Entire Event Management

Concepts

Version 2.2.1

November 2016

This document applies to Entire Event Management Version 2.2.1.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: NCL-CONCEPTS-221-20161117

Table of Contents

1 Concepts and Facilities

This documentation introduces the reader to the Entire Event Management system and provides a broad outline of its facilities and functions.

It covers the following topics:

Introduction	What is Entire Event Management? How it is integrated with other Software AG products. Software requirements.
Functional Overview of Entire Event Management	An overview of the functional areas of Entire Event Management.
Entire Event Management Components	An overview of the components of Entire Event Management.
Entire Event Management Object Types	An overview of the object types which must be defined and maintained when working with Entire Event Management.
User Interface	An overview of the user interface of Entire Event Management.

2 Introduction

This documentation introduces the reader to the Entire Event Management system and provides a broad outline of its facilities and functions.

It contains the following information:

- A definition of Entire Event Management and an outline of the benefits of installing Entire Event Management in your data center;
- An illustration of how Entire Event Management fits into Software AG's 4th generation solution for the data center;
- Software prerequisites for installing Entire Event Management;
- An overview of the functions provided by Entire Event Management;
- A description of the component types that make up Entire Event Management;
- A description of the object types that must be created before working with Entire Event Management;
- An overview of the product's user interface.

The following topics are covered below:

What is Entire Event Management?

Entire Event Management is an online facility for controlling operating systems. It enables a computer operator to control multiple processors, multiple operating systems or both from a single operator console. It responds automatically to system Events defined by the computer operator. Entire Event Management secures the system so that users can access only the parts of the system and the functions for which they are authorized.

Entire Event Management is a powerful tool for controlling and administering a computer center so that it runs smoothly and its facilities are available as often as possible.

The benefits of using Entire Event Management include:

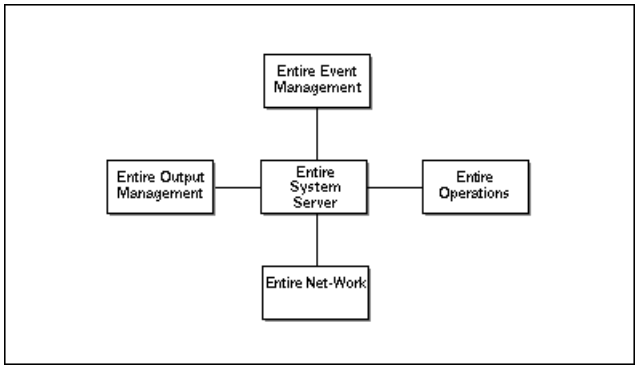
- Replacing standard operating tasks and repetitive operating tables with automatic operator responses;
- Detecting system failure early, thus reducing downtime;
- Selecting messages which have high priority or are of immediate interest;
- Retaining messages for a fixed period of time that is defined by the operator;
- Monitoring individual components of the system;
- Distinguishing messages from different ranges;
- Maintaining and displaying Logical and Physical Consoles;

- Monitoring automatic operator responses;
- Enhancing security by defining flexible Profiles that control access by users and the options available to users;
- Access to operating systems is provided by Entire System Server; no changes are necessary.

Integration with Other Software AG Products

Entire Event Management is part of Software AG's Entire Systems Management (ESM) line of products, the 4th generation solution for integrated control of the data center. The following figure illustrates how Entire Event Management fits into this software architecture:

Entire Event Management in System Automation Tools



- Entire System Server provides the basic technology for access to operating system information and services;
- Entire Net-Work enables access to operating systems located on different physical machines within the computer network;
- Jobs controlled by Software AG's automated job scheduler Entire Operations can be triggered as part of Entire Event Management's automatic operator responses;
- Entire Output Management provides comprehensive handling of reports generated in the system environment.

Software Requirements

Entire Event Management is available for computers running z/OS, MVS/XA, z/VSE and BS2000/OSD operating systems.

The following Software AG products are prerequisites for Entire Event Management:

- Adabas
- Natural
- Entire System Server
- System Automation Tools (SAT)
- Natural Security (optional)
- Entire Net-Work, (optional, for multi-CPU support)
- EntireX Broker (optional when using the API).

3 Functional Overview of Entire Event Management

Entire Event Management consists of the following functional areas:

These are described in more detail below.

