

The Scenarist™ Automated Scenario Generation System

8 January 1994

J. George Caldwell, PhD

Permanent Address:
4570 N.Cerritos Drive
Tucson, AZ 85745
(602)743-0864

Present Address (until fall, 1994):
PO Box 31238
Lilongwe 3, Malaŵi, Africa
(265)733-095

Contents

- I. Introduction and Summary 1
- II. Applications that Use Scenarios 2
- III. Traditional Methodology for Constructing Scenarios 4
- IV. Shortcomings of Previous Scenario-Generation Methodology 4
- V. The Scenarist: A New Tool for Automated Scenario Generation 6
- VI. How to Use the Scenarist 7
- VII. Scenarist Applications 10
- VIII. Sample Output 11
- IX. Scenarist Features and Benefits 12
- X. Scenarist Limitations and Future Development Plans 13
- XI. Acknowledgements 15
- Reference 15
- About the Author..... 16
- Appendix. Detailed Description of Scenarist Capabilities and Functions
..... 21
 - I. Scenarist Data Setup..... 21
 - II. Scenarist Functions: What the Scenarist Does 28

The Scenarist™ Automated Scenario Generation System
J. George Caldwell, PhD

I. Introduction and Summary

A military scenario is a description of various aspects of a conflict or preconflict situation. At one extreme, it may be a qualitative description of the social, economic, and political events leading up to conflict. At another extreme, it may be a detailed quantitative description of the locations, statuses, activities, and plans of military units and equipment just prior to or at some point in time during a battle.

Military scenarios are of interest to military scientists and planners because military analysis, planning, and training are not done in a vacuum, but in a situational context. They take into account many factors, such as tactical doctrine; political goals and constraints; time and resource availability; the geographic terrain on which units are deployed and battle may occur; the capabilities of systems and equipment; and the composition, status, activities, and intentions of friendly and enemy troops. The scenario specifies factors that are important to take into account in an application.

For many applications, the amount of time and effort required to construct a scenario is substantial. A scenario may involve the positioning of a large number of items, and it becomes necessary to examine a large number of relationships among these items and the environment. In the past, although scenario generation has certainly made significant use of automated data processing, the basic process of determining unit and equipment positions often involves a considerable amount of manual effort.

This article describes a new automated tool -- the Scenarist™ -- that has been developed to assist the construction of military scenarios. The

Scenarist is a graphics workstation-based system for generating tactical military scenarios. It uses the technology of knowledge-based systems ("expert" systems) to construct a placement of units and equipment on the battlefield in accordance with rules of tactical doctrine, taking into account geographic terrain features, the mission of friendly forces, and the nature of the enemy threat.

II. Applications that Use Scenarios

There are a number of military science applications areas in which scenarios are needed and used. These areas include force planning; research, development, test and evaluation; operational planning and evaluation; and training and education. These areas are defined as follows:

- o Force planning includes force structure analysis (determination of the organizational form and types of equipment associated with land, sea, and air combat units of a specified size, such as a battalion, carrier task group, or tactical air wing); force level analysis (determination of the number of combat units of the same type to include in a larger force assigned to a military purpose); and force mix analysis (determination of the mix of land, sea, and air units that constitute a force assigned to a military purpose).

- o Research, development, test and evaluation is concerned with concept development, system design and development, system test and evaluation (development test and evaluation and operational test and evaluation).

- o Operational planning and evaluation includes analysis of operational concepts, tactics and doctrine.

o Training and education includes training of military commanders under realistic conditions and education of military science students in warfare concepts and analysis and gaming.

The factors included in a scenario and the amount of detail depend on the application. In general, the positions and activities of the major (high-echelon) units are always specified. What details are included depends on the objectives of the application and on the resources available. For strategic planning, a useful scenario may specify simply the compositions and locations of major units. For a developmental test of a communication system, a scenario may include specification of the locations, activities, and characteristics of thousands of radio transmitters and receivers.

There is a large variety of military applications that use scenarios. What information is needed about the military situation varies widely over these applications, and the characteristics and levels of detail of the scenarios used in these applications vary correspondingly. For this reason, the term "scenario" encompasses a wide variety of situational descriptions. In this article we are concerned with scenarios that are descriptions of the initial positions, activities, and intended movements of military forces from corps echelon down to individual units of equipment. This type of scenario is often referred to as a "tactical scenario," or a "tactical deployment," or a "force laydown."

Although scenarios are used in many applications, their use is particularly important and problematical in test and evaluation applications. In order for the scope of a test or evaluation to be broad, it is necessary to conduct the test or evaluation using a number of scenarios covering a wide range of realistic situations. Previous methods for generating a variety of scenarios have not proved satisfactory for a variety of reasons. The Scenarist overcomes many of the problems encountered in scenario generation.

III. Traditional Methodology for Constructing Scenarios

While a number of automated procedures have been employed in the construction of scenarios, the traditional methodology is essentially manual. There is no well-established general theoretical conceptual framework for scenario description, classification, or construction. Current means of scenario generation are largely ad hoc, heuristic, and labor-intensive. Traditional methods typically involve a subjective assessment of what factors (variables, level of detail) should be included in a scenario in order to promote achievement of the application objectives, and the judgmental specification of the locations, activities, and statuses of the units or equipment.

For large-scale evaluations of systems in a combined-arms context, the principal means of developing scenarios is manual. The US Army Training and Doctrine Command (TRADOC) Analysis Center (TRAC) located at Fort Leavenworth works with 21 Army proponent agencies to develop what is called a Standard Scenario (previously a "SCORES" scenario). The process is costly and time consuming: up to two years may be required to develop a Standard Scenario.

