chemical Industries. Unit Processes Series, Organic Chemical Industries, Volume 1. Ann Arbor Science, Michigan. 146p.

Hetrick, D.M. and McDowell-Boyer, L.M. (1978) TOX-SCREEN: A multimedia screening level model for assessing the potential fate of toxic chemicals released to environmental media. In Proceedings, 1983 Summer Computer Simulation

Conf., Simulation Councils, Inc., La Jolla, CA., pp.464-469.

Hetrick, D.M. and McDowell-Boyer, L.M. (1984) User's Manual for TOX-SCREEN: A Multimedia Screening-level Program for Assessing the Potential Fate of Chemicals Released to the Environment. ORNL-6041, Oak Ridge National Laboratory and EPA-560/5-83-024, USEPA, Washington, D.C.

IAEA (1983) INTERTRAN: A System for Assessing the Impact from Transporting Radioactive Material. A guide prepared by A-M Ericsson and M. Elert for the International Atomic Energy Agency. IAEA-TECDOC-287.

ICF (1984a) The RCRA Risk-Cost Analysis Model. Phase III Report. Submitted to the Office of Solid Waste, Economic Analysis Branch, U.S. Environmental Protection Agency. March 1, 1984. ICF Incorporated, International Square, 1850 K Street, Northwest, Washington, D.C.

ICF (1984b) The RCRA Risk-Cost Analysis Model. Phase III Report. Appendices. Submitted to the Office of Solid Waste, Economic Analysis Branch, U.S. Environmental Protection Agency. March 1, 1984. ICF Incorporated, International Square, 1850 K Street, Northwest, Washington, D.C.

INFUCHS (1985) Informationssystem fuer Umweltchemikalien, Chemieanlagen und Stoerfaelle (Information System for Environmental Chemicals, Chemical Plants and Accidents). Umweltbundesamt, Berlin. (In German)

IRPTC (1978) International Register of Potentially Toxic Chemicals: Its Present State of Development and Future Plans. Ambio 7:5/6, 275-277.

Jones, G.P. and Barrow, R.N. (1973) Risk Analysis in Hazardous Materials Transportation: Vol.1. University of Southern California.

Kaden, S. (1983) Analysis of regional water policies in open pit mining areas -- a multicriteria approach. In: A.Wierzbicki and M.Grauer, [eds.] Interactive Decision Analysis and Interpretative Computer Intelligence. Springer Verlag. pp.218-226

Kleindorfer, P.R. and Vetschera, R. (1985) Risk-Cost Analysis Model for the Transportation of Hazardous Substances. Final Report Vol. I. October. Ludwig Boltzmann Institute, Vienna.

Kleindorfer, P.R. and Vetschera, R. (1985) Risk-Cost Analysis Model for the Transportation of Hazardous Substances. Final Report. Vol. II. October. Ludwig Boltzmann Institute, Vienna.

Kreglewski, T. and Lewandowski, A. (1983) MM-MINOS--An Integrated Interactive Decision Support System. Collaborative Paper, CP-83-63, International Institute for Applied Systems Analysis, Laxenburg, Austria.

Majchrzak, J. (1984). Package DISCRETE for multicriteria optimization and decision making problems with discrete alternatives. IIASA conference on Plural Rationality and Interactive Decision Processes, 16-26 August 1984, Sopron, Hungary.

Majchrzak, J. (1985) DISCRETE - An interactive package for discrete multicriteria decision making problems. Report, June 1985. Systems Research Institute, Warsaw, Poland.

McAllister, D.M. (1980) Evaluation in Environmental Planning. Assessing Environmental, Social, Economic, and Political Trade-offs. The MIT Press, Cambridge, Massachusetts. 307p.

McDowell-Boyer, L.M. and Hetrick, D.M. (1982) A Multimedia Screening-Level Model

for Assessing the Potential Fate of Chemicals Released to the Environment, ORNL/TM-8334. Oak Ridge National Laboratory.

NFPA (1977) Fire Hazard Properties of Flammable Liquids, Gases, and Volatile Solids. NFPA 325M. National Fire Protection Association.

Otway, H. and M. Peltu [eds.] (1985) Regulating Industrial Risks. Butterworths, London and Boston. In press.

Pack, D.H., Ferber, G.J., Heffter, J.L., Telegadas, K., Agnell, J.K., Hoeker, W.H. and Machta, L. (1978) Meteorology of long-range transport. Atmos. Environ., 12, 425p.

