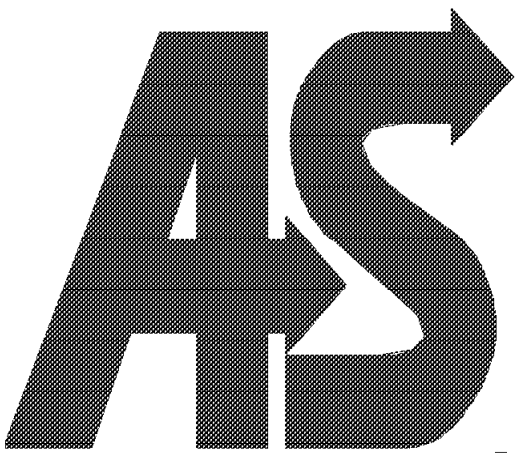
Management Information Systems News



Produced by EAS Product Support working for Workgroup Solutions
EAS is the world-wide centre of competence for AS and Visualizer

Issue 7 - August 1995

AS Version 4.1 Special Edition

August 1995.

This document has not been subjected to any formal review and has not been checked for technical accuracy. Results may be individually evaluated for applicability to a particular installation.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any performance data contained in this document was obtained in a controlled environment based on the use of specific data and is presented only to illustrate techniques and procedures to better understand IBM products. The results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data in their specific environment.

© Copyright International Business Machines Corporation 1995. All rights reserved.
Note to U.S. Government Users — Documentation related to restricted rights — Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Welcome to Application System Version 4 1
The AS Internet home page 2
"Bucking the trend" 3
Announcing the AS Visualizer Command Client 4
The AS Client API 6
AS Client Connections - expanded connectivity and new function 8
CICS/ESA Client feature of AS Version 4 10
Pictograph Attributes in AS Version 4.1 Charts 12
Data Analysis Using AS Version 4.1 14
AS Version 4.1 Performance 20
Application System Version 4.1 - Application Development 22
Now ease-of-use is even easier in AS Version 4 26
External Routines with AS Version 4.1 29
Consolidation Recalculation in Multi-Dimensional Model 30
Using AS with the Hierarchical File System (HFS) 33
AS Version 4.1 Project Management - Consolidator of Choice 37
EAS Technical Bulletins on AS Project Management 39
Info Center/Enhanced in AS Version 4.1 40
Support for Eastern European languages 42

Welcome to Application System Version 4

AS Version 4 lets you combine the power of AS and enterprise data to deliver information to your choice of workstation products or 3270 terminal displays. AS was established as the "intelligent DSS server" for client/server applications by AS Version 3, and in announcing AS Version 4 IBM demonstrates its continued investment in Application System.

This special edition of DSS News gives you some detail of what AS Version 4 contains, including these highlights:

Further exploitation of the Client/Server World

The CICS/ESA Client

Easier Application Development

Data Analysis without programming

Improved performance

Simplified printing

Over 230 customer requirements satisfied.

AS has always kept up-to-date with the technology trends and so you'll also see an article on the Internet. In addition extra capabilities, such as new communication product support for client/server applications, are being added. These new items will be listed on the AS Internet World Wide Web Home Page.

As ever customer feedback is worth its weight in gold to AS Development and Customer Support as it provides guidance, direction and even praise! Here are some comments from customers who have already installed AS Version 4.

"I would like to commend you on the Help (Online reference manual). This is great I can't tell you how great an enhancement this is worth the release upgrade alone."

"It seems that you want to make developers life easier"

"Installation was easy"

"Maintenance is easy"

And finally, my favourite

"AS is going the right direction"

See for yourself.

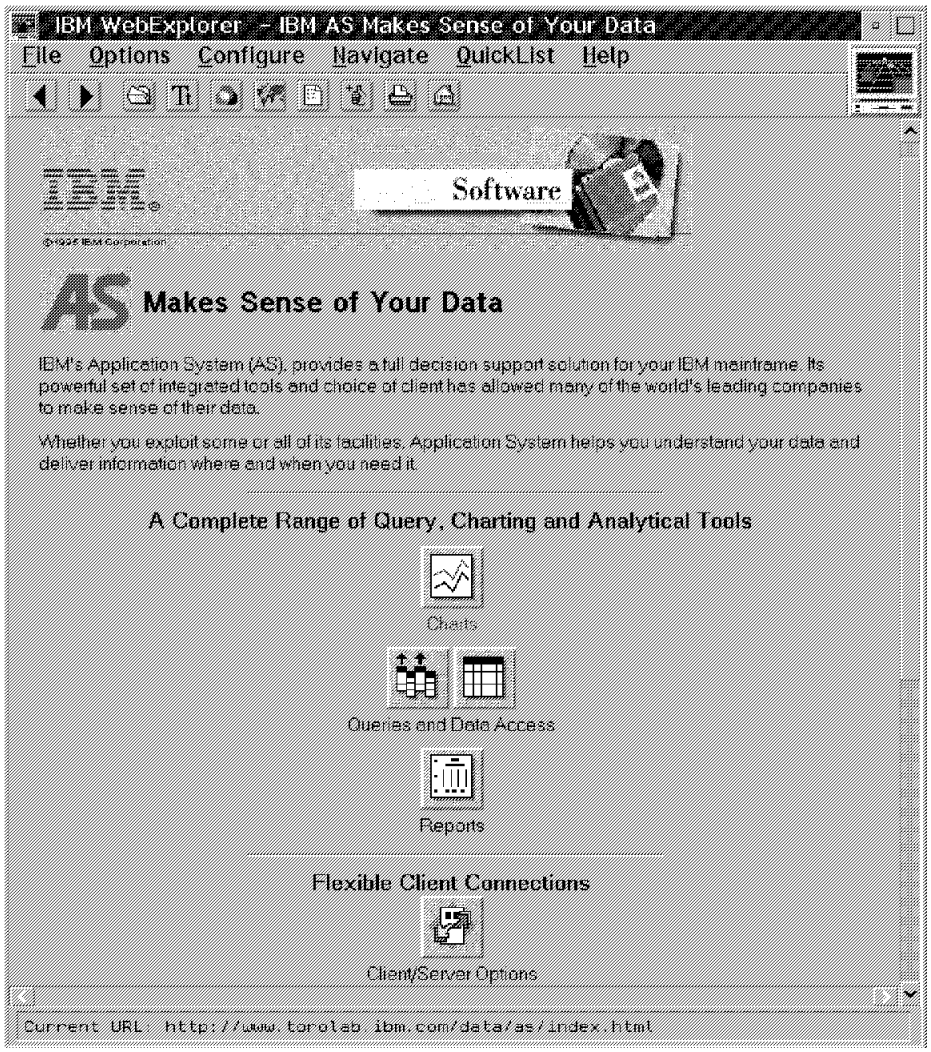
The AS Internet home page

AS has its own pages on the Internet to let you and the rest of the world get information about AS. The pages include icons, hyperlinks and AS screen shots. They are part of the IBM Data Management Software page which can be found from the IBM home page by following the links to Software.

You can get directly to the AS home page using the URL (Web address):

http://www.torolab.ibm.com/data/as/index.html

The Internet provides not only the opportunity to get information on AS, but also the possibility to use AS via the Internet to create pages of information. Here is a glimpse of the AS home page.



"Bucking the trend"

Some people may think it odd that in these days of downsizing from mainframes to smaller machines and personal computing that Application System (AS) continued to sell new licences in 1994, and is continuing to do so very successfully in 1995.

We asked our latest customers for the reasons they decided to "Buck the trend" and stay with a mainframe-based decision support system.

Here are the top five replies:

Business decisions are too critical to just have users "doing their own thing" without having a proper MIS for integrity and support. AS is seen as providing both control and freedom.

Cost of implementing a mainframe solution versus Client/Server. The Return On Investment is not worth it just to have a GUI when AS does it anyway.

The mainframe is already there, let's use it well. Use it differently and use the strengths of mainframe and workstations. AS Client Connection allows function to workstations while exploiting the mainframe.

We want DB2, and AS has the best interface to it, compared to what's on the market. The new Data Analysis functions in AS Version 4.1 make this even better.

Cost of upgrading to latest technology PCs for a large corporation with a high investment in PCs is too much. AS gets the applications started now and allows a gradual, managed move to Client/Server in the future.

These comments and ideas illustrate that the dash to downsizing is not inevitable.

Announcing the AS Visualizer Command Client

On the 24th of May last year the AS team proudly announced the AS Visualizer Command Client for OS/2, a new feature of AS Version 3 Release 2.

For Version 4 we added support for Windows[1], provided national language versions in 10 languages, and made significant functional and usability enhancements.

So what exactly is the AS Command Client?

The AS Command Client is an OS/2 and Windows based PC application with a modern graphical user interface that makes it even easier to develop client/server applications that exploit the analytical power and data access capabilities of AS, the Intelligent Server, from most popular PC products.

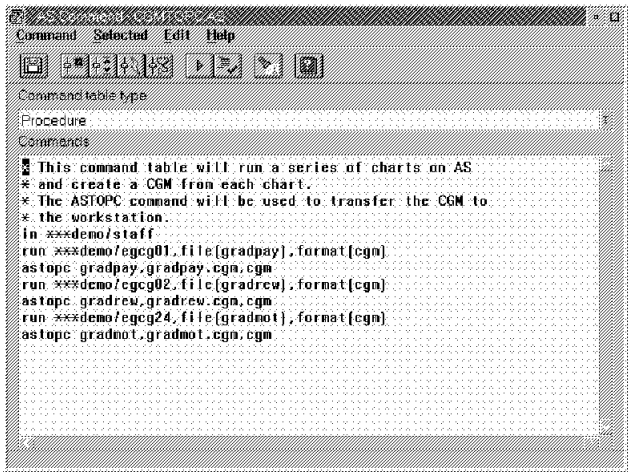
Designed for the Executive

Executives want to work with their latest company information, not to worry about retrieving it. The AS Command Client lets them do just that. Could anything be easier than this?

running an AS Command Client by double-clicking on it

specifying any parameters needed via an optional window

opening the desired spreadsheet or document or chart and watching it being updated with the very latest information, delivered via DDE



The AS Visualizer Command Client Main Window

Could anything be easier? Well, yes it could. A Visualizer Procedure could be used to start both the AS Command Client and the spreadsheet.

[1] Windows is a registered trademark of the Microsoft Corporation

Designed for the Application Developer

Developers of client/server applications want a user interface that lets them specify everything required for the application. The AS Command Client provides the following and more:

Editing on the PC

The developer specifies the AS procedure (or other command language table) that analyses the information to be retrieved. It can either be one that already exists on the mainframe, or it can be held as part of the AS Command Client on the PC. Additionally procedures can be transferred between AS and the AS Command Client, enabling the editing of procedures held on the mainframe

Interactive debugging on the PC

The developer has the same interactive debugging tools for procedures on the mainframe, delivered via a PC window. As this window shows the output from running the command table it can even be used as a means of displaying query results to an end user.

Delivery of information

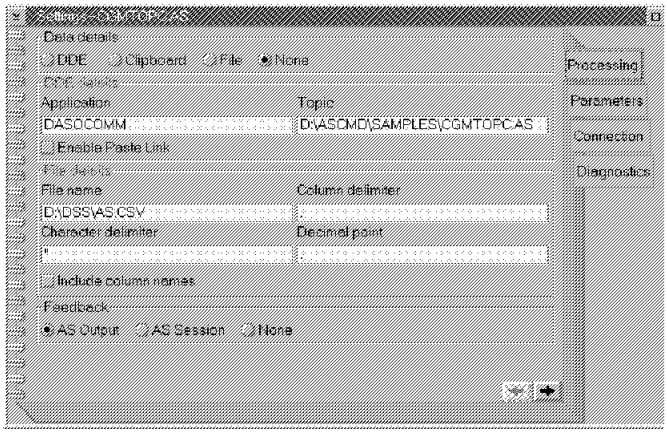
The developer chooses the method to be used to return the information. This can be via DDE, on the PC Clipboard, or in a Comma Separated Values (CSV) file

PC window for parameters

The developer can specify whether the user of the AS Command Client should be prompted for parameter values at runtime.

