

SUPERVISOR

Introduction

Version 5.2

User Guide

V5R2M09

Software Management, Inc.

(951) 359-1087

www.smisupervisor.com

Contents

Contents	3
INTRODUCTION	6
Welcome to SuperVisor Version 5!	7
Making the case	7
Systems Software	7
Application Software	7
SuperVisor bridges the gap	7
Analysis	7
Compatibility with other Software	8
The SuperVisor System	10
SuperVisor SDM	10
Object/Source Reconciliation	10
Object to Source Code Verification	11
Archive Obsolete Objects	11
Test Data Management	12
The SuperVisor Approach	12
Making the Data Set Manageable	12
Tools and Utilities	12
SuperVisor Xdb	14
SuperVisor Assistant	14
The SuperVisor Environment	15
Project Libraries	16
External File Library	16
The SuperVisor Idea	17
Working with Objects	17
SuperVisor Autonomic Processing	17
The Autonomic Control Loop	17
What is the SuperVisor External File	18
The SuperVisor External File is simply the IBM Object Description output file	18
Data resides outside the command process	18
External File represents objects without needing the actual objects	18
Create External File	19
Convert Object Description to External File	19
Edit External File	20
How to specify the External File	21
SuperVisor *LIST	21
SuperVisor Table Service	21
Summary	22
Getting Around in SuperVisor	23
SUPERVISOR	23
Menus	23
SuperVisor SDM	23
SuperVisor Xdb	23
SuperVisor Assistant	24
SuperVisor Commands	24
SuperVisor Help	25
One Product Two Parts	26
SuperVisor SDM (Software Development Manager)	27
In SDM Each Command Stands Alone	27
SDM Command Themes	27
SDM Investigative Analysis	27
Assistant Data Base Structures	27
SuperVisor Xdb (Cross Reference Data Base)	28
Xdb Definition	28
Xdb Data Base Management	28
Xdb Data Base Usage	28
Who uses SuperVisor	29
System Architect	29
System Administrator	29
Operators	29
Production Support	29
Projects	29
SuperVisor Impact	29
What is SuperVisor used for	30
Improve productivity	30
Improves Operations	30
Improves Administration	30
Improves Architecture	30
Improves Production Support	30
Improves Projects	30
GETTING STARTED	31
Getting Started	32
SuperVisor Library List	32

Introduction

Welcome to SuperVisor Version 5

Features common to SuperVisor commands	32
Executing command	33
Here is a question:.....	33
Answer:	34
Verify job library list includes production environment	34
Secondary Question	36
What about files, what files do these programs reference	36
Edit External File	36
Use it for real	37
Edit External File Feature and Function	37
F6= Add.....	38
F18=Filter	39
Summary	41
SUPERVISOR ASSISTANT IS FREE	43
SuperVisor Assistant is FREE	44
SuperVisor Assistant Menu (ASSISTANT)	46
SuperVisor External File Menu (EXTF).....	47
Managing SuperVisor External Files	47
SuperVisor *LIST (LIST).....	55
Managing SuperVisor *LIST.....	55
Create a *LIST	56
Add entries to a SuperVisor *LIST	57
Work with commands using SuperVisor *LIST	62
Execute commands using a SuperVisor *LIST	66
Retrieve Information.....	68
Window Menu (WINDOW).....	70
Working with Window API's	70
Table Service	71
Relational Data Base 3 rd form normalization	73
What is Table Service for.....	73
Database Design	73
Detail Record or Row.....	73
Extended Factors	73
Title Fields	74
Comment Fields	74
The Hierarchy	74
Table Service Levels	74
Table Service Data Base	74
Create your own instance of the Table Service Database	75
Editing Tables.....	77
Define your own column headings	81
Control the number of available views	82
Displaying Table Service In a Window	83
Using Table Service In Applications	85
RPG Example.....	86
Output Queue Management (OUTQMAIN).....	87
Archiving Spool Files	88
Create an Archive File	89
Archive an Output Queue	90
Archive a User's Spool Files	92
Generate a Spool File from an Archive	93
What is an Output Queue Archive File?	95
Archive Job Logs	95
Execute the archive	95
View data in the archive.....	96
Generate a new report.....	97
SuperVisor Archive supports all report types.....	98
Utility Menu (UTILITY).....	99
Time Saving Tools	100
SUPERVISOR	102
SYSTEM VALUES	102
SuperVisor System Values.....	103
1 EDTSYSSIZ - Edit System Size	103
2. Edit Default File Editor (SUPFED).....	105
4 EDTUSROPT - Edit User Options	106
5. EDTEXIT - Edit Exit List on Enter.....	109
6. Edit Main Menu Attention Menu	109
5 SUPFED - Edit File Editor System Value	109
6 TSTEDTDF - Edit SuperVisor Test Default Profile	111

Introduction

Welcome to SuperVisor Version 5

7 Edit Remote Location.....	111
SuperVisor Remote Location Central.....	111
Easy to use.....	112
Ease of access.....	112
What is a SuperVisor Remote Location.....	112
In Summary.....	112
SUPERVISOR	114
COMMON INTERFACES	114
Understanding SQU, *LIST, and *EXTF.....	115
Advanced Structures within the SuperVisor program.....	115
Structured Query Utility (SQU).....	115
The SQU Parameter.....	116
The SQUKEEP parameter.....	116
SQU Sort Specifications.....	116
Examples.....	118
SuperVisor LIST (*LIST).....	119
What is a SuperVisor *LIST.....	119
Managing a SuperVisor *LIST.....	119
Using a SuperVisor *LIST.....	119
*LIST in SuperVisor Commands.....	120
*LIST Files and Personal *LISTS.....	120
External Files (*EXTF).....	121
External File Name Limitation.....	121
Managing SuperVisor External File.....	122
Using SuperVisor External Files.....	122
External Files in SuperVisor Commands.....	123
External File Objects.....	124
External File Details.....	124
More on building External Files.....	125
Is this possible?.....	126
Back to basics.....	128
SuperVisor Remote Location.....	130
SuperVisor Remote Location Central.....	130
Easy to use.....	130
Ease of access.....	130
What is a SuperVisor Remote Location.....	131
In Summary.....	131
Edit Remote Location (EDTRMTLOC) Command.....	131
Summary.....	137
Remote Location Commands.....	137

INTRODUCTION

Welcome to SuperVisor Version 5!

SuperVisor 5 provides a powerful framework designed to enable IT to exercise tighter control over OS400 operations and applications. SuperVisor provides comprehensive analyses and automates complex systems management tasks.

Making the case

Why use SuperVisor when the As/400 and iSeries have been out for so long and we have gotten along fine without it. The basic reason to use SuperVisor is we can do better. While SuperVisor offers powerful tools for understanding and managing the iSeries, it's biggest strength is in integration. Building interfaces and filling gaps.

Software management falls into two categories.

1. System Software
2. Application Software

Systems Software

System software comes from IBM, 3rd party tools, and custom programming. System software is either part of the operating system, interacts with the operating system, or enhances the operating system.

Application Software

Application software provides solutions to support the running of the business. Application software can be provided by a 3rd party software house, it could be custom written, or be a combination of both. It is made up programs written in one or more high level programming languages and is used to manipulate data. These applications may exist on one or more computer systems.

SuperVisor bridges the gap

It sounds like between system software and application software groups, everything has been covered. But there is a middle ground not being addressed by either group.