Pasquill, F. (1961) The estimation of the dispersion of windborne material. Meteorol. Mag. 90: 33.

Pasquill, F. and Smith, F.B. (1983) Atmospheric Diffusion: Study of the Dispersion of Windborne Material from Industrial and Other Sources. Ellis Horwood. 480p.

Reeves, M. and Cranwell, R.M. (1981) User's Manual for the Sandia Waste-Isolation Flow and Transport Model (SWIFT). Release 4.81. SAND81-2516, NUREG/CR-2324.

Rhoades, R.E. (1978) As Assessment of the Risk of Transporting Gasoline by Truck. Battelle-Pacific Northwest Laboratories, PNL-2133, UC-71.

Salas, J.D., Delleur, J.W., Yevjevich, V. and Lane, W.L. (1980) Applied Modeling of Hydrologic Time Series. Water Resources Publications, Littleton, Colorado.

Sax, N.I. (1975) Dangerous Properties of Industrial Materials. Van Nostrand Reinhold Co., New York.

Simmons, J.A. (1983) The Risk of Catastrophic Spills of Toxic Chemicals. SAI Services, McLean, Virginia, 1973, prepared for the University of California, LA.

Simpson, E.A. (1982) The protection of water quality within the European Economic Community. In: R.Perry [ed.]: Effects of Waste Disposal on Groundwater and Surface Water (Proceedings of the Exeter Symposium, July 1982). IAHS Publ.no. 139, pp.3-11

Smith, J.H., Mabey, W.R., Bohonos, N., Holt, B.R., Lee, S.S., Chow, T.W., Bomberger, D.C. and Mill, T. (1977) Environmental Pathways of Selected Chemicals in Freshwater Systems. EPA 600/7-77-113, U.S. Environmental Protection Agency.

Technica (1984) The SAFETI Package. Computer-Based System for Risk Analysis of Process Plant. Vol. I: Overall Description. Technica Ltd., Tavistock Sq., London.

Technica (1984) The SAFETI Package. Computer-Based System for Risk Analysis of Process Plant Using a Simplified Classical Method. Vol.II: Theory. Technica Ltd., Tavistock Sq., London.

Technica (1984) The SAFETI Package. Computer-Based System for Risk Analysis of Process Plant Using a Simplified Classical Method Vol. IV: Test Cases. Technica Ltd., Tavistock Sq., London.

Technica (1984) The SAFETI Package. Computer-Based System for Risk Analysis of Process Plant. Appendices I-IV. Technica Ltd., Tavistock Sq., London.

UNEP (1984a) The International Register of Potentially Toxic Chemicals (IRPTC) of UNEP. Its Origins and Activities. UNEP/IRPTC, Geneva.

UNEP (1984b) International Register of Potentially Toxic Chemicals (IRPTC) Legal File. Vol 1 & 2. UN, Geneva.

USEPA (1979) Industrial Source Complex (ICS) Model User's Guide, Vol. I,II EPA Report No. EPA-450/4-79-030/31. U.S. Environmental Protection Agency, Research Triangle Park, North Carolina.

USEPA (1980b) Hazardous Waste and Consolidated Permit Regulations. Federal Register, May 1980, Vol. 45, No.98. Book 2.

Weigkricht, E. and Winkelbauer, L. (1986) Knowledge-based Systems: Overview and Selected Examples. WP-86-xx. International Institute for Applied Systems Analysis, A-2361 Laxenburg, Austria. Forthcoming.

Wierzbicki, A. (1983) A Mathematical Basis for Satisficing Decision Making. Mathematical Modeling USA 3:391-405.

Winkelbauer, L. (1985) Advanced Decision-oriented Software for the Management of Hazardous Substances. Part IV: Industrial Production Systems--Process Level. Collaborative Paper CP-85-xx. International Institute for Applied Systems Analysis, Laxenburg, Austria. Forthcoming.

Wood, E.F., Ferrara, R.A., Gray, W.G., and Pinder, G.F. (1984) Groundwater Contamination from Hazardous Wastes. Prentice Hall, Englewood Cliffs.

Zach, R. (1978) TERMOD II: An interactive code for analyzing intake of radionuclides by man through terrestrial pathways. Report AECL-6306, Environmental Research Branch, Atomic Energy of Canada Limited, Whiteshell Nuclear Research Establishment, Pinawa, Manitoba. 34p.