The user is prompted for the values required in a window built by the AS Command Client using the business terms for the parameters supplied by the developer.

The parameter values can be passed to the procedure when it is executed either using session variables and the Substitute command, introduced in AS Version 3.2, or by the @PASS facility.



The AS Visualizer Command Client Settings

Getting started

The new AS Version 4 book, the Really Useful Manual - SH45-5543, teaches you how to use the AS Command Client using real life scenarios. We even include the spreadsheets and wordprocessing documents they use with the AS Command Client.

The Client Connections manual - SH45-5565, discusses the AS Command Client and its data delivery mechanisms in more detail.

The AS Client API

The previous article described the AS Visualizer Command Client, which is a major expansion of the access to AS from the workstation.

The complete AS Client Connections package for the OS/2 and Windows environments consists of:

AS Workstation Support
- AS DDE Server
- AS table extract program
- AS Client Application Program Interface (API)
- AS Client API sample client programs

The AS Visualizer Command Client

The AS Workstation Support can be installed on its own, and is also a pre-requisite for the AS Visualizer Command Client.

AS Workstation Support was introduced for OS/2 workstations in AS Version 3.2, via a PTF, and has been further developed for Version 4.1.

The *DDE Server* and *table extract* programs are a pair of useful utility programs for users wishing to run AS command procedures from the workstation, and transfer the results to the workstation.

The AS Client API provides a development environment for users wishing to develop custom applications in C.

Features of the AS Workstation Support for AS Version 4.1 include:

New AS commands ASTOPC and PCTOAS can be run in AS on the mainframe when it is being driven from the workstation. The commands will transfer data in either direction between AS and Comma Separated Values (CSV) files on the workstation. The command is flexible enough to support most variations of Comma Separated Values files. In addition to moving CSVs, ASTOPC will transfer Computer Graphic Metafiles(CGMs) from AS to the workstation, and can be used with new options on the AS RUN command which will save DRAW command output in the form of a CGM.

ASTOPC and PCTOAS are particularly useful when a number of files need to be built and transferred in one procedure - for example when setting up data on a workstation for executive information systems.

A number of new functions for Client API developers

- High level functions will move CSV and CGM files between AS and workstation, in a manner similar to ASTOPC and PCTOAS.
- Functions can import and export AS command tables. Procedure, Query, QS, Crosstab, Report, Edit, Graph, Image, Model, Update and Xtabulate command tables are supported.
- Data can be automatically converted between numeric and alpha during data transfer.

- TSO/E and VM commands can be executed from the workstation, both before AS is started and during the AS session.
- Functions are available to return lists of defined server profiles, supported host code pages, AS table types and so on, to help windowed applications build selection lists.

IBM APPC Networking Services for Windows is supported for APPC connections between mainframe and Windows workstations. It is a development of Networking Services/DOS, and is more suited to the AS Workstation Support for Windows.

An install program is used to setup the workstation. The workstation code continues to be shipped on the AS VM and MVS tapes, as compressed files. It is downloaded, and the install program is then used to setup the workstation desktop.

Install programs are available in U.S. English, Brazilian Portuguese (except in Windows), Danish, Dutch, Spanish, French, German, Italian, Swedish, Japanese and Korean.

Error messages are available for problems detected by AS Workstation Support. They are available in U.S. English, Brazilian Portuguese, Danish, Dutch, Spanish, French, German, Italian, Swedish, Japanese and Korean.

Full details of the AS Workstation Support will be available in the AS Client Connections Manual, SH45-5565.

AS Client Connections - expanded connectivity and new function

New function has been added to AS Client Connections since it became available with AS Version 4 Release 1, improving the access to mainframe AS fros OS/2 and Windows[2] workstations.

APPC communications via Attachmate[3] supported by AS Client Connections for Windows.

APPC communications via Rumba[4] supported by AS Client Connections for Windows.

The AS Command Client supports an Initial AS Command which is executed before the AS procedure.

IBM Personal Communications/3270 Version 4 for OS/2 has become available and is supported.

There have been performance improvements, and a number of minor problems have been fixed.

More details of the new function follow...

The new function is available via new function Program Temporary Fixes (PTFs) for AS V4R1 and AS V3R2.

New function Apar PN68913 for AS Version 4.1 - March 1995

This was shipped as PTF UN74635 for MVS or UN74639 for VM. It is also included in New function Apar PN72042 July 1995.

PN68913 added support for APPC communications via Attachmate.

New function Apar PN72042 for AS Version 4.1 - July 1995

This was shipped as PTF UN78830 for MVS or UN78835 for VM. It adds support for Rumba APPC communications, Initial AS Command and improves performance transferring AS procedure tables between workstation and mainframe.

New function Apar PN72041 for AS Version 3.2 - June 1995

This was shipped as PTF UN78391 for MVS or UN78392 for VM. It adds support to AS Client Connections for Windows for APPC communications using Attachmate, Rumba, and IBM APPC Networking Services for Windows.

---

[2] Windows is a registered trademark of Microsoft Corporation

[3] Attachmate is a registered trademark of Attachmate Corporation

[4] Rumba is a registered trademark of Wall Data Incorporated

Attachmate

Communications support in AS Client Connections for Windows is expanded with the fix for new function Apar PN68913 for AS Version 4.1 or PN72041 for AS Version 3.2 The Attachmate APPC Client (for use with EXTRA![5] for Windows) is now supported.

For full details of the Attachmate APPC support, install the PTF, and download and unpack the distribution files. See README.DOS in the DASPWWIN distribution file.

Rumba

Communications support in AS Client Connections for Windows is expanded with the fix for new function Apar PN72042 for AS Version 4.1 or PN72041 for AS Version 3.2 The Wall Data[6] Rumba APPC Engine is now supported.

For full details of the Rumba APPC support, install the PTF, and download and unpack the distribution files. See README.DOS in the DASPWWIN distribution file.

AS Client Connections connectivity with Windows

In the Windows environment, the following products can now be used to provide the workstation to mainframe connection:

IBM APPC Networking Services for Windows
IBM Personal Communications/3270 Version 3 or later in Windows mode.
Attachmate APPC Client (for use with EXTRA! for Windows)
Rumba APPC Engine

AS Client Connections connectivity on OS/2

IBM Personal Communications/3270 Version 4 for OS/2 was recently announced. It provides emulator support on OS/2 workstations, and it includes the Server Requester Programming Interface (SRPI) support used by AS Client Connections. It can be used for workstation to mainframe communications by Visualizer for OS/2, AS Version 3.2 Client API and AS Version 4.1 Client Connections.

Personal Communications/3270 Version 4.0 for OS/2
Personal Communications AS/400 and 3270 Version 4.0 for OS/2

OS/2 Communications Manager/2 (CM/2) continues to be available. AS Client Connections can use CM/2 for workstation to mainframe communications using either SRPI or APPC.

Initial AS Command in AS Command Client

The AS Command Client has been extended to allow an Initial AS Command to be specified. It is specified on the Processing Settings Page, and is executed by the AS Command Client:

Before the Command Client procedure is run
After any Session Variable Values from the Parameter Settings are set.

---

[5] EXTRA! is a registered trademark of Attachmate Corporation

[6] Wall Data is a registered trademark of Wall Data Incorporated

The Initial AS Command may be up to 248 characters, and may contain multiple AS commands joined with the AS newline character.

The Initial AS Command is particularly useful when the Command Client is running AS procedure tables such as CROSSTAB. The Initial AS Command can be used to set the IN table before the CROSSTAB table is run.

Another example is to use Initial AS Command to load a workstation Comma Separated Value (CSV) file into AS. It can then form the input to the AS commands being run. For example, the Initial AS Command would be:

pctoas shares.tab,shares,csv(names)_in shares

where _ is the AS newline character. PCTOAS creates AS table *shares* from workstation table *shares.tab* and the IN command then makes *shares* the AS IN table.

Performance changes loading AS procedures

The code for moving AS procedures between the workstation and AS has been rewritten. This code is used is used by the AS Command Client and by AS Client API functions DasGetCommandTable and DasPutCommandTable. Performance is improved significantly, and in addition, these functions will no longer change the AS IN table.

CICS/ESA Client feature of AS Version 4

The AS CICS/ESA Client is an optional feature of AS Version 4. It allows access to AS from a CICS/ESA screen.

The CICS "end" consists of four transaction programs which perform 3270 screen handling. These send and receive 3270 data streams to a server program running in an APPC/MVS address space. This server runs the MVSAS code.

Because this is a client/server implementation, with AS running in an APPC/MVS address space, the expression “AS under CICS” is not strictly correct.

When comparing MVSAS access using TSO/E with the use of the AS CICS/ESA feature, consider the following items.

Single point of logon

For installations which have most of their end user applications running under CICS, providing TSO/E access just for AS is a costly overhead. Removing this can mean that the network definition is simpler, administration of users is easier, and end users are not exposed to the non-application TSO/E (ISPF) interface.

Application integration

Facilities exist which enable other CICS transactions to start the AS CICS/ESA transactions, passing an initial AS command. Using this, you can make AS an integrated part of your applications, and to the user the interface is not apparent.

An additional facility allows the end user to suspend the AS session and continue with other CICS transactions.

Performance Tuning

By using facilities of MVS/APPC, you can tune the address spaces running AS to give good performance without affecting other MVS jobs. Additionally, it is possible using APPC Classes to give different priorities to different types of AS usage. For example, you could give different priorities to long running project management sessions.

You can tune interactive TSO/E address spaces, however this is complex, particularly distinguishing between the different types of work users may perform with the same TSO/E user ID. Most installations use TSO/E for administration and application development with appropriate low priorities.

Comparison tests, performed by the AS Development and Customer Support group, show that the perceived response time as experienced by an end user for AS CICS/ESA client is no different than that when using TSO/E.

Flexibility

The APPC/MVS address space need not necessarily run on the same MVS image as the CICS/ESA system running the AS CICS transactions. Also the AS CICS transactions may run in one CICS Application Owing Region (AOR), whilst another CICS Terminal Owing Region (TOR) controls the user terminals.

Continuity

To the end user, the appearance of AS when run from CICS/ESA is the same as when run from TSO/E. One minor restriction, which you may use as an advantage, is that certain TSO/E commands and external programs (invoked by facilities like the APL, HOST TSO, HOST ISPEXEC, PGM, REXX commands and the HOST function) do not work in this environment. In addition, the IC/E feature is not supported when using the AS CICS/ESA client.

Documentation

For full details of the AS CICS/ESA Client feature, see the chapter "Enabling AS from CICS/ESA" in *AS: Supporting and Customising (MVS)*, SH45-5547. For installation instructions, refer to *AS Version 4 Program Directory*; for diagnostic information, refer to *AS: Diagnostic Guide*, SH45-5559.

Pictograph Attributes in AS Version 4.1 Charts

Picto what?