IV. Shortcomings of Previous Scenario-Generation Methodology

From a test and evaluation viewpoint, the existing methodology for generating scenarios has several shortcomings. These include the following:

- o Expensive and Time-Consuming. The manual process for generating scenarios is costly and time consuming.
- o Limited Scope. Because of the time and effort involved both to specify the general features of a scenario and to develop a corresponding detailed tactical deployment, few scenarios are

generated for input to tactical combat or other military system evaluation models. The result is that the evaluative scope of a system evaluation based on a single scenario is restricted. It becomes difficult to estimate what the system performance will be in a wide range of circumstances.

o Ill-Defined Basis for Inference. The typical process for developing a scenario is to use human experts (military analysts, tacticians). A problem that arises with this approach is that, because no well-defined mathematical or probabilistic framework underlies the scenario development, the results obtained using such scenarios for evaluation are conditional on the scenario actually used, and the methods of mathematics or statistics cannot be used to extrapolate the results to a broader population of scenarios. In the absence of a solid mathematical theory of scenarios, the relationship of the generated scenario to a well-defined population of scenarios is not known. In general, there is no general-purpose automated methodology currently available for generating probability samples of scenarios.

o Lack of Reproducibility. With manual methods, two different teams of experts will likely generate different scenarios, even given the same requirements for the scenario. Such variation may be similar to that occurring in real battlefield deployments and may even be advantageous in some applications, especially training. The presence of this type of uncontrolled variation may be problematic in test and evaluation applications, however, where systematic procedures, controlled variation, and reproducibility are important.

o Difficult to Validate. Manually generated scenarios are difficult to validate. Perhaps the best validation of a manually generated scenario is the reputations of the experts who constructed the scenario. The problem that arises is that it is difficult to validate

the generated scenario ex post, that is, by looking at the scenario after the fact. Ex post validation depends largely on verification that all of the many interrelationships among the entities of the scenario are reasonable. If modifications are made to a scenario, it is necessary for experts to reexamine the entire scenario again, to validate the modified scenario.

o Nonparametric and Non-Model-Based Approaches. Scenarios are often specified in detail (by specifying the location and activity of every force element (unit or equipment)) in the absence of a mathematical model of tactics or force deployment. Borrowing from statistical terminology, this type of description may be referred to as a "nonparametric" representation. A problem with nonparametric representations is that they are often awkward to work with. The nonparametric specification is not an efficient, model-based framework for describing either the scenario or the scenario generation process (in terms of a manageable number of entities, variables, relationships, or procedures). Without a model-based framework, efficient, convenient, easy-to-use means for rapidly generating, modifying, or even classifying or describing scenarios are not available.

It was in recognition of the drawbacks of the existing methodology for scenario generation -- costly, time consuming, limited scope, ill-defined basis for inference, difficulty in validation, nonparametric representation -- that we proposed to attempt the development of automated means for scenario generation. There are a variety of mathematical and statistical approaches that may be used to generate scenarios. The tool described in this article is based on the technology of knowledge-based systems.

V. The Scenarist: A New Tool for Automated Scenario Generation

In consideration of the shortcomings of previous scenario-generation techniques, a multi-year research and development effort was recently undertaken to develop improved methodology and tools for scenario generation. The first product of this effort -- a prototype computer-based automated scenario-generation system named the Scenarist -- has been completed. The Scenarist is a knowledge-based ("expert," "rule-based") system for generating tactical military scenarios on a graphics workstation. Using rules stored in a "knowledge base," the Scenarist determines the placement of units and equipment from corps level down to individual items of equipment. The rules specify tactical doctrine for placing units and equipment. The system is designed to incorporate rules that take into account geographic terrain features, the mission of the friendly forces, and the nature of the enemy threat.

The current prototype runs on an 80386-based microcomputer equipped with an MS-DOS operating system, 640 kilobytes of memory, a VGA or EGA video monitor, a mouse, a laser printer, and at least six megabytes of hard disk space.

Major funding for the Scenarist development program was provided by the US Army Communications-Electronics Command (CECOM), through a contract to Vista Research Corporation under the Small Business Innovation Research (SBIR) program. Following an initial three-year development effort, a working prototype of the Scenarist was delivered to CECOM for use in generating scenarios to assist evaluation of electronic warfare systems and concepts. Additional development work is planned, at which time a fully operational, commercially supported product will be available.

VI. How to Use the Scenarist

To use the Scenarist, a user must decide what units and equipment are to be deployed, and on what terrain. The user specifies the location of the

high-echelon units and the system automatically positions the low-echelon subordinate units and equipment in accordance with rules stored in the knowledge base. To use the system, the user must set up data files that describe the units, the terrain, and the placement rules. In the unit descriptions the user specifies initial (suggested, preliminary) relative positions for all of the unit's subordinate units and equipment ("items"). The system then repositions the items in accordance with the rules. It identifies items whose initial placements are not suitable and illustrates the repositioning process on the video screen. The locations of the repositioned items may be stored in a file for input to another program (e.g., a system evaluation program that accepts scenario data), or they may be printed on a line printer. The system can output the screen graphic display to a laser printer.

For test and evaluation applications, the user would typically set up an experimental design in which he would vary the terrain. Depending on the application, he might also vary the unit composition, the enemy threat, or the placement rules.

The current prototype of the Scenarist is a "deterministic" system, in the sense that it generates a single, reproducible scenario for a particular unit placed at a particular location on a particular map. If it is desired to introduce random variations into the generated scenarios, the user may select probability samples of terrain or locations.

Scenarist Data Requirements. To use the Scenarist, the user must specify three kinds of data: (1) map data: digital mapping data that describe the geographic features of an area; (2) unit data: data on the composition, structure, and locations of the higher-echelon units to be deployed in the area; and (3) placement rules: rules for placing lower-echelon subordinate units and equipment, taking into account terrain features, the missions of friendly forces, and the nature of the enemy threat. The system displays the higher-echelon units on the map in the specified locations, repositions lower-echelon subordinate units and

equipment in accordance with the rules, and displays the repositioned subordinate units and equipment.

The map data are stored in digital "map files." The map data specify geographic characteristics such as terrain type (e.g., water, urban, forest), altitude, and the presence of roads. The unit data (stored in "unit files") specify the numbers, types, and positions of all subordinate units ("subunits") and equipment within a unit, the location of the unit boundaries, and the unit mission. The rules, which are stored in a "knowledge base," specify how the subunits and equipment within a unit should be positioned. They are specified in accordance with tactical doctrine, and take into account the information contained in the map and unit data files.

Scenarist Functions. Once the user has placed a higher-echelon unit on a map, the system allows for three types of placement of subordinate units and equipment. First, all of the subunits and equipment are placed on the battlefield (displayed on the map on the video monitor screen) without reference to terrain features, friendly mission, or enemy threat. This initial placement may be arbitrary, but is typically consistent with tactical doctrine in the absence of any knowledge about the terrain features, the friendly mission, or the enemy threat. It is called a "canonical placement." Second, the user may execute the rules stored in the knowledge base to modify the canonical placement (i.e., reposition lower-level subordinate units and equipment) to make it consistent with tactical doctrine, taking into account terrain features, friendly mission, and enemy threat. Finally, the user may review the placement of each unit or equipment, and modify it if desired. He may simply "override" the rule-based placement, or modify the rules in the knowledge base and re-execute them. The user may then print the locations of all of the units, subunits and equipment or write them in a file for input to another computer program (e.g., a system-evaluation simulation model that accepts scenario data). In addition, he may print a black-and-white copy of the (color) video monitor screen on the laser printer.