Message Selection and Logging

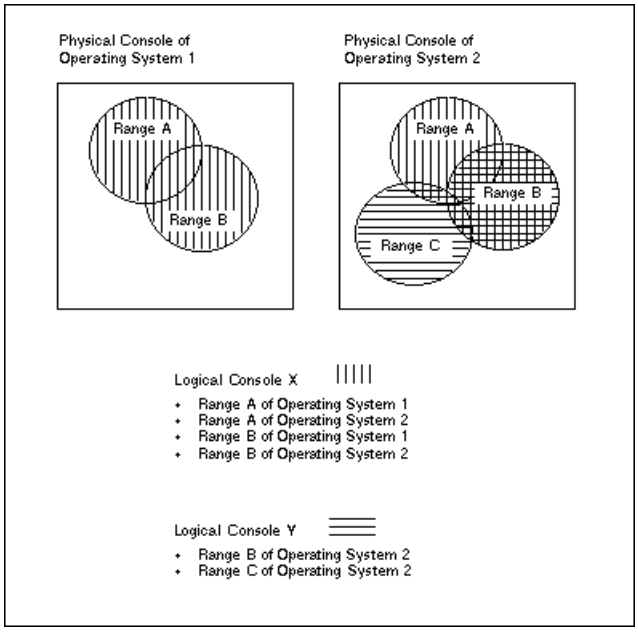
In addition to the conventional views of all messages from a single operating system, Entire Event Management displays selected messages from selected operating systems. Messages from multiple operating systems can be displayed together; irrelevant messages can be excluded from display.

In Entire Event Management, the conventional operator console of an operating system is called a “Physical Console”. It displays all the messages from one operating system. Entire Event Management enables the user to switch the operator console to any Physical Console, thus providing quick and easy access to every Physical Console in a computer network.

Using Entire Event Management, you can select from multiple Physical Consoles the messages that are relevant to a particular topic and you can log those messages for display in a Logical Console. A Logical Console can be restricted to a particular subsystem; it need not be restricted to a single operating system. Any Logical Console can be viewed at the operator console in the same way as a Physical Console. The display at a Logical Console can be restricted to messages with high priority.

A Logical Console provides a selective view of messages from one or more operating systems. In this way, Entire Event Management achieves centralized control of different operating systems.

Message Logging



Uses of Logical Consoles

The concept of Logical Consoles provides a wide range of uses. Some examples are listed below:

- Important messages can be extracted from one or more operating systems and logged in one Logical Console;
- All messages relevant to particular subsystems on various operating systems in the Entire Event Management environment can be selected and logged in a Logical Console, for example:
  - Messages from Adabas databases;
  - VTAM messages;
  - TP system messages;
  - z/OS messages in general;
- Irrelevant messages can be suppressed.

Benefits of Logical Consoles

Benefits for the computer operator of logging messages to Logical Consoles include:

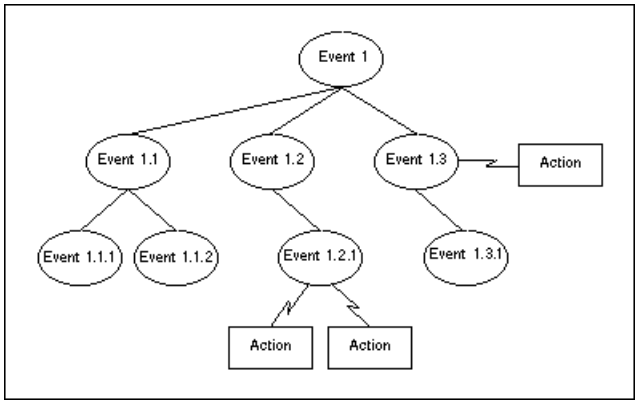
- Easy monitoring and control of a particular subsystem, facilitating faster response when abnormal situations or problems arise;
- Ability to control a departmental computer to which no particular operator is assigned.

Automatic Operator Responses

Automatic operator responses to system Events are achieved by defining Automation Rules. An Automation Rule links an Action to an Event. The Action is triggered by the occurrence of the Event. An automatic operator response consists of one or more Actions that are triggered by an Event. An Event occurs if a message satisfies the conditions set in the Event definition, this means that parameters such as job name, message ID, or character strings appear in the message.

Events can be made inter-dependent by defining an Event Tree that relates them. The following figure illustrates this concept:

Event Tree



An Event Tree is a special kind of decision tree. Events on a higher level must occur before dependent Events on a lower level can occur. Any path in this tree is active if each constituent Event has occurred. For example, in the figure above, Event 1.2.1 can only occur when Event 1 and Event 1.2 have already occurred.