Have you seen those graphs in newspapers where cars piled on top of each other show how many road accidents there have been each year? Or where the bar representing Germany has the German flag above it? Graphs which use related pictures to represent information are called pictographs. An example is shown below, where code defects are represented by a picture of a bug:

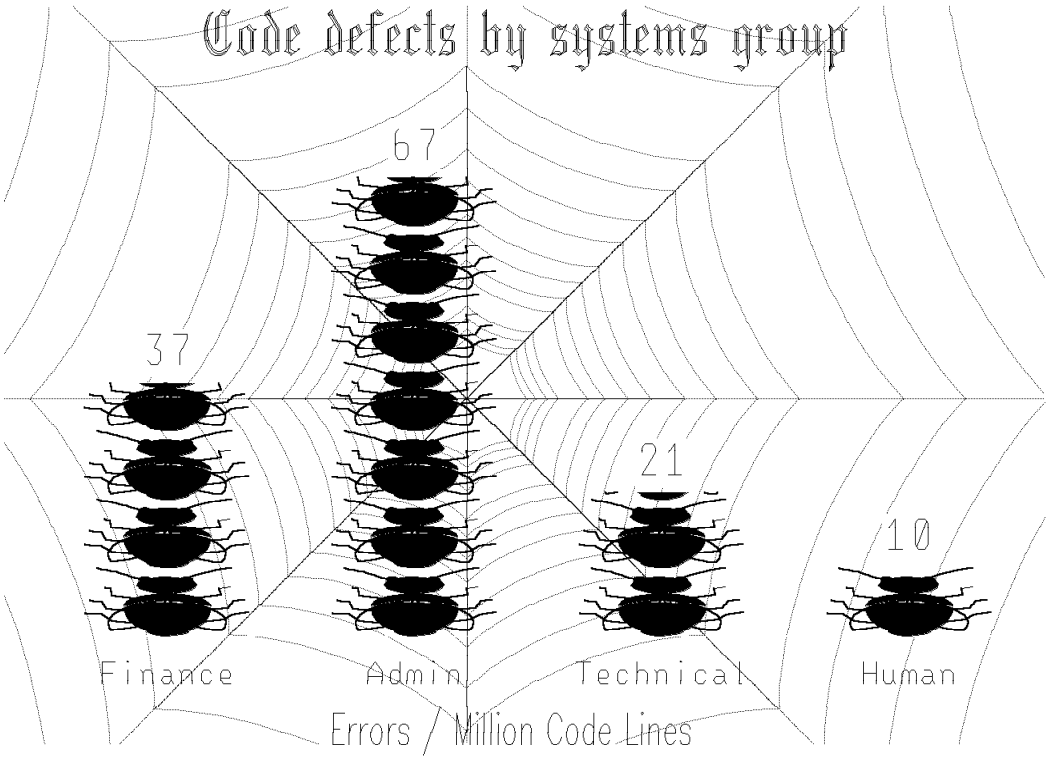


Figure 1. Example Pictograph

Pictograph Attributes

AS Version 4.1 can now produce pictographs. The Contents pulldown in Charts now has an extra option on it called Pictograph Attributes. Selecting it gives you the Pictograph Attributes window:

Figure 2. Pictograph Attributes window

In this window, different styles of pictograph can be defined, and the pictures to be used can be selected. The pictures must be ADMGDF files; press PF11 for a selectable list.

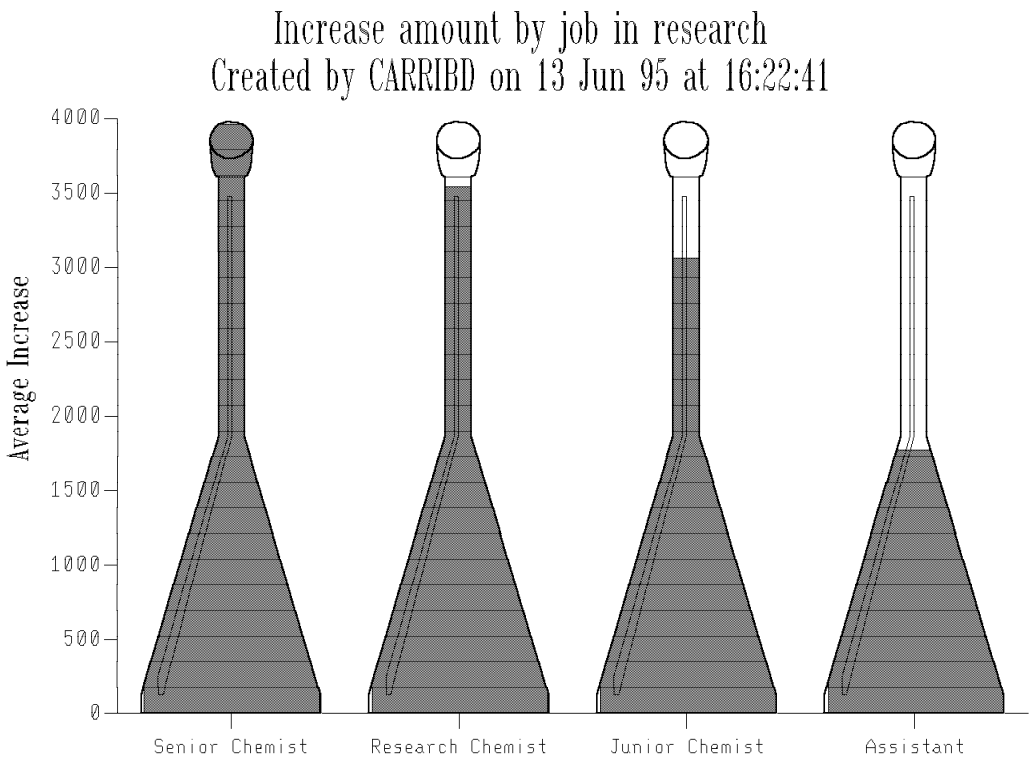


Figure 3. Example Pictograph

The above example shows a chart with the Mask pictograph style. For more examples, see the demo suite by typing *Demo Chart* in AS.

Clip Art

To help you produce pictographs, AS supplies a clip-art library of pictures. This is split into eight main areas:

1. Business
2. Masks
3. Flags
4. People
5. Geometry
6. Transport
7. Icons
8. Sport

For more details, see *Demo Chart - Display AS Clip-Art library*

Data Analysis Using AS Version 4.1

A whole new world

Did you know that there is a whole new application in AS Version 4.1? It is called Data Analysis, and can be used to easily create and execute quite sophisticated analyses of data. It has the normal AS action bar and windows interface, as well as a prompted and editor interface. There are also some new AS commands so that Data Analysis can be used non-interactively.

What is an 'analysis of data'? Data Analysis provides multidimensional summarisation of data by accumulating the data in groups. For example, we can ask Data Analysis to add up the salaries of our staff, breaking the results down by which parts of the company they work in. Data Analysis will provide us with the total salaries for all people who work in Computing, all people who work in the Head Office, and so on.

What about the results? Data Analysis creates an output table from the results of the analysis, and can provide links to other components of AS for displaying the results. Data Analysis output can be fed straight into Report and Charts, or into AS procedures for further processing. For client/server applications, Data Analysis output can be fed straight into PC applications such as a spreadsheet, wordprocessor or Visualizer.

Is Data Analysis similar to XTabulate? Data Analysis is similar to the existing Xtabulate application. The main differences are:

1. Data Analysis provides a more flexible user interface
2. Data Analysis provides results in a more useful form
3. Data Analysis provides links to other parts of AS

The Data Analysis designer

How do I define my analysis? The Data Analysis application can be invoked by selecting Data Analysis from the Facilities pulldown, or by using the Crosstab command. The main screen looks like this:

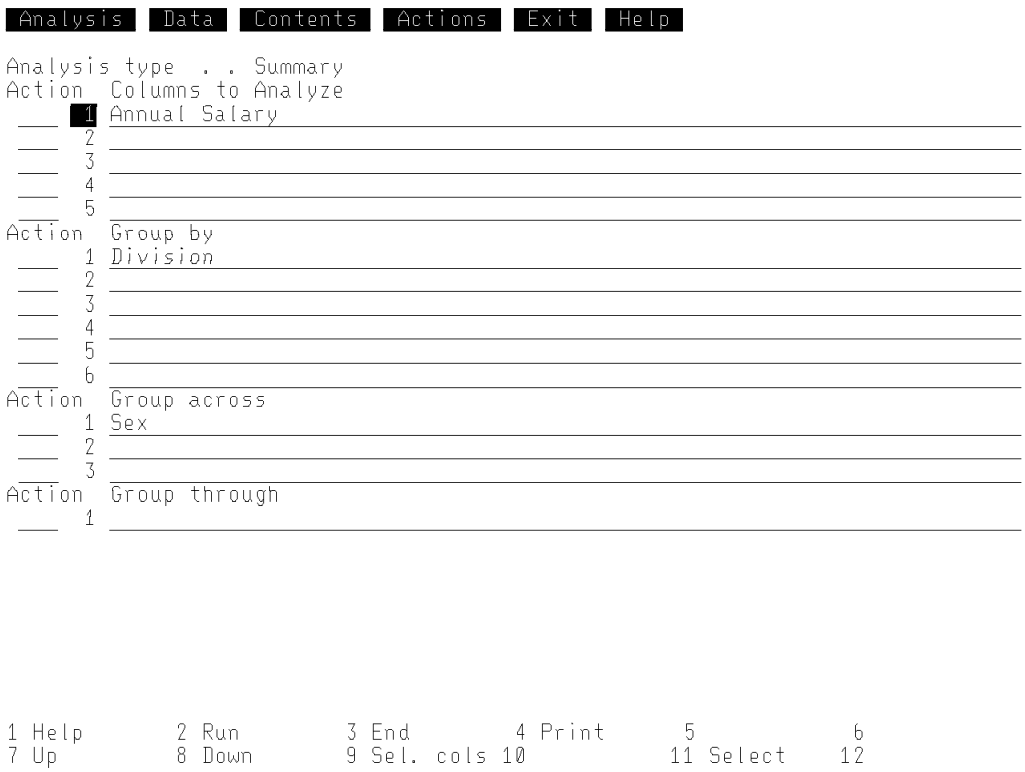


Figure 4. The Data Analysis screen

The 'Columns to Analyse' area is where column information is entered for columns that are to be accumulated in some way. In the example used above, salaries are being accumulated, so 'Annual Salary' has been entered here.

The 'Group By', 'Group Across' and 'Group Through' areas are where column information is entered for columns which are to be used for grouping. In the example used above, the parts of the company are used for grouping by, and we are splitting between males and females across the page.

If the 'Column Identifier' area (the numbers) is selected when no windows are open, a context menu is displayed listing the options applicable to that column. These options are also available from the Action Bars.

An example

What about a simple example? Start up Data Analysis by selecting it from the facilities pulldown, or by using the Crosstabulate command. Select PF9=Select Columns, and select ***Demo/Staff as the IN table. A column list is displayed. Select the columns shown above; the column is routed to the identifier which is currently highlighted on the design screen. Select PF2=Run. Data Analysis will execute your analysis, and produce a table containing the results. To see the results, type VIEW. The results look like this:

```
 Data  Dictionary  Display  Exit  Help
VIEW                #TEST001
Division                 F          M Action
Computing          116,568     56,867
Head Office        412,064    508,259
Production         433,500    885,246
Research           135,347    210,106
Sales              761,237    870,897


 ?
 1 Help       2 Bottom     3 End        4 Print      5 Recall     6 Top
 7 Up         8 Down       9 Switch    10 Left      11 Right     12 Command
```

Figure 5. Results from simple analysis

There is a row for each value of the Division column, and a column for each value of the Sex column. Filling in the grid is the appropriate total Annual Salary.

Going further

How do you perform more sophisticated analyses? The Action Bars in Data Analysis are the same as in all other applications in AS. The Contents Action Bar contains application specific options, and gives you access to windows that allow you to specify more advanced analyses. The windows available are:

1. The Column Accumulation window
2. The Output Order window
3. The Global Options window
4. The Analysis Type window

The Column Accumulation window

How do you control how columns are accumulated? You use the Column Accumulation window. This window contains three options:

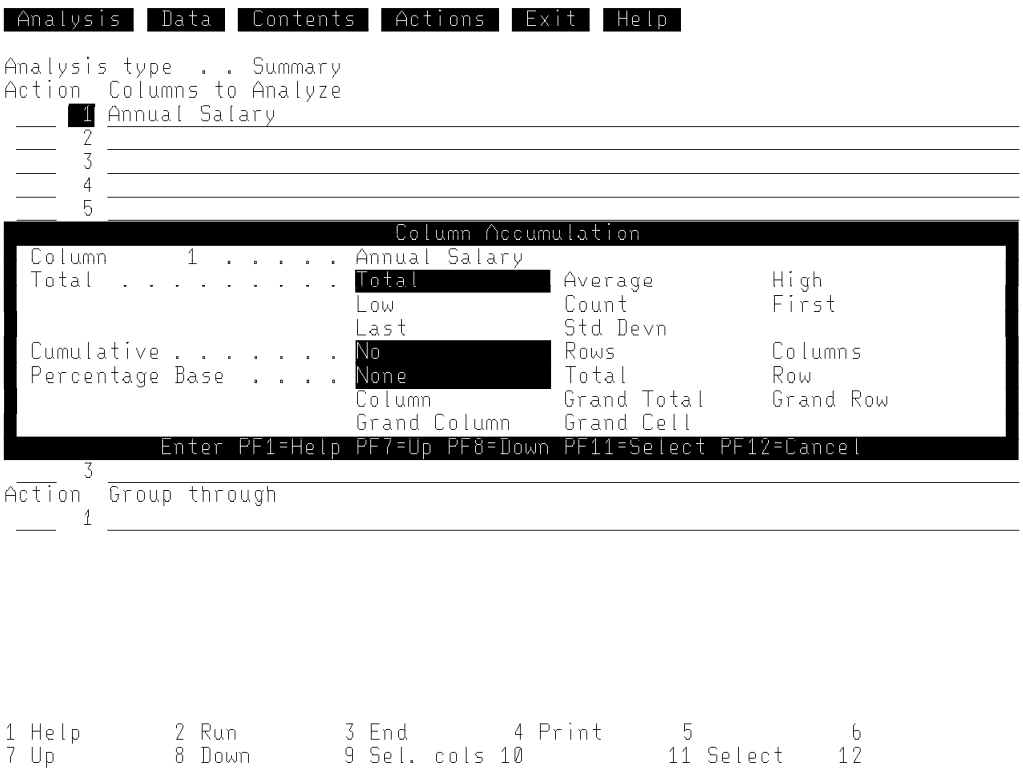


Figure 6. The Column Accumulation window

The Output Order window

How do you control the order of results? You use the Output Order window. This window contains options which allow sequencing the results table by either the values or the results in the results table.