The system provides the user with the ability to "zoom" on the map to observe a higher level of detail. The amount of labeling displayed on the screen may be varied, to keep the "clutter" below a desired level.

VII. Scenarist Applications

The Scenarist can assist the development of scenarios for any application in which units and equipment should be positioned on the battlefield in a manner that takes into account geographic features, friendly mission, and enemy threat. It is useful in any application in which the validity of the subordinate unit and equipment positioning is particularly important. The system enables the generation of scenarios having a high level of "face validity," since the positioning is done in accordance with rules derived directly from tactical doctrine. The positioning is valid with respect to all factors and relationships embodied in the rules of the knowledge base.

The amount of time and effort required to set up the Scenarist varies depending on the availability and amount of map, unit, and rule data. If these data are available in a form that can be readily input to the Scenarist and if there are not a large number of units involved, the setup time would be on the order of days. If none of the required data are available in suitable form, and if a large number of maps and units is involved, data assembly and preparation could take months. The setup and operation of the Scenarist is similar in nature and level of effort to the setup and operation of a tactical combat model. The required effort depends heavily on the level and amount of detail (map, unit, and rules) to be included in the system. This in turn depends on the nature of the application for which the generated scenarios are desired. As with any computer program utilizing a large amount of data and offering a variety of functions, a user's first application will require significantly more time than later applications.

The benefits of using the Scenarist are the ability to automatically generate scenarios having a high level of face validity, and the ability to easily modify scenarios (maps, unit locations, unit compositions, rules) once the initial data assembly and system setup are accomplished. In short, the principal benefit of the Scenarist is its ability to generate high-validity tactical force laydowns quickly on a variety of digital-terrain maps.

The Appendix describes the capabilities and features of the Scenarist in greater detail.

VIII. Sample Output

Figures 1-6 illustrate some sample screen and printer output from the Scenarist. Figure 1 presents a screen showing the Beqaa Valley in Lebanon. Figure 2 shows two hypothetical units placed on this map; the geographic labels were suppressed for clarity. The "Beqaa Valley" data set was generated by hand and was hypothetically contrived to incorporate data characteristics that would permit thorough testing of all of the Scenarist features and clear illustration of rule processing. To show the sequence of rule and search processing in a small number of steps, a very low resolution map was used (two kilometers per map cell). This simplified example included only three rules for determining the suitability of a radar unit's position.

Figures 3 and 4 illustrate the repositioning of a radar unit using rules, in the "Beqaa Valley" example. These figures display the "cell" map, in order to illustrate the sequence of the rule processing and search sequence in a very clear, simple example (large map cells). Figure 3 shows the canonical (initial) placement of a unit. The canonical placement is unsuitable since it results in a radar unit's being placed in a lake, when one of the rules specifies that a radar may not be located in water (i.e., in a

map cell whose "terrain type" is water). The system applies all of the rules to all of the radars and identifies the unsuitably located radar. A search is initiated for a suitable location. Figure 4 shows the unit relocated in a neighboring map cell whose terrain type is not water. Figure 5 shows a hard-copy printout of the video display.

Figure 6 shows a TRAILBLAZER unit displayed on terrain near Spearfish, SD. Figure 6 was produced as part of a test of the Scenarist system. The digital terrain mapping data (30-meter resolution) were extracted from digital mapping data provided as sample data with the US Army's GRASS Geographic Information System. The rules for positioning TRAILBLAZER were derived from US Army field manuals, as were specifications of the unit compositions and command structures. In this test application, a total of eight rules were specified to determine the locational suitability of TRAILBLAZER units. These rules addressed a variety of considerations, such as proximity to the division front, accessibility of the terrain, and line-of-sight between units.

IX. Scenarist Features and Benefits

The Scenarist was developed using the classical methodology of systems engineering (requirements analysis, synthesis of alternatives, selection of a preferred alternative, detailed design, implementation and test) and the modern software engineering discipline (top-down, structured design). The software development was conducted using elements of the US Department of Defense standard for software development, DOD-STD-2167A, and a "rapid prototyping" approach. The Scenarist is coded in the C programming language. The code is highly modular, allowing for easy future modification and maintenance.

The design of the Scenarist automated scenario generator is based on a conceptual framework that includes parametric representations of units that are well-suited to the formulation of placement rules. The model is

highly object-oriented, with all of the unit and map data placed into carefully designed data structures for easy conceptual understanding and program manipulation. The TRAILBLAZER test illustrated the feasibility of applying the system to a real-world system using available mapping data and tactical doctrine contained in field manuals. The difficulty of validating manually generated scenarios was mentioned earlier. Validation is not easy even for model-generated scenarios, in which case the model must be validated. One of the advantages of a rule-based system for generating scenarios is that the model validation is somewhat easier for this type of model than some others (such as simulation); the reason for this is that the model possesses a high degree of face validity, since a rule-based system is simply a mechanism for building a scenario directly from all of the known rules about deployments.

The Scenarist approach to scenario generation exhibits a high degree of "face validity," since the entities of the model (unit and map parameters) are closely related to real-world units and terrain. Moreover, the system's map and unit representation has been demonstrated to lend itself easily to the formulation of rules based on information in field manuals. The validity of the approach and the system are readily demonstrated by a review of the rules and observation of the rule-processing sequence on the screen.

X. Scenarist Limitations and Future Development Plans

As mentioned, major funding for development of the current prototype of the Scenarist was provided through a contract from CECOM, as part of the Small Business Innovation Research program. Although the prototype is operational, it is best characterized as a "research tool." It is not a commercially supported software system. Very little of the development effort to date was allocated to development of a sophisticated user interface or fancy graphics. The primary concern in

the development was demonstration of the feasibility and desirability of using a rule-based expert system as a basis for automated scenario generation.