The following types of Action can be triggered by an Event; multiple Actions for a single Event can be defined:

- Display in a Logical Console an information box containing text;
- Issue operator commands;
- Submit a job;
- Send a message to users, WTOs or Logical Consoles;
- Execute a Natural program;
- Start an Entire Operations job network.

Authorization

Access to Entire Event Management objects and functions can be restricted by assigning a Profile to the User.

Access to Objects

A Profile can forbid access to any Entire Event Management object or can grant access on four levels: display, modify, add and delete. This ensures enhanced security of sensitive subsystems within the Entire Event Management environment.

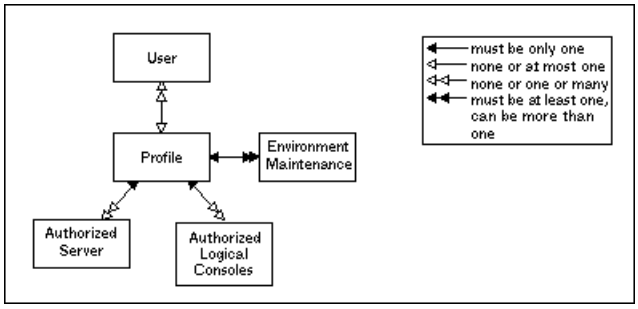
Access to the Defined Environment

Access to the defined environment of Entire Event Management can also be limited. In a Profile, you can define:

- which Logical Consoles can be accessed;
- which Entire System Server Server and Entire Event Management Server can be accessed;
- which security-sensitive functions can be used such as issuing operator commands and controlling the Entire Event Management Servers.

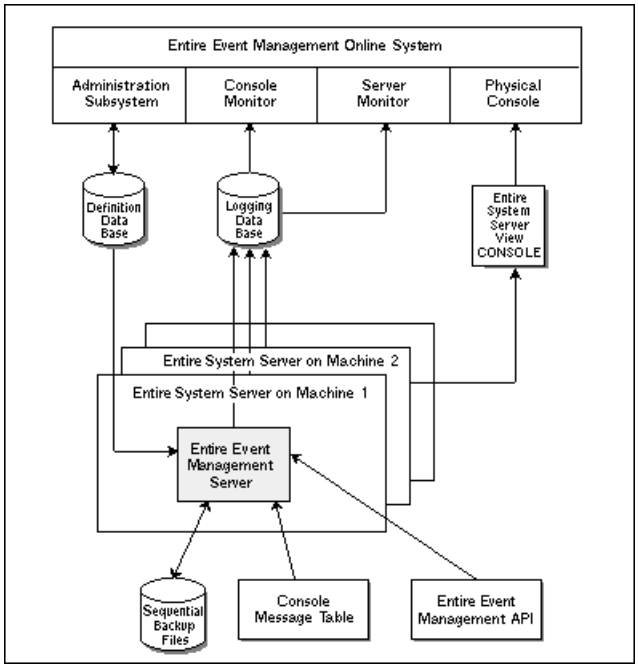
Entire Event Management's security concept is: what a user can see, the user can use.

Object Relationship Diagram for Authorization



4 Entire Event Management Components

The Entire Event Management system consists of components of the following types:



These component types are described below. In this documentation, component types are written with initial capital letters.

Physical Console

A Physical Console is the normal console of an operating system and displays all messages from that operating system. It is also identified by the Entire System Server Node number of the operating system. The user can switch to each Physical Console and view all the messages from each supported operating system in a network.

For a Physical Console to function, Entire System Server must be installed in the appropriate operating system and an Entire System Server Node object must be defined within Entire Event Management.

Server

A Server searches its Physical Console for messages that are inside the Included Message Ranges, but outside the Suppressed Message Ranges that are specified in a Logical Console definition. It then routes these messages to the Logging Database for display in that Logical Console.

The Server also checks for Events as defined by Automation Rules. When an Event occurs, the Server executes the appropriate Actions.

