MAINVIEW® VistaPoint™ User Guide

Version 1.1.04

April 30, 2001

Restricted Rights Legend

BMC Software, Inc.
2101 CityWest Blvd.
Houston TX 77042-2827
USA

Contacting BMC Software

You can access the BMC Software Web site at **http://www.bmc.com**. From this Web site, you can obtain general information about the company, its products, special events, and career opportunities. For a complete list of all BMC Software offices and locations, go to **http://www.bmc.com/corporate/offices.html**.

USA and Canada

Address	BMC Software, Inc. 2101 CityWest Blvd. Houston TX 77042-2827
Telephone	713 918 8800 or 800 841 2031
Fax	713 918 8000

Outside USA and Canada

Telephone	(01) 713 918 8800
Fax	(01) 713 918 8000

Customer Support

You can obtain technical support by using Response Online™ (comprehensive information from the Web) or Response On Demand™. To expedite your inquiry, please see “Before Contacting BMC Software,” below.

Response Online

You can obtain technical support from BMC Software 24 hours a day, seven days a week by accessing the technical support Web site at **http://www.bmc.com/support.html**. From this site, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

Response On Demand

In the USA and Canada, if you need technical support and do not have access to the Web, call 800 537 1813. Outside the USA and Canada, please contact your local support center or your local sales office for assistance.

Before Contacting BMC Software

Before you contact BMC Software, have the following information available so that a technical support analyst can begin working on your problem immediately:

- product information
  - product name
  - product version (release number)
  - license number and password (trial or permanent)
- operating-system and environment information
  - machine type
  - operating system type, version, and service pack or program temporary fix (PTF)
  - system hardware configuration
  - serial numbers
  - related software (database, application, and communication) including type, version, and service pack or PTF
- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
  - product error messages
  - messages from the operating system, such as `file system full`
  - messages from related software

Contents

Figures

Tables

About This Book

MAINVIEW VistaPoint, or VistaPoint for short, is an application workload monitor. VistaPoint reports transaction performance of user-defined MVS, CICS, DB2, or IMS workloads. Workloads can be combined together to form applications that span products, regions, subsystems, and OS/390 images. VistaPoint provides a set of views that report application and workload performance with differing levels of detail.

VistaPoint works with at least one BMC Software MVS, CICS, or database-monitoring tool. MAINVIEW for CICS, DB2, IMS, and OS/390 work singularly or together with VistaPoint. These products monitor their respective targets to offer product-specific views that are integrated with VistaPoint and provide additional detail about a workload's transaction performance.

How This Book Is Organized

This book is organized as follows. In addition, a glossary of terms and an index are included at the end of the book.

Description	Chapter/Appendix
introduces VistaPoint; describes monitors, workloads, and applications	Chapter 1, “Introducing MAINVIEW VistaPoint”
describes the three different data collection periods: real time, interval, and session	Chapter 2, “Viewing Performance over Different Periods”
describes how to create workload definitions and incorporate them into VistaPoint applications	Chapter 3, “Creating VistaPoint Workloads”
describes MAINVIEW contexts. A context is a frame of reference for the data you display in views	Chapter 4, “Setting View Contexts”

describes several ways to manage view data	Chapter 5, “Managing View Data”
Describes how to use the MAINVIEW GraphManager facility to convert view data into charts	Chapter 6, “Creating MAINVIEW VistaPoint Charts”
Describes how to obtain historical performance reports through the submission of batch jobs	Chapter 7, “Generating and Managing Batch Reports”
Describes MAINVIEW conventions used to display numeric and graphic data in views	Appendix A, “View Conventions”
Describes the calculated values appearing in VistaPoint views and the views of the products that work with VistaPoint	Appendix B, “Calculated Values”

Recommended Reading

Occasionally, you may need more information about a topic described in this book. If you do, you should be able to find the information in *Getting Started with MAINVIEW VistaPoint*. If you want a quick overview of MAINVIEW commands, the *MAINVIEW Command List* lists them on a laminated card.

Related Reading

BMC Software products are supported by several types of documentation:

- online and printed books
- online Help
- release notes and other notices

In addition to this book and the online Help, you can find useful information in the publications listed in the following table. As “Online and Printed Books” on page xv explains, these publications are available on request from BMC Software.

Category	Document	Description
Installation documents	*OS/390 and z/OS Installer Guide*	provides instructions for installing and maintaining CPO- or SMP-packaged BMC Software products
	MAINVIEW Common Customization Guide	describes how to set up the operating environment for MAINVIEW products to your site’s requirements
	MAINVIEW Administration Guide	describes how to manage and maintain the operating environment for MAINVIEW products at your site

Category	Document	Description
User documents	*Using MAINVIEW*	describes how to use the common MAINVIEW interface
	Getting Started with MAINVIEW VistaPoint	presents a series of short exercises designed to introduce new users to MAINVIEW VistaPoint
Release documents	*MAINVIEW VistaPoint Release Notes*	describes the enhancements and fixes included in the current version of the product

Online and Printed Books

The books that accompany BMC Software products are available in online format and printed format. If you are a Windows or Unix user, you can view online books with Acrobat Reader from Adobe Systems. The reader is provided at no cost, as explained in "To Access Online Books." You can also obtain additional printed books from BMC Software, as explained in "To Request Additional Printed Books."

To Access Online Books

Online books are formatted as Portable Document Format (PDF) files. You can view them, print them, or copy them to your computer by using Acrobat Reader 3.0 or later. You can access online books from the documentation compact disc (CD) that accompanies your product or from the World Wide Web.

In some cases, installation of Acrobat Reader and downloading the online books is an optional part of the product-installation process. For information about downloading the free reader from the Web, go to the Adobe Systems site at **http://www.adobe.com**.

To view any online book that BMC Software offers, visit the support page of the BMC Software Web site at **http://www.bmc.com/support.html**. Log on and select a product to access the related documentation. (To log on, first-time users can request a user name and password by registering at the support page or by contacting a BMC Software sales representative.)

To Request Additional Printed Books

BMC Software provides printed books with your product order. To request additional books, go to **http://www.bmc.com/support.html.**

Online Help

The MAINVIEW VistaPoint product includes online Help. In the MAINVIEW VistaPoint ISPF interface, you can access Help by pressing **F1** from any ISPF panel or MAINVIEW window.

Release Notes and Other Notices

Printed release notes accompany each BMC Software product. Release notes provide current information such as

- updates to the installation instructions
- last-minute product information

In addition, BMC Software sometimes provides updated product information between releases (in the form of a flash or a technical bulletin, for example). The latest versions of the release notes and other notices are available on the Web at **http://www.bmc.com/support.html**.

Conventions

This book uses the following general conventions:

Item	Example
information that you are instructed to type	Type **SEARCH DB** in the designated field. Type **search db** in the designated field. (Unix)
specific (standard) keyboard key names	Press **Enter**.
field names, text on a panel	Type the appropriate entry in the **Command** field.
directories, file names, Web addresses	The BMC Software home page is at **www.bmc.com**.
nonspecific key names, option names	Use the HELP function key. KEEPDICTIONARY option
MVS calls, commands, control statements, keywords, parameters, reserved words	Use the SEARCH command to find a particular object. The product generates the SQL TABLE statement next.

Item	Example

code examples, syntax statements, system messages, screen text	`//STEPLIB DD`
	
	`The table` *`table_name`* `is not available.`

emphasized words, new terms, variables	The instructions that you give to the software are called *commands*. In this message, the variable *file_name* represents the file that caused the error.

This book uses the following types of special text:

Note: Notes contain important information that you should consider.

Warning! Warnings alert you to situations that could cause problems, such as loss of data, if you do not follow instructions carefully.

Tip: Tips contain useful information that may improve product performance or that may make procedures easier to follow.

Chapter 1 Introducing MAINVIEW VistaPoint

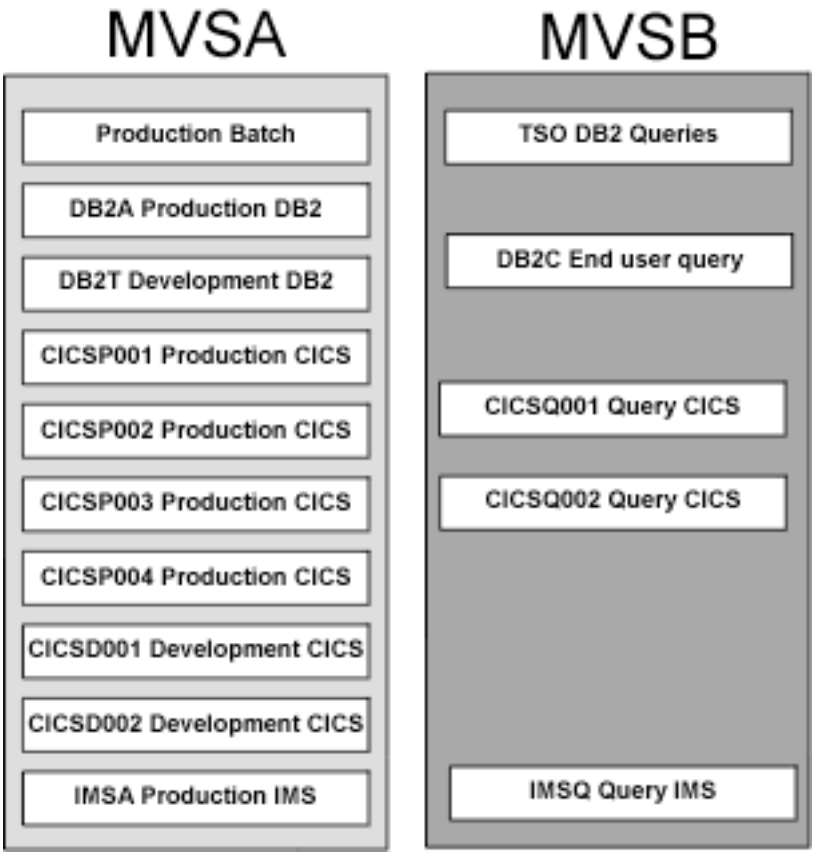
MAINVIEW VistaPoint is an application workload monitor. VistaPoint reports response-time performance of OS/390, CICS, DB2, and IMS transactions and jobs. VistaPoint regards a transaction as a measurable event with a defined start and end time.

You can monitor the performance of selected regions or subsystems by defining them as workloads with BMC Software performance tools for OS/390, CICS, DB2, or IMS. These products work singularly or together with VistaPoint to monitor their respective targets and provide workload transaction data to VistaPoint.

You can combine workloads into a VistaPoint application. By creating applications composed of related workloads, VistaPoint gives you the capability to display transaction performance across products, regions, subsystems, and OS/390 images in a single view.

Figure 1-1 shows a simple example of CICS regions and DB2 and IMS subsystems operating in two OS/390 images. Some regions and subsystems are devoted to production work and others run development tasks. OS/390 batch jobs are running in MVSA and TSO jobs are running in MVSB.

Figure 1-1 Mixture of Products Operating in Two MVS Images



In this example, you can create VistaPoint workloads that represent this site's CICS, DB2, IMS, or OS/390 environments in a single OS/390 image or across a sysplex. For example, sites running CICS tasks with embedded SQL statements can create CICS and DB2 workloads that can be combined together in an application. The ability to combine related workloads together gives you the capability to get a performance overview with VistaPoint and then shift to another BMC Software product for more detail about individual subsystem performance.

VistaPoint provides a set of views that report transaction performance with differing levels of detail. VistaPoint's views present the aggregate performance of workloads by the application of which they are a member. MAINVIEW for CICS, MAINVIEW for DB2, and MAINVIEW for IMS offer product-specific views that are integrated with VistaPoint. Each product's views provide additional detail about the performance of their targets, depending upon which monitors are active.

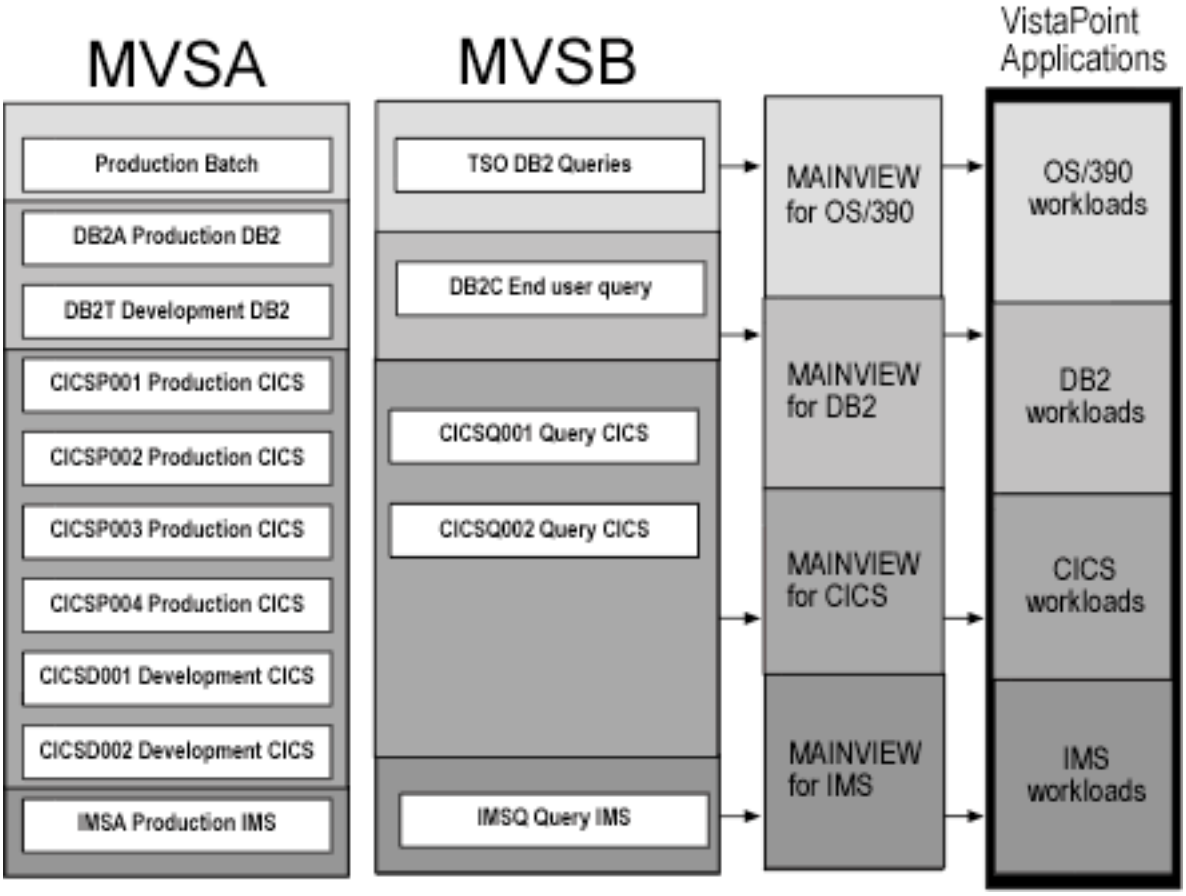
VistaPoint reports performance according to whether a workload's transactions meet a service-level objective composed of the following criteria:

- Transactions must complete within a predefined response-time goal set as the maximum for acceptable performance.
- A specified minimum percentage of the total reported transactions must complete within the response-time goal.

Integrating VistaPoint with Other MAINVIEW Products

Figure 1-2 shows BMC Software MAINVIEW products monitoring their respective regions or subsystems from the example shown in Figure 1-1. You create workloads for the transactions you want to monitor from each target. You can combine workloads into a common application based upon whether targets have shared tasks or resources. In this example, the DB2, CICS, IMS, and OS/390 production regions or subsystems are logical candidates to define as workloads and group together as an application.

Figure 1-2 Monitoring Transactions with Workloads and Applications



You can select the transactions to monitor from a product's targets and set the service-level objective in a definition for each workload. A single workload definition can be applied to multiple targets across multiple OS/390 images. Conversely, you can define multiple workloads for the same target.

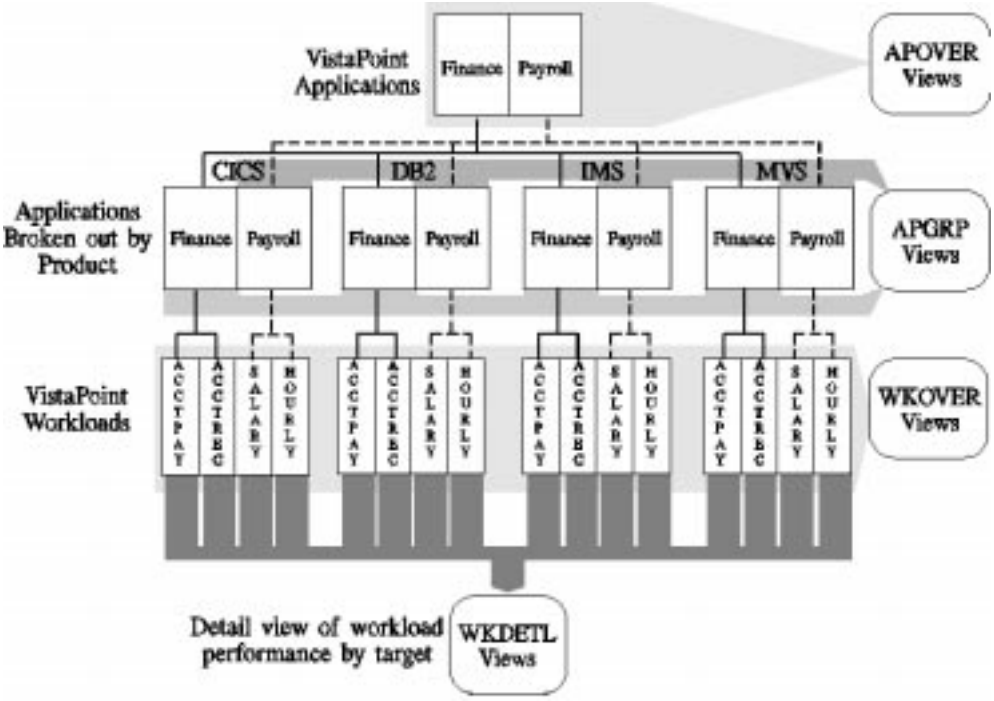
Each product's workloads become part of a common VistaPoint application by being associated to the same composite name in their workload definition. MAINVIEW for OS/390 workloads are incorporated into a VistaPoint application by creating a composite workload with the same name as the application.

As part of a workload definition, you set a daily time range to monitor the response time of transactions that occur in the targets included in the workload. Also, you have the choice of including queuing in the measured response time of CICS and IMS transactions.

Monitoring Applications with VistaPoint

Figure 1-3 illustrates a simple hierarchy of two VistaPoint applications: Finance and Payroll. Both applications are composed of CICS, DB2, IMS, and MVS workloads. Also, Figure 1-3 shows the corresponding VistaPoint view to see details of transaction performance associated with each level of the application and workload hierarchy.

Figure 1-3 VistaPoint Applications and Views



At the top are the Finance and Payroll applications composed of CICS, MVS, IMS, and DB2 workloads. The VistaPoint APOVER view, shown in Figure 1-4 on page 1-6, gives an overview of each application's transaction performance. The graph shows each application's combined performance with respect to the service-level objectives defined for the workloads in the application. The higher the percentage, the closer the application is to meeting or exceeding service-level objectives.

Figure 1-4 VistaPoint APOVER View

```
08NOV2000 08:28:18 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
>W2 =APOVER==========(PRODUCTN=*======)08NOV2000====08:28:02====MVVP==========4
CMD Appl           Interval %Obj  <    80%- 90%- >    Total Worst  Best   Avg
--- --------       0.....50...100 80%  90%  100% 100% Wklds %Obj   %Obj   Resp-
    PAYROLL  77.83 **********        1    5    2    1    10  67.35  92.09  4.54
    ALLBTCH  93.05 **********             2    3         5  88.45  97.23  1.75
    FINANCE  96.00 *************               1         1  96.00  96.00  2.00
    TSOFINAN 100.0 **************              1         1 100.0  100.0   0.00
```

To the right of the graph, APOVER shows the distribution of an application's workloads across the percentage range of the service-level objective. At a glance, you can see whether an application has any workloads that are performing poorly. In the case of the Payroll application shown in Figure 1-4, one workload is performing at less than 80%. The **Worst % Obj** field shows that this workload is meeting only 67.35% of its service-level objective.

You can *hyperlink* to another VistaPoint view to get more information by moving your cursor over a field that is shown under a highlighted column heading and pressing **Enter**. Hyperlinks are associations between views based upon the field that you select from the original view.

The four types of VistaPoint views are connected together with hyperlinks. VistaPoint has predefined hyperlinks that give a logical progression of views, each with more detailed information about the fields you select. The resulting view filters data and restricts the display to a specific application or workload from which you hyperlinked in the original view.

Beneath the application overview, Figure 1-3 on page 1-5 shows that the workloads are grouped by product: CICS, DB2, IMS, and MVS. The APGRP view, shown in Figure 1-5, shows the combined performance of each product's workloads that are part of an application. By default, the view presents the worst performing workloads in descending order.

Figure 1-5 VistaPoint APGRP View

```
08NOV2000 08:40:43 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =APGRP============(PRODUCTN=*=======)08NOV2000====08:40:42====MVVP==========4
CMD Appl     Type        Avg % Obj      Worst Best  Resp    Tran     Total
--- -------- ----        0.....50...100 % Obj % Obj Time--- Total--- Wklds
    PAYROLL  CICS  90.50 **********     77.3  97.7    2.75    11216     4
    PAYROLL  DB2   91.35 ***********    91.4  91.4    0.83     4395     1
    PAYROLL  MVS   93.22 ***********    92.2  92.2    1.45    12674     1
    PAYROLL  IMS   95.45 *************   95.5  95.5    3.75     3333     1
```

Figure 1-5 shows only the PAYROLL application. The other applications have been filtered out of the APGRP view by hyperlinking from APOVER. Hyperlinking is accomplished by moving the cursor directly over the **Avg % Obj** field of the row listing PAYROLL's performance and then pressing **Enter**.

The APGRP view **Total Wklds** field shows the distribution of product workloads in PAYROLL. IMS, MVS, and DB2 each have a single workload. All three workloads are above 90% of objective.

There are four CICS workloads in PAYROLL. Although the combined CICS workload performance is still above 90%, the **Worst Obj** field shows that one workload is at 77.3% of objective. You want to focus the remainder of VistaPoint views on the CICS workloads that belong to PAYROLL.

The WKOVER view gives a performance summary of a target's workloads that belong to VistaPoint applications. Each row of the WKOVER view represents the combined performance of all the workloads that belong to the listed target. In Figure 1-6, only CICS regions are shown because of the hyperlink from the APGRP view. Again, the hyperlink from APGRP restricts the data shown in WKOVER to a specific product.

Figure 1-6 VistaPoint WKOVER View

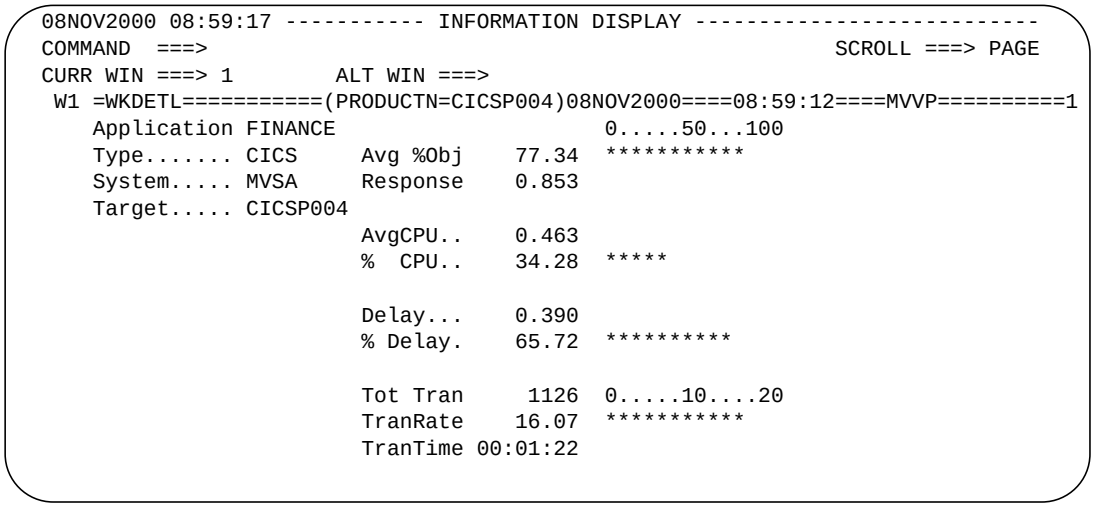
```
08NOV2000 13:20:43 ---------- INFORMATION DISPLAY --------------------------
 COMMAND  ===>                                                SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =WKOVERR==========(PRODUCTN=*=======)08NOV2000====13:20:31====MVVP=========17
 CMD Appl     Target   Type       Realtime %Obj  Resp     Avg     Tran     Tran
 --- -------- -------- ----       0.....50...100 Time---- CPU---- Total--- Rate-
     PAYROLL  CICSP004 CICS  77.18 ***********       1.55   0.42     1123     22
     PAYROLL  CICSP001 CICS  92.97 **************    0.82   0.09    45710    109
     PAYROLL  CICSP003 CICS  94.23 **************    0.75   0.06     3312     19
     PAYROLL  CICSP002 CICS  97.62 **************    0.29   0.02     6763     33
```

In Figure 1-6, the workload defined for region CICSP004 is performing worse than the other workloads. Although there is only a single workload shown for each region in this example, understand that WKOVER gives a summary of all workloads defined as part of the PAYROLL application. You can have more than one workload defined for each region.

Hyperlinking from WKOVER takes you to the WKDETL view, shown in Figure 1-7 on page 1-8. WKDETL expands individual line entries of WKOVER. WKDETL presents details of an application's transaction performance for a single product target.

The second column shown in Figure 1-7 displays the calculated statistical performance derived from transactions that occurred during the sampling interval. In this example, the worst average response time is due to delay time. At this point, you normally shift to the views of the MAINVIEW products that work with VistaPoint to get more information about the performance of a specific target.

Figure 1-7 VistaPoint WKDETL View



Notice the window information line immediately above the data fields in Figure 1-7. Region CICSP004 is displayed next to PRODUCTN. When you display the WKDETL view, you have narrowed the scope of VistaPoint view *context* to a single region or subsystem. You are now looking at the CICSP004 subset of data within the PRODUCTN context.

A context is a frame of reference that represents the products, MVS images, and targets whose transaction data appears in a VistaPoint view. In the examples so far, you have been in the PRODUCTN *single system image* context composed of MVS, CICS, DB2, and IMS workloads. A single system image gives you the ability to show performance combined from multiple targets, images, and products into a single view.

The *Using MAINVIEW* book explains how to set VistaPoint contexts. Just as you can hyperlink between VistaPoint views to get more information, you can shift contexts to expand or restrict the number of targets providing data displayed by a view.