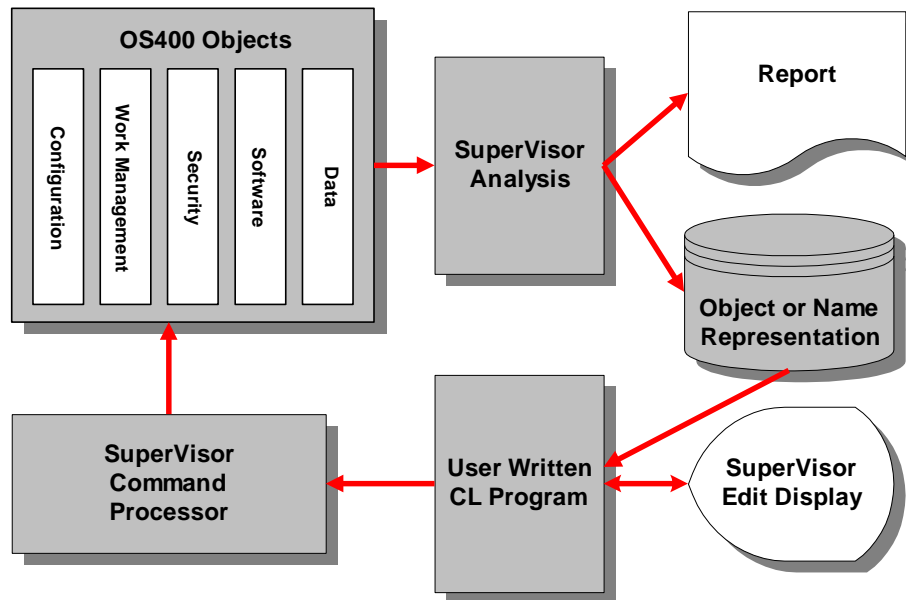
System software provides functionality to the computer system, and an infrastructure that allows the application to run. At that point, the systems group hands control over to the application group. The application group provides business solutions, designing files and writing programs. While each group has meet it's responsibility, but something is missing. Between these two groups is a functional gap. Filling this gap between the systems and applications, while also improving understanding and productivity, is the purpose of the SuperVisor product.

Analysis

SuperVisor uses the object orientation of the operating system to extract information about any aspect of the operating environment. A key to SuperVisor functionality is its ability to store representations of objects or names managed by the operating system. These representations are stored as temporary data assets that can be controlled using SuperVisor commands. SuperVisor provides a rich set of facilities to create, edit, modify, match, merge and delete, records in these temporary files. This means that iterative and nested analyses can be performed against any set of objects. Furthermore, these representations can be used to implement changes and corrections to objects on the system. Developers can create custom solutions in CL using SuperVisor and IBM commands.

Introduction

Welcome to SuperVisor Version 5



Examples of creating custom solutions using SuperVisor appear in this manual and in the Command Reference.

Compatibility with other Software

SuperVisor is designed to work with 3rd party tool sets as well as IBM supplied systems management software. SuperVisor contains more than 500 tools and utilities that can be access from a command line or used in programs you write. Developers have full access to SuperVisor data assets through CL and SuperVisor provides CL commands that can extract data from external physical files and use them in SuperVisor commands.

In some cases SuperVisor is the missing link to making other software work right. For example, to properly implement a change management system, source and object contents for production and development environments must be correct at the on set. In other words, SuperVisor provides an independent audit of the system and application integrity. SuperVisor analyses and maintenance commands can automate the process of reconciling source and objects as well as properly setting production and development environments. Then after the change management system has been used for a period of time, SuperVisor can be used to audit these environments to make sure the change management rules have not been violated.

Another example is the use of high availability software. A mirrored system will not work if modifications have altered the data assets that should be mirrored and the software has not been reconfigured properly. An analysis can be performed with SuperVisor that determines which objects require mirroring and which ones do not exist on the mirrored system.

To make sure that a naming conflict doesn't exist between SuperVisor and a 3rd party tool set run the following SuperVisor command:

```
CMPOBJD OBJ(SUPERVISOR/*ALL xxxxx/*ALL) MISSING(*NONE)
```

This command will compare all objects in the library SUPERVISOR with all objects in the library xxxxx.

Where xxxxx is the of another library name.

Introduction

Welcome to SuperVisor Version 5

MISSING (*NONE) specifies that only objects found in more than one library will be reported.

Introduction The SuperVisor System

The SuperVisor System

SuperVisor is one product made up of 2 parts. Each of these products can be purchased separately or together. This includes the following;

1. SuperVisor SDM (Software Development Manager)
2. SuperVisor Xdb (Cross Reference Data Base)

A third product that is provided free of charge is called SuperVisor Assistant. SuperVisor Assistant is a collection of functions that provide basic support capabilities used as a foundation to SuperVisor SDM and Xdb.

SuperVisor SDM

Software Development Manager provides a wide range of tools you use to enhance your ability to manage the many different aspects of tasks required to support a well organized iSeries installation. A typical iSeries installation includes one or iSeries servers or Logical Partitions. These systems are used for both software development and distribution, as well as end user production environments that use the iSeries to run end user business applications.

Software Development Manager is used to enhance the ability to exploit iSeries system resources, control access and system usage. It does this in specific ways by providing complete projects and in general by providing tools and utilities you can use to develop projects of your own.

```
MAIN                               SuperVisor5(tm) Main Menu                               System: SMI5
Select one of the following.
SuperVisor SDM "Software Development Manager"                Authorized
                                                                *PERM
    3. Object/Source Reconciliation
    4. Archive Obsolete Objects
    5. Test Data Management
    6. Tools and utilities
                                                                More...
Selection
===>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs    F16=Main menu         F18=Work with output
F21=Print menu help
```

Object/Source Reconciliation

This is a vital first step to any and all iSeries modernization projects. No matter what modernization effort you are about to undertake:

- Implementation of Change Management Systems
- Upgrade code to ILE
- Webfacing projects
- Conversion to J2EE

Introduction

The SuperVisor System

All these projects have the same thing in common. All of these projects will fail, unless you ensure that you are using the correct source code. It is vital that the time and effort required by your situation be made to ensure the following key facts are known.

- The source code used to create each object is identified and verified.
- That each deployment of the application is also verified to accurate copies of the original application.

Object to Source Code Verification

Software development and deployment are difficult tasks that go beyond the actual coding and testing that takes place. Ensuring that all of the thousands of different component parts of the application are accounted for is not always the first priority. Whatever the project being undertaken today is, we can be assured that it is not the first time the application has undergone a radical change.

Everyone is always confident that under their watch, due diligence has been given to the production source is protected. But the truth is that all applications have a history and there has been a time when expedience needed to solve specific problems caused normal channels of change control to be circumvented.

So not only is it vital to your next project that the object source reconciliation take place, using SuperVisor it is practical to do.

Archive Obsolete Objects

Before beginning any major project that deals with large numbers of objects it is important to take a moment consider ways to make the overall project more manageable. One of the ways to reduce the size of a project is to reduce the size of the project.

What is meant by this is that overtime development makes a steady progression forward solving problems and completing projects. It is not uncommon that a new project introduces objects to the system that are more modern and replace existing objects that had the same or at least similar functions. What is uncommon is taking the time to remove from the system, those objects that are now circumvented by the new.