The Scenarist development program has been highly successful in demonstrating the feasibility of using a rule-based expert system for automated scenario generation. Moreover, the system has a sufficiently high level of functional capability and ability to represent a high level of detail that it can be used today to generate scenarios for practical applications. Because of the high degree of complexity of the system and the significant effort required to define unit parametric representations and rules, however, the system is not yet ready for easy use by a variety of users in an "operational" setting. To make the system easy to use, a significant additional development effort is planned in the user-interface and graphic areas.

The user interface of the current prototype is limited. Although the system makes use of windows, menu and mouse features, there are numerous areas in which future development would significantly facilitate use of the system. These areas for future development include (1) porting the software to "high-end" graphics workstations; (2) development of an interface with Defense Mapping Agency (DMA) CD-ROM map data files (Digital Terrain Elevation Data and Digital Feature Analysis Data); (3) development of an interface with DMA map image data (for background maps); (4) development of a rule editor; (5) mouse control of unit definition and editing; (6) mouse control of unit repositioning (i.e., creation of an ability to "drag" units and equipment to new locations with the mouse instead of requiring the user to input coordinates using the keyboard); (7) development of improved graphics and easy-to-use procedures for map zooming, panning, and aggregation. It is planned to accomplish these and other improvements in the system, at which time an operational version of the Scenarist will be released.

The object-oriented parametric structure of units developed for the Scenarist would lend itself well to dynamic simulation, that is, to the development of a rule-based tactical combat model. Also, the system could be extended to conduct situation assessment. The current design determines subunit and equipment locations given the location of a higher-echelon unit. With additional development, the system would enable the user to locate identified subunits or equipment and apply rules to specify likely identifications and locations for higher-level units containing the identified items.

XI. Acknowledgements

The author gratefully acknowledges the support provided by the US Army Communications-Electronics Command for the Scenarist development. Many individuals participated in the planning and implementation of the system. Mr. John Cervini was the CECOM project officer for the SBIR Phase I effort, and Dr. Frank Elmer was the CECOM project officer for the SBIR Phase II effort. Scenarist development project staff included the author, Ms. Sharon Hoting, Mr. Bill Goodhue, Dr. Bill Rasmussen, Mr. Fletcher Aleong, Mr. Eric Weiss, Dr. Marty Diamond, and Mr. Chris Caldwell.

Reference

1. Scenarist Automated Scenario Generation System: Final Report, Final Report for Project, "Research in Artificial Intelligence for Noncommunications Electronic Warfare Systems (Contract No. DAAB07-89-C-P017)," Vista Research Corporation, Sierra Vista, Arizona, August 31, 1991.

About the Author...

J. George Caldwell is President of Vista Research Corporation, developer of the Scenarist automated scenario generation system. He holds a B.S. degree in Mathematics from Carnegie-Mellon University (1962) and a Ph.D. degree in Statistics from the University of North Carolina at Chapel Hill (1966). In his Ph.D. dissertation, he developed the best-known class of error-correcting codes for correcting both synchronization and additive errors in digital communication. He has served as a consultant in statistics, economics, operations research, and systems and software engineering; research manager; and professor of statistics. He previously served as Manager of Research and Development for the US Army Electronic Proving Ground's Electromagnetic Environmental Test Facility at Fort Huachuca (Sierra Vista), Arizona, and professor of statistics at the University of Arizona, Tucson, Arizona. He is currently (January 1994) consulting on a project funded by the US Agency for International Development in Malaŵi, Africa.

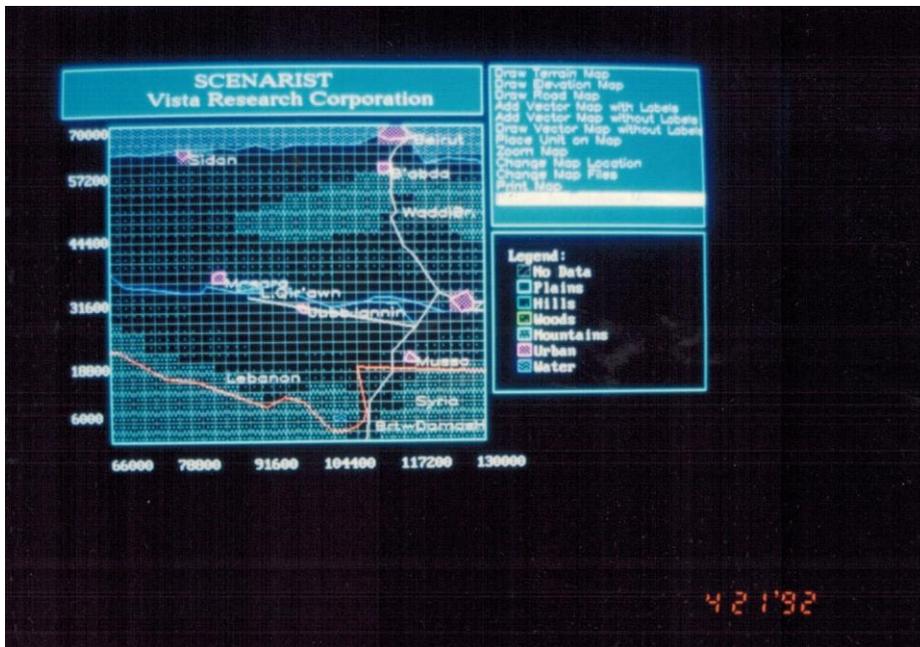


Figure 1. Beqaa Valley in Lebanon ("vector" map).