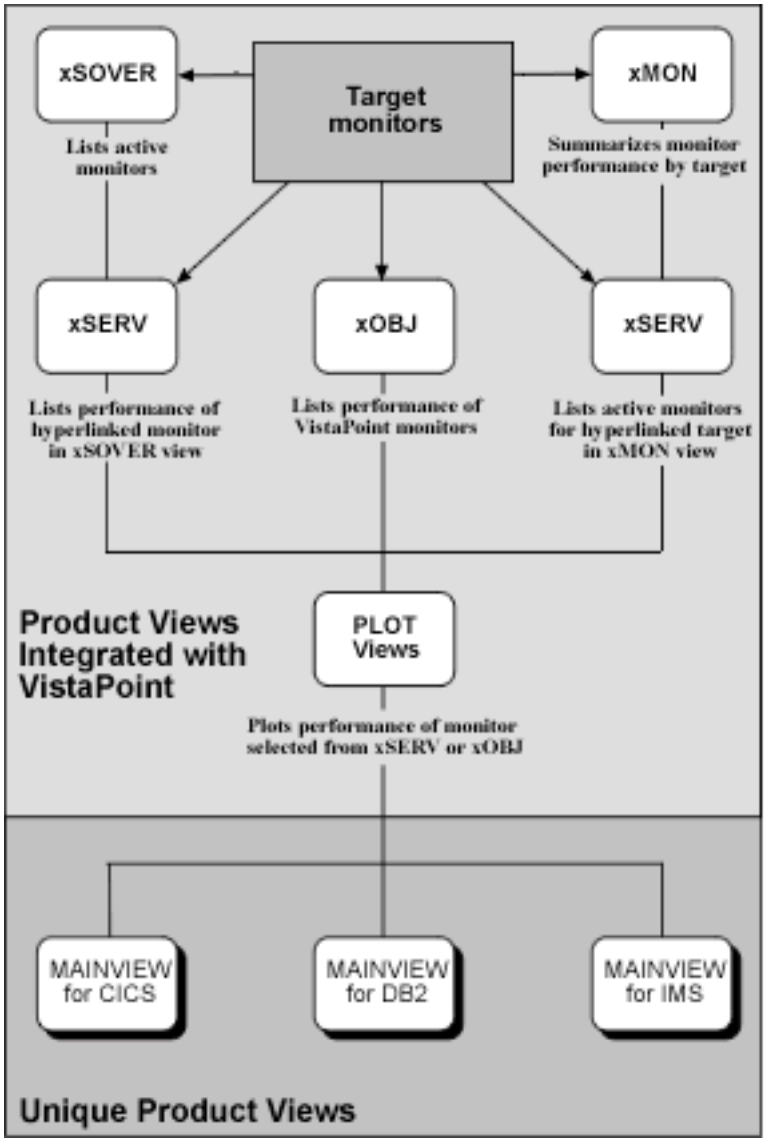
Viewing Workload Performance with Other MAINVIEW Products

MAINVIEW for CICS, DB2, and IMS provide additional views when they work with VistaPoint. Each product has a set of similar views. These views give you the capability to display performance data collected from multiple regions or subsystems across multiple OS/390 images.

The top shaded area of Figure 1-8 on page 1-10 shows the initial five views available to these MAINVIEW products when they work with VistaPoint. Based on the BMC software BBI-3 environment, these views give you the capability to present data gathered across a sysplex with multiple windows from a single screen. Normally, you shift to these views after using VistaPoint to narrow the data you want to display to a single product.

The bottom shaded area of Figure 1-8 represents the views that show data from a single region or subsystem for a particular product. These views use the BMC Software BBI-2 environment to present full-screen panel displays. After narrowing product problems to specific regions or subsystems with the integrated VistaPoint views, you shift to these views to get details about transactions.

Figure 1-8 Product Views Integrated with VistaPoint



Product views integrated with VistaPoint follow a common naming convention. The X prefix of each name identifies the product to which the view belongs by the first letter of the product name. For example, IMON is a MAINVIEW for IMS view. CMON and DMON are equivalent views of MAINVIEW for CICS and DB2, respectively.

In Figure 1-8, the views are arranged in order, top to bottom, from the most general to the most specific. As in the case of VistaPoint views, hyperlinks associate views together. Typically, each successive hyperlink displays another view that expands the information that you selected from the previous view.

Starting at the top of Figure 1-8, both the xSOVER and xMON views give a performance overview. Both views show the status of monitors collecting data from the workloads within your view's context. xSOVER lists all active monitors from the workloads within your view's context. xMON summarizes monitor performance of a single region or subsystem. You start with either view, depending on whether you have already narrowed your focus to a single target or not.

The xSOVER view, shown in Figure 1-9, summarizes the performance of similar monitors that are collecting data from all workloads within your view's context. The number of monitors of a specific type are shown in the **Count** field at the right of Figure 1-9.

Figure 1-9 xSOVER View

```
08NOV2000 13:47:31 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =DSOVER============DB2P=====*========08NOV2000====13:47:31====MVDB2========26
CMD Serv  Parm           Avg % Warning  Avg      Avg      Area Count
--- ----- --------       0.....50...100 Value--  Warning- ---- -----
    DSUT  ECDSA     94.4 *************     85.00    90.00 STOR     3
    DSUT  CDSA      64.4 *********         58.00    90.00 STOR     4
    @ELTD FINANWRK  57.4 ********          60.05     0.20 WKLD     2
    @ELTD DB2PTEST  48.2 *******           45.05     0.30 WKLD     1
    @ELTD TOUGH2DO  40.9 *******           40.05     0.50 WKLD     4
    MXTC            27.1 ****              25.71    80.00 TASK     1
    @RSTM PAYROLL    5.5 *                  5.05     1.00 WKLD     4
    @RSTM ACCTRET    5.5 *                  5.05     1.00 WKLD     3
    @RSTM ACCTRECV   5.0 *                  5.05     1.00 WKLD     2
```

The graph shown in Figure 1-9 displays how close each monitor's average performance is to the warning threshold set for the monitor. The higher the percentage, the closer the monitor's data is to meeting or exceeding its warning threshold.

The xMON view, shown in Figure 1-10 on page 1-12, summarizes monitor performance by target. The two graphs shown in Figure 1-10 show the number of monitors that are exceeding their warning threshold and the average percent warning value for all monitors in the listed target.

Figure 1-10 xMON View

```
08NOV2000 13:48:47 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =DMON=============(PRODUCTN=*=======)08NOV2000====13:48:47====MVDB2=========2
CMD Target   Actv     Number in Warn        Avg % Warning  Maximum
--- -------- Mntrs    0.....10....20        0.....50...100 % Warning
    DB2A         6  0                  32.5 *****               32.5
    DB2T         7  3 **               30.5 ****              2170.5
```

The xSERV view, shown in Figure 1-11, is the default view you see after hyperlinking from xSOVER or xMON. The xSERV presents monitor data from different sources, depending upon whether you hyperlinked from xSOVER or xMON. In the case of the hyperlink from xSOVER, xSERV expands the summary performance of the monitor you selected.

Figure 1-11 shows the xSERV view that results from a hyperlink from the xMON view. In this case, the xSERV view lists all monitors that are active in the target you selected from xMON.

Figure 1-11 xSERV View

```
08NOV2000 13:51:28 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =DMON=====DSERV===(PRODUCTN=*=======)08NOV2000====13:51:03====MVDB2========13
CMD Serv  Parm             % Warning    Curr     Warn     Area Target
--- ----- --------       0.....50...100 Value--- Value--- ---- --------
    @ELTD JEBA08   426.8 *************+     2.13     0.50 WKLD DB2T
    @ELTD WORK2    426.8 *************+     2.13     0.50 WKLD DB2T
    @ELTD ACCTRECV 213.4 *************+     2.13     1.00 WKLD DB2T
    DSOPN           32.5 *****              65.00   200.00 DSYS DB2A
    DSOPN           30.5 ****               61.00   200.00 DSYS DB2T
    SQLAC            0.0                     1.47  3000.00 USER DB2T
    SQLAC            0.0                     0.00  3000.00 USER DB2A
    COMSY                                    0.08          USER DB2T
    THDAC                                    4.00          USER DB2T
    COMSY                                    0.02          USER DB2A
    @ELTM DB2AWORK                           0.00     0.50 WKLD DB2A
    @ELTM JEBA08                             0.00     0.50 WKLD DB2A
    THDAC                                    3.00          USER DB2A
```

Although the xSERV view resulting from a hyperlink from xSOVER is not shown, it presents the same format and data shown in Figure 1-11. This example shows the xSERV view resulting from a hyperlink from xMON.

Each line represents the performance of a single monitor named in the **Serv** field at the left of the view. The **Parm** field immediately to the right of the **Serv** field displays the name of the VistaPoint workload or a specific monitor parameter.

Looking at Figure 1-8 on page 1-10 again, notice the xOBJ views that seem to be out of the normal hyperlink pathways of other product views. The xOBJ view is a specialized xSERV view. It restricts monitor performance to the monitors VistaPoint uses to measure workload performance.

Figure 1-12 COBJ View

```
08NOV2000 13:53:31 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =COBJ==============CICSP01==*========08NOV2000====13:53:31====MVCICS=======10
CMD Workload Avg           % <= Resp Goal Resp  Goal Tran  Composite Target
--- Name---- Resp---       0.....50...100 Goal-    % Count Workload- --------
    ACCTRECV   2.001  35.7 *****           0.85   95   163 FINANCE   CICSP01
    PAYROLL    1.567  74.3 ************    1.00   90    52 PAYROLL   CICSP01
    ACCTPAY    0.596  88.8 *************   0.85   95   222 FINANCE   CICSP01
    CICSP01T   1.588  98.4 **************  2.00   90   410 TESTP01   CICSP01
    BILLING    0.877 105.7 **************+ 1.00   90   662 PAYROLL   CICSP01
```

COBJ is a hyperlink destination from the VistaPoint WKOVER view. This view extends VistaPoint views to a product's workloads. COBJC shows transaction performance of individual CICS workloads.

The **% Resp** field shows the percentage of transactions completing within the response-time goal set in the workload definition. The higher the percentage, the larger the number of transactions that are completing within the response-time limit.

If you look at Figure 1-8 on page 1-10 again, you see that both hyperlink paths and the xOBJ view converge at the plot views integrated with VistaPoint. A plot view displays a 10-line graph of monitored performance over the previous 10 data-collection periods.

Figure 1-13 on page 1-14 shows a graph of a DB2 elapsed time monitor (@ELTD). Looking at the fields shown in the right column of Figure 1-13, you can see that his plot shows the elapsed time of transactions occurring in the WORK2 workload that is part of the PAYROLL application. Each line represents the average elapsed time of transactions in one-minute data-collection intervals.

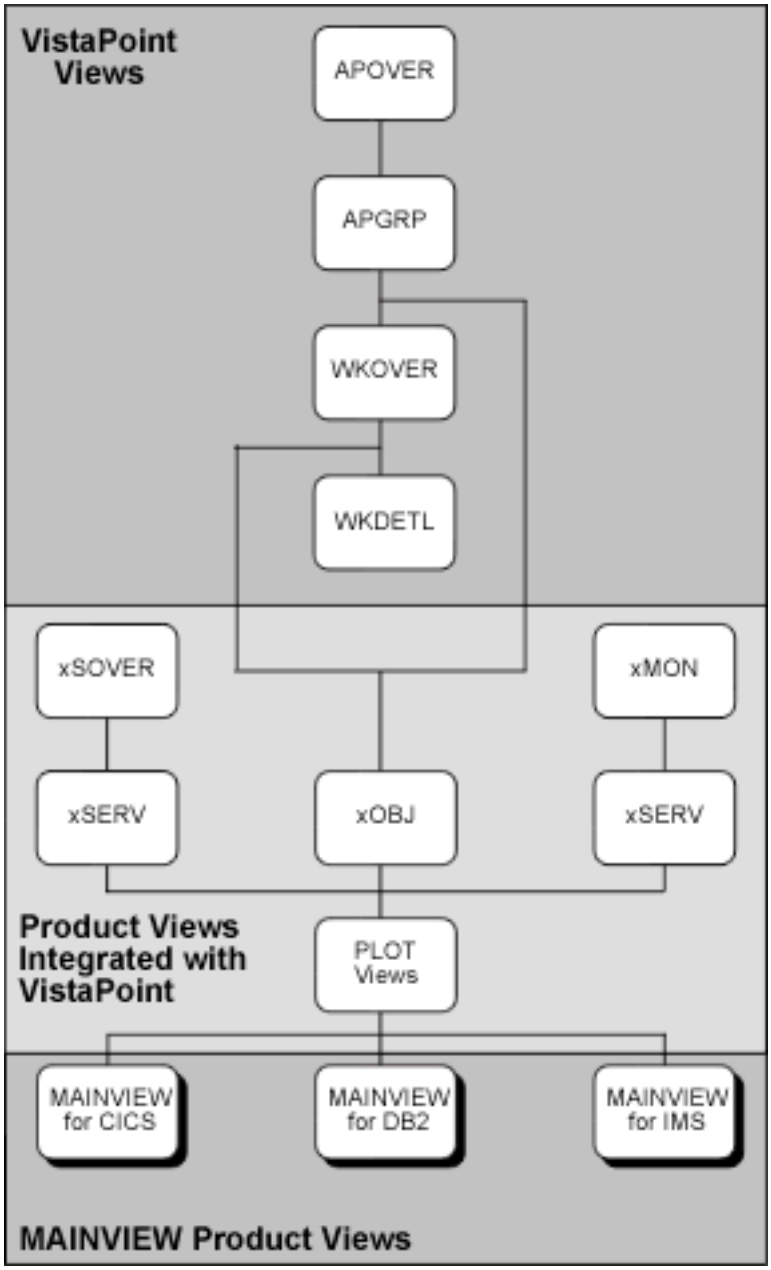
Figure 1-13 Plot View

```
08NOV2000 14:47:28 ----------- INFORMATION DISPLAY --------------------------
 COMMAND  ===>                                                SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =DMON=====@ELTD==(PRODUCTN=DB2T====)08NOV2000====14:46:04====MVDB2=========1
    >>USERS   |.....|......|  Response  Events Parm..... WORK2
    14:46:00  **********         5.44      8  Appl..... PAYROLL
    14:45:00  ***                1.48     16  Warning..     0.50
    14:44:00  **********         5.78      6  Max/Min..  Maximum
    14:43:00  ****               2.24     19   Value...    68.87
    14:42:00  **                 1.28     18   Time.... 14:22:00
    14:41:00  *                  0.63      7  Graph Max     7.19
    14:40:00  *****              2.68      4  Target... DB2T
    14:39:00  ***********        6.11     20  Descript. work only on db2l
    14:38:00  ******             2.97     10
    14:37:00  *                  0.68      9  Samples..       60
              |.....|......|  Response  Events Period... 00:10:00
    Total     **************       7.19    498  Samp Int. 00:01:00
    Prev Pd   *********            4.43     91  Start.... 13:46:00
    Curr Pd   ******               2.91    117  Elapsed.. 01:00:00
```

A hyperlink field is located in the upper left corner of plot views . In the plot view shown in Figure 1-13, the **USERS** field links you to MAINVIEW for DB2 users view. When you use this hyperlink, you can look at individual product views. The hyperlink to individual product views lets you use BMC Software MAINVIEW products to get the most detailed information about your target's performance.

Figure 1-14 on page 1-15 summarizes a typical sequence of views that you might use to monitor the performance of the regions or subsystems at your site. You begin with VistaPoint to identify the product and possibly the target that needs attention by examining the performance of transaction response times. Starting with the APOVER view, each successive view narrows the focus of transaction performance until you have reached the WKDETL view. At this point, you should have a good idea about which target and product are experiencing problems.

Figure 1-14 Typical Diagnostic Sequence of VistaPoint Views



After you tentatively identify the product and target that are experiencing problems, you shift to the product views integrated with VistaPoint. These views allow you to identify the source of the problem by the type of monitors that are exceeding their warning thresholds. Again, you follow a sequence of views until you reach the plot views. When you have reached a plot view, you should be able to identify the product, target, and type of problem you are experiencing. Finally, you hyperlink to individual product views to find the source of a problem.

Chapters 2 and 3 of the book expand on the introductory information given in this chapter. Chapter 2, “Viewing Performance over Different Periods” explains the common features of the MAINVIEW window interface used by VistaPoint. Chapter 3, “Creating VistaPoint Workloads” explains the different classes of VistaPoint views.

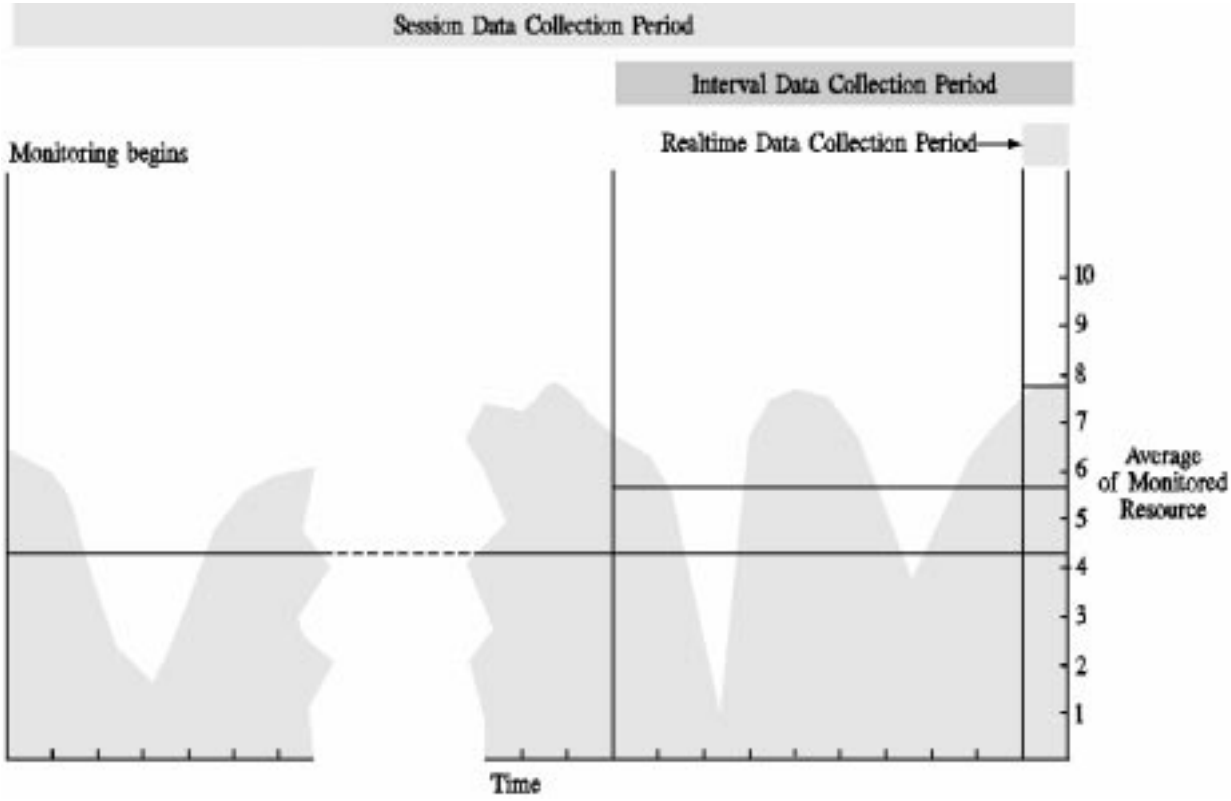
Chapter 2 Viewing Performance over Different Periods

VistaPoint, and the products that work with it, displays current data from three different data-collection periods: real time, interval, and session. Each period combines performance data collected over distinct time intervals. VistaPoint, and the products that work with it, provides a set of views that show the performance of monitored values over these time periods.

This chapter discusses real time, interval, and session views. Tables list the views of VistaPoint and the products that work with it for each data-collection period. A separate section discusses how to display historical data in VistaPoint or other product views.

Figure 2-1 on page 2 shows an example of a performance parameter that is monitored over time. At any instant, the value of the monitored parameter depends on the sampling rate and whether its data represents a count, an average, a rate, or condition.

Figure 2-1 MAINVIEW Data Collection Periods



The top of Figure 2-1 shows real-time, interval, and session data-collection periods. As expected, the real-time data-collection period presents data monitored over the most recent period, typically the last 60 to 90 seconds. An interval data-collection period represents collection over a longer period of time, typically up to 15 minutes. A session data-collection period represents data collected over the entire span of time that a value is monitored.

The horizontal lines represent the average values over the data-collection periods. Notice the difference between real-time, interval, and session averages. Each data-collection period summarizes the performance of the monitored value within that time frame. Figure 2-2 on page 5 shows how cluster views can be used to get an overview of performance across all three periods from a single view.

Note: The actual duration for the interval collection period is determined by the IRRI parameter in BBIISP00. The default value is 15 minutes. The data is reset every 15 minutes on the quarter-hour. BBIISP00 is located in the BBPARM PDS. The duration period for real time data collection is equal to the IRRI value divided by 10. By default, then, it equals 1.5 minutes. Session data is reset every twenty-four hours at midnight.

Normally, you shift between views that display data collected over different time periods when you need to see the performance of your applications or workloads from a different perspective. If you want to see the current performance of a workload or application, real-time views present data that represents near instantaneous performance. Interval views represent data collected over the near-term, typically 15 minutes. Session views give you the entire breadth of performance, representing all performance data collected since monitoring began. The length of a session depends on the time range specified to monitor the performance parameter.

Viewing Immediate Performance with Real-time Views

A real-time period presents the most immediate workload data. Typically, a real-time period is 90 seconds long. VistaPoint and other product views present data taken from the last completed real-time period.

The following two tables show real-time views for VistaPoint and other MAINVIEW products.

Table 2-1 MAINVIEW Product Real-time Views

Product	Monitor Overview	Target Overview	Objective Monitor	Monitor Detail
MAINVIEW for CICS	CSOVERR	CMONR	COBJR	CSERVR
MAINVIEW for DB2	DSOVERR	DMONR	DOBJR	DSERVR
MAINVIEW for IMS	ISOVERR	IMONR	IOBJR	ISERVR

Table 2-2 VistaPoint Real-time Views

Application Overview	Application by Product	Workload Overview	Workload Detail
APOVERR	APGRPR	WKOVERR	WKDETLR

Viewing Near-Term Performance with Interval Views

Interval data presents near-term performance of your workloads. Typically, the interval is 15 minutes. But VistaPoint and other MAINVIEW products display data from the *current interval*, which can be shorter than the entire interval length.

If you look at Figure 2-1 on page 2 again, you can see an interval is composed of periodic segments. However, the data you see in a view depends on the *current interval*. A current interval represents the cumulative period in the active interval when a view query requests data. Interval performance is based on the data from the completed portion of the interval at the time of a view query.

The following tables list interval views for VistaPoint and the products that work with it.

Table 2-3 MAINVIEW Product Interval Views

Product	Monitor Overview	Target Overview	Objective Monitor	Monitor Detail
MAINVIEW for CICS	CSOVER	CMON	COBJ	CSERV
MAINVIEW for DB2	DSOVER	DMON	DOBJ	DSERV
MAINVIEW for IMS	ISOVER	IMON	IOBJ	ISERV

Table 2-4 VistaPoint Interval Views

Application Overview	Application by Product	Workload Overview	Workload Detail
APOVER	APGRP	WKOVER	WKDETL

Viewing Entire Performance with Session Views

A session represents the current length of time that a monitor has been actively collecting data. When you look at a session view, you are seeing a performance summary of your workloads over the entire period that a target's transactions have been monitored.

The following tables list session views for VistaPoint and the products that work with it.

Table 2-5 MAINVIEW Product Session Views

Product	Monitor Overview	Target Overview	Objective Monitor	Monitor Detail
MAINVIEW for CICS	CSOVERS	CMONS	COBJS	CSERVS
MAINVIEW for DB2	DSOVERS	DMONS	DOBJS	DSERVS
MAINVIEW for IMS	ISOVERS	IMONS	IOBJS	ISERVS

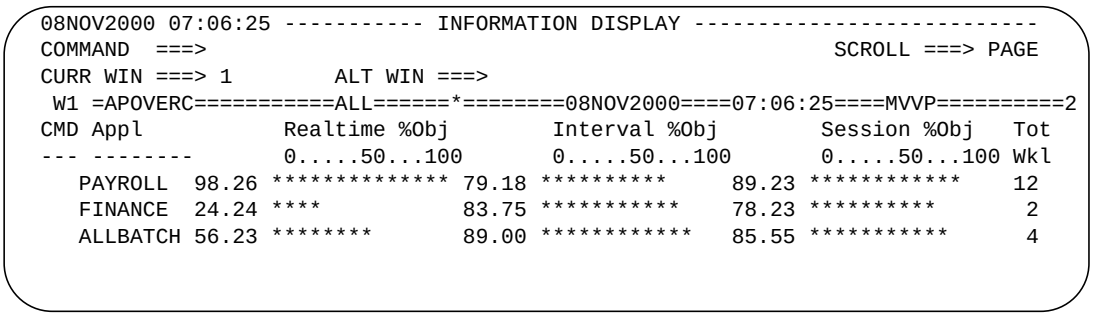
Table 2-6 VistaPoint Session Views

Application Overview	Application by Product	Workload Overview	Workload Detail
APOVERS	APGRPS	WKOVERS	WKDETLS

Getting an Overview with Cluster Views

Figure 2-2 shows the APOVERC cluster view. A cluster view summarizes an application's real time, interval, and session performance with three graphs.

Figure 2-2 APOVERC Cluster View



Cluster views present a summary of an application's performance over the three data-collection periods that can be shown in VistaPoint views.

The following tables list cluster views for VistaPoint and the products that work with it.

Table 2-7 MAINVIEW Product Cluster Views

Product	Monitor Overview	Target Overview	Objective Monitor	Monitor Detail
MAINVIEW for CICS	CSOVERC	CMONC	None	CSERVC
MAINVIEW for DB2	DSOVERC	DMONC	None	DSERVC
MAINVIEW for IMS	ISOVERC	IMONC	None	ISERVS

Table 2-8 VistaPoint Cluster Views

Application Overview	Application by Product	Workload Overview	Workload Detail
APOVERC	APGRPC	WKOVERC	WKDETLC

Displaying Historical Data

Interval records are retrieved from historical data sets and displayed in any of the views listed in the previous tables. Historical data gives you the capability to compare current performance to historical performance collected on different dates and times.

You can select historical records by date, time, and duration with the MAINVIEW TIME command. *Getting Started with MAINVIEW VistaPoint* includes a tutorial that explains how to use the TIME command.

The syntax of the TIME command is shown in the following example.

```
TIME date time [duration|NEXT|PREV]
```

You can enter the date and ending time of the historical data you want to display in VistaPoint views. You must enter the date and time in the same format shown on the window information line of the window that will display the historical view.

The duration is the length of time that you want to collect data from a historical data set. A duration ends at the interval that contains the time specified by the TIME command.

The syntax for the duration parameter is *nnnnu*, where:

nnnn indicates the number of hours, minutes, or intervals in the duration.

u indicates the unit of time:

I	intervals
M	minutes
H	hours interval
=	keeps the duration at its current value.

The NEXT and PREV parameters display historical intervals immediately after and before the interval you initially selected with the TIME command.

If you prefer to work with MAINVIEW dialog boxes rather than commands and parameters, simply enter **TIME** from any Command line. Figure 2-3 shows the Set Time Frame dialog box displayed after entering the TIME command without parameters.

Figure 2-3 MAINVIEW Set Time Frame Dialog Box

```
-------------------------------- SET TIME FRAME --------------------------
COMMAND  ===>


  Requested Time Frame:

    End Date ===> *           (*, =, or ddmmmyy)
    End Time ===> *           (*, =, or hh:mm)
    Duration ===> 1I          (*, =, nnnnI, nnnnM, nnnnH, NEXT, or PREV)


  Data in the Requested Time Frame:

    Interval ===> 10M         (Length, in minutes, of one interval)
    End Date ===> 08NOV2000   (End date of data)
    End Time ===> 09:07       (End time of data)
    Duration ===> 10M         (Minutes spanned by data)

  Type END to set the window's requested time frame
       CANcel to quit without setting
```

By completing the fields of the Set Time Frame dialog box, you select the ending date, time, and duration of the historical data you want to display in your views just as you do with the TIME command.

Managing Historical Data Sets

You will receive an error message if you request a view with data that is outside the time or date range stored on any of the historical data sets. The DSLIST view shows allocated historical data sets. Take a look at DSLIST when you are unsure of the range of data available for the historical views that you want to display. DSLIST displays the date and time range of data stored on each data set. The current status of the data set is also shown.

Note: You must be in a target context to see the status of historical data sets with DSLIST. Figure 2-4 shows the DSLIST view with a list of current historical data sets. In this example, the window information line shows that VistaPoint is the active product and CICSP01 the current context when the DSLIST command was issued.

Figure 2-4 DSLIST View Listing Historical Data Sets

```
 08NOV2000---14:38:16------------ INFORMATION DISPLAY -----------------------
 COMMAND  ===>                                                SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
>W1=DSLIST==============CICSP01==*======08NOV2000====14:38:17======MVVP=======4===
 C DDNAME   From Date  Time  To Date    Time  Rec Status Pending  Data set name
 - ------   ---------- ----- ---------- ----- --- ------ -------  --------------
   HISTDS00 06NOV2000 18:00 06NOV2000   08:00 Yes Closed ******** BBM20.LGS1
   HISTDS24 07NOV2000 08:00 07NOV2000   18:00 Yes Closed ******** BBM20.LGS1
   HISTDS02 07NOV2000 18:30 07NOV2000   08:00 Yes Closed ******** BBM20.LGS1
   HISTDS03 08NOV2000 08:00 08NOV2000   14:30 Yes Active ******** BBM20.LGS1
```

You can manage your historical data sets with actions available from the DSLIST view. DSLIST provides two types of actions:

- Command line actions

ADD	Adds a new historical data set.
SAVE	Saves current historical data set configuration to the PARMFILE data set.