There are many reasons for this but chief among that this sort of cleanup is beyond the scope of the current project. As we invest more time and effort into systems today, more and more time is invested in analysis of obsolete code before it can be determined that the code is obsolete. Automated Change Management systems are unable to distinguish valid dependencies and ones that are obsolete. In fact, Change Management does not include any ability to mark objects as obsolete. You can delete objects, but when you do all information that the object ever existed is also removed, along with all its history stored within the Change Management Tools.

The “Archive Obsolete Objects” sub-system provides a way to delete obsolete objects, and keep an audit trail that the object once existed. This is done, without keeping the actual object. . If information about an obsolete object is needed at a later date, this system will make it available.

Using this system, obsolete objects are organized into an Archive Project. Once all obsolete objects being archived have been identified, the objects will be removed from production, by moving them to the project archive library. The objects are retained by saving the archive saving the archive library to a save file. If space permits the save files can remain on-line. Archive save files can be saved to tape and deleted. If it becomes necessary to access the obsolete objects for some reason, the Archive System can be used to restore the objects.

Introduction

The SuperVisor System

This system is designed to provide the ability to:

- ✓ Identify objects that are obsolete
- ✓ Perform archive analysis
- ✓ Lock Project to prevent further changes
- ✓ Archive obsolete objects
- ✓ Restore objects at a later date

Test Data Management

SuperVisor Test manages environments and data for testing applications. By entering some fairly simple specifications, the user can create test data repositories, development environments and test data sets. Most of the research and work involved in setting up test environments is done by SuperVisor, enabling test managers and programmers to focus on testing rather than setting up the test.

Test Data Management is a process that needs to extract data from the production system(s) and have the following capabilities.

- Provides the ability filter and select data in a logical way that builds a representation of the production data that is only a fraction of the original size.
- Transfer the data to the development system to become the test data repository.
- Distribute the repository to the different development environments.
- Provide data for repetitive testing.

You will notice these are not steps 1, 2, 3 and 4. That is because the way these steps need to be developed depends on the nature and mix of systems, applications and environments that make the unique software development takes place for you.

The SuperVisor Approach

The SuperVisor approach to provide tools that can be linked together to build completed systems is what makes SuperVisor Test an elegant solution for test data management. As you use SuperVisor Test you will recognize that it uses many SuperVisor commands to provide the different functions and features of the system.

This component based approach then applies to SuperVisor Test. Using the commands provided in SuperVisor Test you will be able to enter and manage copy rules at the file level. A collection of copy rules becomes a SuperVisor Test Policy when combined with a library list hierarchy that equate to the different environments making up the change management promotion path.

The final integration to your environment takes place when you write some simple CL programs that integrate SuperVisor Test to your production and development environments.

Making the Data Set Manageable

The key to a successful test database is the management of the data. The main reason for just restoring the entire production backup into the test environment is size. Size is a combination of number of megabytes as well as number of records. Size affects both the disk resource used and the time required to copy data.

Rather than creating a sophisticated data model, SuperVisor uses the relationships already established by the operating system and lets the user define just those relationships necessary for copying data. *(Creating a data model is usually a huge effort. Data models must also be rigorously maintained. All too often data models are out of date when it's time to refresh the development environment.)*

Tools and Utilities

Introduction The SuperVisor System

SuperVisor SDM Tools and Utilities is unique in its implementation of programming tools for system analysis and development. By providing a standardized set of data base files, SuperVisor commands can be linked in new and different ways to create new tools to meet many different and unique programming requirements, all without any need for any high level programming.

To the software architect, this brings the capability to build software tools using standard components, much in the same way a PC can be build using off the shelf, cards and devices.

Introduction

The SuperVisor System

SuperVisor Xdb

The Cross Reference Data Base is a system of programs which has the ability to scan the different objects making up an application and converting the information into a relational data base. This presentation of the application allows someone to research the details of the application using powerful queries to display and print the detailed relationships that exist in the application.

```
MAIN                               SuperVisor5(tm) Main Menu                               System: SMI5
Select one of the following.
SuperVisor Xdb "Cross Reference Data Base"                               Authorized
                                                                              *PERM
    15. Work with Xdb Definition
    16. Manage the Xdb Data Base
Use the Xdb Data base
    19. Work with object references
    20. Work with objects where used
    21. Work with files
    22. Work with fields
    23. Work with programs
    24. Work with System/36 OCL
                                                                              More...
Selection
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs    F16=Main menu        F18=Work with output
F21=Print menu help
```

SuperVisor Assistant

SuperVisor Assistant is a set of utilities that enable users to automate AS/400 operations. Before writing either SuperVisor Xdb or SDM, the tools included in SuperVisor Assistant were written. SuperVisor Assistant provides many of the structures used as a foundation for the more advanced processes used by SuperVisor Xdb and SDM.

Introduction

The SuperVisor System

```
MAIN                               SuperVisor5(tm) Main Menu                               System: SMI5
Select one of the following.
SuperVisor Assistant               Authorized
                                   *FREE
    27. External File
    28. SuperVisor *LIST
    29. Table Service
    30. Tools and Utilities
SuperVisor Configuration
    33. Edit authorization code
    34. Configure Remote Location
    35. Edit system values
    36. Select commands by name
                                   Bottom
Selection
====>
F3=Exit      F5=Next Menu- *SELECT   F6=Display Message   F10=Command entry
F12=Cancel  F14=Work with jobs      F16=Main menu       F18=Work with output
F21=Print menu help
```

The SuperVisor Environment

SuperVisor V5R2 supports any AS/400 at OS/400 version 5.1 or above. It supports Native, ILE, System36, and System/38 development models. Most SuperVisor analysis and cross-referencing functions are language independent. When source needs to be interpreted, SuperVisor supports all versions of RPG, COBOL, CL and OCL.

SuperVisor objects reside in the library SUPERVISOR. Any job, whether interactive or batch, that executes a SuperVisor command, must have the libraries SUPERVISOR, QGPL and QTEMP on the library list. Once you do this SuperVisor is fully integrated into your system and can be used as you would any iSeries command.

The SuperVisor library, and all its objects, is owned by the user profile SMI and reside in the library SUPERVISOR. This user profile is created during the initial load of SuperVisor and should not be deleted unless the SuperVisor library is deleted first. The user profile SMI is only used to own SuperVisor objects and does not require any special authorities.

All SuperVisor physical and logical files have *PUBLIC authority *ALL. All other SuperVisor objects have *PUBLIC authority *NORMAL.

This organization provides the following capabilities:

- The SUPERVISOR library is owned by the user profile SMI. This ensures that updates to SuperVisor will operate under known conditions, making the creation of future release load programs possible. The user profile SMI exists solely to own the SuperVisor objects.
- SuperVisor commands execute using *USER authority, with just a few exceptions.
- The objects found in the SuperVisor library SUPERVISOR may be used by any user. However, if the user does not have the proper authority to the objects selected by the SuperVisor command, no information to the command will be available.
- SuperVisor contains no Machine Instruction Programs and will operate under any AS/400 security level.

Introduction The SuperVisor System

Project Libraries

SuperVisor Project Libraries are created automatically when using the SuperVisor project subsystem commands. Project libraries provide a place to store important information specific to the projects you create when using these special systems included in the SuperVisor program.

It is not necessary to include these libraries in your jobs library list. The names of these libraries are part of the project definition and are accessed implicitly.

Library	Description	Purpose
SUPERRDTA	Object Source Reconciliation Project Library	Stores all project information created when using the Object/Source Reconciliation Project commands.
SUPERADTA	Archive Obsolete Objects	Stores all project information created when using the Archive Obsolete Objects Project commands.
SUPERTST	Test Data Management Project Library	Stores all project information created when using the Test Data Management Project commands.