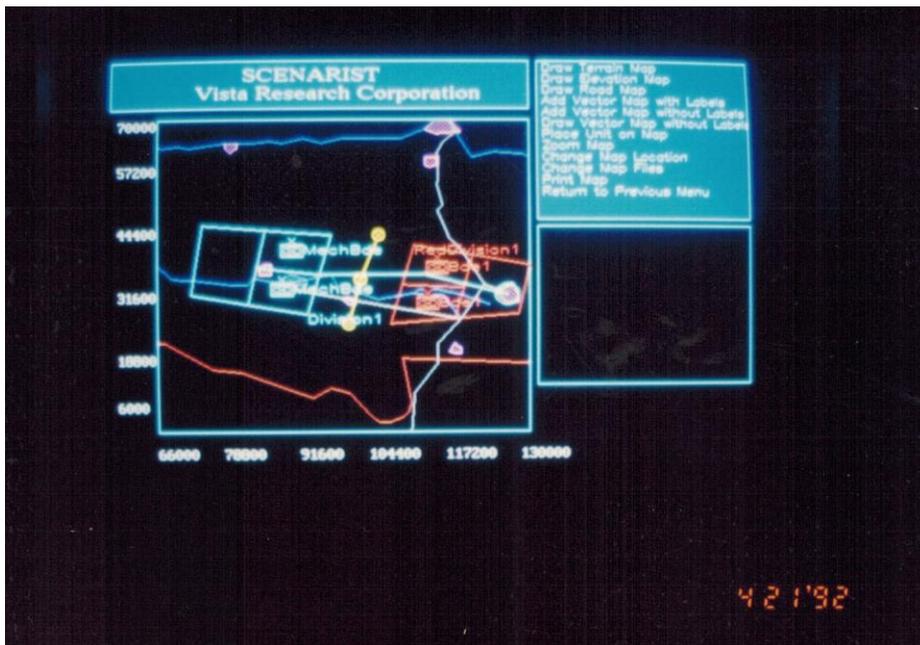


Figure 2. Two Divisions Placed in the Beqaa Valley ("vector" map, labels suppressed).

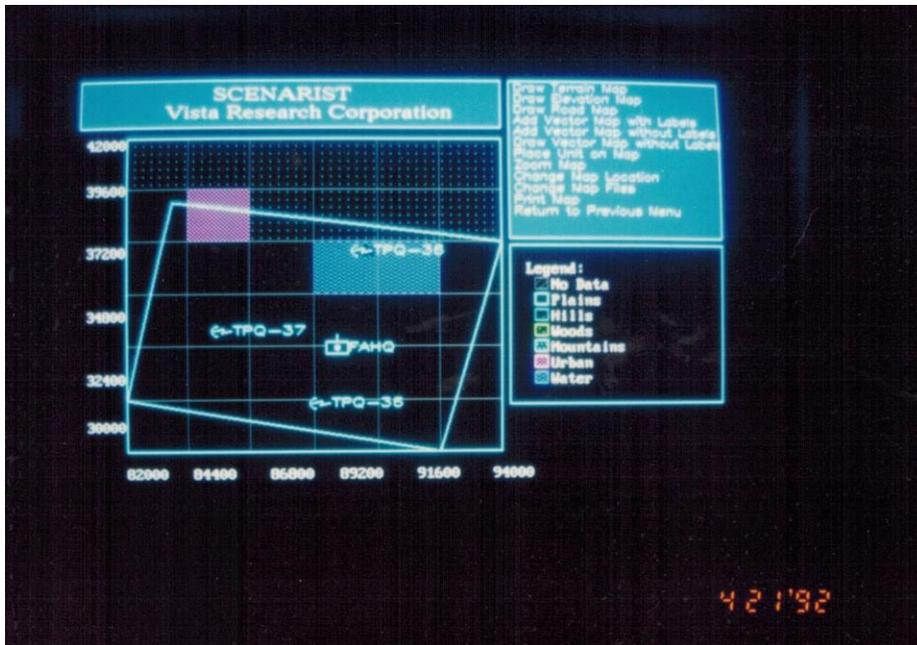


Figure 3. Initial ("Canonical") Placement of Radar Units ("cell" map; note location of radar unit in lake, depicted as a "water" terrain-type cell)

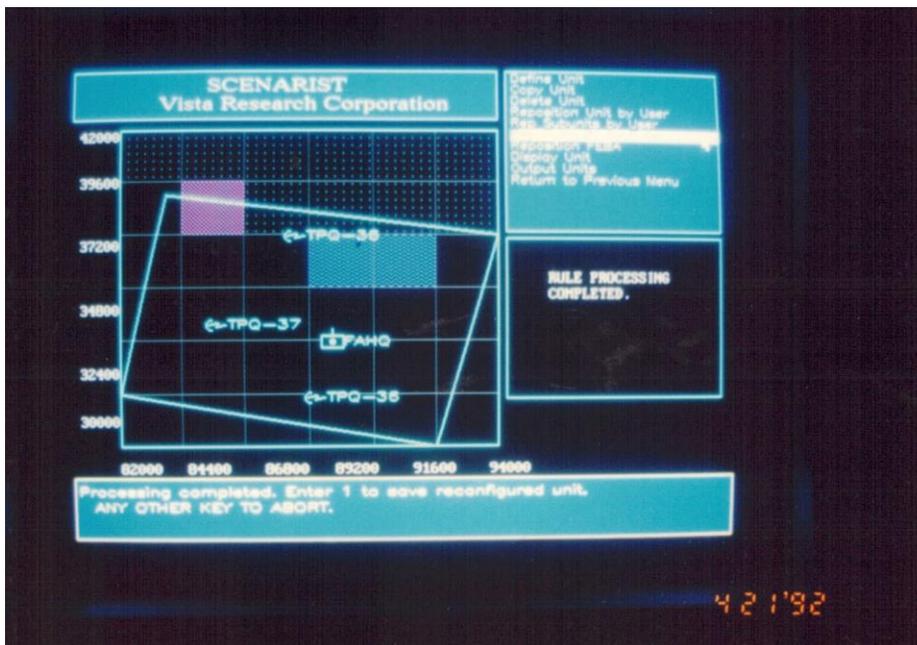


Figure 4. Repositioned Radar Unit, After Applying Positioning Rules Stored in Knowledge Base (note that radar has been moved from water terrain-type cell to land terrain-type cell)

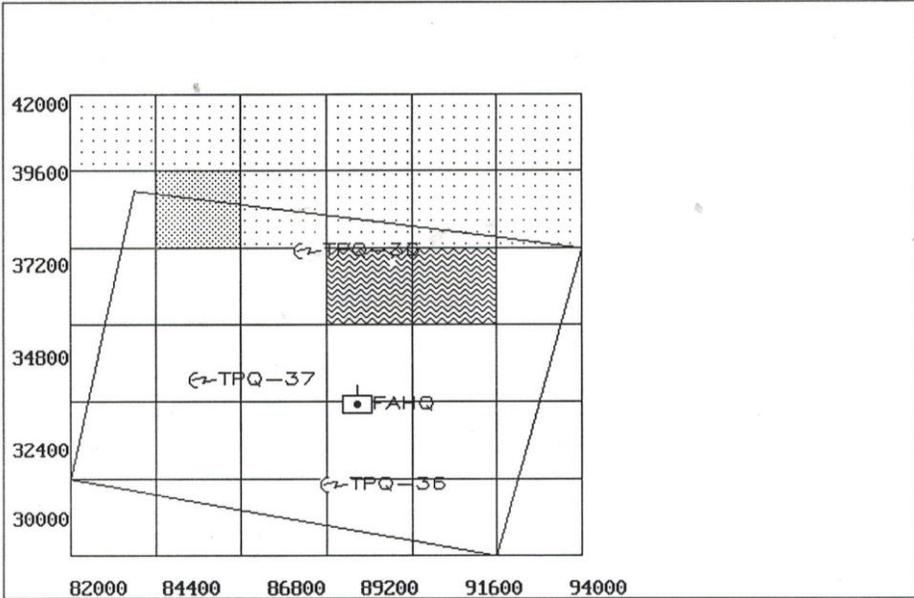


Figure 5. Hard-copy Printout of Screen Display of Figure 4.