- Line actions

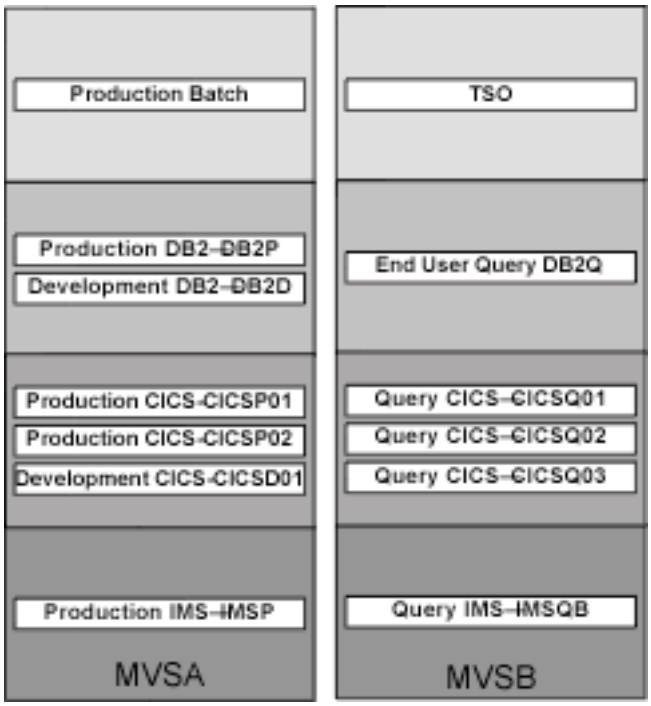
D (DEALLOCATE)	Deallocates an historical data set.
F (FORCE)	Selects the next data set for recording.
O (ORDER)	Toggles a data set's recording eligibility to Yes or No.
Q (QUIESCE)	Restricts read/write access to a data set.
R (EMPTY)	Empties an historical data set.
U (UNQUIESCE)	Reinstates read/write access to an historical data set.

Chapter 3 Creating VistaPoint Workloads

This chapter describes how to create workload definitions and incorporate them into VistaPoint applications.

Figure 3-1 on page 2 shows a transaction environment composed of OS/390, CICS, DB2, and IMS regions and subsystems running on two OS/390 images. CICS and DB2 have production and development work running on MVSA. Query work is done on MVSB for CICS, DB2, and IMS. OS/390 production batch and TSO are split between the two images.

Figure 3-1 CICS, DB2, and IMS Running on Two MVS Images



In this example, you have a heterogeneous integrated transaction environment. Transactions no longer occur within the domain of a single product. Typically, DB2 is a server to CICS and IMS client transactions. An accurate performance assessment in this environment requires the ability to combine performance from separate products performing shared tasks. Also, you need the ability to combine related performance occurring simultaneously in separate MVS images.

BMC Software MAINVIEW products that work with VistaPoint provide the capability to monitor transaction performance for the IBM products shown in Figure 3-1. MAINVIEW for CICS, DB2, IMS, and OS/390 individually monitor their target's transaction response times.

You build workload definitions that describe how you want these products to monitor their respective targets. In an ISPF dialog, you fill in fields with the values that you want for your workload definition.

As part of the definition, you assign a composite name to which a workload belongs. VistaPoint combines performance from similarly named composite workloads into a common application. VistaPoint presents application views that allow you to see the combined performance of workloads sharing common tasks.

Defining Workloads

MAINVIEW for CICS, DB2, and IMS have individual workload definition dialogs to create VistaPoint workloads. This section explains the overall process of creating workload definitions.

MAINVIEW for OS/390 composite workloads are combined in a VistaPoint application without completing a dialog. See “Defining OS/390 Workloads” on page 3-15 for more details.

The following steps outline the general procedure to create VistaPoint workloads.

1. Plan the workload definition.
2. Move to the product workload definition dialog.
3. Obtain an edit lock on the UBBPARM data set.
4. Complete the workload definition dialog.
5. Save the workload definition.
6. Install the workload.

In the following sections, each of these steps is explained in more detail.

Planning Workload Definitions

Before you define workloads, BMC Software recommends that you spend time establishing some conventions when using VistaPoint. This section explains several considerations that you should account for when you define workloads.

Place Workload Definitions in a Common UBBPARM Data Set

With the exception of MAINVIEW for OS/390, each product that works with VistaPoint stores its workload definitions in separate members of the UBBPARM data set. MAINVIEW for CICS workload definitions are saved in member BBKTWK00. MAINVIEW for DB2 and IMS workload definitions are stored in members BBPTWK00 and BBFTWK00, respectively.

BMC Software recommends that workload definition members be placed in a common BBIPARM library accessible to all PASs. The number of workload definitions can be kept to a minimum if they are placed in a shared parameter library. Otherwise, duplicate workload definitions must be created for each linked PAS to ensure VistaPoint applications report performance from all targets. Also, maintenance is easier if updates can be made to a single workload definition without the requirement of making simultaneous changes to similar definitions in different parameter libraries.

Establish Consistent Naming Conventions

BMC Software recommends consistent names for VistaPoint workloads and applications. This is particularly important if your site cannot place workload definitions in a common BBIPARM data set. In this case, you must be able to easily identify similar workloads in different BBIPARM data sets when you update your definitions.

VistaPoint, and the products that work with it, can sort and filter views by workload and application name. Consistent names make sorting and filtering views much easier to find the data you want to display.

Workload names should be a shorthand notation that represents the work performed by the target. For example, CICSPAY1 identifies CICS payroll transactions running on region 1.

Note: A workload name can be a maximum of eight characters.

Application names should indicate a common functional characteristic of the workloads that are part of an application. For example, a PROD suffix to an application name indicates the workloads are monitoring production regions or subsystems. Likewise, an application named FINANCE indicates that a workload's targets are completing financial transactions.

Note: Application names can be a maximum of eight characters.

Identify Critical Workload-Monitoring Periods

VistaPoint monitoring occurs over a daily cycle. During the course of a day, there are periods when transaction performance is not at risk because of the relatively light load on the system. During those periods, you may choose not to monitor your VistaPoint workloads.

Other daily periods are critical. The demands on system resources lead to contention and, invariably, transaction delays. BMC Software recommends that you monitor all of your workloads during your site's daily periods when adequate transaction performance is essential.

Workloads belonging to the same application should have identical monitoring periods. Because VistaPoint views show combined workload performance in applications, having different monitoring periods may result in some of an application's workloads becoming inactive. When this occurs, you must be aware of the fact that different monitoring periods reflect differences in the workload composition of the application.

Moving to the Product Workload Definition Views

There are two basic ways to start the workload definition dialog: starting from the product for which you want to define a workload, or through VistaPoint. VistaPoint lets you start from a MAINVIEW SSI context of All and proceed through a series of views until you have selected a product target. When you have reached the workload definition list, the context shifts to the individual target context. This method lets you select targets within your current SSI context for which you want to define a workload.

Proceeding directly to the workload definition list is the preferred method when you are already within a target context. Both methods of moving to the workload definition list are explained in this section.

The following screens show the view sequence that you follow to select a product target from VistaPoint. In Figure 3-2, the window information line identifies the VistaPoint MAIN view within an SSI context called PRODUCTN. You cannot define workload definitions within an SSI context.

Figure 3-2 VistaPoint MAIN View

```
08NOV2000 09:49:10 ----------- INFORMATION DISPLAY ---------------------------
COMMAND  ===>                                                 SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =MAIN=============(ALL=*=======)08NOV2000====09:49:09====MVVP==========6
CMD View Name  Description
--- ---------- ------------------------------
    Admin      Administrative Views
    Cluster    Trend Application views
    Ezvista    VistaPoint Easy Menus
    Interval   Interval Application views
    Realtime   Realtime Application views
    Session    Session Application views
    Transact   CICS, DB2, IMS Monitor Summary
    User       User created views
```

Selecting the **Admin** option takes you to the ADMIN where you can select one of the administrative views. You choose this option if you need to define product (CICS, DB2, IMS, MVS) workloads that belong to VistaPoint applications. Also, the Admin view provides options for defining VistaPoint security and displaying the status of historical data sets.

Figure 3-3 shows the ADMIN view. The top two options are views that list the products whose performance can be monitored with VistaPoint workloads.

Figure 3-3 VistaPoint ADMIN View

```
08NOV00 04:18:18 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =ADMIN============(ALL=*=======)08NOV2000====04:18:18====MVVP==========4
CMD View Name  Description
--- ---------- ------------------------------
    APPLDEF    Application Definition
    APPLDEFZ   Application Defn Summary
    DSLIST     List Historical data sets
    SECURITY   Security Views
```

Figure 3-4 shows the APPLDEFZ view.

Figure 3-4 APPLDEFZ view

```
08NOV2000 08:50:51 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
W1 =APPLDEFZ============(ALL=*=======)08NOV2000====04:18:18====MVVP==========4
CMD Area       Attatched Description
--- ---------- Targets-----------------------------
    CICS             6  Application definition for CICS
    DB2              3  Application definition for DB2
    IMS              2  Application definition for IMS
    MVS              2  Application definition for MVS
```

Figure 3-5 on page 3-7 shows active CICS targets from the APPLDEF view. This view expands the product you selected from the previous APPLDEFZ view. APPLDEF shows individual region names and the ID of the MVS system in which they operate. You are still within the VistaPoint SSI ALL context

Figure 3-5 APPLDEF View - CICS Regions and SSI Targets

```
08NOV2000 08:50:57 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1 =APPLDEFZ=APPLDEF=(ALL=*=======)08NOV2000====08:50:45====MVVP==========6
CMD Area     SSI      SSI      Description
--- -------- Target   System   ------------
    CICS     CICSP01  MVSA     Application definition for CICS
    CICS     CICSP02  MVSA     Application definition for CICS
    CICS     CICSD01  MVSA     Application definition for CICS
    CICS     CICSQ01  MVSB     Application definition for CICS
    CICS     CICSQ02  MVSB     Application definition for CICS
    CICS     CICSQ03  MVSB     Application definition for CICS
```

Selecting a target by hyperlinking on an item listed beneath the Area or SSI Target fields takes you to the MAINVIEW for CICS CWKLDDEF view shown in Figure 3-6. You are now in a target context and are able to define a workload.

Figure 3-6 MAINVIEW for CICS CWKLDDEF View - Current Workload Definitions

```
08NOV2000 09:51:14 ----------- INFORMATION DISPLAY --------------------------
COMMAND  ===>                                                SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
>W1 =CWKLDDEF==========CICSP01==*========(00 BROWSE        )====MVCICS=======25
CMD Workload Composite Target   System   Description               Sta Resp %Tr
--- Name---- Name----- -------- -------- ------------              --- ---- ---
    ACCTPAY  FINANCE   CICSP*   *        Accounts payable          Act 0.30  90
    ACCTRECV FINANCE   CICSP*   *        Accounts receivable       Act 0.30  90
    ACCTRET  FINANCE   CICSP*   *        Accounts return           Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.    Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing             Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions     Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01      Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing              Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans.  Act 1.00  95
```

MAINVIEW for DB2 and IMS have similar WKLDDEF views. For all three products, this view lists current workload definitions.

Providing that you are in a target context, moving to the WKLDDEF view from a product is simple. You can enter **xWKLDDEF** on the COMMAND line within any product view and move directly to the xWKLDDEF view itself.

Completing the Workload Definition Dialog

When you initially display the WKLDDEF view of the product that you selected, you are in browse mode. You can look at a listing of workload definitions but you cannot alter them.

When you type EDIT on the COMMAND line, an edit lock is placed on the BBIPARM workload member containing the current workload definitions of the product that you selected. You can now enter primary or line commands to add, delete, change, or update workload definitions.

Table 3-1 shows both the line and primary commands available after an edit lock is obtained on workload definitions.

Table 3-1 Workload Definition Commands

Command	Description
Primary Commands	
ADD	creates a new workload definition
CANcel	cancels any changes made to the definition during an edit session by refreshing the workload definition with the current version in storage
SAVE	saves any changes made to the workload definition and maintains the current edit session
Line Commands	
ADD	creates a new workload definition by using the selected definition as a template
CHAnge	updates an existing workload definition
DELete	deletes a workload definition
INStall	updates the runtime version of the workload definition in the CAS to start transaction monitoring
UNDelete	cancels the DELete command and retains the workload definition if issued before the SAVE command

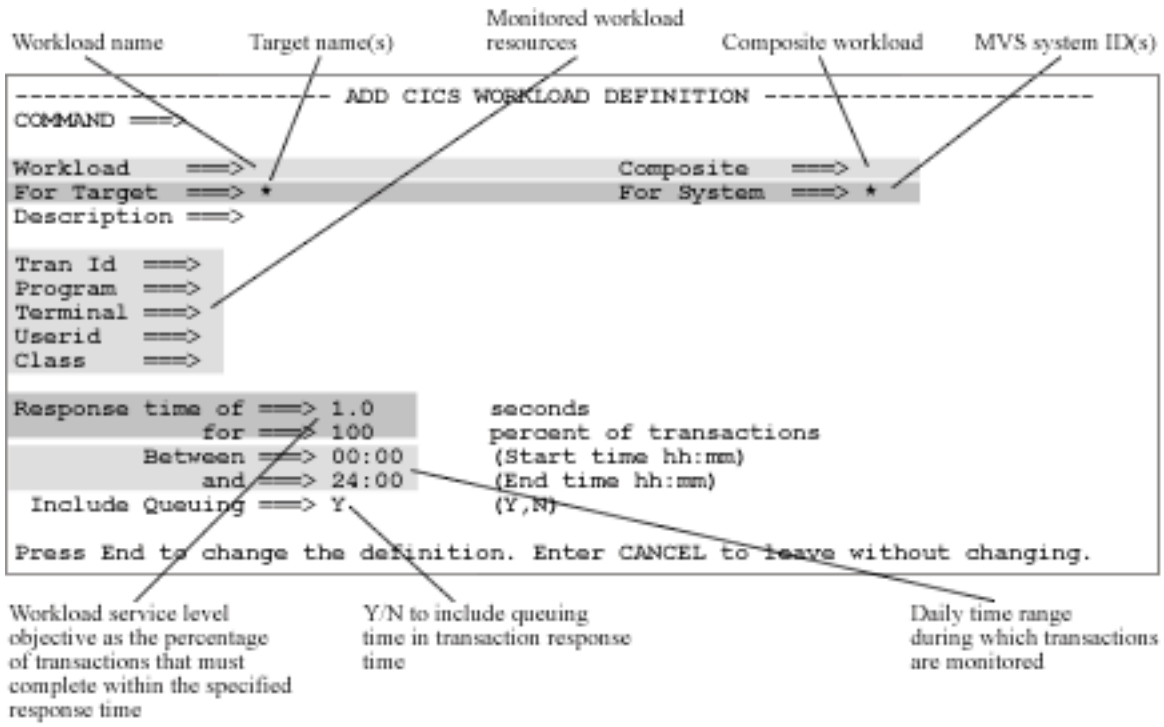
If you enter the ADD primary or line command, the Workload Definition dialog is displayed, as shown in Figure 3-7 on page 3-9. Invoking ADD from the COMMAND line gives you a default dialog for defining a new workload.

Use the ADD line command instead if you prefer to use an existing workload definition as a model to create a new definition. Simply type **add** in the **CMD** field next to the workload definition that you want to use as your model for your new definition.

```
CMD Workload Applcatn Target   System   Description  Resp %Tr Start
--- Name---- Name---- -------- -------- ------------ ---- --- Time-
add ACCTING  CICSP*   *        *        All Accounting  0.5  80 07:00
```

Make changes to the fields of the Workload Definition dialog and save the definition with a new workload name.

Figure 3-7 Example of a MAINVIEW for CICS Workload Definition Dialog



The dialog fields that you complete to define a workload are shown in Figure 3-7. Each field is described separately in the following sections.

Workload and Composite Names

Enter the selected names for the workload and composite. Names can be up to eight characters. Wildcard characters (*, ?, +) are permitted.

The composite workload name is used by VistaPoint to combine workloads together into an application. Related CICS, OS/390, DB2, and IMS workloads must have the same composite name if you want to monitor their combined transaction performance in a VistaPoint application.

Filtering and sorting data is easier from VistaPoint or other product views if you use a consistent prefix for your workload names. Again, BMC Software recommends that you establish a common naming convention for your site's workloads. Refer to “Establish Consistent Naming Conventions” on page 3-4 for naming recommendations.

Target and System ID

Enter the complete target name and system ID if you want to restrict the workload to a single target and OS/390 image.

Use wildcard characters if you want to expand the definition to include multiple targets and OS/390 images.

Resources

Enter the type of resources to be monitored by the workload. Multiple entries for each resource can be specified with either a comma or space between them.

Wildcard characters (*, ?, +) are supported.

Refer to MAINVIEW for CICS, DB2, and IMS libraries for complete listings of the resources monitored by each product.

MAINVIEW for CICS:

Tran ID 4-byte transaction ID

Program 8-byte name of the load module invoked by the transaction

Terminal 4-character terminal name that initiates a transaction

User ID 8-character name of the user signed-on to the terminal that initiates a transaction

Class 2-digit transaction class (01-10) for CICS. Class zero (00) selects all transactions not belonging in classes 01-10.

MAINVIEW for DB2:

Plan 1 to 8-character plan name

Auth ID 1 to 8-character authorization ID

Conn ID 1 to 8-character connection name

Corr ID 1 to 12-character correlation ID

Location 1 to 16-character location name

Conn Type One of the following connection types:

TSO	IMS
CICS	BATCH
CAF	IMSMPP
IMSBMP	IMSTBMP
IMSCTL	DLI
SYSSERV	APLSERV
UTIL	Blank for all connection types

MAINVIEW for IMS:

Tran ID 4-byte transaction ID

Class 001 to 020 transaction class

Program 1 to 8-byte name of the load module invoked by the transaction

PSB 1 to 8-character name of IMS program specification block (PSB)

Region 1 to 8-character job name of the IMS region

Region ID 1 to 3-character name of the IMS region

Terminal 1 to 8-alphanumeric LTERM name

User ID 1-8 character sign-on user ID

Tran Type Specifies IMS transactions by the following types:

ALL	All IMS transaction types are monitored
DB2	Transactions that access DB2, excluding DBCTL
DLI	Transactions that make DL/I database calls
FP	Transactions that make Fast Path database calls

Prog Type Specifies transaction monitoring by the following IMS program types:

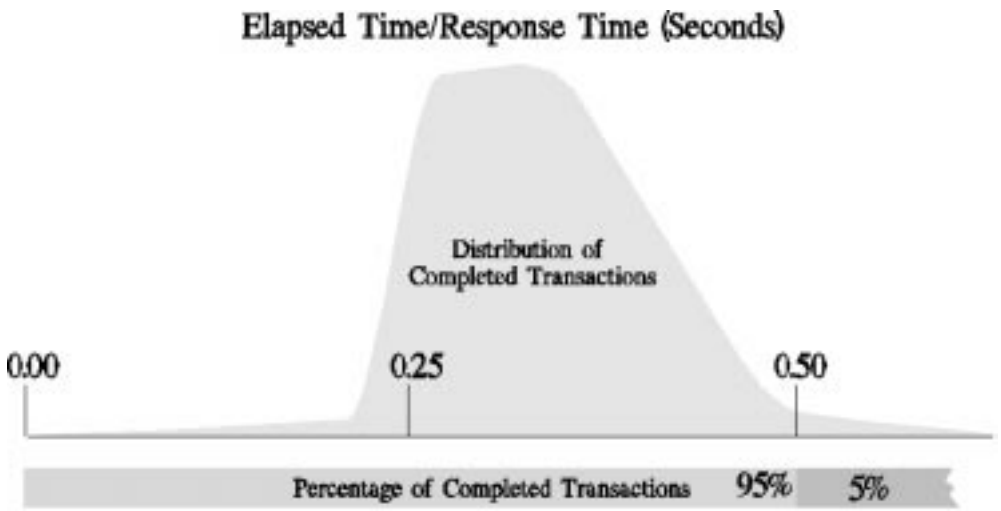
MDP	Message-Driven Program
MPP	Message Processing Program
TPI	CPI-C driven program
DBT	DBCTL thread
NOTDBT	All non-DBCTL threads

Service-Level Objective

A service-level objective specifies the minimum acceptable performance of a workload. You set a service-level objective based on your assessment of the minimum percentage of all transactions that must complete within a specified response time for the tasks that occur in a workload.

Figure 3-8 shows an example of a service-level objective. Transaction response times are distributed over time with the cumulative percentage of completed transactions shown beneath the horizontal axis.

Figure 3-8 Example of a Service-Level Objective



In Figure 3-8, 95% of transactions must complete within 0.5 seconds for the workload to meet its service-level objective. In this example, exactly 95% of transactions are completed within the 0.5 second response-time goal. The workload is performing at 100% of objective.

If more than 95% of transactions complete within the 0.5 second response-time goal, the workload is exceeding its service-level objective and is operating beyond 100%. Conversely, if less than 95% of transactions complete within 0.5 second, the workload is operating at less than 100% of objective.

Workloads that are members of the same application can have different service-level objectives. VistaPoint, and the products that work with it, normalize reported values to assure consistency.

Monitoring Time Range

A workload's transaction response-time monitoring occurs within a daily period. You set the beginning and ending times in hours and minutes (hh:mm) that you want a workload to be monitored.

The hourly range is expressed as a 24-hour clock (00-24). Minutes range from 00-60.

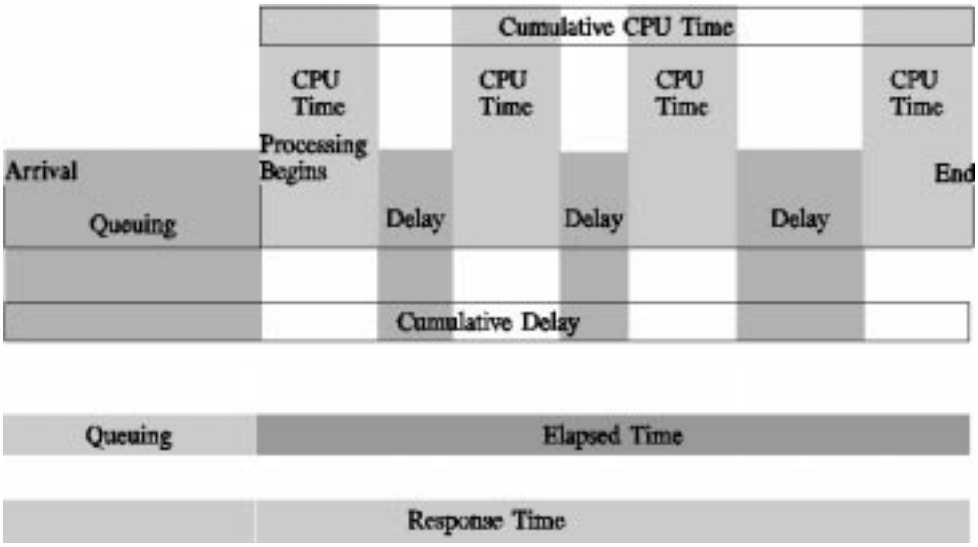
Start and stop times can cross midnight. For example, you can set both the start and stop times to 08:00. Workload monitoring occurs over a 24-hour period between 08:00 and 08:00 of the next day. Every morning at 08:00 hours, workload monitoring resets and begins again.

Response Time Queuing

You can include or exclude queuing time in CICS or IMS transactions. Both products use response time (@RSTM) or elapsed time (@ELTM) monitors, depending on whether you enter Yes or No in the **Include Queuing** field.

Figure 3-9 shows the difference between VistaPoint elapsed- and response-time transaction monitoring.

Figure 3-9 Difference between Response Time and Elapsed Time



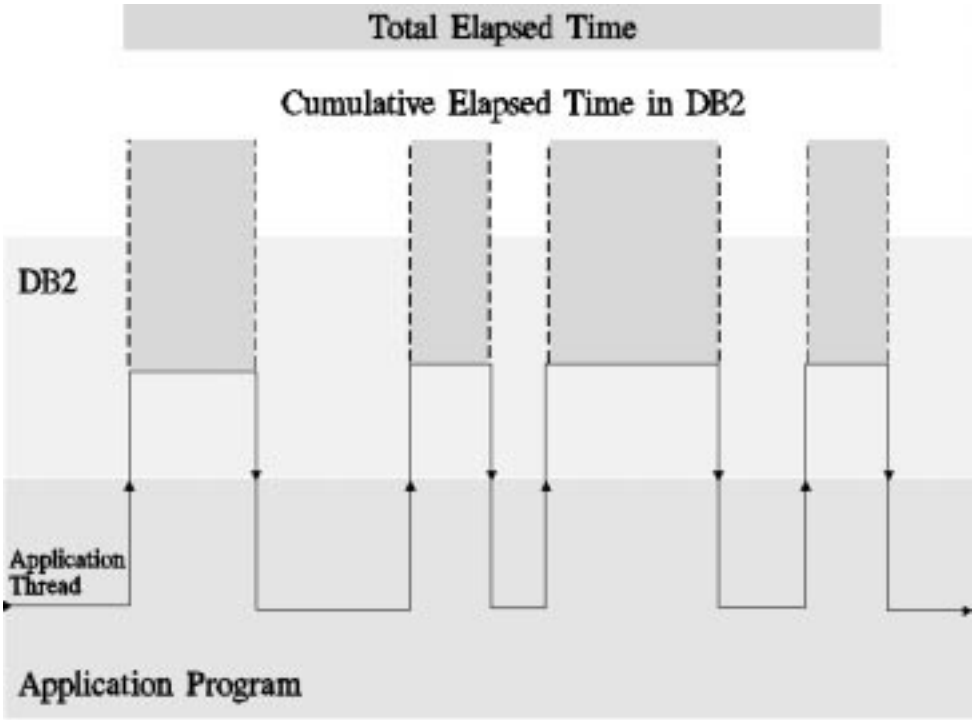
With CICS or IMS, response time includes queuing and the processing period of a transaction. Elapsed time includes only the processing period.

VistaPoint considers total transaction time to be the sum of CPU time and delay. CPU time is the sum of the relatively short intervals in the processing period when the transaction undergoes CPU processing. Delay is the sum of queuing and the processing intervals *not* spent in CPU processing.

You have the choice of two types of elapsed time monitoring of DB2 workloads: total elapsed time or cumulative DB2 time monitoring. If the **In DB2 time only** field is set to Y, VistaPoint uses the @ELTD monitor to report cumulative DB2 processing time. Otherwise, the @ELTM monitor is used to report total elapsed time.

Figure 3-10 shows a time line of an application program. Processing occurs within the application program or as a thread to DB2.

Figure 3-10 DB2 Transaction Monitoring



Total elapsed-time monitoring is the period between a thread's initial entry and final exit from DB2. The other form of DB2 monitoring restricts elapsed time to the cumulative periods spent by the program in DB2.

Defining OS/390 Workloads

VistaPoint monitors OS/390 performance from composite workloads defined with the BMC Software MAINVIEW for OS/390 product. Refer to the *MAINVIEW for OS/390 Customization Guide* for information about defining composite workloads. Also, *Getting Started with MAINVIEW VistaPoint* includes a step-by-step procedure to create OS/390 composite workloads for VistaPoint.

VistaPoint incorporates OS/390 composite workloads into applications based on the workload name. The name of the composite workload must be the same as the VistaPoint application to which you want it to belong.

All workloads that belong to a OS/390 composite workload must have defined service objectives if their performance will be reported as part of a VistaPoint application. VistaPoint only reports performance from composite workloads with service objectives.

Saving and Installing the Workload Definition

After you have completed the workload definition, you must save it and install it for it to become active.

Type **SAVE** or press your END key from the WKLDDEF view to save the workload definition that you modified or added. Type **SAVE** to retain the edit lock on the workload definition member in the BBIPARM data set to make further changes.

After a workload definition is saved, it is still not active. You must install it. Type INStall next to the listed workload definition to start monitoring transaction response time if you are within the defined time range. The WKLDDEF status field, **Sta**, changes from **Ina** to **Act** to indicate the that workload is now active.