The Global Options window

How do you control input and output? You use the Global Options window. window. This window contains options which allow you to specify the format and content of the results table. It also allows you to specify how the results should be displayed. For example, using this window we could have specified in the Data Analysis specification that we wanted the previous set of results displayed as a chart:

Figure 7. Chart from simple analysis

The Analysis Type window

You have missed out the Analysis Type window. What is that? Not only does Data Analysis allow you to design analyses of data, but it also provides some pre-set analyses, where you only provide the column information. The window has five options:

1. Summary is the option which allows you to design your own analysis.
2. Cumulative will perform three analyses: total, total cumulative to row totals and total cumulative to column totals.
3. Descriptive will perform six analyses: total, count, high, low, average and standard deviation.
4. Frequency will perform counting
5. Percentage will perform four analyses: total, percentage of row totals, percentage of column totals and percentage of grand total.

The Data Analysis commands

What about the commands mentioned earlier? There are some new AS commands which allow you to use Data Analysis without using the user interface. These commands match the analysis types in the Analysis Type window:

1. Xsummarise performs a specified analysis
2. Cumulate performs a cumulative analysis
3. Descriptive performs a descriptive analysis
4. Frequency performs a frequency analysis
5. Percentage performs a percentage analysis

Step by step Data Analysis

Does anything help me create an analysis? As with all applications in AS, there is a Create option which will take you through the creation of an analysis step by step. Select Create from the Analysis pulldown.

The intelligent server

Can I use Data Analysis in client/server applications? Data Analysis uses Crosstabulate language statements to define the analysis. These statements can be seen by selecting the Editor option from the Actions Action Bar. These statements are accepted by the AS Command Client, which means that analyses can be defined on a PC. Execution of these analyses will take place on the mainframe, with only the results table being transferred back to the PC. This table can then be fed directly into a PC product using DDE. The following example is shipped as a sample with the AS Command Client. It is the CROTAB.AS sample. See *AS: The Really Useful Manual (SH45-5543)* for further details.

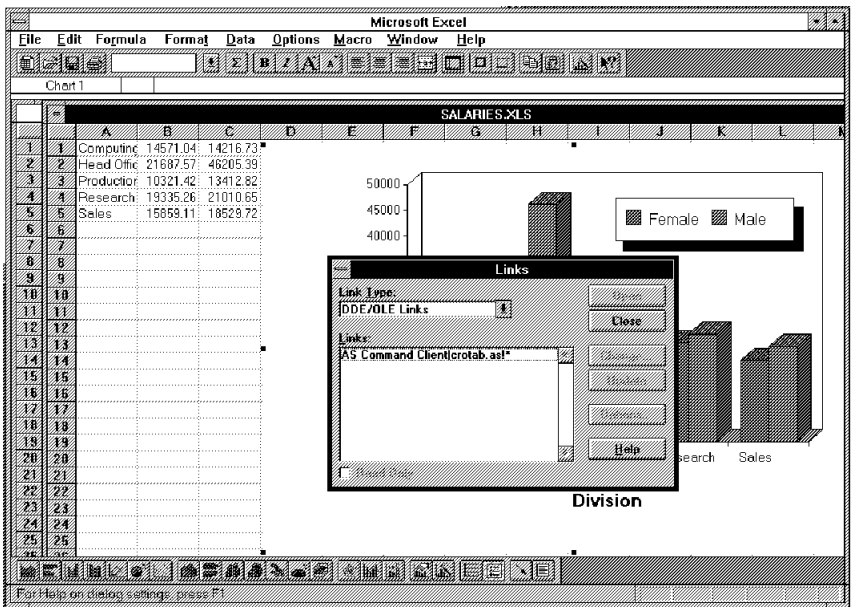


Figure 8. AS Command Client Sample

What you need to use it

Data Analysis looks good. What do I need to use it? Data Analysis is a completely new application in AS Version 4.1, so you will need this version of AS. Data Analysis has also been written in C, so you will need a C runtime library. This can be C/370, or LE/370.

Conclusion

Can you summarise the benefits of using Data Analysis? Data Analysis is:

Easy to use, with intuitive action bars and windows interface in the standard AS interface.

Powerful, allowing sophisticated analyses to be performed easily.

Ideal for client/server applications, since multidimensional summarisation of data reduces traffic volumes.

AS Version 4.1 Performance

AS Version 4 uses less hardware resource and that by itself justifies upgrading from any previous level of AS. So your customers get all the new features of AS Version 4 and improved response time virtually FREE!

More information about AS Version 4 performance is available in two documents:

AS Version 4.1 Performance Measurements
AS Performance Guide

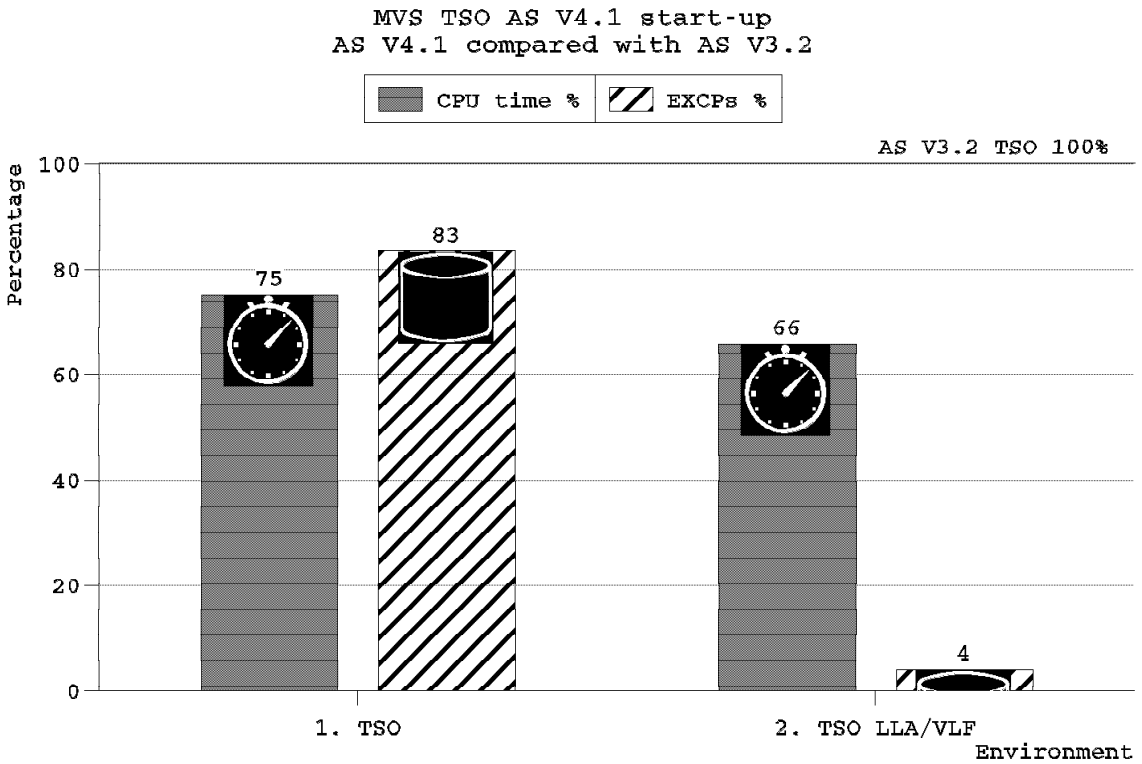
AS Version 4.1 Performance Measurements

This document gives information about AS Version 4.1:

Performance Improvements
System Tuning Enhancements
Measurements for New Environments

It does not go into detail about how to *use* AS Version 4.1 so the AS Version 4.1 manuals should be consulted for details of the new features.

It contains many charts which graphically compare AS Version 4.1 with AS Version 3.2. For example the following chart compares start-up time for MVSAS:



MVS TSO/E AS Version 4.1 Start-up

Performance Improvements:

Measurements presented show:

AS use of DFSORT for sorting — up to 84% reduction in EXCPs
AS use of GDDM HPA — up to 68% saving in CPU time
MVS AS Start-up Time — up to 96% reduction in EXCPs
DB2 interface — up to 45% saving in CPU time
AS Charts — up to twice the speed
AS SRQ Interface — up to 28% saving in CPU time

Performance improvements achievable by AS customers will depend on the AS applications that they use and their system set up, so these figures must be used for general guidance only.

System Tuning Enhancements:

MVSAS Use of LLA/VLF
MVSAS Measurement and Tuning

Measurements for New Environments:

AS CICS/ESA Client
AS OpenMVS HFS Support

You can obtain this and other performance related files by contacting your local IBM representative.

AS Performance Guide

This manual is entirely new for AS Version 4.1 and provides performance related information for the VM and MVS systems programmer, the database administrator and the application developer.

It is intended to help them develop efficient AS applications. It contains detailed advice about how to get the best possible performance from AS. It covers:

designing and coding for efficiency
taking advantage of AS performance features
optimising the use of relational databases
exploiting the features of MVS/ESA and VM/ESA

The guide is available as:

Form number SH45-5569-00

Application System Version 4.1 - Application Development

Introduction

AS Version 4 application development allows you to simplify development and improve maintenance and usability of AS applications.

This article shows how AS Version 4 has helped SHOWPAN users.

AS can now automatically provide the display of contextual help, and handle the display of Radio Buttons and Check Boxes.

Automatic handling of contextual help

In AS Version 3, SHOWPAN applications which built and displayed Help windows had to:

Calculate where the help window was to be displayed

Open the help window at the calculated position

Load contextual help into an array, for display in the window

Display the panel until the user wished to discard the Help

Handle PF7 and PF8 requests to scroll the text in the Help window

Shut the Help window when the display was no longer required.

AS Version 4 does all this work for you, using the **OPENHELP** function.

One function to handle Contextual Help

With the OPENHELP function you only need to define the help text for each part of the screen. AS looks after the positioning, filling, managing and scrolling the Help window for you.

The OPENHELP function tells AS the name of the table containing the help text, for example, with the table called *Explain*:

```
LET !x=OPENHELP('Explain')
```

Display of the help window

When you request help, AS shows a Help window immediately, as part of the same SHOWPAN call. Similarly, when a *PF12=Cancel* request is made, the Help window is removed and the screen redisplayed without control returning to the controlling procedure.

PF keys other than *PF7=Forward, PF8=Back, PF12=Cancel*, and *PF1=Help* are ignored while a Help window is on display.

AS displays the Help window surrounded by a reverse video turquoise border. The top border shows the Help window title. The window is displayed immediately below the area for which help is requested, or above if there are insufficient lines below. The PF key definitions in Help window bottom borders are mouse sensitive, and AS handles the scrolling of help windows automatically.