External File Library

The results of many SuperVisor commands can be placed into data base files. The External File library SUPEREXTF is provided as a default location whenever SuperVisor is directed to output information to a data base file.

Library	Description	Purpose
SUPEREXTF	SuperVisor External File	This library is used as a default location to store different SuperVisor External Files. This library is simply a default, SuperVisor External Files can be placed in any iSeries library.

The SuperVisor Idea

The User Guide has covered a lot of material and there is still much more to go. But before we get too far into a description of different SuperVisor capabilities, this section will be a description of the SuperVisor idea.

Working with Objects

The SuperVisor Idea is really very simple. Provide a better way to manage the iSeries information that goes beyond object attributes. The most successful product on the iSeries is Programming Development Manager or PDM. This idea continues when using WDS and Remote System Explorer and is the management of work using object attributes. The iSeries object being the way iSeries presents itself to the real world.

- Work with objects by Library WRKLIBPDM
- Work with objects by Object WRKOBJPDM
- Work with files by member WRKMBRPDM

Note: Files and members being a more complex object.

The SuperVisor Idea is to take the iSeries idea one more level.

- Work with objects by External File EDTEXTF

The External File is the SuperVisor Idea for Object Management on the iSeries. What makes the External File powerful is that objects in the External File are associated based on relationships beyond object type and attribute.

SuperVisor Autonomic Processing

When SuperVisor was being developed we did not know about another idea being used today. "Autonomic Processing" as presented by IBM. While the IBM focus for "Autonomic Processing" is the management of hardware, when you look at that definition, the artifact you need on the iSeries to make the idea of "Autonomic Processing" work, is the SuperVisor External File.

The SuperVisor "External File" standardizes data collection from many complex processes, as object descriptions. The object being the key component to the way the iSeries stores information.

Next SuperVisor provides hundreds of commands that perform complex analysis which can place the results of the analysis into an External File.

Finally SuperVisor includes additional commands which can use the External File as input so that automatic actions can be built

The Autonomic Control Loop

IBM defines the "Autonomic Control Loop" below

The Control loop is the system by which events can be detected and dealt with. The process involves four steps:

1. **Monitor:** First, the system looks for the events, detected by the sensor from whatever source -- be it a log file or an in-memory process. The system uses the knowledge base to understand what it's looking at.

Introduction

The SuperVisor Idea

2. **Analyze:** When an event occurs, the knowledge base contains information that helps to determine what to do about it.
3. **Plan:** After the event is detected and analyzed, the system needs to determine what to do about it using the knowledge base. The symptom database might have information, or a central policy server might determine the action to take.
4. **Execute:** When the plan has been formulated, it's the effector that actually carries out the action, as specified in the existing knowledge base.

Source: IBM Website "An autonomic computing roadmap"

Use SuperVisor to build the unique "Autonomic Process" to meet your specific needs, using commands included in the SuperVisor. Program by building process of bring together commands using and then re-using the SuperVisor External File.

What is the SuperVisor External File

You talk about output to External File or use External File for input to SuperVisor commands. What exactly is this External File and how is it different than the IBM output file.

The SuperVisor External File is simply the IBM Object Description output file.

The SuperVisor External File is the standard IBM Object Description output file format QLIDOBJD. The SuperVisor External File has a few additional attributes.

1. There is both a physical file and logical view.
 - a. **External File name is limited to 8 characters**
 - b. External Logical file is named by appending a '1' to the end of the physical file name.
2. The logical view is keyed by Object Name, Library, Type and Attribute.
3. All date fields in the logical view are divided into separate year, month and day fields.

Note: You can still use the DSPOBJD and output to the physical file portion of the External File. Contrast the IBM DSPOBJD command with the SuperVisor LSTOBJD command.

Data resides outside the command process

The important thing about the SuperVisor External File is that it is used by SuperVisor as an output file, meaning that SuperVisor commands can place the results of an analysis in a file.

- Everyone is used to that.

What is different is the ability to specify the External File as input to a command as well. When specified as input, it is the records in the file that will be processed as representations of objects.

External File represents objects without needing the actual objects.

To take the mystery out the External File, compare it to a library. Using the IBM Work with Objects using PDM (WRKOBJPDM) command, view the contents of the library SPOOBJ. This is a huge difference between External Files and anything else.

Introduction

The SuperVisor Idea

Executing the following command:

```
Parameters or command  
==> wrkobjpdm spoobj
```

Execute the command to display the list display.

```
Work with Objects Using PDM SMI  
  
Library . . . . . SPOOBJ Position to . . . . .  
Position to type . . . . .  
  
Type options, press Enter.  
2=Change 3=Copy 4=Delete 5=Display 7=Rename  
8=Display description 9=Save 10=Restore 11=Move ...  
  
Opt Object Type Attribute Text  
GENPO *PGM RPGLE Generate Test Purchase Orders  
PAA01A *PGM RPGLE Enter/Update Purchase Order  
PAA01B *PGM RPGLE Duplicate Purchase Order  
PAA02A *PGM RPGLE Enter Approved Purchase Orders  
PAA03A *PGM RPGLE Enter Purchase Order Receipts  
PAA05A *PGM CLLE Print all Open Purchase Orders  
PAA05B *PGM RPGLE Print All Open Purchase Orders (Batch  
PAA05C *PGM RPGLE Print Purchase Order  
  
More...  
  
Parameters or command  
==>  
F3=Exit F4=Prompt F5=Refresh F6=Create  
F9=Retrieve F10=Command entry F23=More options F24=More keys
```

The objects being displayed can only be objects in the library, filtered by object attributes.

Create External File

The Create External File creates the file SPOOBJ in the library SUPEREXTF.

```
Type command, press Enter.  
==> CRTEXTF EXTF (SUPEREXTF/SPOOBJ) MBROPT(*REPLACE) BASED_ON(LSTOBJD) TEXT('Li  
brary SPOOBJ objects')
```

Convert Object Description to External File

Next use the Convert Objects to External File (CVTOBJEXTF) command.

Introduction

The SuperVisor Idea

```
Type command, press Enter.  
====> CVTOBJEXTF OBJ(SPOOBJ/*ALL) OUTF(SPOOBJ) OUTL(SUPEREXTF) OUTMBROPT(*REPLAC  
E)
```

Note: Yes, the CVTOBJEXTF command will automatically create the external file if it does not already exist.

Edit External File

The Edit External File (EXTEXTF) command allows you to work with the object descriptions.

```
Type command, press Enter.  
====> edtextf superextf/spoobj
```

Executing the command will cause the Edit External File list screen to display.

```
EDTEXTF          Edit External File
                                     Position . . . .
Object . . . . . SPOOBJ      Library SPOOBJ objects
Library . . . . . SUPEREXTF  Created by SuperVISOR(tm) system

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source      4=Remove entry
5=Display program file 6=Print      7=Clear Source Ref  8=Edit Source Ref
Object      Object      Object      Object      User      Object
Opt  Name      Library     Type        Attribute   Attribute  Level
-----
GENPO      SPOOBJ     *MODULE     RPGLE
GENPO      SPOOBJ     *PGM        RPGLE
NATMNU     SPOOBJ     *FILE       DSPF                    51191821
NATMNU     SPOOBJ     *MENU       DSPF
NATMNU     SPOOBJ     *MSGF
PAA01A     SPOOBJ     *MODULE     RPGLE
PAA01A     SPOOBJ     *PGM        RPGLE
PAA01ADEV  SPOOBJ     *MODULE     RPGLE

                                     More...

F3=Exit      F4=Prompt    F6=Add      F5=Refresh
F10=Command entry  F11=Next view  F23=More options  F24=More keys
```

This view differs from the WRKOBJPDM screen in that this information only represents objects in a library, while WRKOBJPDM is the objects in the library.