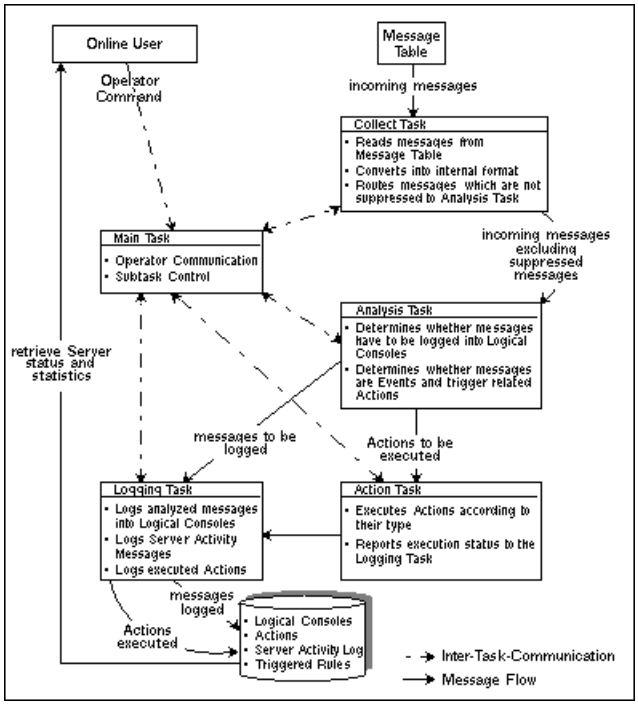
A Server must be defined in the Entire Event Management environment and must be started under Entire System Server in the corresponding operating system for each operating system to be controlled by Entire Event Management.

The Server consists of several service tasks running as subtasks under Entire System Server:

- The Main Task attaches the other subtasks and controls them via inter-task communication. When an online user issues a Server control command such as SHUTDOWN SERVER or REFRESH statistics, the Main Task processes the command by telling the subtasks to terminate or report their statistics. The Server statistics are written to the (Act) Console in the Logging Database, from which they can be retrieved by the online user.
- The Collect Task reads all incoming messages from the Console Message Table and converts them into internal format. It checks the messages against a list of IDs of suppressed messages, which is maintained as a Server parameter. Finally, it routes messages that are not suppressed to the Analysis Task.
- The Analysis Task checks the messages against the conditions defined in the Message Ranges. In this way, it selects messages which must be logged to a Logical Console and determines whether the messages are Events and trigger Actions. It routes messages to be logged to the Logging Task and Actions to be executed to the Action Task.
- The Logging Task logs analyzed messages to their corresponding Logical Consoles; logs messages related to Automation to the (Aut) Automation Console; optionally logs messages which are not yet covered by the currently defined Logical Consoles to the (Udf) Undefined Console; logs Server Activity messages to the (Act) Activity Console for each Server start, restart or renew and logs Server protocol messages to the (Log) Console. All these Logical Consoles are located in the Logging Database.
- The Action Task executes Actions according to their type. It routes the result of the executed Actions to the Logging Task which logs this result to the (Aut) Automation Console in the Logging Database.

The following figure illustrates the flow of information through a Server:

Server Information Flow



Server Monitor

The Server Monitor is a reporting facility that provides information on the activities of a Server. This includes a Statistic Monitor, which shows the status of Server tasks and the information flow between the various service tasks as well as an Activity Monitor which shows which Logical Consoles are currently being served by the Server and which are the most recent messages routed to these Consoles.

Console Monitor

The Console Monitor lists all Logical Consoles defined in the User's Profile and shows the most recent messages routed to these consoles. You can access the Logical Console and display its messages by invoking line commands. The user can scroll up and down through the messages, scan for strings and enter selection criteria for most fields.

Logging Database

The Logging Database stores all messages which are routed by any Server to a Logical Console, all Events and executed Actions in the (Aut) Automation Console, all Server Activity messages in the (Act) Activity Console and all Server protocol messages in the (Log) Console (see the section ***Server***).

How long a message remains in the Logging Database depends on the specifications made in the Logical Console. The system provides a batch utility which can be started every day. It deletes all messages from the Logical Console, for which the time limit has been exceeded. For more information, see *Maintaining the Logging Database* in the *Installation and Customization on Mainframes* documentation.

Administration Subsystem

The Administration Subsystem maintains all Entire Event Management objects, for example, all definitions necessary to build Logical Consoles and to respond to certain messages automatically. Authorization definitions are also maintained here. For a detailed explanation of the objects maintained, see the section ***Entire Event Management Object Types***.

Definition Database

The Definition Database holds all object definitions: User IDs, Profiles, Calendars, Entire System Server Node numbers, Server Parameters, Message Ranges, Logical Consoles, Logical Console Layouts, Automation Rules, Events and Actions.