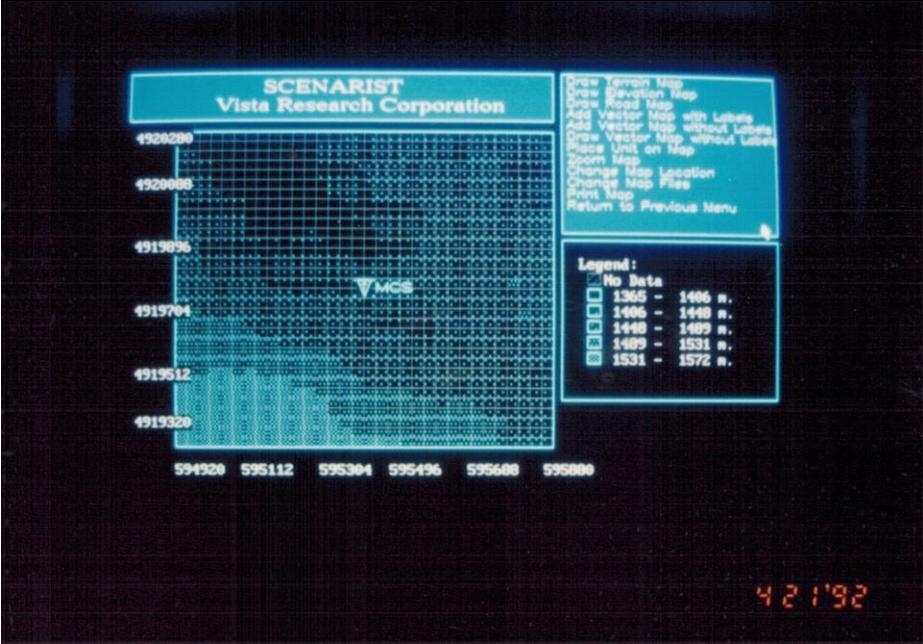


Figure 6. Illustration of TRAILBLAZER Unit on High-Resolution "Cell" Map (30-meter resolution).

Appendix. Detailed Description of Scenarist Capabilities and Functions

I. Scenarist Data Setup

There are two basic phases in using the Scenarist: "setup" and "processing." "Setup" is concerned with assembly of the map, unit, and rule data required or desirable for a particular application, and entry of these data into data files that can be accessed by the Scenarist. "Processing" is concerned with the execution of Scenarist system functions to process map and unit data, including the execution of rules to reposition units and equipment.

Figure A1 presents a summary description of the two phases of Scenarist activity. The steps involved in the system setup (data preparation) phase are as follows.

Map Data Preparation

A principal feature of the Scenarist is its ability to position units and equipment taking into account geographic terrain features. In order for it to accomplish this function, it is necessary for the user to prepare digital terrain maps that describe the geographic features of the area in which the units are to be deployed. The current version of the Scenarist accepts two kinds of map data: "cellular" map data and "vector" map data. The cellular map data specify terrain characteristics (terrain type, elevation, and road access) for every cell of a rectangular grid superimposed over the geographic area of interest. The vector map data are used to display geographic features on maps to make them more understandable; these data include political boundaries, roads, rivers, locations of urban areas, and geographic labels. The cellular map data are accessed by the rules in

the process of repositioning units and equipment. The vector map data are used only for display purposes; they are not used in the analysis.

The vector and cellular map data are stored in data files. The user creates the vector map file with a data editing program ("data editor," either a line editor or a screen editor). The cell map data can also be prepared using a data editor, but more typically would be derived from existing digital mapping files, such as those produced by the Defense Mapping Agency and other digital mapping organizations.

A key design feature of the Scenarist is its ability to process mapping data at varying levels of resolution, matched to the size of the units being repositioned. For example, when positioning large (high-echelon) units, the geographic area of interest may comprise hundreds or thousands of square kilometers, and there is no need for high-resolution map data. Sufficient information for positioning large units is contained in low-resolution maps. Moreover, the amount of computer processing time and storage required to position many units and equipment using high-resolution cellular mapping data would be prohibitive. In this case, the system uses a cellular map in which the map cells are large (e.g., a kilometer on a side). On the other hand, in positioning individual items of equipment in a small geographic area it is desirable to use the highest available resolution map data. In this case, the system uses a cellular map in which the map cells are small (e.g., 30 meters on a side).

Figure A1. Scenarist Data Flow and Major Functions

System Setup (Data Preparation)

Map Data Preparation:

- Obtain digital mapping data (terrain type, elevation, roads)
- Reformat
- Prepare series of aggregate maps (varying resolutions)

Unit Data Preparation:

- Develop parametric representations of military units ("canonical" configurations)
- Specify command structure

Placement Rules Preparation:

- Specify placement rules ("suitability" conditions)
- Program rules in CLIPS or C

System Processing (Scenario Generation)

<u>Inputs</u>	<u>Processes</u>	<u>Outputs</u>
Map data	Display units on maps	Subunit and equipment positions
Unit data	Reposition subunits and equipment	(to printer or file)
Placement rules		Unit on map display (on screen or hardcopy to laser printer)

Provided with the Scenarist system are computer programs that may be used to produce, from high-resolution mapping data, a series of successively lower-resolution (aggregated) map files. During the course of the Scenarist processing, the user may select a level of aggregation (more specifically, a degree of resolution) that is appropriate for the size of the units being processed, so that positioning decisions are made efficiently, without utilizing an unnecessarily high level of map detail and computer processing time.