The Help window is removed, and the underlying screen refreshed, when you press PF key 12, or if you make a help request for a different area. If the screen contains invalid data, such as alphabetic values entered in a numeric area, the Help window remains until either the error is corrected, or you press CLEAR.

Structure of the Help table

AS determines which lines of text in the Help table to display from the internal name of the panel on which the help request was made, and the box name of the area for which help is requested.

The table holding the help text has the following structure:

This column	Contains this information	Key	Type, and maximum size
PanelName	The internal name of the panel for which help is requested.	1	A 20
BoxName	The box name of the area for which help is requested.	2	A 20
Index	The order of the help text: 1 is line 1, 2 is line 2, and so on. You can use Index 0 for your window title.	3	N 9
HelpText	The text. Its size helps determine the width of the window.		A 72

You can use one table for many different help texts as they can be made unique by different panel names, and box names.

Selection of help text

When you press PF key 1, PF key 13, or Mouse button 2, to request help, AS searches for text in the Help table in the following order:

1. Rows with major keys @CURSOR PANEL, and @CURSOR BOX.
2. Rows with major keys @CURSOR PANEL, and ' '.
3. Rows with major keys ' ', and ' '.

You can use these rules to define default help for areas with no box name, as well as for undefined areas.

Example

For an example of the use of automated Help in SHOWPAN, select ‘Using the OPENHELP function’ under ‘Procedure logic’ in DEMONSTRATE.

For a more detailed description of the use of automated Help with SHOWPAN, see *Application System Developing Applications* (SH45-5545) and also *Application System Reference* (SH45-5544).

Screen controls — Radio Buttons and Check Boxes

Radio Buttons and Check Boxes are standard CUA controls that let you select options from a list. AS provides a set of functions to define a set of clones as Radio Buttons or Check Boxes, and to set and query the status of these controls.

Radio Buttons

Use Radio Buttons when you need to select a single option from a list of options. A Radio Button set is a clone set, of which one (and only one) is always 'On'.

If you select a Radio Button that is 'Off', it is set 'On' and the previously 'On' button is set 'Off'. If the selected Radio button is already 'On', no changes are made.

Check Boxes

Check Boxes are appropriate when selecting from a set of options of which any number can be selected. A Check Box set is a set of clones of which any number may be 'On'.

If you select a Check Box that is 'Off', it is set 'On', and if 'On' it is set 'Off'.

AS shows Radio Buttons and Check Boxes that are 'On' in Reverse Video, and those that are 'Off' in Neutral. If the terminal has no EDS support, Radio and Check buttons that are 'On' are displayed Bright.

To select a Radio Button or Check Box, position the cursor and press Enter, PF key 11, PF key 23, or Mouse button 1.

Radio Buttons, Check Boxes

Once you have defined a set of Radio Buttons or Check Boxes, AS takes care of remembering, and highlighting, the selected options.

Using Radio Buttons

The **OPNRADIO** function tells AS to treat a clone set as a set of Radio Buttons:

```
LET !x=OPNRADIO('Main','Channels')
```

A Radio Button set must always have one button 'On', and by default, when first displayed, the first Radio Button will be 'On'. This default can be overruled by specifying a different initial setting as a parameter in the OPNRADIO function. Here, for example,

```
LET !x=OPNRADIO('Main','Channels',4)
```

the 4th Button will be 'On' when the Button set is first displayed.

OPNRADIO function

The display of Radio buttons is entirely looked after by AS. When users select a different setting, AS automatically turns 'Off' the previous setting, and turns 'On' the new one, with no special code required in your procedures.

AS also provides functions to let a SHOWPAN procedure work directly on a Radio Button set:

Turning on a Radio Button - **ONRADIO**

Discovering which Radio Button is ‘On’ - **QRYRADIO**

Finding whether a new Radio Button has been selected
- **@CHANGED RADIO**

See the on-line help for the full syntax.

Procedures using Radio Buttons would now typically look like this:

```
⋮
    LET  !x=ONRADIO(‘Main’,‘Channels’)
    DO  UNTIL(!x=2;14)
      LET  !x=SHOWPAN(‘Main’)
    NEXT
    LET  !x=QRYRADIO('Main',‘Channels’)
⋮
```

Using Check Boxes

The **OPNCHECK** function. tells AS to treat a clone set as a set of Check Boxes:

```
LET !x=OPNCHECK('Main','Attributes')
```

When first displayed, all Check Boxes in a set are ‘Off’. This default setting can be overruled by preselecting entries.

OPNCHECK function

The display of Check Boxes is entirely looked after by AS. When users select a Check Box, AS automatically turns the Check Box ‘On’ if it was previously ‘Off’, and turns it ‘Off’ if it was previously ‘On’, with no special code required in your procedures.

AS also provides functions to let a SHOWPAN procedure work directly on a Check Box set:

Turning a Check Box ‘On’ - **ONCHECK**

Turning a Check Box ‘Off’ - **OFFCHECK**

Discovering which Check Boxes are ‘On’ - **QRYCHECK**

Discovering whether a Check Box has been turned ‘On’ or ‘Off’
- **@CHANGED CHECK**

See the on-line help for the full syntax.

Examples

For an example of the use of Radio Buttons in SHOWPAN, select ‘Radio button functions’ under ‘Procedure logic’ in DEMONSTRATE, and for an example of the use of Check Boxes select ‘Check Box functions’.

For a more detailed description of the use of Radio Buttons and Check Boxes, see *Application System Developing Applications* (SH45-5545) and also *Application System Reference* (SH45-5544).

Code savings

Use of the OPENHELP, OPNRADIO and OPNCHECK functions can result in dramatic code savings for applications that previously used Version 3 functions.

Now ease-of-use is even easier in AS Version 4

The AS End User Interface is one of the aspects of AS that sets it ahead of other mainframe applications in terms of usability. It is a state-of-the-art interface which includes an action bar, option lists (pull-down menus), contextual pop-up help and dialog windows.

Now AS Version 4 gives you the chance to use these facilities directly from commands and procedures, and to customise them for your own applications.

Two new commands bring a lot of powerful function to AS users for minimal coding. The **Actionbar** command and the **Window** command let you customise action bars and call End User Interface windows respectively.

Actionbar command

The Actionbar command can be used to:

Define option lists
Add user defined options
Define help text for the options
Include standard option lists
Switch between user and standard action bars

```
Reports  Chart  Data  Exit
Weekly Report
Monthly Report
Annual Report
Remove
   7 Act Reset
   9 Act User,Reports
  11 Act Weekly Report,Run Weekrep,Help(This is the normal Tuesday report,whic
     h is sent to Personnel.)
  12 Act Monthly Report,Run Monthrep,Help(Run this on the first Monday followi
     ng the 6th of each month.)
  13 Act Annual Report,Run Annrep,Help(This report should be run each January
     after year end cut-off has been completed.)
  15 Act User,Chart
  17 Act Weekly Chart,Run Weekcha
  18 Act Monthly Chart,Run Monthcha
  19 Act Annual Chart,run Anncha
  21 Act Standard(data)
  22 Act Standard(exit)
  24 Act Display(user,1,2)
```

```
1 Help      2 Run       3           4 Print     5 Recall    6 Toggle
7 Up        8 Down      9          10           11 Select   12 Command
```

Action bars like this can be created to your own specification.

Users and application developers can use the Actionbar command to replace the standard action bar at the top of the AS command screen. Frequently used reports, procedures, charts etc. can be assigned to an option in a pull-down option list. Actions which are inapplicable or infrequently used, can be left out. The action bar can therefore be set up, not only for users' personal preferences, but also for different levels of usage, new user, regular user, expert user, and so on.

For application use, users can be prompted by displaying an action bar with the relevant option list already opened and, if desired, with a default option highlit, so that the user need only press Enter to start the next action.

Here are some samples of using the Actionbar command. User specified text is in mixed case.

ACTIONBAR USER,Mybar,HELP(You may choose.....)

ACTIONBAR Monthly Report,RUN MONREP1,HELP(This report is.....)

ACTIONBAR DISPLAY(USER)

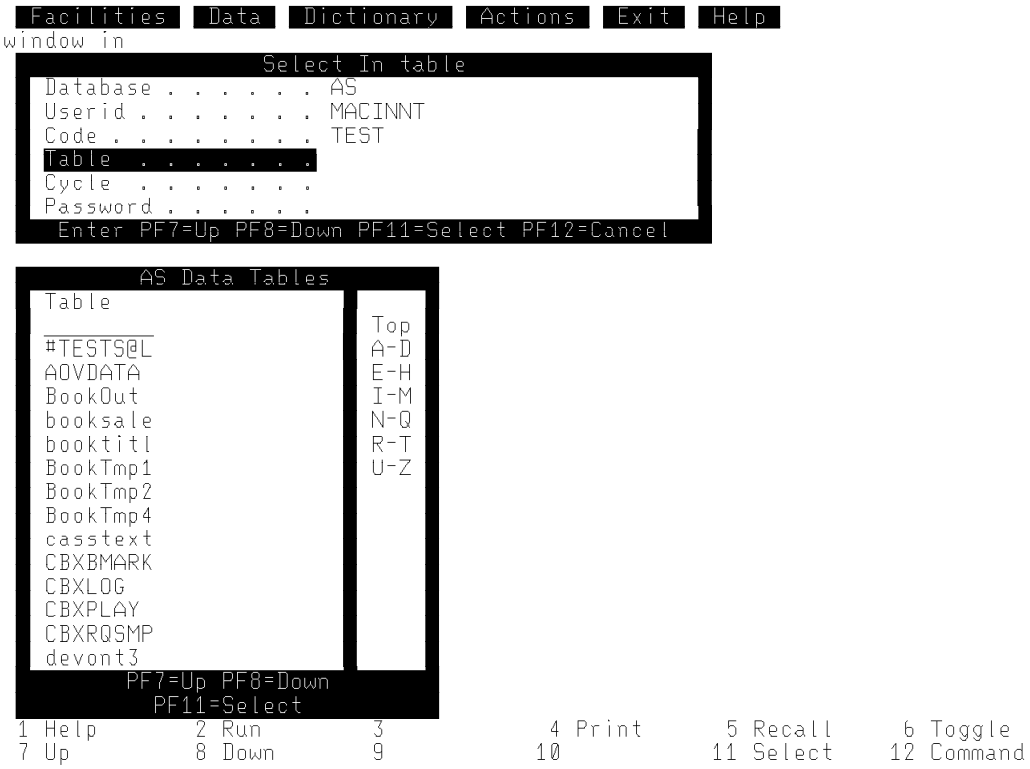
ACTIONBAR DISPLAY(STANDARD)

Full syntax of the Actionbar and Window commands is, of course, available on-line in AS Version 4.

Window command

The Window command will be a great benefit to developers who need to code Showpan procedures to contain actions such as selecting a table from a list. Previously the developer would have to define the correct areas in an image table, create and fill a data table with the table names, fill an array to display the data and code the scrolling and user selection actions.

Now this can be replaced with just one command. AS handles scrolling, user selection, on-line help, and so on. In our example of selecting an IN table, all that is needed is the command **WINDOW IN**. AS calls the End User Interface and presents the appropriate dialog windows as if the user had requested the action from the Data option list of the Action bar as in the example here.



In addition to selecting In tables, a number of actions from the standard Action bar can be called using the Window command. For example:

Select include
Select rows
Housekeeping
Create workplace

Actionbar and Window are just two of the new commands available in AS Version 4.1 which increase ease-of-use for both application developers and end users. I am sure they will be welcomed for their simplicity and power.

External Routines with AS Version 4.1

Routines in AS offer the advantage that commonly used code can be held in one place in a command table. Often, though, commonly used code is required by more than one command table and until AS Version 4 it was necessary to hold copies of routines with each command table that needed them.

With AS Version 4 it is now possible to call routines stored in different command tables. By saving code duplication and holding commonly used routines in one place, they become easier to find and maintain.

Three new commands make it possible to use routines stored in a different command table: **RTLOAD**, **RTUSE** and **RTDROP**. RTLOAD loads all, or a subset, of the routines from a command table into memory. The RTUSE command is used to 'activate' or tell AS to make use of a set of RTLOADed routines. The RTDROP command discards the loaded routines from memory.

How do I tell AS which external routines to use?