There plenty of different screen views, or function keys and/or options to support all kinds of work. User options with special features are allowed for as well.

Since this is only records in a file it is easy to change or alter the information.

- Add more object records
- Delete object records
- Clear source file references

Introduction

The SuperVisor Idea

- Edit source file references to new values.

How to specify the External File

Keep in mind that the External File is a data base object that includes data that can be processed. Specifying the External File is keeping clear which level you are working at. The distinction is very similar to DDM file objects. Are you processing the DDM object file, or the remote objects described by the DDM file?

Process the External File objects:

```
LSTOBJD OBJ(SUPEREXTF/SPOOBJ) TYPE(*FILE) SUBTYP(*DBF)
```

Process the External File entries:

```
LSTOBJD OBJ(SUPEREXTF/SPOOBJ) TYPE(*EXTF)
```

Note: Keep in mind that the External File name is limited to 8 characters.

SuperVisor *LIST

As great as the External File is, there is a problem. You cannot store object descriptions for something that does not exist. SuperVisor *LIST is a second data base structure supported by SuperVisor that can also receive the results of a process, or be used as input to drive additional processing.

A SuperVisor *LIST entry has no attributes and must be unique. It is not possible to store duplicate *LIST entries. These two characteristics make a SuperVisor *LIST a complement to the External File. Like the External File SuperVisor commands can output to a SuperVisor *LIST. You can also use a SuperVisor *LIST as input to a command.

A SuperVisor *LIST is also made up of a physical file and logical view. Each *LIST is represented as a separate member. When you add or remove a SuperVisor *LIST, you are actually adding and removing members in these files. SuperVisor *LIST entries are records in the file.

Process the SuperVisor *LIST objects:

```
LSTOBJD OBJ(SUPERVISOR/ULSTP00) TYPE(*FILE) SUBTYP(*DBF)
```

Process SuperVisor *LIST members:

```
LSTPFA FLE(SUPERVISOR/ULSTP00)
```

Process the SuperVisor *LIST entries:

```
EDTLIST LIST(*SELECT)
```

Note: Keep in mind that the physical SuperVisor *LIST is a physical file and logical view. Each list is a member in the file. SuperVisor *LIST entries are records in the file, which is a single field 10 characters in length. A *LIST entry has no attributes and SuperVisor *LIST can represent a group of anything.

SuperVisor Table Service

Table Service is a third data base that comes with SuperVisor. This data base is a place to store new information that can then be integrated into your programming. Table Service is more complex than either

Introduction

The SuperVisor Idea

the External File or a SuperVisor *LIST and is used to organize information into a 4 tiered hierarchy, much like the library, object, file, member found in the iSeries. Use Table Service to store and organize system values and processing options.

Summary

So the “SuperVisor Idea” is to expand Programming Management, beyond object and type as provided by PDM. To do this SuperVisor includes 3 data base structures that are integrated into the SuperVisor programming.

- External File
- SuperVisor *LIST
- Table Service

It is the integration of these data base structures into the more than 500 commands included in the SuperVisor program that makes SuperVisor unique. The flexibility to use common formats as both input and output to existing commands that provides the unique ability to create new commands using simple CL programs, or even no programming at all.

Support for these structures is provided by commands included in the SuperVisor Assistant subsystem.

Introduction

Getting Around in SuperVisor

Getting Around in SuperVisor

SUPERVISOR

To use SuperVisor add the library SUPERVISOR to your library list and execute the command SUPERVISOR. This will display the SuperVisor Main Menu.

Menus

```
MAIN                               SuperVisor5(tm) Main Menu                               System: SMI5
Select one of the following.
SuperVisor SDM "Software Development Manager"                               Authorized
                                                                              *PERM
    3. Object/Source Reconciliation
    4. Archive Obsolete Objects
    5. Test Data Management
    6. Tools and utilities
                                                                              More...
Selection
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs    F16=Main menu         F18=Work with output
F21=Print menu help
```

SuperVisor contains more than 500 commands and processes. The menu system organizes these commands according to the analysis or maintenance function to be performed. As shown above the main menu is divided into 3 main sections. Keying in the command SUPERVISOR from any command entry line will redisplay the main menu.

SuperVisor SDM

Page 1 of the main menu provides access to the Software Development Manager options.

SuperVisor Xdb

Getting Started

Page 2 on the Main Menu presents the options needed to use the Cross Reference Data Base

```
MAIN                               SuperVisor5(tm) Main Menu                               System: SMI5
Select one of the following.
SuperVisor Xdb "Cross Reference Data Base"                               Authorized
                                                                           *PERM
    15. Work with Xdb Definition
    16. Manage the Xdb Data Base
Use the Xdb Data base
    19. Work with object references
    20. Work with objects where used
    21. Work with files
    22. Work with fields
    23. Work with programs
    24. Work with System/36 OCL
                                                                           More...
Selection
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs    F16=Main menu        F18=Work with output
F21=Print menu help
```

SuperVisor Assistant

Page 3 on the Main Menu provides access to Assistant, as well as special functions used to manage the SuperVisor product.

```
MAIN                               SuperVisor5(tm) Main Menu                               System: SMI5
Select one of the following.
SuperVisor Assistant                               Authorized
                                                                           *FREE
    27. External File
    28. SuperVisor *LIST
    29. Table Service
    30. Tools and Utilities
SuperVisor Configuration
    33. Edit authorization code
    34. Configure Remote Location
    35. Edit system values
    36. Select commands by name
                                                                           Bottom
Selection
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs    F16=Main menu        F18=Work with output
F21=Print menu help
```

SuperVisor Commands

Nearly all of SuperVisor menu options execute a SuperVisor command. This means that the same function may also be performed by executing a command instead of having to go to a particular menu. This also means that SuperVisor functions may be called in CL programs that combine several functions. In essence systems programmers can create their own specialized tools using SuperVisor commands.

Each SuperVisor menu has a work with commands option. This option will present a "Work With Commands" display for that particular SuperVisor component. In addition, the SuperVisor command SUPERCMD will display a list of SuperVisor commands.

Getting Started

```
Alphabetical Command Menu (SUPERCMD)

Type choices, press Enter.

Command Group . . . . . *ALL          *ALL, ADHOC, BEGIN...
```

This brings forward the following display.

```
SUPERCMD          SuperVISOR Commands

Type Option, Press Enter          Position . . . . .
1=Select 5=Display

Opt Command      Text Description

  ADDDATE        ANALYSIS - Add days to date
  ADDJOBdle      MANAGEMENT - Add *JOBd Library List Entry
  ADDLIST        MANAGEMENT - (L) Add List
  AGEOUTQ        MANAGEMENT - (Q) Age Printer Queue to ODD/EVEN
  ALCMMBRG       MANAGEMENT - Allocate Multiple Members *GENERIC
  ALCMOBJG       MANAGEMENT - Allocate Multiple Objects Generic
  ANLRPGSRC      ANALYSIS - Analyze RPG Source Programs
  ANZJOBd        ANALYSIS - Analyze Job Descriptions
  ANZS36         ANALYSIS - Analyze System/36
  ANZUSRp        ANALYSIS - Analyze User Profiles
  ARCOUTr        MANAGEMENT - (Q) Archive Printer Output Queue
  ARCSPLF        MANAGEMENT - (Q) Archive Spool File
  ASKDCeCMT      TABLE SERVICE - Ask Code Comments

                                                    More...

F3=Exit  F10=Command entry  F11=Text sequence  F12=Return
```

The command group parameter selects a subset of the commands to display. *ALL will list all SuperVisor commands.