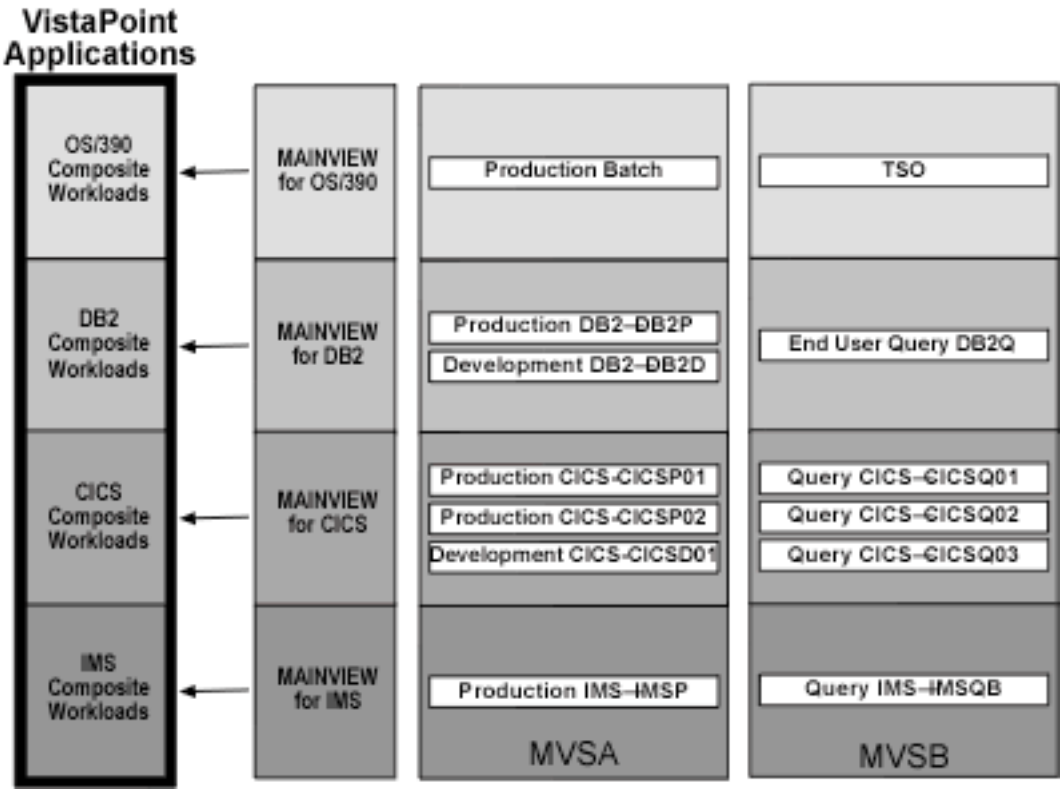
```
>W1 =CWKLDDEF==========CICSP01==*========(00 EDIT          )====MVCICS=======10
CMD Workload Composite Target   System   Description              Sta Resp %Tr
--- Name---- Name----- -------- -------- ------------             --- ---- ---
ins ACCTPAY  FINANCE   CICSP*   *        Accounts payable         Act 0.30  90
```

If you updated an existing workload definition with the CHAnge line command, the INStall command deactivates the current definition and then starts monitoring with parameters set in the updated version of the workload.

Creating VistaPoint Applications

Figure 3-11 shows an example of a VistaPoint application. In this example, the shaded production targets represent workloads within these targets specified as members of the VistaPoint application.

Figure 3-11 Example of an Application Composed of VistaPoint Workloads



The characteristic that associates workloads together is the common composite name specified in their respective definitions. Based on their attributed association, VistaPoint combines data from the member workloads and presents the calculated values in application views. VistaPoint views present the aggregate performance of the workloads that are members of an application.

The calculated values for transaction performance are derived from the service-level objectives specified for the workloads in the application. Refer to Appendix B, “Calculated Values” for more information about performance data shown in VistaPoint views and the views of the products that work with VistaPoint.

Maintaining Workload Definitions

Occasionally, you may need to make changes to your workload definitions or delete them if they are no longer needed. This section shows how to update and delete current definitions that are displayed in the Workload Definition dialogs of the products that work with VistaPoint.

Updating Workload Definitions

When you make changes to subsystems that are monitored by VistaPoint, you must make corresponding changes to their workload definitions. The accuracy of VistaPoint depends on how close its definitions reflect your site's actual environment.

Changing a workload definition is a process of modifying values in the workload definition dialog and then saving the results. This section explains how to modify existing workload definitions.

Figure 3-12 shows that an edit lock has been set against the CICS workload definitions listed on the CWKLDDEF view.

Figure 3-12 Selecting an Existing Workload Definition to Change

```
 08NOV2000 13:09:08 ----------- INFORMATION DISPLAY ---------------------------
 COMMAND  ===>                                                SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =CWKLDDEF==========CICSP01==*========(00 EDIT          )====MVCICS=======10
CMD Workload Composite Target   System   Description               Sta Resp %Tr
--- Name---- Name----- -------- -------- -----------               --- ---- ---
    ACCTPAY  FINANCE   CICSP*   *        Accounts payable          Act 0.30  90
cha ACCTRECV FINANCE   CICSP*   *        Accounts receivable       Act 0.30  90
    ACCTRET  FINANCE   CICSP*   *        Accounts return           Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.    Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing             Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions     Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01      Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing              Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans.  Act 1.00  95
```

Use the CHAnge line command to select the workload definition that you want to modify. Figure 3-13 on page 3-18 shows an example of the application workload definition dialog with the current values listed for the ACCTRECV workload. You can make changes to any field except the workload name.

Figure 3-13 Updating an Existing CICS Workload Definition

```
---------------------- CHANGE CICS WORKLOAD DEFINITION -------------------
COMMAND ===>

Workload    ===  ACCTRECV                Composite   ===> FINANCE
For Target  ===> CICSP*                  For System  ===> *
Description ===> Accounts receivable

Tran ID  ===> *
Program  ===> *
Terminal ===>
Userid   ===>
Class    ===>

Response time of ===> 0.3        seconds
             for ===> 95         percent transactions
         Between ===> 08:00      (Start time hh:mm)
             and ===> 15:00      (End time hh:mm)
 Include Queuing ===> Y          (Y,N)

Press End to change the definition. Enter CANCEL to leave without changing.
```

After updating the definition, press the END key to retain any changes you made. Updates that you made are pending and are not yet saved. If you want to discard the updates, issue the CANCEL command instead.

After issuing the END or CANCEL command, you are returned to the CWKLDDEF view. If you have pending updates, the window information line shows an edit status of `(00 EDIT MOD)`, indicating that a workload definition has unsaved modifications. If you return to the CWKLDDEF view with the CANCEL command, you are still in an edit session although your updates have been discarded.

Any changes that you made to a workload definition can be saved or cancelled from the CWKLDDEF view. If you want to save your updates and exit from the CWKLDDEF view, press your END key. If you want to save updates and maintain an edit lock on the workload definitions, enter **SAVE**. Issuing the CANCEL command discards any unsaved updates and removes the edit lock.

Entering the INStall line command against the updated definition deactivates the original workload. VistaPoint stops monitoring the workload's transactions. Transaction monitoring immediately begins with the updated workload definition if you are within the specified daily time range. Otherwise, transaction monitoring begins at the onset of the next daily time range.

Deleting Workload Definitions

You can delete workload definitions in a two-step process of issuing the DELete line command and saving your editing changes to the workload definition data set. Until your editing changes are saved, deleted definitions can be recovered with the UNDelete command. “Recovering a Deleted Workload Definition” on page 3-20 explains how to recover workload definitions marked for deletion.

Figure 3-14 shows the MAINVIEW for CICS CWKLDDEF view listing current workload definitions. The edit lock is active and the DELete line command can be entered against the listed workload definitions.

Figure 3-14 CWKLDDEF with an Edit Lock Active for the Workload Definitions

```
 08NOV2000 13:09:08 ----------- INFORMATION DISPLAY ---------------------------
 COMMAND  ===>                                              SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =CWKLDDEF==========CICSP01==*========(00 EDIT          )====MVCICS=======10
CMD Workload Composite Target   System   Description               Sta Resp %Tr
--- Name---- Name----- -------- -------- ------------              --- ---- ---
    ACCTPAY  FINANCE   CICSP*   *        Accounts payable          Act 0.30  90
del ACCTRECV FINANCE   CICSP*   *        Accounts receivable       Act 0.30  90
    ACCTRET  FINANCE   CICSP*   *        Accounts return           Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.    Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing             Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions     Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01      Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing              Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans.  Act 1.00  95
```

In Figure 3-14, invoking the DELete line command against ACCTRECV selects the workload definition for deletion. It is still possible to recover ACCTRECV until the SAVE command is issued.

Figure 3-15 shows that ACCTRECV has been removed from the listing of current workload definitions.

Figure 3-15 CWKLDDEF after Deleting a Workload Definition

```
08NOV2000 13:09:08 ----------- INFORMATION DISPLAY --------------------------
 COMMAND  ===>                                              SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =CWKLDDEF==========CICSP01==*========(00 EDIT         )====MVCICS========9
CMD Workload Composite Target   System   Description              Sta Resp %Tr
--- Name---- Name----- -------- -------- ------------             --- ---- ---
    ACCTPAY  FINANCE   CICSP*   *        Accounts payable         Act 0.30  90
    ACCTRET  FINANCE   CICSP*   *        Accounts return          Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.   Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing            Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions    Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01     Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing             Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans. Act 1.00  95
```

Enter **SAVE** on the **COMMAND** line to permanently delete the workload definition and any other changes that you made in this editing session.

Recovering a Deleted Workload Definition

If you mistakenly delete a workload definition you meant to keep, you can undelete and recover the definition. Deleted workload definitions can be recovered with the UNDelete line command. But, undeleting a workload definition is possible only if you have not yet saved your changes to the BBIPARM data set. The workload definition is permanently deleted and cannot be recovered after the delete request is saved.

Use the PARM DELETED(*) command with UNDelete to recover workload definitions. PARM DELETED(*) identifies workload definitions that have been previously marked for deletion in the current editing session. After you issue PARM DELETED(*), deleted workload definitions are displayed on the WKLDDEF screen and are colored differently from other workload definitions.

Figure 3-16 and Figure 3-17 on page 3-21 show the CWKLDDEF screen before and after the ACCTPAY workload definition has been marked for deletion. Workloads are no longer listed on a WKLDDEF screen after they have been selected for deletion.

Figure 3-16 Before Deleting ACCTPAY Workload Definition

```
 08NOV2000 13:09:08 ----------- INFORMATION DISPLAY --------------------------
 COMMAND  ===>                                            SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =CWKLDDEF==========CICSP01==*========(00 EDIT         )====MVCICS========9
CMD Workload Composite Target   System   Description              Sta Resp %Tr
--- Name---- Name----- -------- -------- -----------              --- ---- ---
del ACCTPAY  FINANCE   CICSP*   *        Accounts payable         Act 0.30  90
    ACCTRET  FINANCE   CICSP*   *        Accounts return          Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.   Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing            Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions    Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01     Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing             Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans. Act 1.00  95
```

Figure 3-17 After Deleting ACCTPAY Workload Definition

```
 08NOV2000 13:09:08 ----------- INFORMATION DISPLAY --------------------------
 COMMAND  ===>                                            SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =CWKLDDEF==========CICSP01==*========(00 EDIT         )====MVCICS========8
CMD Workload Composite Target   System   Description              Sta Resp %Tr
--- Name---- Name----- -------- -------- -----------              --- ---- ---
    ACCTRET  FINANCE   CICSP*   *        Accounts return          Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.   Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing            Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions    Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01     Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing             Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans. Act 1.00  95
```

Figure 3-18 on page 3-22 shows the same CWKLDDEF screen after issuing the PARM DELETED(*) command. ACCTPAY is listed again, but is highlighted to indicate its delete-pending status.

Figure 3-18 Identifying Deleted Workloads with the PARM DELETED(*) Command (Part 1)

```
 08NOV2000 13:09:08 ----------- INFORMATION DISPLAY --------------------------
 COMMAND  ===>                                              SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =CWKLDDEF==========CICSP01==*========(00 EDIT         )====MVCICS========9
CMD Workload Composite Target   System   Description              Sta Resp %Tr
--- Name---- Name----- -------- -------- ------------             --- ---- ---
und ACCTPAY  FINANCE   CICSP*   *        Accounts payable         Act 0.30  90
    ACCTRET  FINANCE   CICSP*   *        Accounts return          Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.   Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing            Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions    Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01     Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing             Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans. Act 1.00  95
```

Entering the UNDelete line command recovers workloads with a pending delete status.

Figure 3-19 Identifying Deleted Workloads with the PARM DELETED(*) Command (Part 2)

```
 08NOV2000 13:09:08 ----------- INFORMATION DISPLAY --------------------------
 COMMAND  ===>                                              SCROLL ===> PAGE
 CURR WIN ===> 1        ALT WIN ===>
 >W1 =CWKLDDEF==========CICSP01==*========(00 EDIT         )====MVCICS========9
CMD Workload Composite Target   System   Description              Sta Resp %Tr
--- Name---- Name----- -------- -------- ------------             --- ---- ---
    ACCTPAY  FINANCE   CICSP*   *        Accounts payable         Act 0.30  90
    ACCTRET  FINANCE   CICSP*   *        Accounts return          Act 1.00  95
    DIRECT   PAYROLL   CICSP*   *        Direct employee trans.   Act 1.00  95
    BILLING  FINANCE   CICSP*   *        Debit billing            Act 0.10 100
    CICSALL  BBCICS    *        *        CICS All transactions    Act 0.50  85
    QUERY1   QUERY     CICSQ01  *        CICS queries CICSQ01     Act 0.25  95
    CICSP01  CICSP01   CICSP01  *        CICSP01 only transactions Act 1.00 100
    CICSTEST TEST0131  CICS*    *        CICS testing             Act 0.30  95
    INDIRECT PAYROLL   CICSP*   *        Indirect employee trans. Act 1.00  95
```

After undeleting a workload, the line color returns to normal for the line listing the deleted workload definition. This indicates the workload definition has been recovered.

Chapter 4 Setting View Contexts

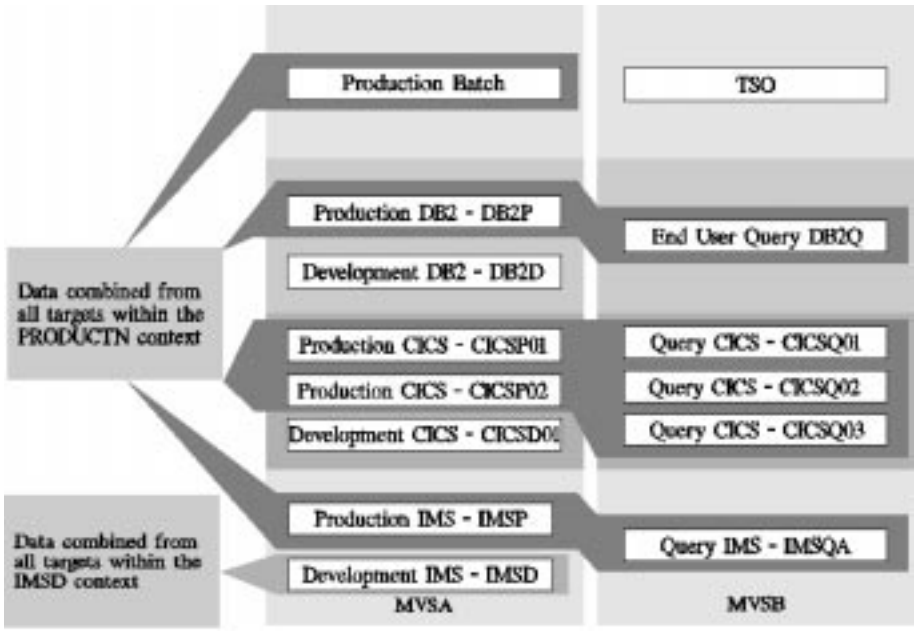
A context is a frame of reference for the data that you display in views. A context can be limited to a single target or expanded to include all recognized targets providing data to VistaPoint. In the case where you are viewing multiple targets, you can set the scope of your views to display a selected target within a context.

The default context is ALL, which includes all monitored targets providing CICS, DB2, IMS, or OS/390 data appearing in VistaPoint views. Contexts that include more than a single target are called single system image (SSI) contexts. An SSI context functions as a high-level filter that selects the data appearing in your views by the following criteria:

- target name (CICS region, IMS or DB2 subsystem, OS/390
- products attached to a coordinating address space (CAS)
- BMC Software MAINVIEW product
- product attached to a product address space (PAS)

Figure 4-1 on page 4-2 shows the difference between different types of contexts.

Figure 4-1 SSI and Single-Product View Contexts



In Figure 4-1, the single-target context named IMSD limits data shown in VistaPoint views to a single IMS development subsystem. Data from the CICS, DB2, OS/390, and IMS production targets is combined into views within the SSI PRODUCTN context. You can also set the scope of an SSI context to restrict views to a single target.

Using Single System Image Contexts

View contexts are particularly important for VistaPoint and the MAINVIEW products that work with it. Because VistaPoint views display data from multiple targets, products, and systems, SSI contexts are essential.

As you navigate between VistaPoint views and the views of the products that work with VistaPoint, frequently you need to set another context. This happens most often when you want to look at data from a target that is not in your current context, or when you want to focus on a single target within an SSI context.

This chapter explains how to set a context using MAINVIEW commands. There is a separate section that explains how to limit the scope of an SSI context.

Changing Products, Contexts, and Targets with the CONtext Command

The CONtext command is the fastest way to switch between contexts, products, and views. The syntax of the command is

CONtext ***target_name product_identifier view_name***

where:

target_name

*	Specifies the default system name to which Plex Manager is attached
=	Accepts the current target set for the active window.
SSI name	Name of a single system image context listed on the CONACT view.
Target name	Context name identified for each product listed on the PLEX view.

product_identifier

CMF	CMF MONITOR Online
MVCICS	MAINVIEW for CICS
MVDB2	MAINVIEW for DB2
MVEMGR	MAINVIEW Alarm Manager
MVIMS	MAINVIEW for IMS
MVMVS	MAINVIEW for OS/390
MVVP	MAINVIEW VistaPoint
PLEXMGR	PLEX MANAGER

view_name

Name of a view associated with the product identified in the Product name parameter

The CONtext command establishes the current context, product, and target for the active window. Subsequent views displayed in the same window inherit these values; however, new windows do not. Unless the context for new windows is specified with the SETD command, new windows assume the context, product, and target established at the beginning of the session.

Setting a Context with the SET Commands

MAINVIEW provides two different SET commands to change a context. The SET and SETD commands use separate ISPF dialogs. You replace the existing values with new values and save your changes.

SETD changes the default settings of new windows, but does not affect the context of the current window. SET, on the other hand, is similar to the CONtext command. You can change products, contexts, and targets by changing field values in an ISPF dialog.

Setting a Default Context with the SETD Command

You should use the SETD command when you need to establish a default context for any new windows you open. After that, all new windows automatically reflect the context, product, and scope specified with the command.

SETD is useful when several windows need to be opened to a new context. Instead of issuing a CONtext command for each new window, you can use SETD once to set all future windows with the default settings that you choose.

Existing windows are not affected by the values set with the SETD command unless you reset a window with the CLEar command. In that case, the open window assumes the default settings specified with the SETD command.

The SETD command displays the Set Default Context, Product and Scope dialog shown in Figure 4-2 on page 4-5. You need to fill in the three fields shown beneath the **Default Window Parameters** field.

Figure 4-2 Set Default Context, Product, and Scope Dialog

```
-------------------- SET DEFAULT CONTEXT, PRODUCT AND SCOPE ---------------
COMMAND  ===>


  Default Window Parameters:

  Context     ===> PRODUCTN
  Product     ===> MVVP
  Scope       ===> *


  Type END to set default window parameters
       CANcel to quit without setting
```

Type over any of the displayed settings with any changes that you want and press your END key to define them as default settings. You can cancel any changes and return to the current window by issuing the CANcel command.

Context Field: The **Context** field displays the SSI context or the name of the target active in the current window.

Product Field: The **Product** fields displays the product identifier of the product whose view is active in the current window. Refer to “Changing Products, Contexts, and Targets with the CONtext Command” on page 4-3 for a list of MAINVIEW product identifiers.

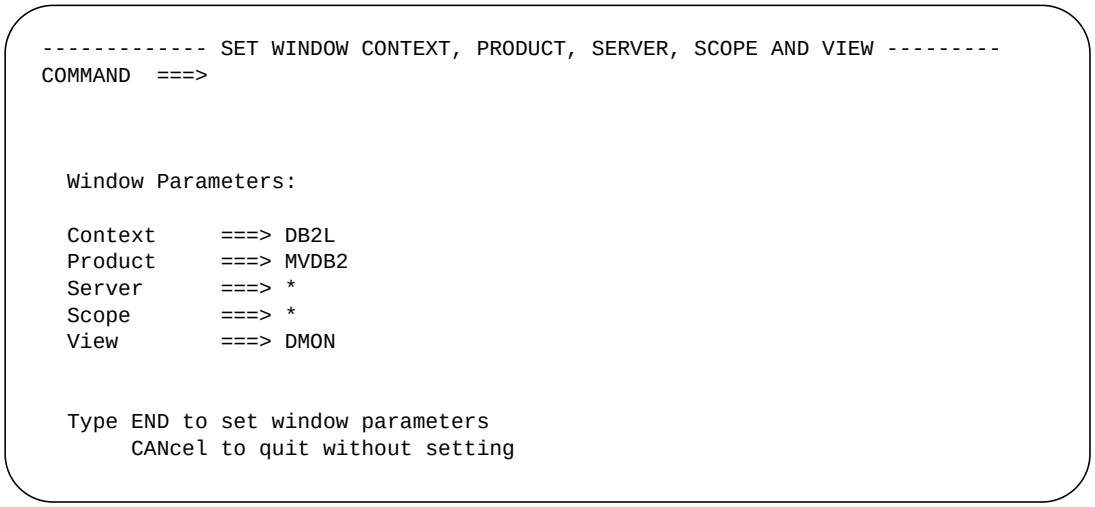
Scope Field: The **Scope** field displays the scope of the SSI context currently active in the window. An asterisk (*) appears in the **Scope** field if the SSI context and the scope are the same.

Using SET to Change Contexts

The SET command provides an ISPF dialog to specify the context, product, server, scope, and view of the current window. SET is functionally equivalent to the CONtext, SCOpe, and VIEW commands when all values are specified from its dialog fields.

Figure 4-3 on page 4-6 shows the Set Window Context, Product, Server, Scope, and View dialog displayed after the SET command is used.

Figure 4-3 Set Window Context, Product, Server, Scope and View Dialog



Type over any of the displayed settings and press END to activate the new settings. The window is redisplayed with updated data, and the changes are reflected in the window information line. Issue the CANcel command to cancel any changes you made in the SET dialog.

Context Field: The **Context** field displays the SSI context or the name of the target active in the current window.

Product Field: The **Product** fields displays the product identifier of the product whose view is active in the current window. Refer to “Changing Products, Contexts, and Targets with the CONtext Command” on page 4-3 for a list of MAINVIEW product identifiers.

Server: The **Server** field identifies the name of the product address space (PAS) if multiple PASs are operating within a single MVS image. The **Server** field is important in situations where targets with the same name are accessible to multiple PASs.

Scope: The **Scope** field displays the scope of the SSI context currently active in the window. An asterisk (*) appears in the **Scope** field if the SSI context and the scope are the same.

View: The **View** field specifies the name of the view to appear in the active window after saving the values specified with the SET command.

SET establishes the current context, product, and target for the active window. Subsequent views displayed in the same window inherit these values; however, new windows do not. Unless the context for new windows is specified with the SETD command, new windows assume the context, product, and target established at the beginning of the session.

Using SCOPE to Narrow the Focus of an SSI Context

Figure 4-1 on page 4-2 shows the difference between SSI and target contexts. The PRODUCTN SSI context includes all targets doing production work. The IMSD context shows data from a single IMS target.

You can switch back and forth between an SSI and a target context using the MAINVIEW commands discussed earlier in this chapter. Alternatively, you can stay within an SSI context and selectively narrow its focus when you want to see the performance of a single target.

You can use the SCOpe command to restrict views within an SSI context to show data from a target. The syntax of the SCOpe command is

SCOpe *target* | *

where:

target Name of an active target within the current SSI context.

* Reverses the effect of the target parameter and broadens the SSI context to include all targets without updating the data.

Note: You use the * parameter only after you have previously issued the SCOpe command to focus on specific target data within an SSI context.

Using the example shown in Figure 4-1 on page 4-2 again, you can issue the command scope cicsp01 to restrict a view within the PRODUCTN context to show data from the CICSP01 region. The window information line, shown immediately above the view's data fields, shows the SSI context has been limited in scope.

```
>W1=APOVER=========(PRODUCTN=CICSP01=)08NOV2000==10:53:43====MVVP=========9
```

You can redisplay all the targets again by issuing SCOpe with a wildcard character (*) to select all the original targets included in the context.

Chapter 5 Managing View Data

This chapter describes several ways to manage view data. The first section explains several MAINVIEW commands that regulate the refresh rate of a view. The second section describes how to use alternate forms of a view when you want to use the same query data for all views. Using filters to restrict data shown in views is the topic of the last section.

Updating and Locking View Data

Occasionally, you may need to temporarily change the refresh rate of a view. You may want to increase the refresh rate to get more detail about moment-to-moment changes in a view's data. Or you may want to decrease the refresh rate to see larger blocks of time for trend analysis. In either case, a view should be refreshed automatically without the need to press **Enter**.

There are other situations where you need to stop view updates completely. Typically, you observe unusual conditions shown in a view that you would like to examine in more detail. However, views are updated periodically and the original data is overwritten before you have time to do any analysis. You need to be able to *freeze* a view's data and prevent it from being updated.

This section describes several MAINVIEW commands that control how data is refreshed in a view.

Changing the Refresh Rate of Views

The ASU (automatic screen update mode) command allows you to temporarily override the default view refresh rate defined in your MAINVIEW profile. Using the ASU command, you can set views in unlocked windows to automatically refresh their data without having to press the **Enter** key.

The syntax of the ASU command is

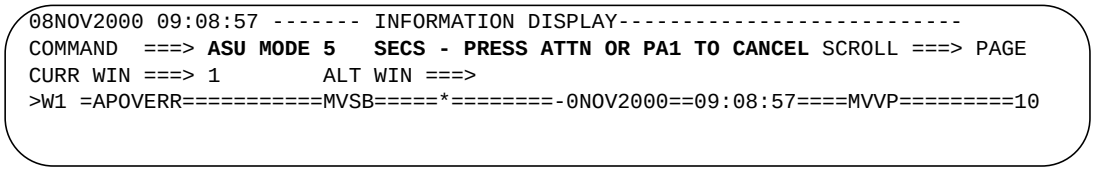
ASU *nnn*

The *nnn* parameter specifies a view's refresh rate. You can specify a range from 3 to 999 seconds. Leading zeros are not required for values less than 3 digits.

Note: If you do not specify the *nnn* parameter with the ASU command, views are refreshed at the default rate defined in your session profile.

After you enter the ASU command, a message is displayed on the **COMMAND** line indicating the refresh rate of the view. You should see the **Time** field of the window information line increment by the length of the refresh rate that you specified with the ASU command.

Figure 5-1 ASU Command



When a view is cyclically refreshed using the ASU command, all keys are inactive except the Attention function key; you cannot dynamically update a view by pressing **Enter**. Commands cannot be issued because the keyboard is locked.

Automatic screen update mode remains active until you end it. Depending on the type of terminal that you are using, you can end automatic screen updates with the following keys:

- For a SNA-attached terminal, press the ISPF ATTN key.

- For a non-SNA-attached terminal, press RESET and then press **PA1**.

Note: If you accessed VistaPoint through a VTAM or EXCP session using MAINVIEW Alternate Access, press the **PA1** key instead of the ISPF ATTN key.

Locking and Unlocking Views

The LOCK command maintains the present data displayed in a view and prevents it from being updated. You cannot dynamically update a view by pressing **Enter**.

LOCK is useful for situations when you want to compare data from the same view. Normally, you lock a view in one window and compare it to the same view in another window whose data is periodically updated.

The LOCK command is entered without parameters.