The RTLOAD command loads a set of routines from an AS command table. Routines can be loaded from any valid type of command table, and used wherever they are valid. For example, routines from an Update command table can be loaded in a Report command table or a Procedure command table.

To load a set of routines you must specify the table you are loading the routine(s) from, the routine(s) you wish to load, and an identifier, or handle by which the group of routines is to be known. Here, for example, the set of routines whose names begin 'HX' are loaded from the table RTable:

```
RTLOAD  rtn1,RTable,RTN(HX )
```

If no routine selections are provided with RTLOAD, every routine in the table will be loaded:

```
RTLOAD  RR,RTNSET
```

When the table contains both routines and other AS commands, the other AS commands are ignored by RTLOAD.

RTLOAD, RTUSE and RTDROP are active commands which are not controlled by AS logical commands such as WHEN and IF. Routines that have been RTLOADed remain available throughout the AS session within a single application code. This means that you could RTLOAD your commonly used routines at the start of a session and subsequently RTUSE them, when required from any of your command tables.

How do I use them?

Once you have RTLOADed routines into memory they remain available until discarded by RTDROP. Before being referenced and used by a command table, they must first be activated by the RTUSE command. Here, for example, the set of routines RTLOADed earlier, and given the handle rtn1, are made available for use:

```
RTUSE  rtn1
```

How do I get rid of them when I no longer need them?

RTDROP command is used to drop a set of external routines previously loaded by RTLOAD.

How are PSSAVE and PSLOAD affected by external routines?

PSSAVE uses external routines no differently from other routines.

Once you have used PSSAVE to save a processing step, you can remove any RTLOAD and RTUSE statements that were used for that step, just as you can remove local routines. (Remember to check first, though, that the routines are not CALLed in any other processing step in the application!) As with other routines, if any changes are made to External routines then all processing steps that used those routines and were saved using PSSAVE should be run through the PSSAVE process again.

What if new routines are added to the external routine

library? Do I have to PSSAVE again?

The only reason to create a new PSLOAD module is if one of the routines your application calls has been changed and you wish to make use of the new code. If you continue to use the old PSLOAD module, it will continue to operate unchanged.

What are the benefits of this new function?

The new RT commands make it possible to establish a common set of routines for use in many different command tables. By saving code duplication and holding commonly used routines in one place, they make routines easier to find and maintain.

Consolidation Recalculation in Multi-Dimensional Model

Consolidation prior to AS Version 4.1

AS Business Planning has always allowed users to identify plan items that are to be recalculated at total time within the detail levels of a Multi-Dimensional Model. However the consolidation level results for these plan items were generated by adding all the detail levels together and dividing by the number of levels consolidated. At this point discrepancies can occur for plan items which represent percentages or were derived by some form of formula.

To illustrate the problem Figure 9 on page 31 shows the sort of results produced by Model prior to AS Version 4.1. ***Balance*** and ***Payment*** are plan items read in from a data table. ***Interest*** is calculated as ***Balance / Payment * 100*** and is flagged as a *"recalculate at total time"* plan item. As can be seen from the example the **"Total"** percentage for detail levels(**ZONE:1/AREA: A** & **B.**) have been recalculated from ***Balance*** and ***Payment*** Totals. However for the consolidation level (**ZONE: 1** with no AREA) the values for ***Interest*** have not been calculated correctly. They have been averaged from the results for *"recalculated"* plan items.

ZONE: 1
AREA: A

Plan items	1	2	Total
Balance	1 2	5	6 2
Payment	14	5	64
Interest	13.725	1 .	1 .631

ZONE: 1
AREA: B

Plan items	1	2	Total
Balance	12	567	687
Payment	16	53	69
Interest	13.333	9.347	1 . 44

ZONE: 1

Plan items	1	2	Total
Balance	222	1 67	1289
Payment	3	1 3	133
Interest	13.529	9.674	1 .337

Figure 9. Model Consolidation prior to AS Version 4.1

New feature in AS Version 4.1

In AS Version 4.1 the model is effectively re-executed using the consolidation level results as input data thus producing the correct results for *recalculated* plan items. Figure 10 on page 32 shows the results produced by a model using the new consolidation system. As can be seen the detail level values have not change but the consolidation level results for ***Interest*** show a difference as they reflect a true calculation of ***Balance / Payment * 100*** at this level.

ZONE: 1
AREA: A

Plan items	1	2	Total
Balance	1 2	5	6 2
Payment	14	5	64
Interest	13.725	1 .	1 .631

ZONE: 1
AREA: B

Plan items	1	2	Total
Balance	12	567	687
Payment	16	53	69
Interest	13.333	9.347	1 . 44

ZONE: 1

Plan items	1	2	Total
Balance	222	1 67	1289
Payment	3	1 3	133
Interest	13.514	9.653	1 .318

Figure 10. Proposed Model Consolidation.

The new facility of re-executing the model at consolidation time is controlled via the selection of option "Recalculate" on the revised plan details panel.

```
┌──────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────┐
│                                        Plan Details                                                                          │
│ ┌──────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────┐ │
│ │ Number of Periods. . . . . 12                                                                                            │ │
│ │ Frequency of Totals. . . . 12                                                                                            │ │
│ │ Type of Total. . . . . . . Months                    Quarters                                                            │ │
│ │                                 Years                          Others                                                    │ │
│ │ User Column Headings . . . . Yes          No                                                                             │ │
│ │ Interleave Totals. . . . . Yes            No                                                                             │ │
│ │ Missing Data . . . . . Yes                No                                                                             │ │
│ │ Recalculate. . . . . Yes                  No                                                                             │ │
│ │ Precision. . . . . Long                   Short                                                                          │ │
│ │ Rounding . . . . . Yes                    No                                                                             │ │
│ │ Display Width. . . . . 8                                                                                                 │ │
│ └──────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────┘ │
│                  Enter  PF1=Help  PF11=Select  PF12=Cancel                                                                   │
└──────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────┘
```

Figure 11. Revised PLAN DETAILS panel.

The Consolidation recalculate feature will only be used if the model has plan items with a TOTCH type of "C" (recalculate at Total time) and does not use work plan items (those prefixed with a "!"). If work plan items are detected in a model that is using the new feature a message is issued and processing continues with no consolidation recalculation.

The introduction of the new Consolidation Recalculation feature in AS Version 4.1 provides the user with better control over the results produced by their Multi-Dimensional models.

Using AS with the Hierarchical File System (HFS)

AS Version 4 for MVS/ESA, introduced the ability to use the Hierarchical File System (HFS) of MVS OpenEdition as an alternative to using AS File Servers, and the AS VSAM Common Library. This can reduce the need for data administration, and can improve the performance of some applications, under certain conditions.

When deciding whether to use HFS for AS data storage, consideration should be given to the backup and restore of data within each HFS File System.

Full details of how to install the AS *** (3 star) library into an HFS File System are provided in *Application System: Program Directory (MVS)*. Other AS documentation relevant to HFS can be found in *Application System: Supporting and Customising (MVS)*.

Migration and Definition of AS users

Migration to HFS (from File Servers) involves moving AS data to a pre-defined HFS File System, and making the necessary changes to the AS Names file. This is documented in *Application System: Supporting and Customising (MVS)*.

Before an AS user can access HFS data they should be defined to RACF (or equivalent) as a OpenEdition user. This is done using the RACF **ADDUSER** or **ALTUSER** command:

ALTUSER DAVISR OMVS(UID(678) HOME('/u/AS/davisr'))

In the above example user DAVISR is defined to OMVS with an initial home directory of '**/u/AS/davisr/'**. This would then be the current working directory, for the user's process when the user enters the TSO/E OMVS command.

Note: The directory names are case sensitive - lower case really does mean lower case.

For AS, this default working directory can be overridden using the AS Names file configuration entry **:asfiles.** tag. The HFS data set that the user would allocate data in would then be determined by the "mountpoint" of the HFS file system. For example, if HFS file system "ASPP.ACCOUNTS.HFS" was mounted with the following command:

MOUNT FILESYSTEM('ASPP.ACCOUNTS.HFS')
TYPE(HFS) MOUNTPOINT('/as/accounts')

User DAVISR could then specify :asfiles./as/accounts/davisr and for that AS session all AS tables would be allocated in HFS data set ASPP.ACCOUNTS.HFS (assuming access controls were met). This is an easy way of splitting users across multiple HFS data sets, for the duration of the AS session.

Note: A MKDIR command should be issued by an OMVS authorised user for '/as/accounts' if this is a new directory path. To issue the **MOUNT** command the user must be defined as an OMVS "superuser". For more information on these commands see:

MVS/ESA OpenEdition MVS Command Reference - SC23-3014

Allocating Space for AS HFS Users

When using HFS there is no equivalent of the AS ASADMIN command. Therefore, any user could theoretically fill the complete HFS File System causing AS to fail any subsequent write, for any AS users of that File system, and in some cases AS start up.

The following message(s) are issued to the AS user when a HFS data set fills, for example, during a table copy:

```
AS 1682   Table/File MYTABLE   user library full  47/ 12/ 263
AS 2546   Maximum table size reached
AS 2852   nnnn input table rows processed
```

To determine which AS user may have filled the HFS data set it is necessary to query the directory structure of the HFS data set, using the TSO/E OpenEdition SHELL or ISPF Shell interface.

Backup and Recovery of AS HFS data

When using the AS File Server to store data, program DASMSM is used for backup and restore purposes. For data stored in HFS data sets, the DASMSM program is **NOT** supported.

Data Facility Systems-Managed Storage Hierarchical Storage Manager (DFSMShsm) and Data Facility Systems-Managed Storage Data Set Services (DFSMSdss) provide archive and backup facilities for complete HFS File Systems. HFS data sets can be incorporated into normal DASD backup procedures.

Warning: Currently there is no MVS facility to automatically backup individual files/directories within a HFS data set.

However, it is possible to backup individual or groups of files within a HFS file system, using the OpenEdition SHELL commands and standard MVS/ESA services. The OpenEdition SHELL is an optional MVS feature, based on the UNIX V shell.

Using Shell commands to backup files

The following diagram shows an HFS file system containing two users (DAVISR and DUBLIND). The HFS file system is called ASPP.AS.SUPPORTG and is defined with a MOUNTPOINT of **'/u/AS'**.

Both users have a home directory of **/u/AS/userid**, and AS Names File tags of **:asfiles.HFS /u/AS/userid** where userid is their TSO/E userid. Each user has ONE application code (CCFC for DAVISR and BFR for DUBLIND).

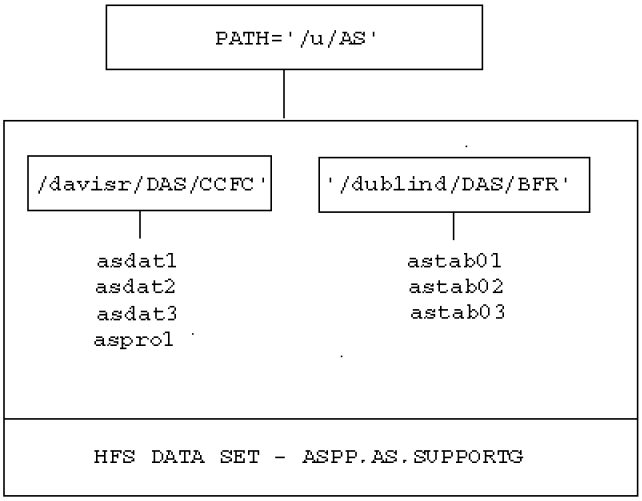


Figure 12. HFS Data Set holding data for two AS users, DAVISR & DUBLIND.

To dump and restore individual files in the file system, there are several SHELL commands that can be used:

tar

cpio

pax

For more details on these commands see:

MVS/ESA OpenEdition MVS Command Reference - SC23-3014

Creating the backup/archive file

In the following examples we are using the **tar** command.

Assuming that the current working directory after entering the shell is **/u/AS**, and the user issuing the tar command has the necessary authority to read the HFS data, the following command(s) can be used:

tar -cvf archive.tar /u/AS/davisr

This creates an archive file called archive.tar (/u/AS/archive.tar) containing all data in the directory path of /u/AS/davisr.

tar -cvf ourarc.tar /u/AS

This creates a archive file called ourarc.tar containing all data in the directory path of /u/AS (all AS users).