SuperVisor Help

SuperVisor provides On-line Help screens that are easily accessible from displays used throughout the system. There are three ways to view help text.

- **Cursor Placement** - Move the cursor to the position on the screen for which you want information. When you press the HELP KEY or F1
- **General Screen** - Pressing help when the cursor is on the command line displays general help information about the screen.
- **Printed Text** - F21 prints all related help information about the display including general and specific information.

Introduction
One Product Two Parts

One Product Two Parts

SuperVisor is single product made up of two parts.

1. SuperVisor Xdb (Cross Reference Data Base)
2. SuperVisor SDM (Software Development Manager)

SuperVisor SDM (Software Development Manager)

SuperVisor SDM should be installed on every iSeries or LPAR in your network. This includes the iSeries or LPAR systems used for software development, production or High Availability systems.

In SDM Each Command Stands Alone

SuperVisor SDM is easy to learn because you only need to learn how to run a single command. While each command includes a variety of parameters, parameters have defaults, so it remains simple to do.

SuperVisor SDM command names sound similar to what they do.

XREFPGMMOD	Cross Reference Program Modules
LSTOBJD	List Object Description

SDM Command Themes

There are many cases where there are a number of commands that perform similar functions or work in similar ways. These commands are grouped by function as found on the different SuperVisor menus, or grouped by name, where the command name denotes function.

SDM Investigative Analysis

It would be great to implement a single command that could do everything we would every want to do. The problem is that it would have so many parameters no one would ever be able to figure out how to use it.

SuperVisor SDM does the next best thing.

Use information from one command to control the processing of the next. Gather information using multiple commands by collecting the information together in a single place. Then use that information to perform the process you really want to do.

Assistant Data Base Structures

SuperVisor Assistant provides a number of different data base structures that provide the foundation to the different SuperVisor commands need to work. These structures include the following.

1. SuperVisor External File
2. SuperVisor *LIST
3. SuperVisor Table Service

By making these known data base structures available to SuperVisor SDM commands, it is possible for you to write CL programs that can perform complex analysis and processes without the need for any High Level Language programming.

These three data base structures are used to receive the results of different SuperVisor commands and can also be used as input to SuperVisor commands. Most importantly, additional commands provide the ability to convert information from one structure to another.

The key to using SuperVisor is found in the understanding and use of these three data base structures.

SuperVisor Xdb (Cross Reference Data Base)

The Cross Reference Data Base should be installed on the iSeries system used to support software development. This iSeries supports developers and includes both the executable application and the application source code.

Xdb Definition

The SuperVisor Xdb is implemented by using an Xdb Definition which lists each iSeries library that will be included in the cross reference. Then for each library a list of source files is given which identify where to look for the source code used to create the application.

Xdb Data Base Management

The Xdb Data Base is created by executing a build process. This process retrieves the Xdb Definition, and then processes each library to build a data base that contains detailed information about each object in the library.

Xdb Data Base Usage

When completed the Xdb Data Base can be access by different commands to extract important information about the application using capabilities of a relational data base to access the application internal details.

Who uses SuperVisor

SuperVisor supports virtually every aspect of iSeries, from software development to production support and operations. Because SuperVisor offers support in such a wide area it is often difficult to get your hands around SuperVisor. So before we go any further it is worth the time to discuss not what SuperVisor does, but how it uses SuperVisor.

System Architect

Any evaluation of SuperVisor would require the participation of the iSeries system architect. This would be the person responsible for the iSeries or iSeries network. It is the architect that would best understand the existing iSeries installation and how bringing in SuperVisor will help the different groups using and supporting the iSeries systems and applications. It is the System Architect that are responsible for the overall design and implementation.

The architect creates the policies that govern the systems and designs the applications that implement those policies.

System Administrator

Systems administrators are responsible for ensuring the iSeries systems run each day as designed. The System Administrator controls access and system integrity. The administrator needs to know that only authorized users have access to the system and that the systems designed by the architect are implemented as designed.

The administrator ensures that both hardware and software systems are current and the proper versions of all software are installed and used by the proper people.

Operators

Operators monitor the system and ensure that all jobs run on time and complete normally. Operators ensure that all exceptions are identified and expedited to the proper production groups.

Production Support

These groups include personnel with expertise in both the system and business applications and are responsible for investigating problems and correcting any exceptions that may occur to the normal processing of the iSeries operation.

This group needs detailed information that explains how the different aspects of the system work and the means to make and implement changes as needed by the operations of system administrators.

Projects

No computer implementation is static. Planned changes take all forms and include hardware and software, systems and application programming. Change to a single object, or the replacement of entire systems.

SuperVisor Impact

All of these groups have different goals and responsibilities. What they have in common is that they all support the iSeries, its users and applications. They all require knowledge and information about the iSeries and the ability to automate processes and procedures to allow them to better do their job.

SuperVisor is an iSeries tool kit used by iSeries professionals to improve their ability to understand and act in ways that would be impossible without SuperVisor.

Introduction

What is SuperVisor used for

What is SuperVisor used for

SuperVisor is used to enhance your use of the IBM iSeries computer. The iSeries is an advanced computer system designed to be the ideal platform for business processing. SuperVisor is an extension to the iSeries allowing iSeries professionals to better manage their system, secure its resources and maximize their time.

Improve productivity

SuperVisor provides significant enhancements to the iSeries by providing access to information that would not be accessible without some serious programming. There will be little need to learn APIs when SuperVisor commands provide the same information in consistent the easy to use command interface.

Improves Operations

The operations group will use tools SuperVisor to improve their understanding of the system operational details. For example, operators will have research tools needed to understand jobs and how they relate as easily as they can now display the system status.

Improves Administration

For the system administrator SuperVisor includes everything that is needed to ensure a well run iSeries installation. Not only will you use SuperVisor commands on a daily basis to monitor the system, SuperVisor will be used for managing system resources, controlling system access and managing your iSeries upgrades.

Improves Architecture

For the System Architect SuperVisor turns plans and ideas into realities. SuperVisor will play the crucial role in the implementation of almost every aspect of the iSeries. Whether you need help with simple tasks such as output queue management to the complex requirement for test data management, the System Architect will benefit the most by having SuperVisor tools available and will soon wonder how systems management was done without it.

Improves Production Support

Production support required by complex iSeries business applications is the most costly aspect of day to day life. The combination of real time analysis provided by SuperVisor SDM and the comprehensive attention to detail provided by the SuperVisor Xdb places the right tools into the hands of the production support staff. Using the Cross Reference Data Base for impact analysis and "Software Development Manager" for real time investigation of the production environment will expedite both production fix and root cause determination.