When a window is locked, the status field of the window information line changes to L. The date and time fields show when the view was last updated.

```
>L1=APOVERR====MVSB===*===08NOV2000=09:59:56===MVVP====10
```

A view's data can be updated in a locked window by

- unlocking the window
- permitting a data refresh but maintaining the window lock

The UNLock command resumes data updates to a view at the original refresh rate. UNLock can be used only for windows that are currently locked.

The UNLOCK command is entered without parameters.

When a window is unlocked, the status field of the window information line changes to W, indicating that the window is available to receive input.

Refreshing a Locked View

The DATARefresh command allows you to update view data but maintain the window lock. DATARefresh is useful when you want to see changes in a view's data and save them by maintaining the window lock.

DATARefresh is entered without parameters.

After you enter the DATARefresh command, data is updated at the time you press **Enter**. The date and time fields of the window information line indicate when the data refresh occurred. The status field indicates that the window is still locked and no further updates will occur until the DATARefresh command is invoked again or the view is unlocked.

Table 5-1 shows MAINVIEW commands that are useful for updating and locking view data.

Table 5-1 Commands to Refresh and Lock View Data

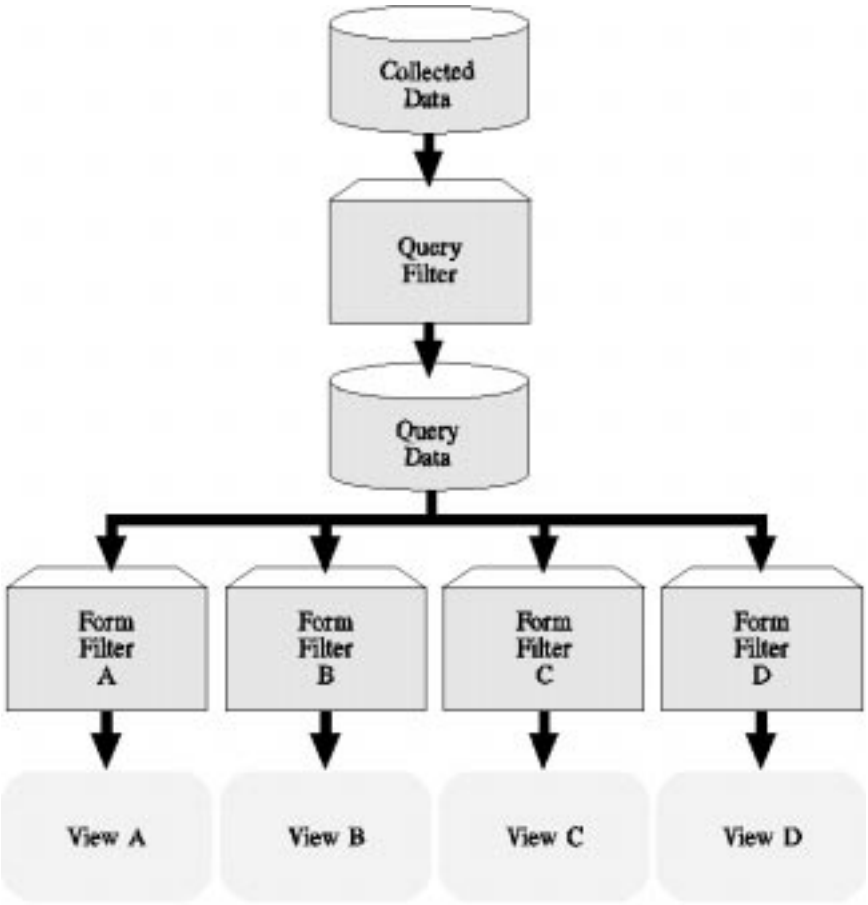
To do this...	Use this command...
Update view data at a fixed rate	ASU *nnn* The refresh rate, *nnn*, is a value between 3 and 999 seconds.
Lock current view data	LOCK
Unlock data shown in a view	UNLock
Refresh data displayed in a locked view	DATARefresh

Displaying Alternate Forms of a View

A view is composed of a query and a form. A query represents a selected portion of all available data that is displayed by a view. A view's form specifies how the query data is summarized and displayed.

Look at Figure 5-2 on page 5-5. When you request a view, a query is made against the collected data. The appropriate data is filtered and stored as query data. Each view's form formats and summarizes the query data.

Figure 5-2 Views Composed of Queries and Forms



Each view contains one form, but query data can be shared by several views. Views that share data from a single query are known as alternate forms. In the example shown in Figure 5-2, views A, B, C, and D are alternate forms of each other.

The principal benefit of using alternate forms is fewer systems resources are required because views use the same query data. Each view does not require an update request to select query data.

You can display alternate forms of a view with the FORM command. The syntax of the FORM command is:

FORM {*view name*} *keyword* | *positional*

The *view name* parameter is required. It is the name of any view that is an alternate form of the current view. You can find alternate forms of your current view with online view help. After you bring up your view's online help, scroll forward until you see the Forms that are valid for this view option. Hyperlink on the option, as shown in Figure 5-3 on page 5-6.

Figure 5-3 Selecting the Alternate Forms Option from View Help

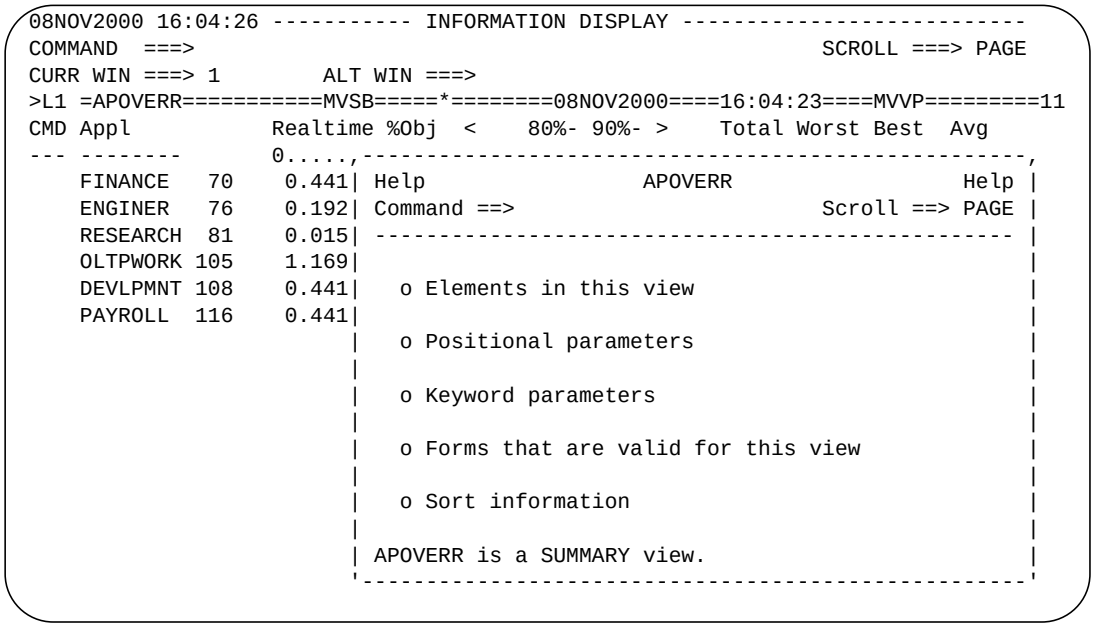
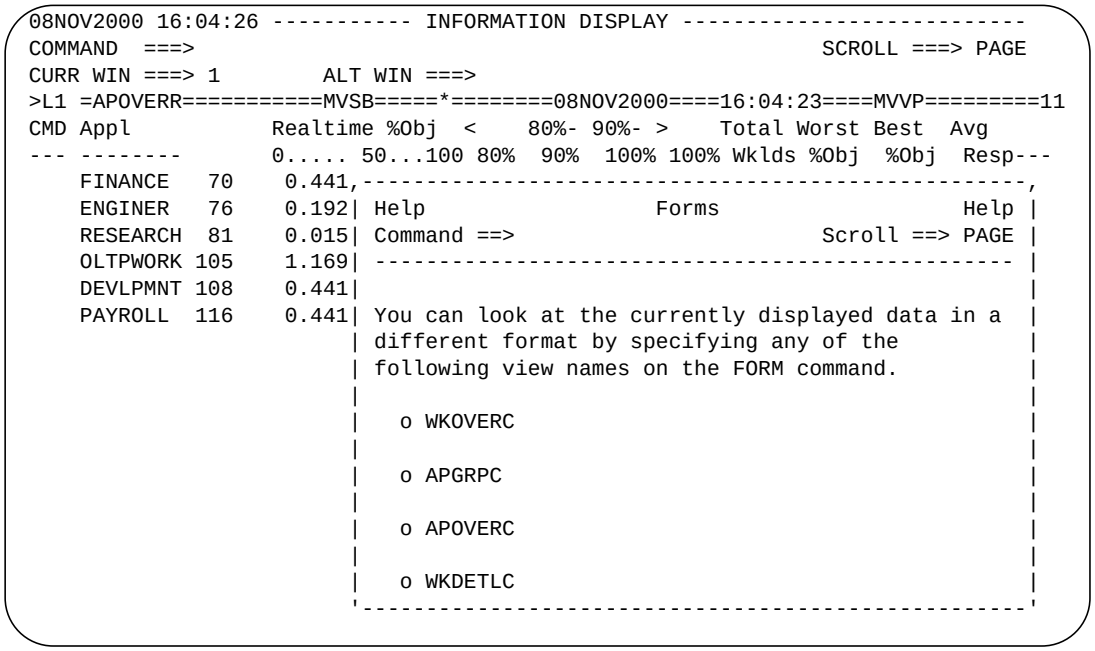


Figure 5-4 shows a pop-up help window listing the names of VistaPoint views that are alternate forms of each other.

Figure 5-4 List of Alternate Forms of a View



The *keyword* and *positional* parameters are optional. Both parameters allow you to apply filters to specific view fields to restrict the query data that appears in the alternate forms of a view. The parameters are useful when you want to limit the amount of data shown in an alternate form.

As in the case of the *view name* parameter, the *keyword* and *positional* parameters are selected from a view's online help. If you look at Figure 5-3 on page 5-6 again, the online help has hyperlink options that list a view's element names that can be used as *keyword* or *positional* parameters.

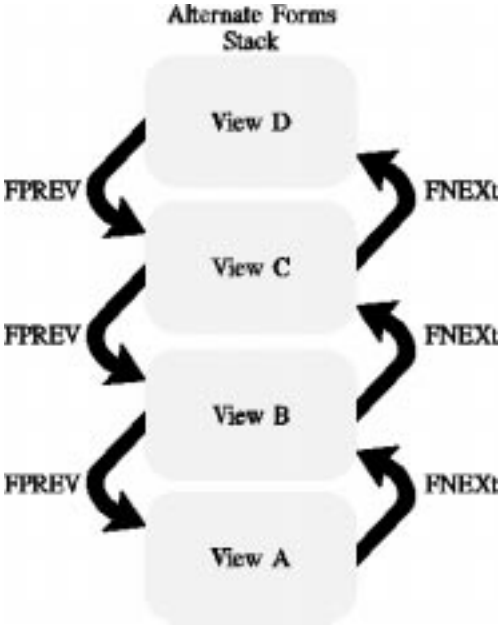
When you display an alternate form of a view, its name field appears in the form field of the window information line. In the example below, APOVERC is the alternate form that results from invoking the FORM command from the WKOVER view.

```
>W1=WKOVER===APOVERC==MVSB===*===08NOV2000==17:26:49===MVVP===9
```

You can display up to 20 different forms against a single query. Each time you select an alternate form with the same query data, the view is added to the forms stack. The alternate forms are serially linked together.

Figure 5-5 shows an example of four views in a forms stack. View A was the original view. The other views were added in order (B, C, and D).

Figure 5-5 Alternate Forms Stack



You can cycle forward or backward through the forms stack with the FNEXt and FPREV commands. Each time you enter either command, you shift to the alternate form immediately above or below the current view in the stack.

When you issue the FNEXt command at the last form in the stack, you return to the first view. The names in the view and form fields of the window information line should be the same.

Use the ENDQuery command to end the forms stack and return to the original view.

Table 5-2 shows MAINVIEW commands that are useful for displaying alternate forms of a view.

Table 5-2 Commands to Display Alternate Forms of a View

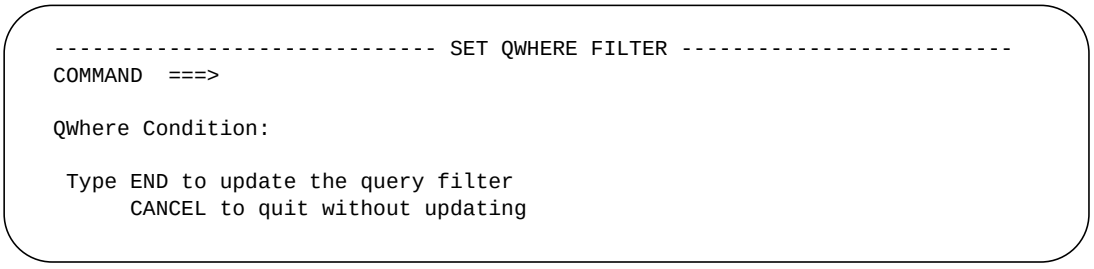
To do this...	Use this command...
Display an alternate form of a view	FORM {*view name*} *keyword* | *positional* The *view name* variable is the name of an alternate form of the current view. The *keyword* and *positional* parameters are optional. They specify filters to restrict the query data that appears in the alternate form.
Cycle forward in a forms stack	FNEXt
Cycle backward in a forms stack	FPREV
Delete a query and a forms stack	ENDQuery

Filtering Data

The MAINVIEW Where and QWhere commands are used to filter view data. The Where command uses existing query data and does not update the view. The Qwhere command updates query data when the filter conditions are set for view data. This section describes how to use both commands to filter view data.

Neither command has parameters. After you enter either command, a dialog appears to set the filter conditions. Figure 5-6 on page 5-9 shows an example of the Set Qwhere Filter dialog that appears after you invoke the QWhere command. A similar dialog appears when the Where command is used.

Figure 5-6 Set Qwhere Filter Dialog



You enter the filter conditions beneath the **QWhere Condition** field and press your END key to return to the view.

A view displays only data that meets the filter conditions you set in the command dialogs. A view resulting from filter conditions set by the QWhere command displays updated query data. Views that result from the Where command use existing query data.

A filter condition contains one or more expressions. An expression is comprised of an element name, an operator, and one or more constants, variables, or pattern values.

Filter conditions are based on the element names of the fields of the view you want to filter. Use the WHATis command to determine the element names of the view fields to which you want to apply filter conditions. The following example shows a filter condition consisting of a single expression to filter all VistaPoint application names that begin with A or B.

```
APWKAPPL IN (A*,B*)
```

APWKAPPL is the element name of the **Appl** field that lists applications by name. IN is an operator that includes data that is true for constant or variable values. In this case, the variables A* and B* select any application name beginning with A or B.

Syntax Rules of Filter Conditions

Filter conditions set by the Where and QWhere commands adhere to common syntax rules. The following syntax rules apply to the filter conditions of either command.

- Wildcard characters can be used to define a pattern value.

  ```
  PAY*  -or-  8?.6  -or-  P*1?
  ```

- A pattern value can be defined only with the following operators:

  ```
  = (equal sign)
  IN
  NOT IN
  ```

- A constant or pattern value defined to the IN or NOT IN operators must be enclosed within parentheses.

  ```
  APPLID IN (MVVP)
  ```

- When multiple constant and/or pattern values are defined to IN or NOT IN operators, commas must separate the values.

  ```
  APPLID IN (C*,MVDB2)
  ```

- With the exception of IN or NOT IN operators, parentheses need not enclose a single expression.

  ```
  APWKIRES > 1.0
  ```

- When two or more expressions are defined, each expression must be enclosed by parentheses and joined with AND or OR operators.

  ```
  (APWKAPPL = T) AND (APWKIOBJ BETWEEN 2 AND 25)
  ```

- Any expression containing the AND or OR operator can itself be used as part of a larger AND or OR expression by enclosing all expressions within parentheses.

  ```
  ((APWKAPPL = T) AND (APWKIOBJ BETWEEN 2 AND 25))
  OR (APWKIRES > 2)
  ```

- The NOT operator can be inserted before an expression to set the inverse condition.

  ```
  NOT (APWKAPPL = J*)
  ```

Filter Condition Operators

The following operators are used with the QWhere and Where commands to set filter conditions:

=	Matches value.

==	Exactly equal to; use when the wildcards (* or ?) are part of the variable.
<>	Does not match value.
NEE	Not exactly equal to.
>	Greater than.
>=	Greater than or equal to.
<	Less than.
<=	Less than or equal to.
AND	Used in an expression containing BETWEEN or NOT BETWEEN operators; used to join expressions with a logical AND statement.
OR	Used to join expressions containing any operator with an OR statement.
IN	Include only true data that meets the criteria specified by variables.
NOT IN	Include only false data that does not meet the criteria specified by variables.
BETWEEN	Include all data that falls between the range specified by variables. The AND operator sets the range by joining one variable to another; the low-end value must be defined first.
NOT BETWEEN	Include all data that does not fall between the range specified by variables. The AND operator sets the range by joining one variable to another; the low-end value must be defined first.

Examples of Filter Commands

The following list gives examples of generic filter conditions set by the QWhere and Where commands.

- Value in element matches one or more constants or patterns only:

  ```
  (element IN (pattern*|constant))
  ```

  -or-

  ```
  (element IN (pattern1*,constant,...,pattern2*))
  ```

- Value in element matches all but the following constants or patterns:

  ```
  (element NOT IN(constant1,constant2,...,pattern*))
  ```

- Value in element falls between two constant values:

  ```
  (element BETWEEN constant AND constant)
  ```

- Value in element is the constant or falls outside of the range of constant values:

  ```
  (element NOT BETWEEN constant AND constant)
  ```

- Value in element1 is equal to the constant, pattern, or element2 value:

  ```
  (element1 = constant|pattern*|element2)
  ```

- Value in element1 is exactly equal to the constant or element2 value:

  ```
  (element1 = constant|element2)
  ```

- Value in element1 is not equal to the constant or element2 value:

  ```
  (element1 <>  constant|element2)
  ```

- Value in element1 is less than the constant or element2 value:

  ```
  (element1 <  constant|element2)
  ```

- Value in element1 is less than or equal to the constant or element2 value:

  ```
  (element1 <=  constant|element2)
  ```

- Value in element1 is greater than the constant:

  ```
  (element1 <=  constant|element2)
  ```

- Value in element1 is greater than the constant or element2 value:

  ```
  (element1 >  constant|element2)
  ```

- Value in element1 is greater than or equal to the constant or element2 value:

  ```
  element1 >=  constant|element2)
  ```

Chapter 6 Creating MAINVIEW VistaPoint Charts

MAINVIEW VistaPoint has the capability to create charts from tabular data normally shown in views. Rather than look at VistaPoint views consisting of rows and columns of numbers, you can convert selected view data into a chart.

VistaPoint uses the MAINVIEW GraphManager facility to convert view data into charts. This chapter explains how to use GraphManager to

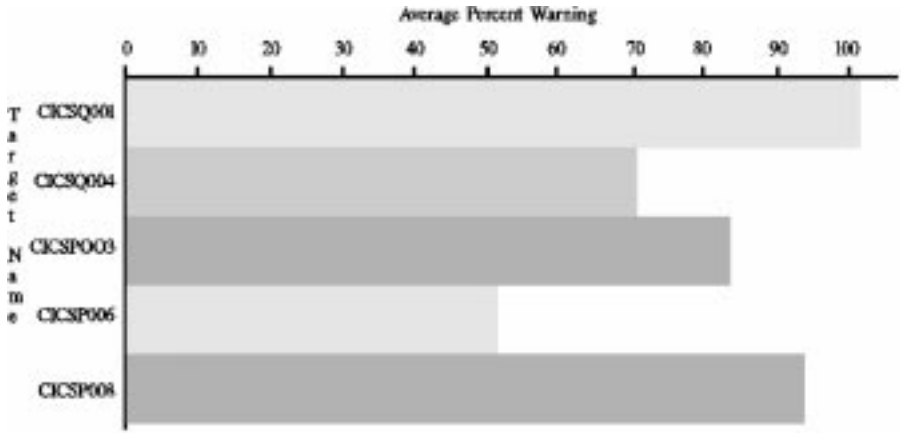
- Display a chart
- Customize a chart
- Print a chart or save it in a picture file for later viewing

Understanding the Different Chart Types

GraphManager produces high- and low-resolution charts. The type of VistaPoint chart you see depends on the type of terminal you are using and whether your site has IBM Graphical Data Display Manager (GDDM) software installed. GraphManager defaults to low-resolution charts if you are using a terminal that cannot support high-resolution displays or if GDDM libraries are not allocated.

Figure 6-1 on page 6-2 shows a high-resolution chart. You can display high-resolution charts if you are using an IBM 3279, 3179G, 3290, 3270-PC terminal, or the equivalent. Also, GDDM libraries must be allocated to display high-resolution charts.

Figure 6-1 MAINVIEW VistaPoint High-Resolution Chart

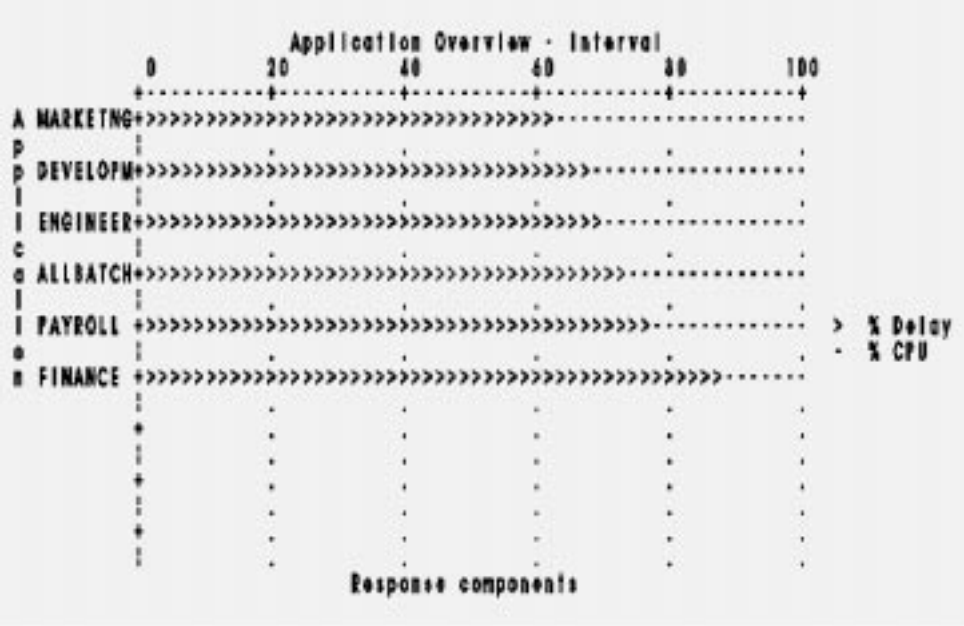


GraphManager uses GDDM to control the format and display options of high-resolution charts. Using GraphManager, you can specify the following characteristics for your high-resolution charts:

- colors, line type, text font, and shading patterns of the graph areas
- scale, range, grid lines, titles, and labels of the X and Y axes

Figure 6-2 shows a typical VistaPoint low-resolution chart. GraphManager uses the ISPF dialog management services to construct low-resolution charts.

Figure 6-2 Low-Resolution Chart of APOVER Response-Time Components



Notice the details of the low-resolution chart shown in Figure 6-2 on page 6-2. A low-resolution chart is monochrome and uses standard keyboard characters to represent the graph's curves and lines.

Displaying Charts

The APOVER and WKOVER views are distributed with predefined charts. A predefined chart has preselected data elements, display attributes, and chart format. By issuing the GRAph command, you can display the default chart for both VistaPoint views.

To display the default chart for a VistaPoint view:

1. Display the view.

2. Enter **GRA** from the **COMMAND** line.

GraphManager automatically creates a high-resolution chart or defaults to a low-resolution chart, depending upon whether you have an appropriate terminal and GDDM allocated.

After a chart is displayed, you can customize it and save the new chart definition. In addition, you can save a displayed chart in a picture file data set for later use.

If a chart is not displayed, a default chart may not be defined for this view. In this case, the message `NO GRAPHABLE COLUMNS` appears in the upper right corner of the view. You need to customize this view to create a default chart for the view.

Customizing Charts

You may want to customize a VistaPoint view to include a chart. For example, you want to create a graph with APGRPR so that it is displayed every time you issue the GRAPH command. Although there are several ways to accomplish this, the recommended procedure is as follows:

Step 1 Display APGRPR.

Step 2 Enter **CUST** on the **COMMAND** line to enter view customization.

Step 3 Select option **G - Graph.**

An example of the dynamic customization window for the Graph option is shown in Figure 6-3.

Figure 6-3 Creating a Chart for the APGRPR View

```
------------------------ VIEW CUSTOMIZATION - APGRPR --------------------------
OPTION  ===> G                                              SCROLL ===> PAGE
Options: (that require column selection)       Other options:
F - Format    M - Move       I - Include       G - Graph      S - Save view
O - Order     R - Repeat     X - Exclude       P - Parameters E - Show excluded
L - Filter    T - Threshold  H - Hyperlink     Z - Summarize  K - Show template
--------------------< Graph - Specify columns for graphing >------------------
X=> A       Chart Type => $SBARH   Library => U       (D-dist, S-site, Userid)
1=> F   5=>     Title  => Applications by Type - R
2=>     6=>     X-axis => Application          Select X-axis and Y-axis columns
3=>     7=>     Y-axis => Realtime % Obj       and enter optional titles.
4=>     8=>                 Preview chart => y (Y/N)
--------------------------------------------------------------------------------
    A        D    E     F              G     H     I        J        K
CMD Appl     Type       Realtime % Obj Worst Best  Resp     Tran     Total
--- -------- ----       0.....50...100 %Obj  % Obj Time---- Total--- Wklds
    TSO      MVS   56.0 ********        56.0  56.0    0.420     2098     2
    OLTPWORK DB2   86.1 ************    86.1  86.1    0.836      911     1
    PAYROLL  CICS  90.9 *************   72.8 103.9    0.836     1111     6
    FINANCE  DB2   95.7 *************   95.7  95.7    0.836     2016     1
    FINANCE  IMS   95.9 *************   95.9  95.9    0.836      711     1
    TESTAPPL IMS  100.0 ************** 100.0 100.0    0.000      987     1
```

Step 4 Follow the instructions shown in the following table to select the elements from a view to create a custom chart.

To change the...	Do this...
Element used for the X axis	Enter a highlighted letter in the **X** field. The highlighted letters are shown above the view columns in the graph customization window.
Element(s) used for the Y axis	Enter the highlighted column letter in the fields 1 through 8. The data you select to appear on the Y-axis must be numeric.
Title of the graph	Enter the new title in the **Title** field.
X-axis label	Enter a description of the data shown on X-axis.
Y-axis label	Enter a description of the data shown on Y-axis.
Chart type	If you know the name of the *chart definition* you want to use, enter its name in the **Chart Type** field. (To see a list of chart definition names and their descriptions, press your HELP key and scroll down until the list is visible.) If you do not know the name, continue following the numbered steps.

Step 5 To test your changes, type **Y** in the **Preview chart** field and press **Enter**. Press your END key to return to view customization.

Step 6 If you are satisfied with your changes, press **Enter** to communicate your changes to ISPF, then press your END key to exit view customization. Be sure to save your modifications by entering **YES** in the **Save changes** field.

Tip: To save a chart definition, the BBTLIB data set must be allocated to your user ID. Typically, BBTLIB is allocated automatically when you access MAINVIEW. If the data set is not allocated to your user ID, see the *MAINVIEW Common Customization Guide* or your system administrator for allocation instructions.

If you want to change the chart type, but do not know which type to use, continue with this procedure.

Step 7 From view customization, type **Y** in the **Preview chart** field and press **Enter** to display the graph.

Step 8 Press **PF1** to enter GraphManager.

The Chart Selection panel is displayed.

Note: For specific information about the fields and options available from this panel, press the HELP key.