This way, using the **tar** command, you can be selective of how much of the HFS data set you wish to backup, and therefore you can backup individual or groups of AS users.

Writing the backup file to MVS

After creating the archive file (**ourarc.tar**) you should use the **OGET** command to write the archive file from the HFS data set, to an MVS data set, and then delete the archived file from the HFS data set:

OGET '/u/AS/ourarc.tar' 'ASPP.ACCOUNTS.BACKUP' BINARY

The data set on MVS can either be a sequential, partitioned or partitioned-extended. However, the OGET command will only dynamically allocate sequential format data sets - partitioned data sets must be pre-allocated. Once data is in an MVS type data set it can be managed according to normal MVS data backup and restore procedures.

Note: The BINARY option is required on the OGET and OPUT commands, to avoid record truncation.

Restoring a backup/archive file

To restore the archive shown in the previous example, first write the archive file from the MVS data set back into the HFS file system,using the **OPUT** command:

OPUT 'ASPP.ACCOUNTS.BACKUP' '/u/AS/ourarc.tar' BINARY

Then use the tar command to restore the contents of the archive. The data is restored at the directory point from which it was archived:

tar -xvf ourarc.tar

Compress the archive file

It is possible using the tar command to compress the archive file as it is created, to save space in the HFS file system. Use the **-z** option:

tar -zcvf ourarc.tar /u/AS/davisr

When working with compressed archives, the **-z** option is always required.

Listing and Selective restore of Archive Files

Using the **tar** command it is possible to list the contents of an existing archive file:

tar -tvf ourarc.tar

It is also possible to selectively restore files from an archive. The following example restores AS procedure ASPRO1 in application code CCFC for user DAVISR, from archive **ourarc.tar**:

tar -xvf /u/AS/ourarc.tar /u/AS/davisr/DAS/CCFC/ASPRO1

Using JCL (BPXBATCH) for backup

It is possible, using the BPXBATCH program to run shell commands against HFS files through MVS JCL. Using the JCL interface saves you the trouble of going into the shell to enter commands for backup and restore.

The following JCL invokes the BPXBATCH program to create an archive (mybackup.tar) of the files in path /u/AS/davisr. The second step uses IKJEFT01 to invoke the OGET command to write the archive to the MVS data set **DAVISR.BACKUP.TAR**:

```
//ASBACKU JOB (ASADMIN),PROGRAMR,MSGCLASS=X
//OMVSARC EXEC PGM=BPXBATCH,
//    PARM='SH tar -cvf /u/AS/davisr/mybackup.tar /u/AS/davisr'
//STDOUT     DD  PATH='/u/AS/davisr/mybackup.tar',
//              PATHOPTS=(OWRONLY,OCREAT,OTRUNC),PATHMODE=SIRWXU
//STDERR     DD  PATH='/u/AS/davisr/errors',
//              PATHOPTS=(OWRONLY,OCREAT,OTRUNC),PATHMODE=SIRWXU
//BPXBA      EXEC  PGM=IKJEFT 1
//SYSTSPRT DD SYSOUT=
//SYSTSIN   DD
 OGET '/u/AS/davisr/mybackup.tar' 'DAVISR.BACKUP.TAR' BINARY
```

Using JCL (BPXBATCH) for restore

To restore the archive file into the HFS system, and re-build the **/u/AS/davisr** path, the following JCL can be used:

```
//ASBACKU JOB (ASADMIN),PROGRAMR,MSGCLASS=X
//BPXBA      EXEC  PGM=IKJEFT 1
//SYSTSPRT DD SYSOUT=
//SYSTSIN   DD
 OPUT 'DAVISR.BACKUP.TAR' '/u/AS/davisr/mybackup.tar' BINARY
//OMVSARC EXEC PGM=BPXBATCH,
//    PARM='SH tar -xvf /u/AS/DAVISR/mybackup.tar'
//STDERR     DD  PATH='/u/AS/davisr/errors',
//              PATHOPTS=(OWRONLY,OCREAT,OTRUNC),PATHMODE=SIRWXU
```

Program IKJEFT01 is executed first to read the MVS data set and load it into the HFS data set. BPXBATCH is then run

AS Version 4.1 Project Management - Consolidator of Choice

The reasoning behind the project management enhancements introduced in AS Version 4.1 was to make AS the ***Consolidator of Choice***. For this to be true, the following functions have to be available to the end user (project planner):

1. The ability to read data from PC packages. In particular from some of the popular PC packages such as Project Manager Workbench[7] (also known as Project Workbench[8]), CA-SuperProject[9], and Microsoft Project for Windows[10].
2. Once the PC projects have been imported into AS, you need an easy method of merging (or combining) them (or other individual AS projects) together to form a consolidated super-project. Once the projects have been merged, they can then be analysed together using the normal functions of AS project management.
3. After the analysis, the user will want a simple way of summarising or condensing the data so that reports can be created for upper management.
4. Once the data has been processed and summarised, the user needs the ability to easily select the summary subset for displaying in the various reports.

---

[7] Project Manager Workbench is a registered trademark of the Hoskyns group Plc

[8] Project Workbench is a registered trademark of Applied Business Technology Corporation

[9] CA-SuperProject is a registered trademark of Computer Associates International Incorporated

[10] Microsoft Project for Windows is a registered trademark of the Microsoft Corporation

5. Once there is a consolidated project, users need an easy method of extracting a subset from it so that individual planners can make changes to their parts of the overall project without locking out all the other planners from the overall project.

All of these consolidation requirements are now satisfied with the new Version of AS (in fact the summarisation requirement was satisfied with the ***Summary Activity*** feature of Version 3.2):

The Archive/Restore functions of AS Version 4.1 permit a project to be archived as a set of CSV tables, and restored from a set of CSV tables. In particular, there is demonstration code provided that supports the restoring of an AS project from a set of CSV tables created via:

Hoskyn's Project Management Workbench
Microsoft Project for Windows
Computer Associates CA-SuperProject

The Archive/Restore functions also support a set of model CSV tables.

The Merge/Split functions of AS Version 4.1 permit sub-projects and super-projects to be processed. Merge and Split will extract a subset of one project (a sub-project) and combine it with another project (super-project). The projects being processing do not need to have identical table structures.

A special case exists where a new project can be created from a subset of an existing project. These functions enable you to create a new project by extracting and combining together a series of standard sub-projects that could be held in some form of template project. This much simplifies the chore of new project creation.

The ability to easily summarise project information was already present in AS. However with Version 4.1, the ability to use standard AS selection criteria has been extended to the Gantt chart. Gantt charts can now have associated Data Specification Tables. This means that a Gantt can be easily created to just show the summary activities within the project plan.

All of these new features go towards making AS a very efficient consolidator of project management information.

Version 4.1 of AS project management also contains the following additional smaller enhancements:

An extension to the Gantt charts so that the date scale information (start date, end date, etc) can be passed as parameters. This makes it much easier for users to build up standard project management procedures that will work with all their projects without having to do individual date customisation.

The ability to produce a Gantt chart from any table containing dates without AS project management having to be installed. In the past, if you wanted to create a Gantt chart from a table you had created, you had to "fool" AS into thinking it was an activity table. Now you can create a Gantt chart directly from any table containing dates. Not only that, but you don't even have to have project management installed!

A new progress reporting option that allows AS to assume that an activity is progressing according to the project plan. Progress reporting can be quite a chore for project planners as they try to keep each activity's progress information current from one week to the next. Now with the new Assume Progress option, you can let AS assume that the activity is continuing to progress normally once you have specified its actual start date. In such a situation, AS will assume the activity is complete once

its calculated finish date (actual start plus duration) is before the Timenow date. If there are some activities that you specifically do not want to have progress assumed, you can turn the option off at the activity level.

An extension to the maximum number of calendar units. You can now have up to 30,000 work units in your calendar.

Project management front-end functions now available at the command level. These include PURGE, ARCHIVE, and RESTORE, together with the new MERGE and SPLIT functions.

New Print/Plot functions introduced into Reports and Charts are also available from within Gantt and Network Drawing.

All in all, Version 4.1 of AS project management provides the end user with a considerable set of functional and usability enhancements.

EAS Technical Bulletins on AS Project Management

EAS has created three new bulletins on AS Project Management, which you can receive by contacting your local IBM representative.

AS Project Management Demonstration Guide - building a house

The first bulletin is an AS Version 4.1 Project Management Demonstration Guide. Designed to illustrate some of the functions and capabilities of AS Project Management, including those new in Version 4, the demo features the building of a house with a detached garage. The house and garage are initially planned separately. The demo covers the following project management functions:

planning the house and garage

integrating the house and garage

resource allocation analysis of the integrated project

costing the project

project summarisation

baselining the project

progress reporting

An ASTRAN file containing the data and reports is also supplied in the package.

AS Project Management Demonstration Guide - managing an IT project

This second guide to demonstrating AS Project Management takes an IT project as its scenario. In this scenario, the project plans are maintained at two different levels. The master (summary) plan is kept separate from the detail plans. Scheduling and progress reporting are done at the detail level and the results summarised into the master plan. The demonstration covers the following scenario:

create a high level master plan using summary activities

schedule and cost the master plan using 'generic' resource groups

split off one of the sub-projects and assign to a project planner

developer the detail sub-project and assign individuals to the activities

schedule and cost the detail sub-project

report progress to the detail sub-project

summarise the detail progress and merge into the master plan

create summary management reports from the master plan

An ASTRAN file containing the data and reports is also supplied in the package.

Concepts and Benefits of AS Project Management

The third EAS Technical Bulletin on AS Project Management is entitled Concepts and Benefits of AS Project Management. It covers the concepts of project planning techniques, network planning, risk analysis, resource allocation, cost control and tracking in general terms plus an overview of the features that AS Project Management provides. Typical application areas by industry and the management planning cycle are also discussed.

The Bulletin also covers the benefits of Project Management in general terms and then the benefits that AS offers for the project manager.

Info Center/Enhanced in AS Version 4.1

Info Center/Enhanced (IC/E) allows you to make inquiries against a range of file types, including large inverted files, and to produce reports and business graphics. IC/E is an optional feature of Application System and consists of three components:-

Query System (QS)

Reporting System (RS)

Automatic Fix Application System (AFAS)

The following enhancements/changes have been made in Version 4.1:

Info Center/Enhanced-general changes

Prerequisite APL2 Version 2

New I/O processor for improved performance

VSAM clusters are no longer needed for IC/E object libraries.

New AP211 utility functions

LE/370 supported provided via PTF (UN74171,MVS and UN74172,VM)

Improvements (QS and RS)

Improved editing

PF4 on the EDIT screen now performs a split/join in the standard XEDIT fashion.

Users can define a continuation character for query procedures defined as tables.

For consistency, PF4 and PF11 in QS have changed places.

Allow users to define file descriptions for any file types

Row specification on output scroll panels Users can define which line number they want as the top row, rather than scrolling up/down a page at a time.

Improvements (RS only)

REPEAT command in DATA
Users can now specify individual columns to be repeated.

Performance improvements for regular reports.

Improvements (QS only)

New function RLIST (rotated list)
Inverts output so that columns become rows and rows become columns. This makes it easier to read output that is normally very wide.

RANGE/CODER
Text descriptions can now be saved with classifications.

PF10 (show log)
The last, most recent, page of the log is now displayed first.

DHHCREAT allows field descriptions with a maximum of 999 characters

New function SEARCHP (search position)
This allows positional specification for individual columns in one or more character fields.

Improvements (AFAS only)

Performance improvements through the use of AP211 and replacement of complete functions rather than individual lines

Most panels have been re-designed for usability

Customisation of most panels via the status file

New options DEBUG, APLCHAR and ENVIRONMENT

New PTF actions HOLD, RELEASE and RESTORE

New error actions STOP and BYPASS

Improved PTF application

Multiple PTF(s) can now be processed and will be automatically sorted into correct prerequisite or dependency order.

The message field has been extended to provide a rolling progress report for the PTF(s) being processed.

A summary display shows the status of each PTF at the end of processing.