Improves Projects

The iSeries business systems in service today are of tremendous value to every organization. This in both the business knowledge stored in the application programming, but in the value they bring to the organization in its ability to do business. Any modernization project which improves the capabilities or our understanding of these business systems is vitally important. SuperVisor brings the professional tools needed to analyze these systems, make changes and implement upgrades, making SuperVisor a key player in any iSeries modernization effort.

GETTING STARTED

Getting Started

SuperVisor is a very comprehensive program and includes support for every fact of live at an iSeries installation. The purpose of this first section is to introduce you to some interesting areas of SuperVisor so you can get an idea of what kind of things you can do. What you may focus on depends on your need and the makeup of in-house and toolsets you already have.

SuperVisor Library List

To use SuperVisor ensure the library SUPERVISOR and QTEMP is in your library list. This makes SuperVisor fully integrated into you system. SuperVisor is command driven and all the commands are in the library SUPERVISOR. Once this is done all you need to do is to learn the names of the commands you want to use.

Features common to SuperVisor commands

SuperVisor SDM is a completely different approach to system analysis, when compare to the Cross Reference or any other iSeries tool available anywhere. Compare to the Xdb, SuperVisor SDM is easy to use in that each command performs a limited function that being and ends with that command. The SuperVisor Cross Reference Programs Modules (XREFPGMMOD) is an example, showing the components common to many SuperVisor commands.

The screen prints for this command have each parameter identified as to the role it places in the command processing.

```

                                X-Ref Programs Modules (XREFPGMMOD)

Type choices, press Enter.

1 Select programs . . . . . PGM
  Library name . . . . . *LIBL
                                + for more values
                                *LIBL
1 Select object type . . . . . TYPE      *PGM
1 Select Sub-type . . . . . SUBTYP      *ALL
1 Select *LIST library search . . LISTLIB *LIBL
1 Module reference library . . . REFLIB  *OBJLIB
2 Module filter . . . . . FILTER        *NONE
2 SuperVISOR *LIST ? . . . . . *NO
2 Create date . . . . . CRTDTE         *ALL
2 Source last change date . . . SRCCHGD *ALL
2 Source last change time . . . SRCCHGT *ALL
3 Calling/Module Report . . . . RPT1    *YES
3 Module/Calling Report . . . . RPT2    *YES
4 Output module/calling programs SELECT *CALLING

                                More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys

```

Getting Started

```

                                X-Ref Programs Modules (XREFPGMMOD)

Type choices, press Enter.

4 SuperVisor *LIST name . . . . . LIST          *NONE
4 SuperVisor *LIST Member option LSTMBROPT      *REPLACE
4 SuperVisor *LIST description . . LSTDSC       *SAME

4 Output External File . . . . . OUTF          *NONE
4  Library name: . . . . . OUTL              SUPEREXTF
4  Member option . . . . . OUTMBROPT        *REPLACE
4 Output External File Text . . . OUTT         *SAME

5 Printer Output Queue: . . . . . OUTQUE
5  Library name: . . . . .                  *LIBL
5 Job Desc. for Submit Job: . . . . . JOBD
5  Library name: . . . . .                  *LIBL

                                Bottom

F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

1. Identify objects to be processed
 - Process objects found in a library
 - Process objects already identified in an External File
2. Filters to select objects
 - Select objects based on a variety of filtering options
3. Special processing
 - In this case, enter the library used to locate the modules being referenced by the selected objects.
4. Output the results
 - Print the results
 - Output results to an External File
 - Output results to a SuperVisor *LIST
5. Job Control Parameter
 - Control if processing should be submitted to batch
 - Specify if reports should be placed in an output queue other than the job default.

Executing command

System analysis on the iSeries or any computer system is similar to word problems in math. First you formulate the question, then choose the SuperVisor commands that will provide you the results.

Note: Much easier then finding an API then writing a program, then building a process structure within which the program can run.

Here is a question:

Getting Started

On the production system, get me a list printing the last used date for all programs using the module PBA24A.

Answer:

To do this two SuperVisor SDM commands will be used.

Command	Description	Action
XREFPGMMOD	Cross Reference Program Modules	Process all programs in the library SPOOBJ. Select only programs using the module PBA24A. Output results to an External File
LSTOBJD	List Object Description	Print the object descriptions found in the specified External File.

Note: Running SuperVisor SDM commands have similar requirements as executing the application. There is no Xdb Definition to define scope used to locate referenced objects. Referenced objects are located using the job library list, just as they do when executing the programs.

Verify job library list includes production environment

The library list for the job is set for executing the application, plus the library SUPERVISOR.

Sequence	
Number	Library
0	
10	SPOOBJ
20	SPOSRC
30	SPOSUP
40	SUPERVISOR
50	QGPL
60	QTEMP

Place named of programs using module PAA05C to External File

```
Type command, press Enter.
====> XREFPGMMOD PGM(SPOOBJ/*ALL) TYPE(*PGM) FILTER(PAA05C) RPT1(*NO) RPT2(*NO)
OUTF(MYEXTF) OUTL(MYLIB) OUTMBROPT(*REPLACE) OUTT('Programs using module PAA05C'
)
```

Process external file and print object usage report

Getting Started

```
Process External File and print Object Usage Report
Type command, press Enter.
==> LSTOBJD OBJ(MYLIB/MYEXTF) TYPE(*EXTF) RPT1(*NO) RPT2(*YES)
```

Secondary Question

How about Service Programs. Is module PAA05C used in any Service Program?

```
Selection or command
===> XREFPGMMOD PGM(SPOOBJ/*ALL) TYPE(*SRVPGM) FILTER(PAA05C) OUTF(MYEXTF) OUTL
(MYLIB) OUTMBROPT(*ADD)
```

This time the XREFPGMMOD processes service programs, then adds the results to what was already in the file MYLIB/MYEXTF.

```
Type command, press Enter.
===> LSTOBJD OBJ(MYLIB/MYEXTF) TYPE(*EXTF) RPT1(*NO) RPT2(*YES)
```

Printing the External File is the same as before

Note: The External File can contain the results of any number of different processes and then it can still be edited manually. As a result the objects in an External File represent a relationship. Basing processing on a relationship is a very powerful capability to have. Especially when it can be created as easily as simply running SuperVisor commands.

What about files, what files do these programs reference

Research is research. Questions are not always complete. As you investigate new information poses new questions. This is something we call “Investigative Analysis” The concept seated in reality that at the start we don’t even know what the right question. It is only after we start down the road that we begin to get a true picture of what we are looking for.

The External File MYLIB/MYFILE contains the object description list for all programs and service programs using the module PAA05C. We can use that to drive the Cross Reference Programs Files (XREFPGMFLE).