If you have a high-resolution terminal, the Chart Selection panel looks like this:

Figure 6-4 High-Resolution Chart Selection Panel

```
------------------------ CHART SELECTION -  $SBARH ----------------- CHART: 1
OPTION  ===>                                          HARDCOPY PFKEY ==> 4

   A - List, select, update chart definitions         MULTIPLE CHART ==> N
   B - Display data item selection list               CHART LOCATION ==>
   C - Redefine current chart specifications           1 L.half   2 R.half
   blank - Generate chart                              3 L.top    4 R.top
                                                       5 L.bottom 6 R.bottom
CHART TYPE  ==> 6  (Enter one of the chart types listed)
CHART TITLE ==> Applications by Type               COMMON HEADING ==> N

1 Line graph   4 Overlay surf 7 Stacked bar  10 3D bar   A Annotation only
2 Scatterplot  5 Histogram    8 Overlay bar  11 3D surf  T Tabular display
3 Stacked surf 6 Multiple bar 9 Pie chart               I ICU (GDDM)

Data items currently selected:
 X-axis: APWKAPPL
 Y-axis: APWKROBJ
```

If you have a low-resolution terminal, the panel looks like this:

Figure 6-5 Low-Resolution Chart Selection Panel

```
------------------------ CHART SELECTION -  $SBARH --------------------------
OPTION  ===>

   A - List, select, update chart definitions          LEGEND POSITION  ==> R
   B - Display data item selection list                (B=bottom,T=top,R=right)
   C - Redefine current axis range and labels
   blank - Generate chart

CHART TYPE  ==> 3   (Enter one of the chart types listed)
CHART TITLE ==> Applications by type

 1 Line graph    2 Scatterplot   3 Stacked bar   4 Overlay bar    5 Pie chart

Data items currently selected:
X-axis: APWKAPPL
Y-axis: APWKROBJ

For hardcopy of a displayed chart, press the ISPF defined PRINT PFK.
```

Step 9 To choose a new chart type, select option **A** from the Chart Selection panel.

The Chart Definition panel is displayed, which contains a list of predefined chart definitions from which to choose.

Step 10 Type an **S** next to the chart definition that you want to display. Press **Enter** twice to see the chart.

Continue selecting and displaying different chart definitions until you find the one that you want to use for APOVER.

If you do not find the definition that you want, or if you find one but want to make changes to it, go to step 11.

If you are satisfied with one of the standard definitions, go to step 12 on page 6-7.

Step 11 If you do not find the chart type that you want in one of the standard definitions, or if you want to change a standard definition, you must create a new chart definition. To do so:

11.A Select a standard definition that is *most similar* to the chart type you want to use.

11.B Press **Enter** to return to the Chart Selection panel.

11.C Change the chart type by entering the number corresponding to the chart type you want in the **CHART TYPE** field.

11.D If you have a high-resolution terminal, select option **C** and make additional changes to your graph.

11.E When you are satisfied with your changes, select option **A** from the Chart Specification panel.

11.F Assign a name to your customized chart in the **CHART DEFINITION NAME** field.

11.G To save the chart in your site-wide chart library, enter an **S** in the **CHART LIBRARY** field. To save the chart in your personal chart library, enter your user ID in the **CHART LIBRARY** field. (You cannot save a customized chart in the distributed library.)

11.H Press END until you return to view customization.

11.I Notice how the chart definition you selected is now displayed in the **Chart Type** field. Make any other changes in the Graph dynamic window and press **Enter**. Go to step 12.

Step 12 Press your END key to save the chart and exit view customization.

The modified chart is now saved with APGRPR and will be displayed every time you enter GRAph from that view.

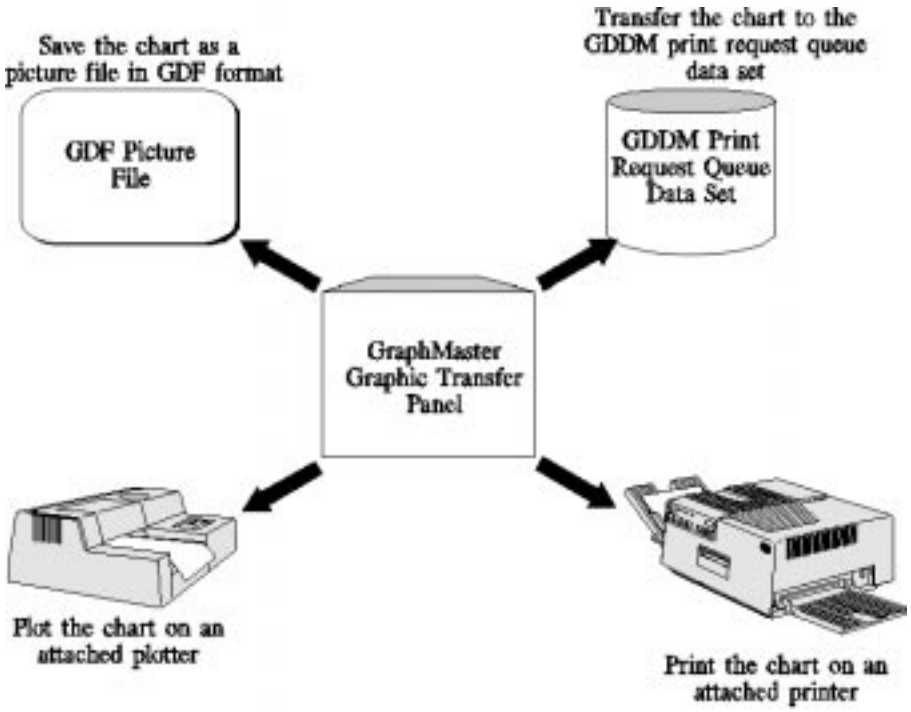
Creating Permanent VistaPoint Charts

Occasionally, you may need a permanent copy of your VistaPoint charts. GraphManager can produce permanent charts that you can save in different formats. You can print or plot the charts to produce hardcopy, or save them as permanent picture files that you can view online with GDDM.

The number of methods to create permanent charts depends on whether you are viewing low-resolution or high-resolution charts. You are limited to printing low-resolution charts through your ISPF print facility.

You have more options with high-resolution charts. Figure 6-6 on page 6-8 shows four different ways to create permanent high-resolution charts. In each case, you select an option from the GraphManager Graphic Transfer panel to create one of the permanent chart types shown in Figure 6-6.

Figure 6-6 Creating Permanent High-Resolution Charts



The remainder of this chapter discusses the four transfer options shown in Figure 6-6.

Saving High-Resolution Charts as Picture Files

You can transfer VistaPoint charts to permanent picture files. The files are converted to a device-independent Graphic Data Format (GDF) and stored as members of a partitioned data set. After that, you can view these permanent charts with the MAINVIEW PICture command.

Step 1 Allocate a permanent graphics PDS if one is not already present. It must be a cataloged PDS with a fixed record length of 400 and can be either blocked or unblocked. The data set should have the following attributes:

RECFM=F or B
LRECL=400

Step 2 Display the chart you want to save as a picture file.

Step 3 Press the HARDCOPY option key to display the Graphics Transfer panel shown in Figure 6-7 on page 6-10.

Step 4 Fill in the fields beneath the **Picture File** area f the Graphics Transfer panel shown in Figure 6-7 on page 10.

- Data set name. Enter the name of your permanent graphics PDS.
- Member name. Enter the member name of the GDF file that holds the screen image of your VistaPoint chart.

or

- Member prefix

Enter a member prefix of up to six characters. A prefix is used instead of a member name to save a series of GDF files. The related files can be displayed sequentially in a slide-show presentation.

A two-digit suffix is appended to the prefix to create a unique member name. A suffix starts with the number 01 and is incremented for each saved member as long as the prefix remains unchanged during the session.

Step 5 Enter **2** on the **Command** line and press **Enter**.

Step 6 Issue the PICture command from any view to see the permanent screen images of your VistaPoint charts.

Transferring High-Resolution Charts to the GDDM Print Request Queue Data Set

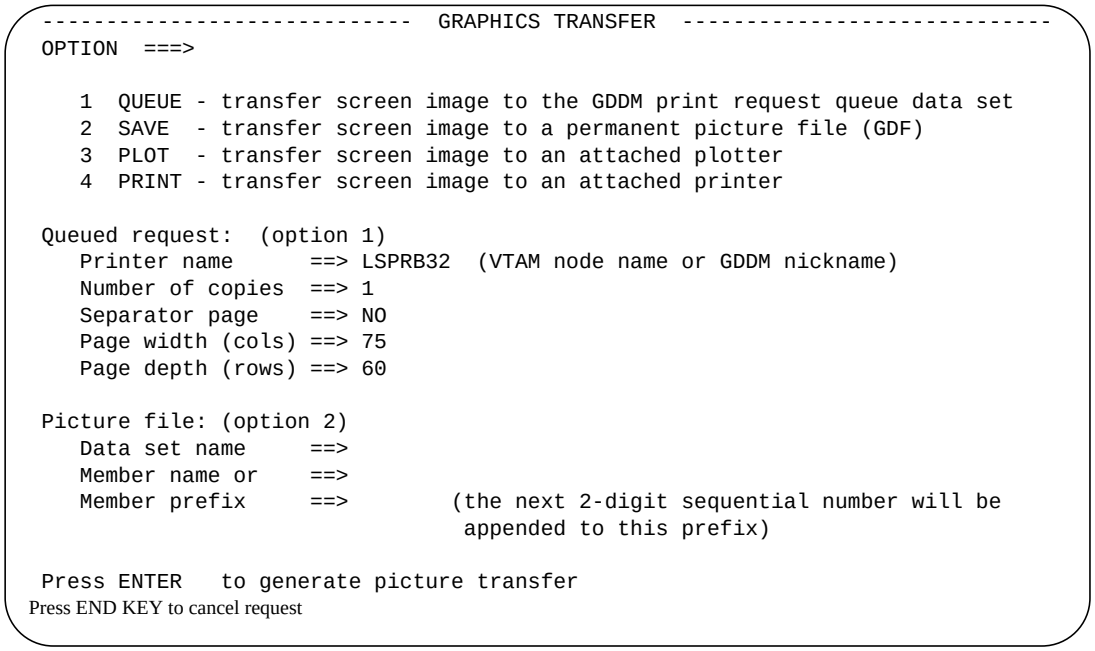
High-resolution graphs can be transferred to the GDDM Print Request Queue data set. The screen image of the chart is sent to the GDDM print request queue on a specific VTAM node or nickname.

Step 1 Display the graph you want to transfer to the GDDM Print Request Queue data set.

Notice the number of the PF key assigned to the **HARDCOPY** option shown in the lower right corner of your chart screen.

Step 2 Press the HARDCOPY PF key (PF4 by default) to display the Graphic Transfer panel shown in Figure 6-7 on page 6-10.

Figure 6-7 Graphics Transfer Panel



Step 3 Fill in the fields beneath the **Queued request** area

Printer name	Enter the VTAM node name or nickname of the printer. You can specify a GDDM nickname and reroute the output to a queued plotter. If you use a nickname, you must allocate the file to the DD name ADMDEFS.
Number of copies	Enter the number of copies you want printed. The default is 1.
Separator page	Enter YES if you want a banner page preceding each printed copy of your chart. The banner page shows the date and user ID. The default is YES.
Page width	Enter the number of columns to set the width of the chart. Each column is equivalent to a single character, and the default is 80 columns.
Page depth	Enter the number of rows to set the depth (height) of the chart. The printed chart is scaled to fit within this depth. The default is 60 rows.

Step 4 Select option **1**, QUEUE, by typing **1** on the **Command line**.

Transferring High-Resolution Charts to an Attached Plotter

You can transfer your chart screen images to an attached plotter. A plotter can be attached directly to a 3179G terminal or a PC workstation.

Step 1 Display the chart you want to print with a plotter.

Step 2 Press the HARDCOPY option key to display the Graphics Transfer panel shown in Figure 6-7 on page 6-10.

Step 3 Select option 3, PLOTTER, and press **Enter**.

Transferring Charts to a Locally Attached Printer

High-resolution charts can be printed to your locally attached printer. A printer can be attached directly to a 3179G terminal or a PC workstation.

Step 1 Display the chart you want to print.

Step 2 Press the HARDCOPY PF key to display the Graphics Transfer panel shown in Figure 6-7 on page 6-10.

Step 3 Select option 4, PRINT, to transfer the chart image to your attached printer.

Printing Low-Resolution Charts

Step 1 Display the chart you want to print.

Step 2 Press your PRINT PF key. You can change or designate your PRINT PF key with option 0, Parameters, from the MAINVIEW Selection Menu.

Step 3 Print the file according to your site's standards.

Figure 6-8 on page 6-12 shows a printed copy of a low-resolution chart. Notice that the printed chart still uses standard keyboards characters to represent the lines and curves of the graph.

Figure 6-8 Printed Low-Resolution Chart

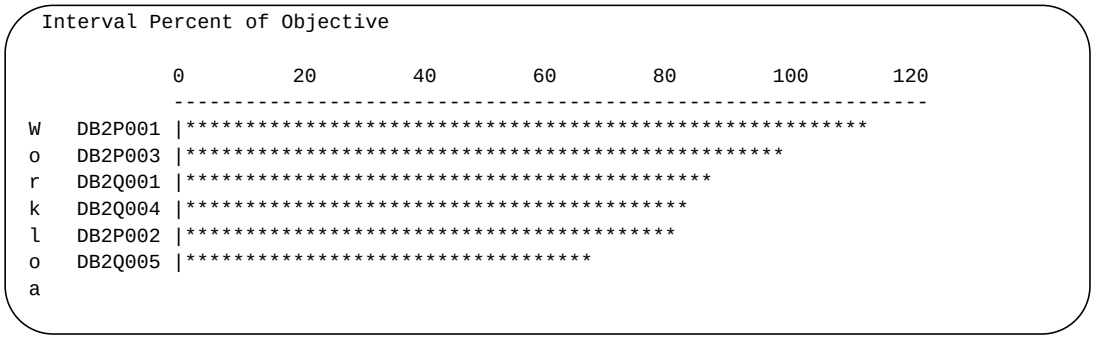


Table 6-1 summarizes the steps in the preceding procedures for creating permanent high-resolution charts. Choose a procedure according to the type of permanent chart you want to create.

Table 6-1 Summary of MAINVIEW VistaPoint Chart Transfer Procedures

To do this...	Follow this procedure...
Send a screen image of a chart to the GDDM Print Request Queue data set.	Type 1 (QUEUE) in the **OPTION** field. Fill in the Queued request: fields on the Graphics Transfer panel. Press **Enter**.
Convert the graph to a permanent picture file and store it as a member in a data set for later use. A *picture file* is a device-independent member that represents a screen image saved in Graphic Data Format (GDF).	If you or your site does not already have a graphics PDS, allocate one on auxiliary storage with these attributes: RECFM=F or FB LRECL=400 Type 2 (SAVE) in the **OPTION** field. Specify the name of the PDS in the **Data set name** field. Specify the member name you want to use in the **Member name** field or a prefix of up to six characters in the **Member prefix** field. Press **Enter**. **Note:** The Member prefix field allows you to logically group a set of related graphs. GraphManager appends a two-digit suffix (01 - 99) to the prefix for each unique graph. For example, if you specified the prefix TAXES, the first graph is saved as TAXES01, the second as TAXES02, and so on. Then, when you are ready to display all of the TAXES graphs, you can use the PICture command to display the graphs in numerical order.
Transfer the graph to an attached plotter. A plotter can be attached to a 3179G or 3270 PC/G(X).	Enter **3** (PLOT) in the **OPTION** field and press **Enter**.
Transfer the graph to a locally attached printer. A printer can be attached to a 3179G or 3270 PC/G(X).	Enter **4** (PRINT) in the **OPTION** field and press **Enter**.

Chapter 7 Generating and Managing Batch Reports

MAINVIEW can obtain historical performance reports through the submission of batch jobs. An ISPF dialog will assist you in generating the JCL to produce MAINVIEW batch reports of your historical data.

This chapter contains the information you need to

- set up the MAINVIEW batch environment
- generate the MAINVIEW batch report JCL
- manage the MAINVIEW batch report JCL members

You may want to submit a job each day to report on some of the key elements of performance for the previous day. The TIME command has several parameters that enable you to specify time frames relative to today. This enables you to submit the same job on a periodic basis without having to change the JCL (i.e., TIME yday 15:00 4h produces a report for every four hours starting at 3:00 pm yesterday).

With this feature, you can enter the command in an ISPF dialog much as you would if you were in an online session. The tabular or detail report is directed to a data set or SYSOUT. The report output is nearly the same format as the online tabular and detailed displays; however, it will show all rows from the query and show as many columns as your data set allows.

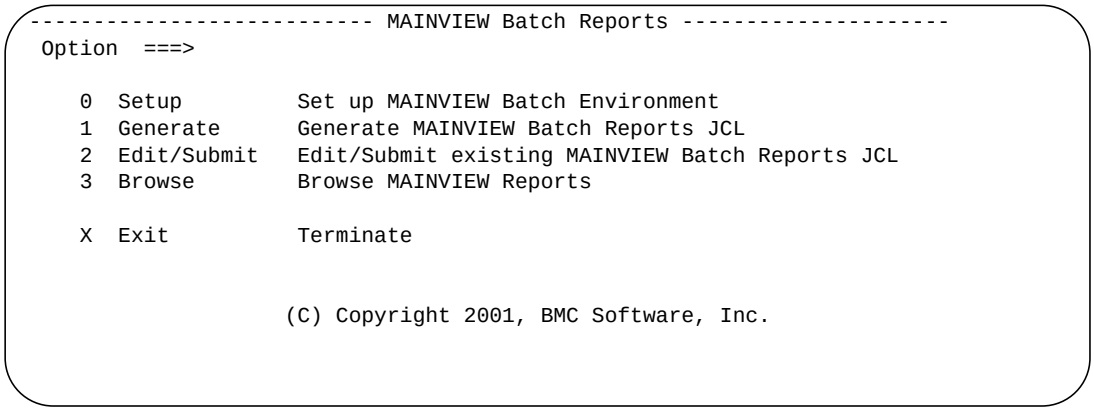
JCL Generation

The ISPF dialog helps generate the JCL for reports you wish to run periodically, and keeps track of previously generated report JCL members. Each report can have up to 16 queries.

To set up JCL for reports, select **MVBATCH**.

The MAINVIEW Batch Reports panel is displayed, as shown in Figure 7-1.

Figure 7-1 MAINVIEW Batch Reports Panel



The MAINVIEW Batch Reports panel provides the following options:

Setup	Displays the MAINVIEW Batch Environment Setup panel where job and report information are recorded. This information is unlikely to change much.
Generate	Displays the Generate MAINVIEW Batch Reports JCL panel where information specific to each report is recorded, saved, and submitted to generate the JCL.
Edit/Submit	Displays the MAINVIEW Batch JCL Member List panel where JCL members are stored. The stored members can be browsed, deleted, edited, and resubmitted.
Browse	Displays the MAINVIEW Batch Reports panel where batch reports are stored. The list shows that the report is either stored in a sequential data set or in a member of a partitioned data set (PDS). The stored reports can be browsed.

Setting Up the MAINVIEW Batch Environment

You will want to set up the MAINVIEW batch environment for your reports. Once you select the setup information, little change will be made from one report to another.

To set up the MAINVIEW Batch JCL:

Step 1 In the MAINVIEW Batch Reports panel, on the **Option** line, type **0**.

The MAINVIEW Batch Environment Setup panel is displayed, as shown in Figure 7-2.

Figure 7-2 MAINVIEW Batch Environment Setup Panel

```
---------------------- MAINVIEW Batch Environment Setup ----------------------
Command ===>

Job Statement Information:
 ===> //USERID   JOB  (ACCOUNT),'NAME'
 ===> //*
 ===> //*
 ===> //*
 ===> //*

Report Title ===>

MAINVIEW Clist Library..... ===> hilevel.BBCLIB
MAINVIEW Clist Name........ ===> MAINVIEW
MAINVIEW BBLINK Library.... ===> hilevel.BBLINK
CAS SSID................... ===> BBCS

Library to save JCL........ ===> 'userid.export.cntl'
Temporary Workfile Unit.... ===> VIO

Press END to save changes and return to the previous panel
Type CANCEL to return to the previous panel without saving changes
```

This panel provides the options for selecting the job and report information you want for the JCL to generate your report. To do this:

Step 2 Under **Job Statement Information**, type a job card that conforms to your installation standards.

Step 3 In the **Report Title** field, type the title you want printed at the top of your reports.

Step 4 In the **MAINVIEW Clist Library** field, type the name of the library containing the MAINVIEW CLIST.

Step 5 In the **MAINVIEW Clist Name** field, type the name of the MAINVIEW CLIST.

Step 6 In the **MAINVIEW BBLINK Library** field, type the name of the MAINVIEW BBLINK library.

Step 7 In the **CAS SSID** field, type the four-character CAS subsystem ID.

Step 8 In the **Library to save JCL** field, type an ISPF library name for the saved JCL.

Step 9 In the **Temporary Workfile Unit** field, type a unit name for the temporary data sets.

Step 10 Press End to save your changes. To return to the MAINVIEW Batch Environment Setup panel without making or saving changes, on the **COMMAND** line, type **CANCEL**.

Generating MAINVIEW Batch Report JCL

Before generating your JCL report, some specific information is necessary.

To add the necessary information:

Step 1 In the MAINVIEW Batch Reports panel, on the **Option** line, type **1**.

The Generate MAINVIEW Batch Reports JCL panel is displayed, as shown in Figure 7-3.

Figure 7-3 Generate MAINVIEW Batch Reports JCL Panel

```
---------------------- Generate MAINVIEW Batch Reports JCL ----------------------
 Command ===>

 JCL Member name        ===> JOBNAME   Replace (Y/N)?  YES
 JCL Member Description ===> Description                              <===
 Report format:         ===> ASIS      (ASIS or CSV)
 Lines/Page:            ===> 60        (ASIS format only)

 Sysout Class ===>
 Or
 Output data set ===> EXPORT.LST
          Volume ===>         (If data set uncatalogued)

 Enter Queries on the lines below.  Each line will write a separate
 report to the output data set.

 Press END to save changes and generate the JCL
 Type CANCEL to return to the previous panel without saving changes
                                                                  More:     +
  ===> TIME LASTWEEKDAY 23:59 1D;APGRP
  ===> TIME *-7;JCPU
  ===> APOVER
  ===> WKBTGTZ
  ===>
```

The Generate MAINVIEW Batch Reports JCL panel contains fields for recording the information and the queries necessary for your JCL.

Step 2 Type the input information, output information, and queries you need for your report.

The fields are as follows:

JCL Member name	Name for this JCL member. Each JCL report can be given a member name and a description.
JCL Member Description	Description of the report. This is optional.
Report format	ASIS or CSV format. Enter ASIS to print reports that look like the screens. Enter CSV for comma-separated fields to download the record to a spreadsheet program.
Lines/Page	For ASIS reports, the heading will be printed on each page. For a continuous report with the heading on the first page only, enter **0**.
Sysout Class	For entering report SYSOUT class.
Output data set	For entering a sequential data set or partitioned data set with a member name for report output. If the data set is not catalogued, supply the volser. Entering a SYSOUT class overrides the data set specification.
Queries	Kinds of data you want to see. Enter queries exactly as you would on the **COMMAND** line in an online session. For example: • **APOVER** prints the APOVER view. • **APOVER;FORM APGRP** prints APOVER with the APGRP form. • **APOVER** on the first line followed by **FORM APGRP** on the second line prints the APOVER report first followed by the APOVER report under the APGRP form. To establish a time frame and duration that are different from the current time, combine that different timeframe and duration with the first command. **TIME LASTWEEKDAY 23:59 1D;APOVER** **Note:** This time period will remain in effect until it is changed by a subsequent query. Through the use of one-per-line commands, additional reports of data can be created from the same timeframe.

The generated JCL executes the MAINVIEW CLIST. The CAS and PAS must be started *before* the JCL is executed

Step 3 Press the End key to save changes and generate the JCL.

The JCL will be presented in an edit session. You are not expected to have the need to make changes.

Step 4 To submit the job, on the **COMMAND** line, type **SUB**.

Step 5 Press **Enter**.

To cancel and return to the MAINVIEW Batch Reports panel without saving changes, type **CANCEL**.

Managing MAINVIEW Batch Report JCL Members

The Edit/Submit option on the MAINVIEW Batch Reports panel opens the JCL member list of generated records. This list provides options to browse, delete, edit, and submit the JCL.

To display the MAINVIEW Batch JCL Member List panel, on the MAINVIEW Batch Reports panel **Option** line, type **2**.

The MAINVIEW Batch JCL Member List panel is displayed with a table of stored JCL members, as shown in Figure 7-4.

Figure 7-4 MAINVIEW Batch JCL Member List Panel

```
----------------------- MAINVIEW Batch JCL Member List ------ Row 1 to 6 of 6
Command ===>                                                 Scroll  ===> PAGE

 Line Commands: B - Browse JCL   DEL - Delete JCL   E - Edit JCL
                SUB - Submit JCL

 Press END to return to main menu
                                                                          >>>
LC  Member   Description                              Date       Time
--- -------- ---------------------------------------- ---------- --------
    WCPU     Weekly CPU Report                        14:01:42   1998/08/21
    DCPU     Daily CPU Report                         06:48:14   1998/09/21
    WDEVICE  Weekly Device Report                     09:06:46   1998/09/22
    DDEVICE  Daily Device Report                      08:45:39   1998/09/18
****************************** Bottom of data ******************************
```

The table is two panels wide. The directional arrows (>>> or <<<), above the list and on the right, indicate that additional JCL member information is available. Use the right scroll key to see information on the right and use the left scroll key to return to the information on the left.

To edit, browse, delete, and submit the JCL in this member list, type

- **E** to edit a member
- **B** to browse the JCL
- **DEL** to delete a member
- **SUB** to submit the job

Note: When a specific JCL member is deleted, the corresponding report data set or member is also deleted.

Batch Report Output Members

The Browse option on the MAINVIEW Batch Reports panel opens the MAINVIEW Batch Report List panel. This panel provides a list of generated online batch members, that can be browsed.

To display the MAINVIEW Batch Report List panel, on the MAINVIEW Batch Reports panel **Option** line, type **3**.

The MAINVIEW Batch Report List panel is displayed, as shown in Figure 7-5 on page 7-8.