New PTF(s) applied panel

Optional use of APL character set

New ACTION field for PTF(s) to be audited, held, released or transferred to the PTF processing panel.

New OBJECT field which displays the modules updated by each PTF

Output can be sorted by Date, PTF name or Object name.

New PTF(s) received panel

New ACTION field for PTF(s) to be audited, held, released or transferred to the PTF processing panel.

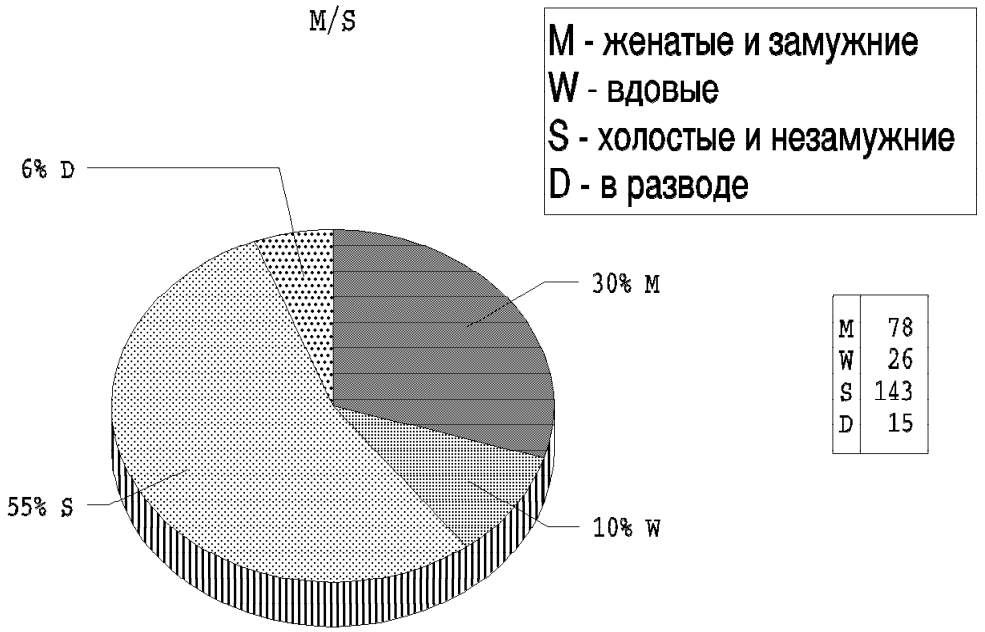
New COMMENTS field to show general PTF information.

PTF(s) can be displayed and control information obtained.

Support for Eastern European languages

GDDM have created 28 new symbol sets to support various Eastern European languages. These will allow you to use these languages with AS applications. For example you can create:

Gantt charts with descriptions in Russian
Histograms with legends in Polish



A pie chart with Russian annotation

This chart shows the marital status of staff on the ***DEMO/STAFF file. It uses symbol set ADMDVCHR (Cyrillic Helvetica Roman medium).

This article gives some more details about using these languages.

GDDM support

GDDM are providing the new symbol sets and associated changes by means of APARs and PTFs to GDDM Version 2.3 and GDDM Version 3.1.1.:

GDDM/MVS Version 3.1.1 PN7 441 GDDM/VM Version 3.1.1 PN7 443
GDDM/MVS Version 2.3 PN7 786 GDDM/VMXA Version 2.3 PN7 788

For full details please read the cover letters supplied with the PTFs.

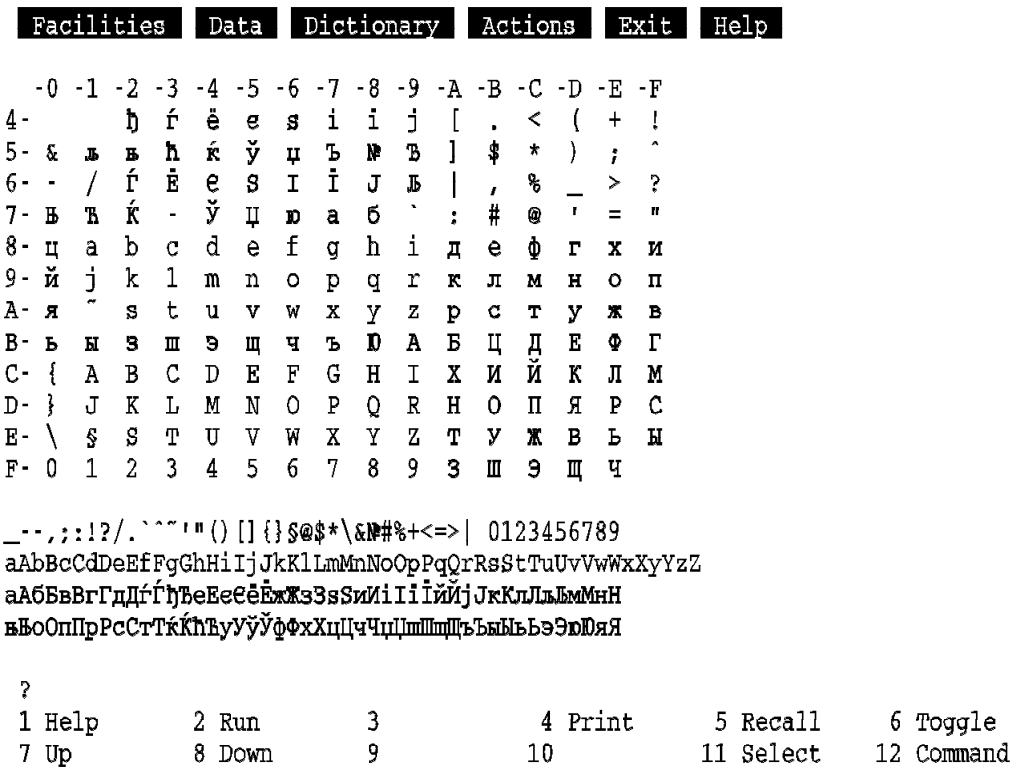
Character sets

GDDM Version 2 and Version 3 already support the Latin character set including the accented characters required for many popular Western European and Scandinavian languages. Two new families of character sets are now supported:

Cyrillic: This is the character set used for the following countries and languages:

Bulgaria
Russia
Parts of former USSR
Parts of Yugoslavia

It contains all the upper and lower case characters required for English in addition to the upper and lower case Cyrillic characters. It does not contain the accented Latin characters required for many other European languages.



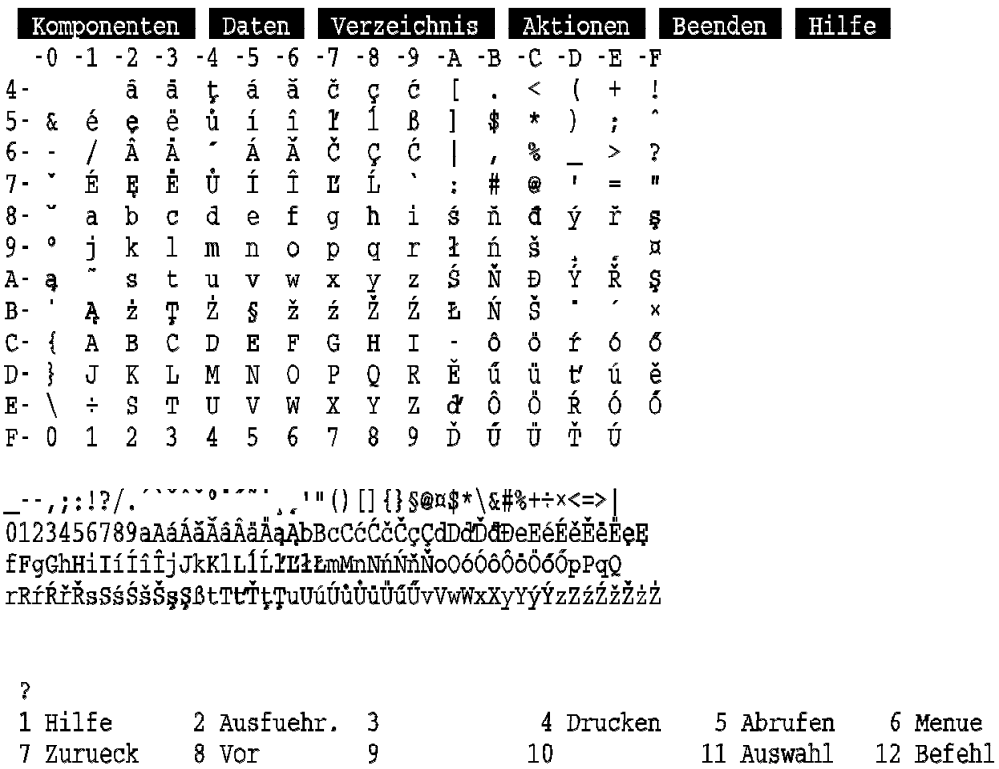
The Cyrillic code page (1025) and character set (1150)

This figure was created by using the SCSAVE command on an AS screen.

Latin 2: This is the character set used for the following countries and languages:

Albania
Czech Republic
Hungary
Poland
Romania
Parts of former USSR
Parts of Yugoslavia
Slovak Republic

It contains all the upper and lower case characters required for English as well as many accented characters. It does not contain the accented Latin characters required for many other Western European languages.



The Latin 2 code page (870) and character set (959 / 1128)

This figure was created by using the SCSAVE command on an AS screen using German (Deutsch).

Compatibility: The character sets are different. In general, data entered using one character set will not display correctly unless the same character set is used. However:

A-Z, a-z, 0-9 will always display correctly.

When the output device cannot display an accented character, GDDM can translate it to the same character without an accent.

When the output device cannot display a Cyrillic character, GDDM can translate it to the Latin phonetic equivalent character.

To enable GDDM translation you must specify the appropriate application code page to GDDM. You can do this by adding a default statement to your ADMDEFS file like this:

DEFAULT APPCPG=1 25

Vector symbol sets

You can use vector symbol sets on AS charts. GDDM supplies the new vector symbol sets with the naming convention **ADMDVctd** where **c** is:

C — Cyrillic
L — Latin 2

and **t** is

C — Courier
H — Helvetica
T — Times New Roman

and **d** is

R — Roman medium
I — Italic medium
S — Roman bold
J — Italic bold

Image symbol sets

The image symbol sets (screen fonts) are provided in two sizes:

9x16 suffix A
9x12 suffix C

They are:

ADMDCII – Cyrillic
ADMDLII – Latin 2

AS support

AS Version 3 and AS Version 4 support the new symbol sets. The rest of this section describes the extent of this support in more detail.

AS Names file

You should use the **:keyboard.** and **:vsscecp.** tags in the selected configuration entry of the AS Names file. The values should be as follows:

```
:keyboard.NOAPL
:vsscecp.ADMDVctd
```

where **ADMDVctd** is a suitable vector symbol set such as **ADMDVCCR**.

Screen fonts

On screens and emulators which support programmed symbol sets (for example the IBM InfoWindow 3472G) you can use the Cyrillic or Latin 2 image symbol sets in an AS session by using the **SCREEN** command with the **FONTS** option:

```
SCREEN FONT(,,,,ADMDCIIC)
```

This allows you to display data using the new character sets without you needing to install new hardware.

Data

Can be successfully input and output provided that the hardware or emulator also supports the character set and code page.

Can be compared for equality.

Can be sorted and compared according to hexadecimal code, but *not* according to appropriate cultural sequence.

Can *not* be folded from lower case to upper case.

Can be displayed on charts *provided* that the appropriate new GDDM vector symbol sets are used.

Accented characters used in other European languages such as in French and in the Scandinavian languages are not all available or are at different code points — this means that AS should be used in an AS supported language which does not use accents. For example you may use English or German.

Month names will not be recognised unless entered in the language that has been specified for the AS session.

Column names are restricted to a subset of the whole character set. However quoted column names may contain any character except for the current AS newline character.

Table names are restricted to the basic non-accented Latin character set.

Syntax

The syntactical characters used in arithmetic expressions are at standard EBCDIC code points, and present no problem.

The 6 AS special characters |, !, $, ¬, #, and @ are the same as the multilingual code page (00500), and present no special problem.