Unit Data Preparation

In order to reposition the subunits and equipment of a large unit, a substantial amount of computer processing is required. A number of possible locations have to be examined for each item to be positioned, and a large amount of "geographic" processing (e.g., line-of-sight determinations) has to be done for each candidate location. In order to process rules quickly for large units, it is important to have available an efficient method of describing military units that "captures" the aspects of a unit that are important to represent in the repositioning rules.

The Scenarist embodies a model-based approach to scenario generation. The foundation for this approach is a parametric representation of military units. The term "parametric representation" means that all military units have a common structure, that is defined in terms of a relatively small number of variables, called parameters. The parameters specify the number and types of subordinate items in the unit. The subordinate items are either other military units, platforms, or equipments.

In the Scenarist computer program, all of the unit parameters are stored together in a collection called a "data structure." The data structure includes the following variables:

- o side
- o echelon
- o type

- o number
- o identification (ID) number
- o parent unit's ID number
- o code
- o parent unit's code
- o name
- o location
- o geographic type
- o numbers, geographic types, and relative locations of subordinate items
- o mission, objective, and approach

The system allows for three sides (usually referred to as "BLUE," "RED," and "GRAY"). "Echelon" refers to the level of the unit in the military command structure: army, corps, division, brigade (or brigade group), regiment (or group), battalion (or squadron), company (or battery or group), platoon (or detachment), section, squad, platform, and equipment. "Type" differentiates units of different structure but the same echelon (e.g., a mechanized infantry battalion vs. a tank battalion). "Number" differentiates units of the same type. "ID number" is a user-assigned numerical unit descriptor. The variable "code" is a 13-digit code that specifies a unit's position in the military command structure. The code is unique for every unit, and is the "key" for retrieving a particular unit's data from the Scenarist unit file.

The "name" is a user-assigned name. "Location" specifies the geographic coordinates of the four corners of the unit (each unit is represented as a four-sided polygon, or quadrilateral). "Geographic type" is a variable that specifies how much internal detail will be represented in a unit. The level of detail may vary considerably. For high-echelon units such as divisions, the internal detail allows for the specification of front (on-line) and rear (reserve) subordinate units covering specified areas. For low-echelon units (e.g., a squad), the unit

may be represented simply as a circle, with no additional internal structure specified.

"Numbers, geographic types, and relative locations of subordinate items" refers to data that specify how many subordinate items of each geographic type are included in a unit and their relative locations within the unit. (These relative locations are the initial, or "canonical" locations.)

"Mission, objective, and approach" refer to the intentions of a unit relative to a geographic area. The missions are identified by number, and may represent any missions of the user's choosing. The objective is represented by an ellipse on the map. The approach is a line from the center of the unit's front to the center of the ellipse defining the objective. These data about the mission, objective, and approach may be referred to in the rules.

The Scenarist contains an interactive capability (video display, mouse, keyboard) for "defining" units, i.e., for entry of all of the data required to specify a unit in the parametric representation used by the Scenarist and storing it in a unit file. To facilitate the unit-definition process, the system enables the user to define "generic" units of a particular echelon and type, and to "copy" the generic-unit data into a "specific" unit of that same type.

Placement Rules Preparation

The Scenarist is a "rule-based" system for positioning units and equipment. In order for it to operate, the user must specify rules that indicate the conditions under which a particular location is suitable for the placement of a low-echelon unit or equipment (the boundaries of high-echelon units are specified by the user). The rules may take into account the characteristics of the terrain (e.g., terrain type, road availability) or relationships to other units and equipment (e.g., distance to other units, or the availability of line-of-sight to certain other units).

To construct the placement rules, the user may review field manuals that describe the tactical doctrine for positioning the units and equipment, and formulate quantitative statements of the placement suitability conditions in terms of variables whose values can be determined from the cellular map data and the unit locations and other characteristics.

The initial version of the Scenarist has been developed with two subsystems for specifying rules. The user may specify the rules either in CLIPS rule syntax or in the C programming language. CLIPS -- the C Language Integrated Production System -- is a knowledge-based system developed by NASA and widely used by US defense contractors and organizations. The user must translate the suitability conditions into either CLIPS or C, and store them in a CLIPS rule file or a C function file.

II. Scenarist Functions: What the Scenarist Does

Figure A2 identifies the functions of the Scenarist system. These functions accomplish the processes identified in the "System Processing" phase of Figure A1. These functions fall into four major categories: functions dealing with system input data, functions for displaying units on maps, functions for repositioning items, and functions for producing system output.

Input Data Functions

The three major categories of input data are map data, unit data, and rule data. With respect to map data, the user is required to provide digital mapping data in files of a prescribed format. As mentioned, the system includes software for aggregating maps. With respect to rule data, the user may specify rules either in C language functions or in CLIPS rule files. The generation of the map files and rule files is accomplished external to the Scenarist system. With respect to unit data, the Scenarist includes an interactive capability for building a Scenarist unit file.

The system input data functions are as follows:

Project Selection. For each Scenarist application, the user sets up a "project file" that identifies the map files, unit files, and rule files that are to be used. At the beginning of each Scenarist run, the user selects a project file. The system then inputs the map, unit, and rule data from the files specified in the selected project file. The user is provided with options for specifying the portion of the map to be displayed on the screen (i.e., map size and location).

The Main Menu. After the data files specified in the project file are input, the "main menu" appears on the screen, and the user is requested to select a menu option.

Define Unit. This function requests the user to input all of the data required to specify a unit. The unit data are values for all of the unit parameters described earlier. The user may define either "specific" (actual) units or "generic" units (exemplar units from which specific-unit "copies" can be made). Once all of the required data have been provided by the user, the defined unit is written into the generic-unit file or the specific-unit file specified in the project file.

Copy Unit. To facilitate the definition of units, the user may "copy" all of the unit data from a previously defined specific or generic unit, and define a new unit by modifying some of the parameter values of the copied unit (i.e., its identification, location, and command-structure relationship to other units).

Delete Unit. This function deletes specified units from the generic or specific unit files.

Display Unit. This function lists all of the units in the specific-unit file, and enables the user to display individual units on the screen.