Figure 7-5 MAINVIEW Batch Report List Panel

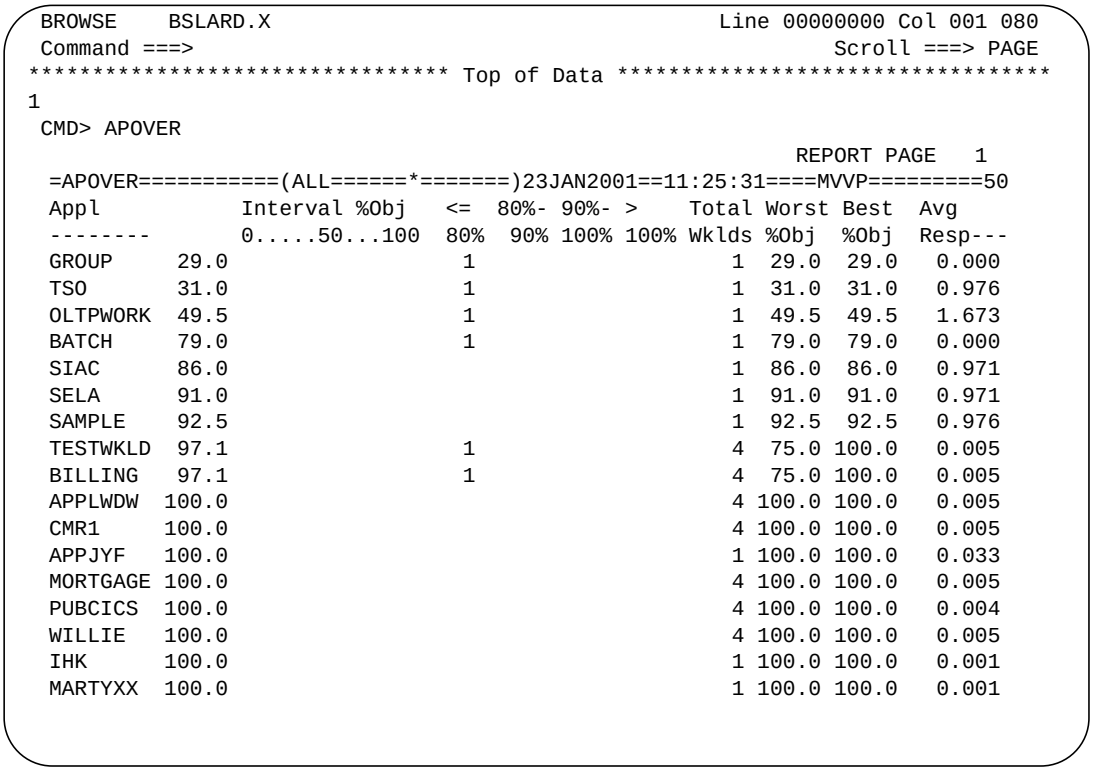
```
------------------------- MAINVIEW Batch Report List ------------ Row 1 of 20
Command ===>                                                  Scroll  ===> PAGE

  S  Select member to view report online

 Press END to return to MAINVIEW Batch Reports Menu

LC  Member   Reports Data Set                                  UserID
--- -------- ---------------------------------------------  -------
    BPA0583A BMVDID.BPA0583.JCL(BPA0583L)                    BMVDID3
    BPA0583B BMVDID.BPA0583.JCL(BPA0583M)                    BMVDID3
    CACHE    BMVJOJ.TESTPO.LST(CACHE)                        BMVJOJ2
    COUPLING BMVJOJ.TESTPO.LST(COUPLING)                     BMVJOJ2
    DEVICES  BBSECH8.JCL.CNTL(DEVZ)                          BBSECH8
    JDELAYS  BMVJOJ.TESTPO.LST(DELAYS)                       BMVJOJ2
    JONJMV   BMVJOJ.ZYYXXX.LST(JONJMV)                       BMVJOJ2
    JOVER    BMVJOJ.COOL                                     BMVJOJ2
    LOST     BMVJOJ.TESTPO.LST(A9)                           BMVJOJ2
    MVBATCH  SYSOUT(*)                                       BBGST09
    MVBATCH1 BMVJOJ.A3DLIB(MVBATCH1)                         BMVJOJ2
    MVBATCH2 BSLARD.X                                        BSLARD1
    MYBATCH  SYSOUT(R)                                       BMVJOJ2
    TERRY1   SYSOUT(A)                                       BBGST06
    TESTJN   BMVJOJ.TESTPO.LST(TESTJ2)                       BBGST12
    TEST2    BBGST05.PRNT.TEST(DATA2)                        BBGST05
    TST1     BMVJOJ.TDSTPO.LST                               BMVJOJ2
```

To browse a report, in the **LC** column on the left side of the report member, type **S**.

In Figure 7-5, the member MVBATCH2 in the BSLARD.X reports data set is selected.

The MAINVIEW Batch Report list panel for BSLARD.X is displayed, as shown in Figure 7-6 on page 7-9.

Figure 7-6 MAINVIEW Batch Report List Panel, BSLARD.X



```
 BROWSE    BSLARD.X                                    Line 00000000 Col 001 080
 Command ===>                                            Scroll ===> PAGE
******************************** Top of Data ********************************
1
 CMD> APOVER
                                                              REPORT PAGE   1
 =APOVER==========(ALL======*=======)23JAN2001==11:25:31====MVVP=========50
 Appl           Interval %Obj   <=  80%- 90%- >    Total Worst Best  Avg
 --------       0.....50...100  80%  90% 100% 100% Wkld %Obj  %Obj  Resp---
 GROUP     29.0                  1                    1  29.0  29.0   0.000
 TSO       31.0                  1                    1  31.0  31.0   0.976
 OLTPWORK  49.5                  1                    1  49.5  49.5   1.673
 BATCH     79.0                  1                    1  79.0  79.0   0.000
 SIAC      86.0                                       1  86.0  86.0   0.971
 SELA      91.0                                       1  91.0  91.0   0.971
 SAMPLE    92.5                                       1  92.5  92.5   0.976
 TESTWKLD  97.1                  1                    4  75.0 100.0   0.005
 BILLING   97.1                  1                    4  75.0 100.0   0.005
 APPLWDW  100.0                                       4 100.0 100.0   0.005
 CMR1     100.0                                       4 100.0 100.0   0.005
 APPJYF   100.0                                       1 100.0 100.0   0.033
 MORTGAGE 100.0                                       4 100.0 100.0   0.005
 PUBCICS  100.0                                       4 100.0 100.0   0.004
 WILLIE   100.0                                       4 100.0 100.0   0.005
 IHK      100.0                                       1 100.0 100.0   0.001
 MARTYXX  100.0                                       1 100.0 100.0   0.001
```

Appendix A View Conventions

This appendix describes MAINVIEW conventions used to display numeric and graphic data in views. These conventions are adhered to by all of the Software MAINVIEW products.

Numeric Values

If a number fits within the column width of a field, it is displayed in its entirety. When a number exceeds the column width of a field, MAINVIEW attempts to fit the number by several methods. The following list shows the methods in the order they are employed to fit a number within a field's column width.

1. Insignificant low-order decimal values are rounded.

2. Low-order digits are truncated. Decimal numbers are converted to integers if necessary.

3. Numbers are expressed in powers of 1000 using the following notation:

K	Kilo (1000)
M	Mega (1,000,000)
G	Giga (1,000,000,000)
T	Terra (1,000,000,000,000)

If a number is still too large to display within the field width, asterisks (*) are displayed in the field to indicate that the numeric value is too large to display.

Graph Values

A value within the lower and upper boundaries of a graph is displayed as a horizontal bar. Depending on your terminal, the magnitude of the graphed value may be depicted as a high-resolution horizontal bar or simply keyboard characters in the case of low-resolution terminals.

A value exceeding an upper graph boundary has a plus sign (+) to indicate the horizontal bar has been truncated at the upper boundary limit. The example below shows a value exceeding the 100% upper boundary limit of a graph displayed on a low-resolution terminal.

```
  W1
=CSERV=============CICSPROD=*========08NOV2000===09:34:53====MVCICS========2
 CMD Serv  Parm             % Warning    Curr     Warn     Area Target
 --- ----- --------       0.....50...100 Value--- Value--- ---- --------
     DSUT  ECDSA     176.0 **************+     88.00     50.00
STOR CICSPROD
```

Appendix B Calculated Values

This appendix describes the calculated appearing in VistaPoint views and the views of the products that work with VistaPoint. Each section of the appendix briefly describes the calculated value. A formula is shown with an example that demonstrates how the value is calculated.

Calculated Values Shown in VistaPoint Views

VistaPoint views show the following calculated values.

Percent Objective

Percent objective expresses the ratio of the current monitored value to the objective goal.

Calculation

$$\text{percent objective} = \left(\frac{\text{percentage successful transactions}}{\text{percentage goal by workload}} \times 100\% \right)$$

Example

Consider the case where a workload had 125 transactions over a sampling period. Only 68 transactions met the response time goal specified in the workload definition. The percentage goal for successful transactions is 80%.

$$\text{percent objective} = 54.4\% = \frac{68}{125} \text{ X } 100\%$$
$$80\%$$

Average Percent Objective

Average percent objective represents the average percentage of successful transactions that occurred in an application's workloads during the sampling period. A successful transaction completes within the response-time goal set in the workload definition.

Calculation

$$\text{average percent objective} = \frac{\sum\left(\text{number transactions} \times \left(\frac{\text{percent successful}}{\text{percentage goal}}\right)\right)}{\sum\left(\text{number transactions}\right)} \times 100\%$$

Example

This example expands on the earlier example shown for percent objective. Workload 1 had 68 successful transactions out of 125 in a sampling period. Again, the defined percentage goal for successful workloads is 80%.

Workload 2 had 40 successful transaction out of 60 in the same sampling period. This workload's percentage goal is 75%.

```
                                    54.4%
Workload 1 =  125 transactions X    ------- = 85.00
                                    80.0%


                                    88.8%
Workload 2 =   60 transactions X    ------- = 71.04
                                    75.0%
               -------                        ---------
               185 total transactions          156.04 weighted successful
                                                      transactions


                                      156.04
average percent objective = 84.35% = -------- X 100%
                                        185
```

Average Response Time

Average response time represents the average length of time for a transaction to complete. It is calculated as a weighted average from the reported transactions occurring in an application's workloads during a sampling period.

Calculation

$$\text{average response time} = \left(\frac{\text{total response time from all workloads with an application}}{\text{number of events}}\right) \text{X } 100\%$$

Example

If the sum of response times for 115 transactions in a sampling period is 2.004 seconds, then the average response time is

```
                                    2.004 seconds    1000 milliseconds
                                    -------------  X -----------------
                                          1                1 second
 average response time = 17.42 msec = ---------------------------------
                                              115 transactions
```

Average CPU Time

Average CPU time represents the average length of time that a transaction spends in CPU processing. It is calculated from the reported transactions occurring in an application's workloads during a sampling period.

Calculation

$$\text{Average CPU Time} = \frac{\text{total CPU time from application workloads}}{\text{total transactions by application}}$$

Example

If the total CPU time is 1.234 seconds for the same 115 transactions in the example given for average response time, then:

$$\text{average CPU time time} = 10.73 \text{ msec} = \frac{\frac{1.234 \text{ seconds}}{1} \times \frac{1000 \text{ millisecond}}{1 \text{ seconds}}}{115 \text{ transactions}}$$

Percent CPU

Percent CPU gives the percentage of average response time that a transaction spends in CPU processing.

Calculation

$$\text{percent CPU} = \left(\frac{\text{average CPU time (seconds)}}{\text{average response time seconds}} \right) \times 100\%$$

Example

```
W1=WKDETL===========(PRODUCTN=DB2P====)08NOV2000====11:32:45====MVVP======1
   Application FINANCE                    0.....50...100
   Type....... DB2      Avg %Obj     77.4  ***********
   System..... MVSA     Response   10.907
   Target..... DB2P
                        AvgCPU..    0.202
                        % CPU...        2

                        Delay...   10.704
                        % Delay.       98  **************

                        Tot Tran       34  0.....10....20
                        TranRate    0.283
                        TranTime 00:02:00


                   0.202
Percent CPU = 2 = ------ x 100%
                   10.907
```

Average Delay

Average delay is the difference between average response time and average CPU time. Delay is the portion of response time not spent in CPU processing. Delay includes queuing for CPU, I/O, or other resources.

Calculation

$$\text{average delay} = \text{average response time} - \text{average CPU time}$$

Example

```
W1=WKDETL============CICSP01==*========08NOV2000====11:51:41====MVVP======1
     Application FINANCE                    0.....50...100
     Type....... CICS     Avg %Obj    105.3  ************+
     System..... MVSB     Response    0.200
     Target..... CICSP01
                          AvgCPU..    0.016
                          % CPU...        8  *

                          Delay...    0.184
                          % Delay.       92  ************
```

```
average delay = 0.184 = 0.200 - 0.016
```

Percent Delay

Percent delay gives the percentage of response time that a transaction spends waiting for system resources. Delay is the difference between response time and CPU time.

Calculation

$$\text{percent delay} = \left(\frac{\text{delay (seconds)}}{\text{response time (seconds)}} \right) \text{X } 100\%$$

Example

```
W1=WKDETL============CICSP01==*========08NOV2000====11:51:41====MVVP======1
     Application FINANCE                      0.....50...100
     Type....... CICS     Avg %Obj    105.3  ************+
     System..... MVSB     Response    0.200
     Target..... CICSP01
                          AvgCPU..    0.016
                          % CPU...        8  *

                          Delay...    0.184
                          % Delay.       92  *************


                        0.184 (seconds)
Percent delay = 92% =  --------------- x 100%
                         0.200 (seconds)
```

Transaction Rate

Transaction rate is the average number of transactions per second over the data collection period.

Calculation

$$\text{transactions per second} = \frac{\text{total number of transactions}}{\left(\text{length of data collection period (minutes)} \div \frac{60 \text{ seconds}}{1 \text{ minute}}\right)}$$

Example

```
<W1=APOVER===========(PRODUCTN=*=======)08NOV2000====11:00:56====MVVP=====8
CMD Appl     Avg     Avg     Avg     Tran    Total
--- -------- Resp--- CPU---- Delay-- Rate--- Trans---
    FINANCE    3.576   0.014   0.000   0.310      186
```

In this example, the data collection period is a 10-minute interval.

```
                        186 transactions
Tran Rate  = 0.310  =  ----------------------------
per second             10 minutes    60 seconds
                                   &div. ----------
                                     1  minute
```

Transaction Time

Transaction time expresses the current cumulative time of the data collection period: real time, interval, or session. Each period has a characteristic calculation.

Calculation

- Real-time Transaction Time

  None. Transaction time always shows the length of the real-time period.

- Interval

  Interval transaction time is the product of the number of real-time periods in the current interval and the length of the real-time period.

- Session Transaction Time

  Session transaction time is the product of the number of real-time periods in the current session and the length of the real-time period.

Example

```
Tot Tran       34  0.....10....20
TranRate    6.120  *****
TranTime 00:01:00
```

Real-time transaction time always shows the length of a real-time period. In the example shown above, the length of a real-time period is 01:00 minute.

```
Tot Tran      139  0.....10....20
TranRate    9.283  *******
TranTime 00:05:00
```

Interval transaction time is calculated as the product of the number of real-time periods in the current interval and the length of a real-time period. In the example shown above, the current interval transaction time is 05:00 minutes, which is the product of five real-time periods of 01:00 minute duration.

```
Tot Tran      634  0.....10....20
TranRate    8.321  ******
TranTime 01:10:00
```

Session transaction time is calculated as the product of the number of real-time periods in the current session and the length of a real-time period. In the example shown above, there have been 70 (01:10:00) real-time periods of 01:00 minute duration.

real-time Percent of Objective

Real-time percent of objective gives the ratio of successful transactions that occurred in a real-time sampling period to the percentage goal specified in the workload definition. A successful transaction completes in less time than the response time goal set in the workload definition.

Calculation

$$\text{realtime percent objective} = \left(\frac{\text{percent successful transactions (realtime period)}}{\text{percentage goal}}\right) \text{ X } 100\%$$

Example

The service-level objective of a workload specifies that 90% of all transactions must complete in 1 second. Over the current real-time period, 47 transactions out of 60 have completed in 1 second or less.

$$\text{realtime percent objective} = 87.03\% = \frac{47 \text{ successful transactions}}{60 \text{ totaltransactions}} \times \frac{100}{0.90}$$

Interval Percent of Objective

Interval percent of objective gives the ratio of successful transactions that occurred in the current interval to the percentage goal specified in the workload definition. A successful transaction completes in less time than the response-time goal set in the workload definition.

Calculation

$$\text{interval percent objective} = \left(\frac{\text{percent successful transactions (interval)}}{\text{percentage goal}}\right) \text{X } 100\%$$

Example

The service-level objective of a workload is 90% of all transactions must complete in 1 second. Over the current interval, 236 transactions out of 298 were completed in 1 second or less.

$$\text{interval percent objective} = 87.99\% = \frac{\frac{236 \text{ successful transactions}}{298 \text{ total transactions}}}{0.90} \times 100$$

Session Percent of Objective

Session percent of objective gives the ratio of successful transactions that occurred in a session to the percentage goal specified in the workload definition. A successful transaction completes in less time than the response-time goal set in the workload definition.

Calculation

$$\text{session percent objective} = \left(\frac{\text{percent successful transactions (session)}}{\text{percentage goal}}\right) \text{X } 100\%$$

Example

The service-level objective of a workload is 90% of all transactions must complete in 1 second. In the current session, 2246 transactions out of 2576 were completed in 1 second or less.

```
                                2246 successful transaction
                                ---------------------------
session                        2576 total transactions
percent =  96.88% = ------------------------------- x 100
objective                                  0.90
```

Calculated Values Shown in Views of Other MAINVIEW Products

The views of the products that work with VistaPoint display the following calculated values.

Number in Warning

Number in warning reports the number of monitors in the current reporting period that are exceeding their warning thresholds.

Calculation

None.

Example

```
W1=DMON==============DB2PROD==*========08NOV2000====04:38:06====MVDB2=====1
CMD Target   Actv     # in Warn             Avg % Warning  Maximum
--- -------- Mntrs    0.....10....20        0.....50...100 % Warning
    DB2PROD     16  4 ****             37.0 *****               37.0
```

Percent Warning

Percent warning expresses the ratio of the current measured value to the warning value specified for the monitor collecting the data.

Calculation

$$\text{percent warning} = \left(\frac{\text{current value}}{\text{warning value}}\right) \text{X } 100\%$$

Example

```
W1=CSERV============(PRODUCTN=*=======)08NOV2000====12:49:34====MVCICS====8
CMD Serv  Parm             % Warning    Curr     Warn     Area Target
--- ----- --------       0.....50...100 Value--- Value--- ---- --------
    @RSTM WORKWORK 205.0 *************+     0.41     0.20 WKLD CICSPROD
                           0.41
percent warning = 205.0 =  ----- x 100%
                           0.20
```

Average Percent Warning (Target)

Average percent warning by target represents a performance summary of the active monitors in a CICS region, DB2 subsystem, or IMS region. The value is calculated by taking the sum of the percent warning of all monitors in the target and dividing by the number of monitors.

Calculation

$$\text{average percent warning by target} = \left(\frac{\Sigma\left(\frac{\text{current value (monitor)}}{\text{warning value (monitor)}}\right)}{\text{total monitors (target)}}\right) \text{X } 100\%$$

Example

DSERVR shows the percent warning for three monitors running in subsystem DB2A.

```
08NOV2000 03:44:06 ----------- INFORMATION DISPLAY ----------------------
COMMAND  ===>                                          SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1
=DSERVR============DB2P=====*========08NOV2000====03:44:06====MVDB2=======9
CMD Serv  Parm             % Warning    Curr     Warn     Area Target
--- ----- --------       0.....50...100 Value--- Value--- ---- --------
    @ELTM          141.0 *************+   113.50    80.50 USER DB2A
    @ELAP          115.0 *************+     0.92     0.80 WKLD DB2A
    DSOPN           59.0 ********         118.00   200.00 DSYS DB2A
```

DMONR shows the average percent warning for the three active monitors running in the target DB2A subsystem.

```
08NOV2000 03:44:09 ----------- INFORMATION DISPLAY ----------------------
COMMAND  ===>                                          SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>
 W1
=DMONR=============DB2P=====*========08NOV2000====03:44:09====MVDB2=======1
CMD Target   Actv     Number in Warn        Avg % Warning  Maximum
--- -------- Mntrs    0.....10....20        0.....50...100 % Warning
    DB2P         3  2 **              105.0 *************      141.0
```

$$\text{average percent warning (target)} = 105.0\% = \frac{(141.0 + 115.0 + 59.0)}{3} \times 100\%$$

Average Percent Warning (Monitor)

Average percent warning by monitor represents the average performance of the active monitors in a target by type. The value is calculated from the quotient of the derived average value and average warning for similar monitors.

Calculation

$$\text{average percent warning by monitor} = \left(\frac{\Sigma\left(\frac{\text{current value (monitor)}}{\text{warning value (monitor)}}\right)}{\text{number of monitors by type}} \right) \times 100\%$$

Example

DSERVR shows the percent warning for each @ELTD monitor running on subsystem DB2A. In this example, assume both monitors reported an equal number of transactions.

```
08NOV2000 03:49:06 ----------- INFORMATION DISPLAY ----------------------
COMMAND  ===>                                            SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>

W1=DSERVR============DB2P=====*========08NOV2000====03:49:06====MVDB2=====9
CMD Serv  Parm             % Warning    Curr     Warn     Area Target
--- ----- --------       0.....50...100 Value--- Value--- ---- --------
    @ELTD          141.0 *************+     1.13     0.80 WKLD DB2A
    @ELTD           83.8 ************       0.62     0.74 WKLD DB2A
```

DSOVERR shows the average percent warning for two @ELTD monitors running in the target DB2A subsystem. Average percent warning is derived from the values shown in the **Avg Value** and **Avg Warning** fields of the view.

```
08NOV2000 03:50:09 ----------- INFORMATION DISPLAY ----------------------
COMMAND  ===>                                            SCROLL ===> PAGE
CURR WIN ===> 1        ALT WIN ===>

W1=DSOVERR===========DB2P=====*========08NOV2000====03:50:09====MVDB2====10
CMD Serv  Parm           Avg % Warning  Avg      Avg      Area Count
--- ----- --------       0.....50...100 Value--- Warning- ---- -----
    @ELTD          113.6 *************+     0.88     0.77 DSYS     2
```

Average value and average warning are intermediate average calculations shown in the DSOVERR view. Both values are used to calculate average percent warning.

Average value and average warning are calculated from each monitor's current value and warning value shown in the DSERVR view.

$$\text{average value} = 0.88 = \frac{(1.13 + 0.62)}{2}$$

$$\text{average warning} = 0.77 = \frac{(0.80 + 0.74)}{2}$$

$$\text{average percent warning (monitor)} = 114.3\% = \frac{0.88}{0.77} \times 100\%$$

Percentage Less Than or Equal to Objective

Percentage less than or equal to objective gives the percentage of transactions with response times less than or equal to the maximum threshold limit set in a VistaPoint workload definition.

Calculation

$$\text{percent <= objective} = \left(\frac{\text{number of transactions <= response time objective}}{\text{transaction count}}\right) \text{ X } 100\%$$

Example

```
W1=DOBJS============(ALL======*=======)08NOV2000====13:53:31====MVDB2====10
CMD Workload Avg          % <= Resp Goal Resp  Goal Tran  Composite Target
--- Name---- Resp---      0.....50...100 Goal-    % Count Workload- -------
   ACCTRECV   1.348  68.4 *********       1.00   95  6531 FINANCE   DB2P
```

In this example, 4467 transactions completed with response-times less than or equal to the 1.00 second objective.

$$\text{Percent <= objective} = 68.4 = \frac{4467}{6531} \times 100\%$$

Although not shown, the number of transactions less than or equal to the response time objective can be derived by the product transaction count x % <= Obj.

$$\text{Number of transactions <= objective} = 4467 = \frac{68.4\%}{100\%} \times 6531$$

Glossary

This glossary defines BMC Software terminology. Other dictionaries and glossaries may be used in conjunction with this glossary.

Since this glossary pertains to BMC Software-related products, some of the terms defined may not appear in this book.

To help you find the information you need, this glossary uses the following cross-references:

Contrast with indicates a term that has a contrary or contradictory meaning.

See indicates an entry that is a synonym or contains expanded information.

See also indicates an entry that contains related information.

action Defined operation, such as modifying a MAINVIEW window, that is performed in response to a command. *See* object.

active window Any MAINVIEW window in which data can be refreshed. *See* alternate window, current window, window.

administrative view Display from which a product's management tasks are performed, such as the DSLIST view for managing historical data sets. *See* view.

ALT WIN field Input field that allows you to specify the window identifier for an alternate window where the results of a hyperlink are displayed. *See* alternate window.

Alternate Access *See* MAINVIEW Alternate Access.

alternate form View requested through the FORM command that changes the format of a previously displayed view to show related information. *See also* form, query.

alternate window (1) Window that is specifically selected to display the results of a hyperlink. (2) Window whose identifier is defined to the ALT WIN field. *Contrast with* current window. *See* active window, window, ALT WIN field.

analyzer (1) Online display that presents a snapshot of status and activity data and indicates problem areas. (2) Component of CMF MONITOR. *See* CMF MONITOR Analyzer.

application (1) Program that performs a specific set of tasks within a MAINVIEW product. (2) In MAINVIEW VistaPoint, combination of workloads to enable display of their transaction performance data in a single view.

application trace *See* trace.

ASCH workload Workload comprising Advanced Program-to-Program Communication (APPC) address spaces.

AutoCustomization Online facility for customizing the installation of products. AutoCustomization provides an ISPF panel interface that both presents customization steps in sequence and provides current status information about the progress of the installation.

automatic screen update
Usage mode wherein the currently displayed screen is refreshed automatically with new data at an interval you specify. Invoked by the ASU command.

batch workload Workload consisting of address spaces running batch jobs.

BBI Basic architecture that distributes work between workstations and multiple OS/390 targets for BMC Software MAINVIEW products.

BBI-SS PAS *See* BBI subsystem product address space.

BBI subsystem product address space (BBI-SS PAS)
OS/390 subsystem address space that manages communication between local and remote systems and that contains one or more of the following products:

- MAINVIEW AutoOPERATOR
- MAINVIEW for CICS
- MAINVIEW for DB2
- MAINVIEW for DBCTL
- MAINVIEW for IMS Online
- MAINVIEW for MQSeries (formerly Command MQ for S/390)
- MAINVIEW for VTAM
- MAINVIEW VistaPoint (for CICS, DB2, DBCTl, and IMS workloads)

BBPARM *See* parameter library.

BBPROC *See* procedure library.

BBPROF *See* profile library.

BBSAMP *See* sample library.

BBV *See* MAINVIEW Alternate Access.

BBXS BMC Software Subsystem Services. Common set of service routines loaded into common storage and used by several BMC Software MAINVIEW products.

border Visual indication of the boundaries of a window.

bottleneck analysis Process of determining which resources have insufficient capacity to provide acceptable service levels and that therefore can cause performance problems.

CA-Disk Data management system by Computer Associates that replaced the DMS product.

CAS Coordinating address space. One of the address spaces used by the MAINVIEW windows environment architecture. The CAS supplies common services and enables communication between linked systems. Each OS/390 image requires a separate CAS. Cross-system communication is established through the CAS using VTAM and XCF communication links.

CFMON *See* coupling facility monitoring.

chart Display format for graphical data. *See also* graph.

CICSplex User-defined set of one or more CICS systems that are controlled and managed as a single functional entity.

CMF MONITOR Comprehensive Management Facility MONITOR. Product that measures and reports on all critical system resources, such as CPU, channel, and device usage; memory, paging, and swapping activity; and workload performance.

CMF MONITOR Analyzer
Batch component of CMF MONITOR that reads the SMF user and 70 series records created by the CMF MONITOR Extractor and/or the RMF Extractor and formats them into printed system performance reports.

CMF MONITOR Extractor
Component of CMF that collects performance statistics for CMF MONITOR Analyzer, CMF MONITOR Online, MAINVIEW for OS/390, and RMF postprocessor. *See* CMF MONITOR Analyzer, CMF MONITOR Online, MAINVIEW for OS/390.

CMF MONITOR Online
Component of CMF that uses the MAINVIEW window interface to present data on all address spaces, their use of various system resources, and the delays that each address space incurs while waiting for access to these resources. *See* CMF MONITOR, MAINVIEW for OS/390.

CMF Type 79 API Application programming interface, provided by CMF, that provides access to MAINVIEW SMF-type 79 records.

CMFMON Component of CMF MONITOR that simplifies online retrieval of information about system hardware and application performance and creates MAINVIEW SMF-type 79 records.

The CMFMON *online facility* can be used to view data in one or more formatted screens.

The CMFMON *write facility* can be used to write collected data as MAINVIEW SMF-type 79 records to an SMF or sequential data set.

CMRDETL MAINVIEW for CICS data set that stores detail transaction records (type 6E) and abend records (type 6D). Detail records are logged for each successful transaction. Abend records are written when an abend occurs. Both records have the same format when stored on CMRDETL.

CMRSTAT MAINVIEW for CICS data set that stores both CICS operational statistic records, at 5-minute intervals, and other records, at intervals defined by parameters specified during customization (using CMRSOPT).

column Vertical component of a view or display, typically containing fields of the same type of information, that varies by the objects associated in each row.

collection interval Length of time data is collected. *See also* delta mode, total mode.

command delimiter Special character, usually a **;** (semicolon), used to stack commands typed concurrently on the COMMAND line for sequential execution.

COMMAND line Line in the control area of the display screen where primary commands can be typed. *Contrast with* line command column.

Command MQ Automation D/S
Command MQ agents, which provide local proactive monitoring for both MQSeries and MSMQ (Microsoft message queue manager). The Command MQ agents operate at the local node level where they continue to perform functions regardless of the availability of the MQM (message queue manager) network. Functionality includes automatic monitoring and restarts of channels, queue managers, queues and command servers. In cases where automated recovery is not possible, the agents transport critical alert information to a central console.

Command MQ Automation S/390

Command MQ component, which monitors the MQM (message queue manager) networks and intercedes to perform corrective actions when problems arise. Solutions include:

- Dead-Letter Queue management
- System Queue Archival
- Service Interval Performance solutions
- Channel Availability

These solutions help ensure immediate relief to some of the most pressing MQM operations and performance problems.