Zanelli, S., Bello, G.C. and Frattini, B. (1984) Classification Scheme for Process Plants. Intermediate Progress Report, November 1984. Joint Research Centre, Ispra.

Zebrowski, M., Rys, T., Skocz, M., and Ziembla, W. (1985) Modeling the Chemical Industry for Management of Technological Hazards and Wastes. Academy of Mining and Metallurgy, Cracow and International Institute for Applied Systems Analysis, Laxenburg, Austria. Unpublished.

Zhao, C., Winkelbauer, L. and Fedra, K. (1985) Advanced Decision-oriented Software for the Management of Hazardous Substances. Part VI: The Interactive Decision-Support Module. CP-85-50. International Institute for Applied Systems Analysis, Laxenburg, A-2361 Austria.

Zhao, C., Winkelbauer, L. and Fedra, K. (1986) An Interactive Graphics-based Data Post Processor: An Eclectic Approach to Decision Support. Collaborative Paper, CP-86-xx. International Institute for Applied Systems Analysis, Laxenburg, Austria. Forthcoming.

APPENDIX:

A1. SUMMARY PROJECT DESCRIPTION:

ADVANCED DECISION-ORIENTED SOFTWARE
FOR THE MANAGEMENT OF HAZARDOUS SUBSTANCES

A1.1. Background

Many industrial products and residuals such as hazardous and toxic substances are harmful to the basic life support system of the environment. In order to ensure a sustainable use of the biosphere for present and future generations, it is imperative that these substances are managed in a ***safe and systematic*** manner. The aim of this project is to provide software tools which can be used by those engaged in the management of the environment, industrial production, products, and waste streams, and hazardous substances and wastes in particular.

A1.2. Project Objectives

The objective of the project is to design, develop, and implement in a demonstration prototype version an ***integrated set of software tools***, building on existing models and computer-assisted procedures. This set of tools is designed for non-technical users. Its primary purpose is to provide easy access and allow efficient use of methods of analysis and information management which are normally restricted to a small group of technical experts. The use of advanced information and data processing technology should allow a more comprehensive and interdisciplinary view of the management of hazardous substances and industrial risk. Easy access and use, based on modern computer technology, software engineering, and concepts of Artificial Intelligence (AI) now permit a substantial increase in the group of potential users of advanced systems analysis methodology and thus provide a powerful tool in the hand of planners, managers, policy and decision makers and their technical staff.

To facilitate the access to complex computer models for the casual user, and for more experimental and explorative use, it also appears necessary to build much of the accumulated knowledge of the subject areas into the user interface for the models. Thus, the interface will have to incorporate a knowledge-based expert system that is capable of assisting any non-expert user to select, set up, run, and interpret specialized software. By providing a coherent user interface, the interactions between different models, their data bases, and auxiliary software for display and analysis become transparent to the user, and a more experimental, educational style of computer use can be supported. This greatly facilitates the design and evaluation of alternative policies and strategies for the management of industrial risk.

The system design combines several methods of applied systems analysis and operations research, planning and policy sciences, and artificial intelligence into one fully integrated software system (Figure A1.1). The basic idea

is to provide direct and easy access to these largely formal and complex methods for a broad group of users.