Figure A2. Scenarist Functions

System Input Data Functions:

- Project Selection
- The Main Menu
- Define Unit
- Copy Unit
- Delete Unit
- Display Unit

Functions for Displaying Units on Maps

- Draw Terrain Map
- Draw Elevation Map
- Draw Road Map
- Add Vector Map with Labels
- Add Vector Map without Labels
- Draw Vector Map without Labels
- Place Unit on Map
- Zoom Map
- Change Map Location
- Change Map File

Functions for Repositioning Items

- Reposition Unit by User
- Reposition Subunits by Rules
- Reposition Subunits by User

Data Output Functions

- Print Map
- Output Unit

Functions for Displaying Units on Maps

The Scenarist displays units on maps to provide the user with a means for locating major units and for judging the reasonableness of rule-based placements. The maps available to be displayed on the screen are those specified by the user in the project file. They include terrain and elevation cell-maps (required), a road map (optional), and a vector (descriptive feature) map. The Scenarist performs all analytical processing using 32-cell by 32-cell portions of maps of varying levels of resolution (degrees of aggregation). The 32-by-32 maps used at various points in an application will vary in resolution, depending on the level of detail specified by the user as appropriate for the size of the items being positioned (i.e., low-resolution maps for positioning large units, high-resolution maps for precise positioning of individual items of equipment).

While cell maps (rather than vector maps or raster-image maps) are typically used in geographic-information-system applications as the basis for analysis, they are "grainy" or "blocky," and are not as pretty for display purposes as vector or raster-image maps. Cell maps are helpful in visualizing how the rules work in repositioning units. Vector maps are more suitable, however, for presentation purposes, since they do not show the map grid cells. The Scenarist provides the user with the option of displaying either cell maps (to monitor the rule repositioning process) or vector maps (for presentation purposes). The functions for displaying units on maps are the following.

Draw Terrain Map. This function displays a terrain cell map.

Draw Elevation Map. This function displays an elevation cell map.

Draw Road Map. This function displays a road cell map.

Add Vector Map with Labels. This function is used to overlay a vector map, including all labels, on a map already on the screen (either a cell map or a vector map with no labels).

Add Vector Map without Labels. This function is used to overlay a vector map, without labels, on a cell map.

Draw Vector Map without Labels. This function is used to display a vector map.

Place Unit on Map. This function draws a specified unit on whatever map is on the screen.

Zoom Map. This function zooms or unzooms the map.

Change Map Location. This function enables the user to extract a different 32-cell by 32-cell portion of the map-file map.

Change Map File. This function enables the user to select a different map file (e.g., a map having a higher or lower level of aggregation).

To date, most of the Scenarist development effort has been invested in developing the expert-system aspects of the Scenarist system, not in developing user-interface systems. For this reason, the input data and map processing functions of the current Scenarist prototype are rudimentary. They are not on a par with the features of commercially supported mapping or geographic information systems. Later versions will develop improved input data editing, map editing, and rule editing capabilities.

Functions for Repositioning Items

The location of a unit is specified in either of two ways. If the unit has no defined superior (parent) unit, then the user must input the map

coordinates of the four corners of its quadrilateral representation. If the user has previously defined the unit's parent unit, however, the location of the unit is implied by the parent unit's location and the relative location of the unit in the parent unit, and the Scenarist hence determines the unit's location automatically from those parameters. The locations of platforms and equipments are always computed automatically from the parent unit's location and the relative locations of those items in the parent unit.

After a unit has been placed on the map, the user may decide to reposition ("move") it. This is done by accessing the function Reposition Unit by User. The user specifies new coordinates for the unit's four corners. If desired, the user may also respecify the unit's mission, objective, and avenue of approach. The system then automatically recomputes the location (corner) coordinates for all of the units in the unit file that are subordinate to the unit being repositioned.

To reposition a unit's subitems in accordance with the rules, the user executes the function Reposition Subunits by Rules. When that function is initiated, the Scenarist determines (according to the rules) the suitability of every "point" subunit, platform, and equipment in the unit (these subitems are called "rule-applicable" subitems). If a subitem's location is not suitable, the system initiates a search for a suitable location. The search is conducted over the nearby cells of the cell map. The search is continued until a suitable location is found or until a prescribed number of unsuitable locations has been examined.

The Scenarist classifies rules into two categories: "local" rules and "global" rules. For local rules, the suitability of an item's location does not depend on the location of any other rule-applicable subitems in the unit. For such rules, subitems can be repositioned sequentially in a single "pass" through the subitems. Once a suitable location is found for an item, it remains suitable no matter how the other items are

repositioned. An example of a local rule is the rule that an item may not be located in water.

For global rules, the suitability of an item's location depends on the locations of other rule-applicable items. For such rules, it is not possible to reposition the items sequentially in a single pass through the items since the suitability of an item's location may be affected by the repositioning of another subitem. Each time an article is repositioned, it is necessary to recheck the suitability of the positions of all of the other items. For global rules, it is hence necessary to process the rules in a simultaneous or iterative manner.

In the Scenarist, the suitability of a location for an item is determined by applying ("firing") the rules of the knowledge base. Search strategies are implemented by means of algorithms programmed in C, not in CLIPS.

When the user processes the Reposition Subunits by Rules function, the display "flashes" each item as its suitability is being examined. If an item's location is unsuitable, the rules that "failed" are shown on the screen, and a search is initiated for a suitable location. The "search path" (sequence of alternative locations considered) is also shown on the screen.

It is possible that the user may be interested in "manually" positioning certain items. This could occur, for example, if it is wished to position units and equipment in a particular prespecified arrangement in order to conduct some sort of comparative evaluation. Or, it may be helpful in understanding how a rule works to fix a particular item in a very unusual place, and observe how the rules position related items around it. In any case, the Scenarist enables the user to manually reposition items, through the function Reposition Subunits by User.

Data Output Functions

There are two data output functions. The Print Map function prints a black-and-white copy of a map on a laser printer. The Output Unit function writes the locations of all of the subordinate units and equipment of a unit in a file or prints this data to a printer.

The preceding discussion has described the principal functions of the Scenarist. There are a few other functions, such as functions to define or relocate the "Forward Edge of the Battle Area (FEBA)," to display the rules in the CLIPS rule file, and to identify the computer hardware system.

FndID(186)

FndTitle(The Scenarist™ Automated Scenario Generation System)

FndDescription(The Scenarist™ Automated Scenario Generation System)

FndKeywords(Scenarist; automated scenario generaton; artificial intelligence; test and evaluation; military scenarios)