Entire Event Management API

The Entire Event Management API enables applications to forward exception messages, so-called events, to the Entire Event Management server for further analysis. The server determines by means of filter and automation rule definitions provided by the administrator whether the event must be logged in the database and which automated actions must be executed.

5 Entire Event Management Object Types

Objects of the following types must be defined and maintained when working with Entire Event Management:

Physical Environment

- Entire System Server Node
- Server Parameters

Logical Environment

- Message Range
- Logical Console
- Logical Console Layout

Automation

- Automation Rule

Authorization

- Profile
- User

Calendar

- Calendar

These object types are described in the following topics:

In this documentation, object types are written with initial capital letters.

Physical Environment

Entire System Server Node

Entire System Server must be installed on every operating system embedded in the Entire Event Management environment. An object of the type "Entire System Server Node" identifies an Entire System Server Server by number within the Entire Event Management environment. Depending on the requirements of your installation, more than one Entire System Server Server can be installed on the same operating system.

Server Parameters

An object of the type “Server Parameters” defines some parameters for a Server (see the section *Server*). The most important of these are the Logical Consoles to be served by the Server. The Server Parameters also set the overall framework for message logging and automation.

Logical Environment

Message Range

An object of the type “Message Range” is a filter for the message-logging process and describes a set of messages to be included in or excluded from display at a Logical Console. A Message Range is the summary of all messages which satisfy all range conditions that have been defined. The same Message Range can be specified for more than one Logical Console.

Messages are selected using conditions that include the job name, the message ID and character strings that appear in the message. The Asterisk Selection Facility is available to select messages according to these conditions. With the help of delimiters, a search can be performed for any character string in the message line.

Representation attributes can be defined for a Message Range: color, prefix and highlight or blink. These serve to distinguish it optically from other ranges.

Logical Console

An object of the type “Logical Console” is defined by a set of Message Ranges and contains all definitions necessary to display messages logged from one or more operating systems.

The Logical Console definition includes its name and the names of the Included Message Ranges to be displayed in the Logical Console component. Logical Console objects can also have message logging, automation, console-layout, life-time and representation attributes. In addition, Excluded Message Ranges can be defined for messages not to be displayed in the Logical Console component. All messages which satisfy the conditions of at least one Included Message Range and are not in an Excluded Message Range are displayed in a Logical Console.

For each operating system to be controlled with the concept of Logical Consoles, a Server must be running. The Server scans all messages from Physical Consoles, selects certain messages according to conditions set in the Included Message Ranges and Excluded Message Ranges of the Logical Console object and routes the selected messages to the Logging Database. The Logical Console component retrieves the messages for display from the Logging Database.

Representation attributes can also be defined for a Message Range in a Logical Console: color, prefix and highlight or blink. These serve to distinguish it optically from other ranges specified for the same Logical Console.

Logical Console Layout

An object of the type "Logical Console Layout" determines what console-related data are displayed and the sequence and width of the columns in which they are displayed. Normally, a Logical Console is always displayed according to the layout definition assigned to it. However, you can temporarily assign any other defined Console Layout by selecting it on the Logical Console screen.

Automation

Automation Rule

An object of the type "Automation Rule" specifies how the system responds automatically to certain Events. An Event is a message which satisfies the conditions specified in an Event definition within the rule.

An Automation Rule is always defined by an initial Event, called the "Root Event", and can also be defined by Events dependent on the Root Event. The Root Event is always linked to an Included Message Range of the Logical Console in which the Automation Rule is defined. The Included Message Range containing the Root Event is called its "owner". Actions are linked to the Root Event or to dependent Events.

Automation Rules are either simple or complex. A simple rule consists of a Root Event and the linked Action(s). A complex rule consists of an Event Tree with a Root Event, dependent (Non-Root) Events and the linked Actions. In a complex rule, Actions are not necessarily linked to each Event of the tree. The figure **Event Tree** is an example of a complex rule.

Automation Rules are always related to a Logical Console. This means that the Root Event of an Automation Rule can only be triggered by a message that appears in the Logical Console for which the Automation Rule is defined. On the other hand, Non-Root Events can be triggered by any message.

Authorization

User

An object of the type "User" identifies a user to Entire Event Management. Each user must be defined to Entire Event Management.

Profile

An object of the type "Profile" defines rights to access Entire Event Management objects relative to specific functions, Logical Consoles and Servers.