Command MQ for D/S

Command MQ for D/S utilizes a true client/server architecture and employs resident agents to provide configuration, administration, performance monitoring and operations management for the MQM (message queue manager) network.

Command MQ for S/390

See MAINVIEW for MQSeries.

COMMON STORAGE MONITOR

Component of MAINVIEW for OS/390 that monitors usage and reconfigures OS/390 common storage blocks.

composite workload Workload made up of a WLM workload or other workloads, which are called *constituent workloads.*

constituent workload

Member of a composite workload. Constituent workloads in a composite usually belong to a single workload class, but sometimes are mixed.

contention Occurs when there are more requests for service than there are servers available.

context In a Plex Manager view, field that contains the name of a target or group of targets specified with the CONTEXT command. *See* scope, service point, SSI context, target context.

CONTEXT command Specifies either a MAINVIEW product and a specific target for that product (*see* target context) or a MAINVIEW product and a name representing one or more targets (*see* SSI context) for that product.

control statement (1) Statement that interrupts a sequence of instructions and transfers control to another part of the program. (2) Statement that names samplers and other parameters that configure the MAINVIEW components to perform specified functions. (3) In CMF MONITOR, statement in a parameter library member used to identify a sampler in the extractor or a report in the analyzer, or to describe either component's processing requirements to the operating system.

coupling facility monitoring (CFMON) Coupling facility views that monitor the activity of your system's coupling facilities.

CPO Customized Product Offering. Delivery and installation technique that allows any combination of BMC Software SMP/E-maintainable products to be distributed on a product tape to a customer and installed quickly. The CPO product tape contains libraries required for product customization and execution, plus SMP distribution libraries and data sets needed for application of SMP maintenance.

current data Data that reflects the system in its current state. The two types of current data are realtime data and interval data. *Contrast with* historical data. *See also* interval data, realtime data.

current window In the MAINVIEW window environment, window where the main dialog with the application takes place. The current window is used as the default window destination for commands issued on the COMMAND line when no window number is specified. *Contrast with* alternate window. *See* active window, window.

DASD Direct Access Storage Device. (1) A device with rotating recording surfaces that provides immediate access to stored data. (2) Any device that responds to a DASD program.

DASD ADVISOR An interactive software tool that diagnoses DASD performance problems and makes recommendations to reduce overall service time. This tool measures and reports on the operational performance of IBM and IBM-compatible devices.

data collector Program that belongs to a MAINVIEW product and that collects data from various sources and stores the data in records used by views. For example, MAINVIEW for OS/390 data collectors obtain data from OS/390 services, OS/390 control blocks, CMF MONITOR Extractor control blocks, and other sources. *Contrast with* extractor.

delta mode (1) In MAINVIEW for DB2 analyzer displays, difference between the value sampled at the start of the current statistics interval and the value sampled by the current analyzer request. *See also* statistics interval. (2) In CMFMON, usage mode wherein certain columns of data reflect the difference in values between one sample cycle and the next. Invoked by the DELta ON command. *See also* collection interval, sample cycle, total mode.

DFSMS (Data Facility Storage Management System) Data management, backup, and HSM software from IBM for OS/390 mainframes.

DMR *See* MAINVIEW for DB2.

DMS (Data Management System) *See* CA-Disk.

DMS2HSM Component of MAINVIEW SRM that facilitates the conversion of CA-Disk, formerly known as DMS, to HSM.

DSO Data Set Optimizer. CMF MONITOR Extractor component that uses CMF MONITOR Extractor data to produce reports specifying the optimal ordering of data sets on moveable head devices.

EasyHSM Component of MAINVIEW SRM that provides online monitoring and reporting to help storage managers use DFHSM efficiently.

EasyPOOL Component of MAINVIEW SRM that provides control over data set allocation and enforcement of allocation and naming standards. EasyPOOL functions operate at the operating system level to intercept normal job processing, thus providing services without any JCL changes.

EasySMS Component of MAINVIEW SRM that provides tools that aid in the conversion to DFSMS and provides enhancement to the DFSMS environment after implementation. EasySMS consists of the EasyACS functions, the SMSACSTE function, and the Monitoring and Positioning Facility.

element (1) Data component of a data collector record, shown in a view as a field. (2) Internal value of a field in a view, used in product functions.

element help Online help for a field in a view. The preferred term is *field help.*

Enterprise Storage Automation
Component of MAINVIEW SRM that integrates powerful event management technology and storage monitoring technology to provide significant storage automation capabilities and solutions. Storage occurrences are defined to generate events in the form of messages that provide an early warning system for storage problems and are routed to MAINVIEW AutoOPERATOR to be viewed.

Event Collector Component for MAINVIEW for IMS Online, MAINVIEW for IMS Offline, and MAINVIEW for DBCTL that collects data about events in the IMS environment. This data is required for Workload Monitor and optional for Workload Analyzer (except for the workload trace service). This data also is recorded as transaction records (X‘FA’) and program records (X‘F9’) on the IMS system log for later use by the MAINVIEW for IMS Offline components: Performance Reporter and Transaction Accountant.

expand Predefined link from one display to a related display. *See also* hyperlink.

extractor Program that collects data from various sources and keeps the data control blocks to be written as records. Extractors obtain data from services, control blocks, and other sources. *Contrast with* data collector.

extractor interval *See* collection interval.

fast path Predefined link between one screen and another. To use the fast path, place the cursor on a single value in a field and press Enter. The resulting screen displays more detailed information about the selected value. *See also* hyperlink.

field Group of character positions within a screen or report used to type or display specific information.

field help Online help describing the purpose or contents of a field on a screen. To display field help, place the cursor anywhere in a field and press PF1 (HELP). In some products, field help is accessible from the screen help that is displayed when you press PF1.

filter Selection criteria used to limit the number of rows displayed in a view. Data that does not meet the selection criteria is not displayed. A filter is composed of an element, an operator, and an operand (a number or character string). Filters can be implemented in view customization, through the PARm/QPARm commands, or through the Where/QWhere commands. Filters are established against elements of data.

fixed field Field that remains stationary at the left margin of a screen that is scrolled either right or left.

FOCAL POINT MAINVIEW product that displays a summary of key performance indicators across systems, sites, and applications from a single terminal.

form One of two constituent parts of a view; the other is query. A form defines how the data is presented; a query identifies the data required for the view. *See also* query, view.

full-screen mode Display of a MAINVIEW product application or service on the entire screen. There is no window information line. *Contrast with* windows mode.

global command Any MAINVIEW window interface command that can affect all windows in the window area of a MAINVIEW display.

graph Graphical display of data that you select from a MAINVIEW window environment view. *See also* chart.

hilevel For MAINVIEW products, high-level data set qualifier required by a site's naming conventions.

historical data (1) Data that reflects the system as it existed at the end of a past recording interval or the duration of several intervals. (2) Any data stored in the historical database and retrieved using the TIME command. *Contrast with* current data, interval data and realtime data.

historical database Collection of performance data written at the end of each installation-defined recording interval and containing up to 100 VSAM clusters. Data is extracted from the historical database with the TIME command. *See* historical data.

historical data set In MAINVIEW products that display historical data, VSAM cluster file in which data is recorded at regular intervals.

HSM (Hierarchical Storage Management) Automatic movement of files from hard disk to slower, less-expensive storage media. The typical hierarchy is from magnetic disk to optical disk to tape.

hyperlink (1) Preset field in a view or an EXPAND line on a display that permits you to

- Access cursor-sensitive help
- Issue commands
- Link to another view or display

The transfer can be either within a single product or to a related display/view in a different MAINVIEW product. Generally, hyperlinked fields are highlighted. (2) Cursor-activated short path from a topic or term in online help to related information. *See also* fast path.

Image log Collection of screen-display records. Image logs may be created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Image log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to the BBI-SS PAS Image log stops when both data sets are filled and the first data set is not processed by the archive program.

The TS Image log is a single data set that wraps around when full.

IMSPlex System Manager (IPSM)
MVIMS Online and MVDBC service that provides Single System Image views of resources and bottlenecks for applications across one or more IMS regions and systems.

interval data Cumulative data collected during a collection interval. Intervals usually last from 15 to 30 minutes depending on how the recording interval is specified during product customization. *Contrast with* historical data.

Note: If change is made to the workloads, a new interval will be started.

See also current data and realtime data.

InTune Product for improving application program performance. It monitors the program and provides information used to reduce bottlenecks and delays.

IRUF IMS Resource Utilization File (IRUF). IRUFs can be either detailed (one event, one record) or summarized (more than one event, one record). A detailed IRUF is created by processing the IMS system log through a program called IMFLEDIT. A summarized IRUF is created by processing one or more detailed IRUFs, one or more summarized IRUFs, or a combination of both, through a sort program and the TASCOSTR program.

job activity view Report about address space consumption of resources. *See* view.

journal Special-purpose data set that stores the chronological records of operator and system actions.

Journal log Collection of messages. Journal logs are created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Journal log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to the BBI-SS PAS Journal log stops when both data sets are filled and the first data set is not being processed by the archive program.

The TS Journal log is a single data set that wraps around when full.

line command Command that you type in the line command column in a view or display. Line commands initiate actions that apply to the data displayed in that particular row.

line command column
Command input column on the left side of a view or display. *Contrast with* COMMAND line.

Log Edit In the MAINVIEW for IMS Offline program named IMFLEDIT, function that extracts transaction (X‘FA’) and program (X‘F9’) records from the IMS system log. IMFLEDIT also extracts certain records that were recorded on the system log by IMS. IMFLEDIT then formats the records into a file called the IMS Resource Utilization File (IRUF).

MAINVIEW BMC Software integrated systems management architecture.

MAINVIEW Alarm Manager (MV ALARM)
In conjunction with other MAINVIEW products, notifies you when an exception occurs. MAINVIEW Alarm Manager is capable of monitoring multiple systems simultaneously, which means that MAINVIEW Alarm Manager installed on one system keeps track of your entire SYSPLEX. You can then display a single view that shows exceptions for all MAINVIEW performance monitors within your OS/390 enterprise.

MAINVIEW Alternate Access
Enables MAINVIEW products to be used without TSO by providing access through EXCP and VTAM interfaces.

MAINVIEW AutoOPERATOR
Product that uses tools, techniques, and facilities to automate routine operator tasks and provide online performance monitoring, and that achieves high availability through error minimization, improved productivity, and problem prediction and prevention.

MAINVIEW control area
In the MAINVIEW window environment, first three lines at the top of the view containing the window information line and the COMMAND, SCROLL, CURR WIN, and ALT WIN lines. The control area cannot be customized and is part of the information display. *Contrast with* MAINVIEW display area, MAINVIEW window area.

MAINVIEW Desktop Version of the MAINVIEW window interface designed to run on OS/2 and Windows workstations.

MAINVIEW display area
See MAINVIEW window area.

MAINVIEW Explorer Product that provides access to MAINVIEW products from a Web browser running under Windows. MAINVIEW Explorer replaces MAINVIEW Desktop.

MAINVIEW for CICS Product (formerly MV MANAGER for CICS) that provides realtime application performance analysis and monitoring for CICS system management.

MAINVIEW for DB2 Product (formerly MV MANAGER for DB2) that provides realtime and historical application performance analysis and monitoring for DB2 subsystem management.

MAINVIEW for DBCTL (MVDBC)
Product that provides realtime application performance analysis and monitoring for DBCTL management.

MAINVIEW for IMS (MVIMS) Offline
Product with a Performance Reporter component that organizes data and prints reports used to analyze IMS performance and a Transaction Accountant component that produces cost accounting and user charge-back records and reports.

MAINVIEW for IMS (MVIMS) Online
Product that provides realtime application performance analysis and monitoring for IMS management.

MAINVIEW for IP Product that monitors OS/390 mission-critical application performance as it relates to IP stack usage. Collected data includes: connections, response time statistics, application availability, application throughput, and IP configuration.

MAINVIEW for MQSeries (formerly known as Command MQ for S/390)
Delivers comprehensive capabilities for configuration, administration, performance monitoring and operations management for an entire MQM (message queue manager) network.

MAINVIEW for OS/390
System management application (formerly MAINVIEW for MVS (prior to version 2.5)). Built upon the MAINVIEW window environment architecture, it uses the window interface to provide access to system performance data and other functions necessary in the overall management of an enterprise.

MAINVIEW for UNIX System Services
System management application that allows you to monitor the performance of the Unix System Services from a MAINVIEW window interface.

MAINVIEW for VTAM
Product that displays application performance data by application, transaction ID, and LU name. This collected data includes connections, response time statistics, application availability, and application throughput.

MAINVIEW Selection Menu
ISPF selection panel that provides access to all MAINVIEW windows-mode and full-screen mode products.

MAINVIEW Storage Resource Monitor (SRM)
Suite of products that assist in all phases of OS/390 storage management. MAINVIEW SRM consists of components that perform automation, reporting, trend analysis, and error correction for storage management in OS/390.

MAINVIEW SYSPROG Services
See SYSPROG services.

MAINVIEW VistaPoint
Product that provides enterprise-wide views of performance. Application and workload views are available for CICS, DB2, DBCTL, IMS, and OS/390. Data is summarized at the level of detail needed; e.g., reports may be for a single target, an OS/390 image, or an entire enterprise.

MAINVIEW window area
Portion of the information display that is not the control area and in which views are displayed and windows opened. It includes all but the first three lines of the information display. *Contrast with* MAINVIEW control area.

monitor Online service that measures resources or workloads at user-defined intervals and issues warnings when user-defined thresholds are exceeded.

MV MANAGER for CICS
See MAINVIEW for CICS.

MV MANAGER for DB2
See MAINVIEW for DB2.

MV MANAGER for MVS
See MAINVIEW for OS/390.

MVALARM *See* MAINVIEW Alarm Manager.

MVCICS *See* MAINVIEW for CICS.

MVDB2 *See* MAINVIEW for DB2.

MVDBC *See* MAINVIEW for DBCTL.

MVIMS *See* MAINVIEW for IMS.

MVMQ *See* MAINVIEW for MQSeries.

MVMVS *See* MAINVIEW for OS/390.

MVSRM *See* MAINVIEW Storage Resource Manager (SRM).

MVSRMHSM *See* EasyHSM.

MVSRMSGC *See* SG-Control.

MVSRMSGD *See* StorageGUARD.

MVSRMSGP *See* StorageGUARD.

MVUSS *See* MAINVIEW for UNIX System Services.

MVScope MAINVIEW for OS/390 application that traces both CPU usage down to the CSECT level and I/O usage down to the channel program level.

MVVP *See* MAINVIEW VistaPoint.

MVVTAM *See* MAINVIEW for VTAM.

MVWEB *See* MAINVIEW for WebSphere.

nested help Multiple layers of help pop-up windows. Each successive layer is accessed by hyperlinking from the previous layer.

object Anything you can manipulate as a single unit. MAINVIEW objects can be any of the following: product, secondary window, view, row, column, or field.

You can issue an action against an object by issuing a line command in the line command column to the left of the object. *See* action.

OMVS workload Workload consisting of OS/390 OpenEdition address spaces.

online help Help information that is accessible online.

OS/390 and z/OS Installer
BMC Software common installation system for mainframe products.

OS/390 product address space (PAS)
Address space containing OS/390 data collectors, including the CMF MONITOR Extractor. Used by MAINVIEW for OS/390, MAINVIEW for USS, and CMF MONITOR products. *See* PAS.

parameter library Data set comprised of members containing parameters for specific MAINVIEW products or a support component. There can be several versions:

- The distributed parameter library, called BBPARM
- A site-specific parameter library or libraries

These can be

-A library created by AutoCustomization, called UBBPARM
-A library created manually, with a unique name

PAS Product address space. Used by the MAINVIEW products. Contains data collectors and other product functions. *See* OS/390 product address space (PAS), BBI subsystem product address space (BBI-SS PAS).

performance group workload
MVS/SP-defined collection of address spaces. *See* service class workload, workload definition.

PERFORMANCE MANAGER
MAINVIEW for CICS online service for monitoring and managing current performance of CICS regions.

Performance Reporter (MVIMS)
MVIMS Offline component that organizes data and prints reports that can be used to analyze IMS performance.

Performance Reporter
Product component that generates offline batch reports. The following products can generate these reports:

- MAINVIEW for DB2
- MAINVIEW for CICS

Plex Manager Product through which cross-system communication, MAINVIEW security, and an SSI context are established and controlled. Plex Manager is shipped with MAINVIEW window environment products as part of the coordinating address space (CAS) and is accessible as a menu option from the MAINVIEW Selection Menu.

pop-up window Window containing help information that, when active, overlays part of the window area. A pop-up panel is displayed when you issue the HELP command.

PRGP workload In MVS/SP 5.0 or earlier, or in compatibility mode in MVS/SP 5.1 or later, composite of service classes. MAINVIEW for OS/390 creates a performance group workload for each performance group defined in the current IEAIPSxx member.

procedure library Data set comprised of members containing executable procedures used by MAINVIEW AutoOPERATOR. These procedures are execute command lists (EXECs) that automate site functions. There can be several versions:

- The distributed parameter library, called BBPROC

- A site-specific parameter library or libraries

These can be

-A library created by AutoCustomization, called UBBPROC

-A library created manually, with a unique name

The site-created EXECs can be either user-written or customized MAINVIEW AutoOPERATOR-supplied EXECs from BBPROC.

product address space

See PAS.

profile library

Data set comprised of members containing profile information and cycle refresh definitions for a terminal session connected to a BBI-SS PAS. Other members are dynamically created by MAINVIEW applications. There can be several versions:

- The distributed profile library, called BBPROF
- A site-specific profile library or libraries

These can be

-A library created by AutoCustomization, called SBBPROF
-A library created manually, with a unique name

The site library is a common profile shared by all site users. The terminal session CLIST creates a user profile automatically if one does not exist; it is called userid.BBPROF, where userid is your logon ID. User profile libraries allow each user to specify unique PF keys, CYCLE commands, target system defaults, a Primary Option Menu, and a unique set of application profiles.

query

One of two constituent parts of a view; the other is form. A query defines the data for a view; a form defines the display format. *See also* form, view.

realtime data

Performance data as it exists at the moment of inquiry. Realtime data is recorded during the smallest unit of time for data collection. *Contrast with* historical data. *See also* current data and interval data.

Resource Analyzer

Online realtime displays used to analyze IMS resources and determine which are affected by specific workload problems.

Resource Monitor

Online data collection services used to monitor IMS resources and issue warnings when defined utilization thresholds are exceeded.

row (1) Horizontal component of a view or display comprising all the fields pertaining to a single device, address space, user, etc. (2) Horizontal component of a DB2 table consisting of a sequence of values, one for each column of the table.

RxD2 Product that provides access to DB2 from REXX. It provides tools to query the DB2 catalog, issue dynamic SQL, test DB2 applications, analyze EXPLAIN data, generate DDL or DB2 utility JCL, edit DB2 table spaces, perform security administration, and much more.

sample cycle Time between data samples.

For the CMF MONITOR Extractor, this is the time specified in the extractor control statements (usually 1 to 5 seconds).

For realtime data, the cycle is not fixed. Data is sampled each time you press Enter.

sample library Data set comprised of members each of which contains one of the following:

- Sample JCL that can be edited to perform specific functions
- A macro that is referenced in the assembly of user-written services
- A sample user exit routine

There can be several versions:

- The distributed sample library, called BBSAMP

- A site-specific sample library or libraries

These can be

-A library created by AutoCustomization, called UBBSAMP
-A library created manually, with a unique name

sampler Program that monitors a specific aspect of system performance. Includes utilization thresholds used by the Exception Monitor. The CMF MONITOR Extractor contains samplers.

SBBPROF *See* profile library.

scope Subset of an SSI context. The scope could be all the data for the context or a subset of data within the context. It is user- or site-defined. *See* SSI context, target.

screen definition Configuration of one or more views that have been stored with the SAVEScr command and assigned a unique name. A screen includes the layout of the windows and the view, context, system, and product active in each window.

selection view In MAINVIEW products, view displaying a list of available views.

service class workload
OS/390- or MAINVIEW for OS/390-defined collection of address spaces.

If you are running MVS Workload Manager (WLM) in goal mode, MAINVIEW for OS/390 creates a service class workload for each service class that you define through WLM definition dialogs.

If you are running MVS 4.3 or earlier, or MVS/SP 5.1 or later with WLM in compatibility mode, OS/390 creates a performance group workload instead of a service class. *See* performance group workload.

service objective Workload performance goal, specified in terms of response time for TSO workloads or turnaround time for batch workloads. Performance group workloads can be measured by either objective. Composite workload service objectives consist of user-defined weighting factors assigned to each constituent workload. There are no OS/390-related measures of service for started task workloads.

service point Specification, to MAINVIEW, of the services required to enable a specific product. Services may be actions, selectors, or views. Each target (e.g., CICS, DB2, or IMS) has its own service point.

The PLEX view lists all the defined service points known to the CAS to which the terminal session is connected.

service request block (SRB)
Control block that represents a routine to be dispatched. SRB mode routines generally perform work for the operating system at a high priority. An SRB is similar to a task control block (TCB) in that it identifies a unit of work to the system. *See also* task control block.

service select code Code entered to invoke analyzers, monitors, and general services. This code is also the name of the individual service.

session Total period of time an address space has been active. A session begins when monitoring can be performed. If the product address space (PAS) starts after the job, the session starts with the PAS.

SG-Auto Component of MAINVIEW SRM that provides early warning notification of storage anomalies and automated responses to those anomalies based on conditions in the storage subsystem.

SG-Control Component of MAINVIEW SRM that provides real-time monitoring, budgeting, and control of DASD space utilization.

single system image (SSI)
Feature of the MAINVIEW window environment architecture that allows you to view and perform actions on multiple OS/390 systems as though they were a single system. The rows of a single tabular view can contain rows from different OS/390 images.

SRB *See* service request block.

SSI *See* single system image.

SSI context Name created to represent one or more targets for a given product. *See* context, target.

started task workload
Address spaces running jobs that were initiated programmatically.

statistics interval For MAINVIEW for DB2, cumulative count within a predefined interval (30-minute default set by the DB2STATS parameter in the distributed BBPARM member BBIISP00) for an analyzer service DELTA or RATE display. Specifying the DELTA parameter displays the current value as the difference between the value sampled by the current analyzer request and the value sampled at the start of the current interval. Specifying the RATE parameter displays the current value by minute (DELTA divided by the number of elapsed minutes).

StopX37/II Component of MAINVIEW SRM that provides enhancements to OS/390 space management, reducing the incidence of space-related processing problems. The StopX37/II functions operate at the system level to intercept abend conditions or standards violations, thus providing services without any JCL changes.

StorageGUARD Component of MAINVIEW SRM that monitors and reports on DASD consumption and provides historical views to help control current and future DASD usage.

summary view View created from a tabular view using the Summarize option in view customization. A summary view compresses several rows of data into a single row based on the summarize criteria.

SYSPROG services Component of MAINVIEW for OS/390. Over 100 services that detect, diagnose, and correct OS/390 system problems as they occur. Accessible from the OS/390 Performance and Control Main Menu. Note that this is also available as a stand-alone product MAINVIEW SYSPROG Services.

system resource *See* object.

target Entity monitored by one or more MAINVIEW products, such as an OS/390 image, IMS or DB2 subsystem, CICS region, or related workloads across systems. *See* context, scope, SSI context.

target context Single target/product combination. *See* context.

TASCOSTR MAINVIEW for IMS Offline program that summarizes detail and summary IMS Resource Utilization Files (IRUFs) to be used as input to the offline components.

task control block (TCB)
Address space-specific control block that represents a unit of work that is dispatched in the address space in which it was created. *See also* service request block.

TCB *See* task control block.

terminal session (TS)
Single point of control for MAINVIEW products, allowing data manipulation and data display and providing other terminal user services for MAINVIEW products. The terminal session runs in a user address space (either a TSO address space or a standalone address space for EXCP/VTAM access).

TDIR *See* trace log directory.

threshold Specified value used to determine whether the data in a field meets specific criteria.

TLDS *See* trace log data set.

total mode Usage mode in CMFMON wherein certain columns of data reflect the cumulative value between collection intervals. Invoked by the DELta OFF command. *See also* collection interval, delta mode.

trace (1) Record of a series of events chronologically listed as they occur. (2) Online data collection and display services that track transaction activity through DB2, IMS, or CICS.

trace log data set (TLDS)
Single or multiple external VSAM data sets containing summary or detail trace data for later viewing or printing. The trace log(s) can be defined as needed or dynamically allocated by the BBI-SS PAS. Each trace request is assigned its own trace log data set(s).

trace log directory (TDIR)
VSAM linear data set containing one entry for each trace log data set. Each entry indicates the date and time of data set creation, the current status of the data set, the trace target, and other related information.

transaction Specific set of input data that initiates a predefined process or job.

Transaction Accountant MVIMS Offline component that produces cost accounting and user charge-back records and reports.

TS *See* terminal session.

TSO workload Workload that consists of address spaces running TSO sessions.

UAS *See* user address space.

UBBPARM *See* parameter library.

UBBPROC *See* procedure library.

UBBSAMP *See* sample library.

user address space Runs a MAINVIEW terminal session (TS) in TSO, VTAM, or EXCP mode.

User BBPROF *See* profile library.

view Formatted data within a MAINVIEW window, acquired from a product as a result of a view command or action. A view consists of two parts: query and form. *See also* form, job activity view, query.

view definition Meaning of data that appears online, including source of data, selection criteria for data field inclusion and placement, data format, summarization, context, product, view name, hyperlink fields, and threshold conditions.

view command Name of a view that you type on the COMMAND line to display that view.

view command stack Internal stack of up to 10 queries. For each command, the stack contains the filter parameters, sort order, context, product, and timeframe that accompany the view.

view help Online help describing the purpose of a view. To display view help, place the cursor on the view name on the window information line and press PF1 (HELP).

window Area of the MAINVIEW screen in which views and resources are presented. A window has visible boundaries and can be smaller than or equal in size to the MAINVIEW window area. *See* active window, alternate window, current window, MAINVIEW window area.

window information line
Top border of a window. Shows the window identifier, the name of the view displayed in the window, the system, the scope, the product reflected by the window, and the timeframe for which the data in the window is relevant. *See also* window status field.

window number Sequential number assigned by MAINVIEW to each window when it is opened. The window number is the second character in the window status field. *See also* window status field.

window status One-character letter in the window status field that indicates when a window is ready to receive commands, is busy processing commands, is not to be updated, or contains no data. It also indicates when an error has occurred in a window. The window status is the first character in the window status field. *See also* window information line, window status field.

window status field Field on the window information line that shows the current status and assigned number of the window. *See also* window number, window status.

windows mode Display of one or more MAINVIEW product views on a screen that can be divided into a maximum of 20 windows. A window information line defines the top border of each window. *Contrast with* full-screen mode.

WLM workload In goal mode in MVS/SP 5.1 and later, a composite of service classes. MAINVIEW for OS/390 creates a workload for each WLM workload defined in the active service policy.

workflow Measure of system activity that indicates how efficiently system resources are serving the jobs in a workload.

workload (1) Systematic grouping of units of work (e.g., address spaces, CICS transactions, IMS transactions) according to classification criteria established by a system administrator. (2) In OS/390, group of service classes within a service definition.

workload activity view
Tracks workload activity as the workload accesses system resources. A workload activity view measures workload activity in terms of resource consumption and how well the workload activity meets its service objectives.

Workload Analyzer Online data collection and display services used to analyze IMS workloads and determine problem causes.

workload definition Workload created through the WKLIST view. Contains a unique name, a description, an initial status, a current status, and selection criteria by which address spaces are selected for inclusion in the workload. *See* Workload Definition Facility.

Workload Definition Facility
In MAINVIEW for OS/390, WKLIST view and its associated dialogs through which workloads are defined and service objectives set.

workload delay view Tracks workload performance as the workload accesses system resources. A workload delay view measures any delay a workload experiences as it contends for those resources.

Workload Monitor Online data collection services used to monitor IMS workloads and issue warnings when defined thresholds are exceeded.

workload objectives Performance goals for a workload, defined in WKLIST. Objectives may include measures of performance such as response times and batch turnaround times.

Index

A

B

C

D

E

F

G

H

I

L

M

N

O

P

Q

R

S

T

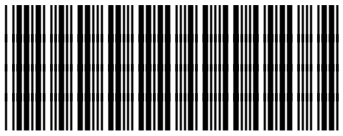
U

V

W

X

Notes



100035693