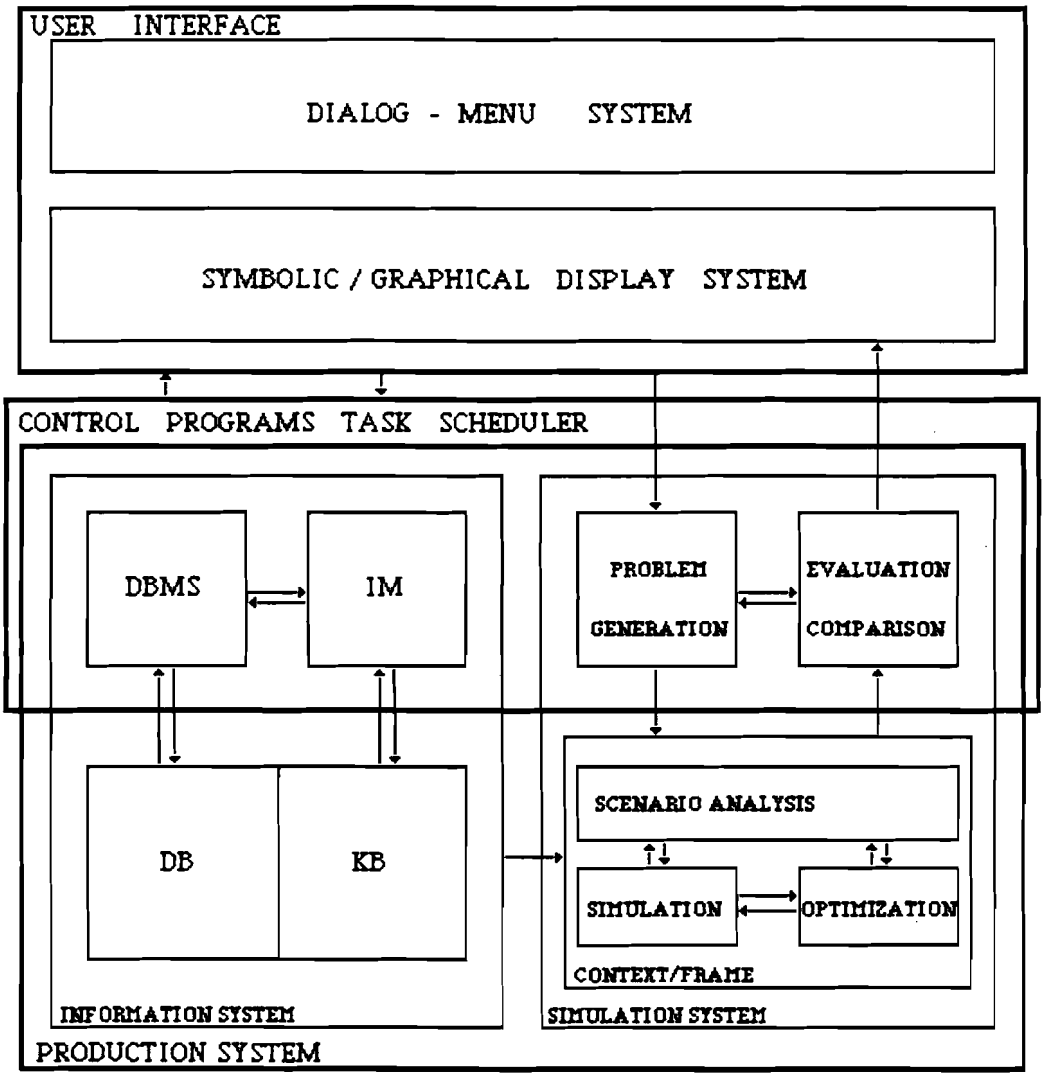


Figure A1.1: Elements of the integrated software system

Conceptually, the main elements of the system are:

- **an Intelligent User Interface,** which provides easy access to the system. This interface must be attractive, easy to understand and use, error-correcting and self-teaching, and provide the translation between natural language and human style of thinking to the machine level and back. This interface must also provide a largely menu-driven conversational guide to the system's usage (dialog - menu system), and a number of display and report generation styles, including color graphics and

linguistic interpretation of numerical data (symbolic/graphical display system);

- **an Information System,** which includes the system's Knowledge and Data Bases (KB, DB) as well as the Inference Machine and Data Base Management Systems (IM, DBMS), which not only summarize application- and implementation-specific information, but also contain the most important and useful domain-specific knowledge. They also provide the information necessary to infer the required input data to run the models of the system and interpret their output. The Inference and Data Base Management Systems (which are at the same time part of the Control Programs and Task Scheduler level) allow a context- and application-oriented use of the knowledge base. These systems should not only enable a wide range of questions to be answered and find the inputs and parameters necessary for the models, but must also be able to explain how certain conclusions were arrived at. For a given application, the data base systems must also perform the more trivial tasks of storing and organizing any interim or final results for display and interpretation, comparison, and evaluation;
- **the Simulation System,** which is part of the Production System and consists of a set of models (simulation, optimization), which describe individual processes that are elements of a problem situation, perform risk and sensitivity analyses on the relationship between control and management options and criteria for evaluation, or optimize plans and policies in terms of their control variables, given information about the user's goals and preferences, according to some specified model of the system's workings and rules for evaluation.

These elements are transparently linked and integrated. Access to this system of models is through a conversational, menu-oriented user interface, which employs natural language and symbolic, graphical formats as much as possible. The systems must be error-correcting and self-teaching, and provide not only a low-cost entry for the casual user, but also have the potential to be custom configured for day-to-day use by users of growing expertise.