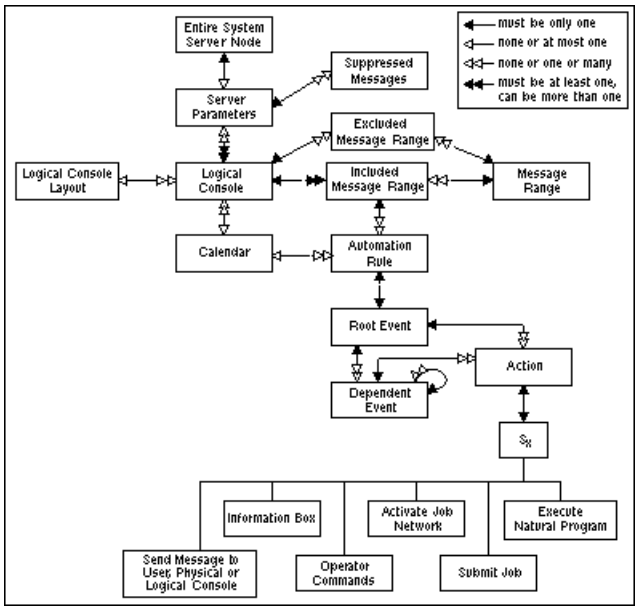
Each User defined to Entire Event Management can be assigned a Profile. A User without a Profile can use all functions and access any object. A User with a Profile is restricted to using those functions and objects allowed by that Profile.

Calendars

Calendars distinguish between working days and non-working days as specified by the user. By linking a Calendar to a Logical Console or an Automation Rule, the User can specify the days when the console or rule should be active.

Object Relationship Diagram for Environment and Automation

The following figure illustrates the relationship between the various types of Entire Event Management objects:



For Authorization Objects, see the object relationship diagram for authorization in the section *Authorization*.

6 User Interface

The user interface to Entire Event Management is an easy-to-use, menu-driven application. It guides the user through object definition and retrieval. When a user logs on to Entire Event Management, the Main Menu displays a list of the available options.

The following topics are covered below:

Entire Event Management Main Menu

```
 11:36:22              *** ENTIRE EVENT MANAGEMENT ***              15.08.93
 Srv    *                        - Main Menu -


    Console Services

     1  Logical Console
     2  Server


    Administration

     3  Environment
     4  Automation
     5  Authorization
     6  Calendars

     .  Exit
     ?  Help
     *  Commands

  Command ===> _____________________________________________________________
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help        Exit  Flip                                        Menu   ↩
```

- The **Console Services** options enable users to view the Logical Consoles and to oversee the activities of the various Servers.
- The **Administration** options enable users to define objects. Object definitions are stored in the Definition Database. Each option leads either to a submenu with further options or to a list of the defined objects of the selected type. From the list screen, the user can modify, display or delete an object of the selected type by entering a two-letter line command in the two-character field preceding the name of the object to be processed. An object can be added by pressing PF2 (Add) or by entering the direct command ADD in the command line.

Moving through Entire Event Management

There are several ways of moving through Entire Event Management:

- **Cursor selection:**
  Use the cursor to directly select options displayed on the screen.
- **Option code input:**
  Use the option code to bypass the menu hierarchy.
- **Command input:**
  Issue direct commands or two-character line commands to navigate and perform functions.
- **PF Keys**
  Navigate and issue commands by pressing PF keys.

Asterisk Selection Facility

Selection is made easy by the use of asterisks in direct commands and key fields to invoke selection lists of objects of the selected type. For example, entering an asterisk in the input field for an object name and pressing Enter displays a list of all available object names of the selected type. Selection lists can be narrowed down by entering a prefix followed by an asterisk. For example, entering ABC* and pressing Enter will display a list of all objects of the selected type whose names begin with ABC.

Online Help

The user interface includes a comprehensive online help system that provides help texts for screens and for input fields of screens. To display the help text for the current screen, press PF1 (Help) or enter a question mark (?) in any input field and press Enter. For an input field that accepts only certain values, enter an asterisk (*) in the field and press Enter. An active help window opens from which you can select a value.

ZOOM Options

A ZOOM function is available to display data for which there is insufficient space in the current screen. Options which can be ZOOMed are preceded by a greater than sign (>). The complete data for these options can be displayed by placing the cursor anywhere on the option heading with the > and pressing Enter. ZOOM options which contain data are preceded by a highlighted plus sign (+).

You will find more detailed information about how to move through Entire Event Management in the *Entire Event Management User's Guide*.