```
Selection or command
===> XREFPGMFLE PGM(MYLIB/MYEXTF) TYPE(*EXTF)
```

This command will process the objects identified by the External File MYLIB/MYFILE and print two reports.

Edit External File

The External file is data base file containing iSeries object description records, which represent iSeries objects.

Here is a brief peek at the contents of a SuperVisor External File

Getting Started

```
Selection or command
==> edtextf mylib/myextf
```

Will display the contents of the External File

```
EDTEXTF          Edit External File
                                     Position . . . .
Object . . . . . MYEXTF      Programs or Srv Pgms using module PBA24A
Library . . . . . MYLIB      Created by SMI SuperVisor5 system

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source      4=Remove entry
5=Display program file 6=Print      7=Clear Source Ref  8=Edit Source Ref
Object      Object      Object      Object      User      Object
Opt  Name    Library    Type      Attribute  Attribute  Level
PAA05B     SPOOBJ     *PGM      RPGL       *ILE
PAA05C     SPOOBJ     *PGM      RPGL       *ILE
PAA06A     SPOOBJ     *PGM      RPGL       *ILE

                                     Bottom

F3=Exit      F4=Prompt    F6=Add      F5=Refresh
F10=Command entry  F11=Next view  F23=More options  F24=More keys
```

Very similar to the DSPOBJD screen and it is used in similar ways. Records in the External File represent objects, without being objects. Records in an External File can represent a group objects for any reason.

You can see how External File got its name. The SuperVisor External File is not just an output file. It exists on its own independent of the commands used to create it, or commands that may use it later on.

This is all for External Files for now. See information in other sections of this manual to learn more about External Files and things you can do with them.

Use it for real

It is important to remember that when using SuperVisor SDM, **the job library list must be set exactly the same as if you were going to execute the programs.** Then add library SUPERVISOR and QTEMP, usually at the end.

In the past your tools worked on the development system and you needed to assume that everything is installed exactly the same on each distribution of the system. With SuperVisor SDM you don't need to assume anything ever again. Ask the question, perform the analysis. You know if things are ok because you are using the same environment the users of the application will use.

Edit External File Feature and Function

Like PDM, the Edit External File (EDTEXTF) command is used to work with and manage iSeries objects.

Getting Started

```
EDTEXTF          Edit External File

                                Position . . . .
Object . . . . . MYEXTF      Programs or Srv Pgms using module PBA24A
Library . . . . . MYLIB      Created by SMI SuperVisor5 system

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source      4=Remove entry
5=Display program file 6=Print      7=Clear Source Ref  8=Edit Source Ref

Object      Object      Object      Object      User      Object
Opt  Name    Library    Type       Attribute   Attribute  Level
PAA05B     SPOOBJ    *PGM      RPGLE      *ILE
PAA05C     SPOOBJ    *PGM      RPGLE      *ILE
PAA06A     SPOOBJ    *PGM      RPGLE      *ILE

Bottom

F3=Exit      F4=Prompt    F6=Add      F5=Refresh
F10=Command entry  F11=Next view  F23=More options  F24=More keys
```

F6= Add

Press F6 to Add records to the External File.

```
EDTEXTF          Edit External File

                                Position . . . .
Object . . . . . MYEXTF      Programs or Srv Pgms using module PBA24A
Library . . . . . MYLIB      Created by SMI SuperVisor5 system

T .....
:  Add records to External File      : urce  4=Remove entry
:  Select one of the following, press enter.  : 8=Edit Source Ref
:    1. Convert Object Description      :      Object
O :    2. Convert Member List           : ibute  Level
:  F12=Cancel                          :
:                                       :
:.....:

Bottom

F3=Exit      F4=Prompt    F6=Add      F5=Refresh
F10=Command entry  F11=Next view  F23=More options  F24=More keys
```

Select 1 "Convert Object Description" to add records based on objects.

Getting Started

```

Convert Object Description (CVTOBJEXTF)

Type choices, press Enter.

Object--Generic* or *ALL . . . . *all      Name, generic*, *ALL, *SELECT
Object Library name . . . . . spobj      Name, generic*, *USRLIBL...
      + for more values      *all
                          sposup
Object type: . . . . . > *ALL      Character value, *ALL...
Sub-Type (Files/Programs): . . . *ALL      Character value, *ALL...
From create date QDATFMT . . . . 000000      Date
To create date QDATFMT . . . . . 999999      Date, 999999
OR From last used date QDATFMT . . 000000      Date
To last used date QDATFMT . . . . 999999      Date, 999999
Remote Object Generic* or *ALL      Name, generic*, *ALL
Remote Library name . . . . .      Name, generic*, *ALLUSR, *ALL
Remote location name . . . . . *NONE      Name, *NONE
      + for more values

                          *NONE
                                          More...

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys

```

Records can be added while editing the file, or by using any SuperVisor command which supports output to the External File. In this example you can see that the Convert Object to External File (CVTOBJEXTF) command is used in both cases.

```

EDTEXTF          Edit External File

                          Position . . . .
Object . . . . . MYEXTF      Programs or Srv Pgms using module PBA24A
Library . . . . . MYLIB      Created by SMI SuperVisor5 system

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source      4=Remove entry
5=Display program file 6=Print      7=Clear Source Ref 8=Edit Source Ref

Object      Object      Object      Object      User      Object
Opt Name      Library     Type        Attribute   Attribute  Level
SPOBNDIR   SPOOBJ      *BNDDIR
SPOJOB    SPOSUP      *JOB
SPOJRN     SPOOBJ      *JRN
SPOJRN1000 SPOBJ      *JRNRCV
SPOJRN2000 SPOBJ      *JRNRCV
SPOJRN3000 SPOBJ      *JRNRCV
UCACL01    SPOSUP      *FILE      LF
UCACP00    SPOSUP      *FILE      PF

                                          More...

F3=Exit      F4=Prompt      F6=Add      F5=Refresh
F10=Command entry F11=Next view  F23=More options F24=More keys

```

F18=Filter

Filters can be used to limit the records being displayed.

Getting Started

```

EXCQRYS                      Edit External File                      9/03/06
1 Field Name Type Length Dec Description
ODAAPI      A          1      Allow change by API: 0-No,1-Yes
ODAPAR      A          6      APAR ID
ODASP       N          2      0 ASP number
ODBPUN      N         10      0 Bytes per unit
ODLCEN      A          1      Change century
ODLDAT      A          6      Change date: format- mmddyy

Type changes, press enter
1=Add 2=Change 3=Copy 4=Delete 9=Insert
Opt Oper Field Name Rel Selection Criteria
AND
IF  ODOBTP      EQ  *MODULE
OR  ODOBTP      EQ  *FILE
OR  ODOBNM      CT  PBA

More...

Bottom

F3=Exit  F4=Fold/Unfold  F5=Refresh  F6=View Statement  F14=Process
    
```

Use any field in the Object Description as a basis for the filter. When completed press F14 to process.

```

EDTEXTF                      Edit External File                      Position . . . .
Object . . . . . MYEXTF      Programs or Srv Pgms using module PBA24A
Library . . . . . MYLIB      Created by SMI SuperVisor5 system

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source      4=Remove entry
5=Display program file 6=Print      7=Clear Source Ref 8=Edit Source Ref

Opt Name      Library      Type      Attribute      Attribute      Level
EVFEVENT      SPOOBJ      *FILE      PF
GENPO         SPOOBJ      *MODULE      RPGLE
MYCLSOURC     SPOOBJ      *MODULE      CLLE
PAA01A        SPOOBJ      *MODULE      RPGLE
PAA01ADEV     SPOOBJ      *MODULE      RPGLE
PAA01AFM      SPOOBJ      *FILE      DSPF
PAA01B        SPOOBJ      *MODULE      RPGLE
PAA02A        SPOOBJ      *MODULE      RPGLE

More...

F3=Exit      F4=Prompt      F6=Add      F5=Refresh
F10=Command entry  F11=Next view  F23=More options  F24=More keys
    
```

F21=Print

A number of reports can be printed based on the objects included in the current select/omit filter.

Getting Started

Depending on your responsibilities, your favorite SuperVisor command may be in Security, system analysis, or in ILE cross references.

Have a particular problem or issue you need to solve. Contact [Software Management, Inc.](#) and explain what you would like to do. We are always happy to point you in the right direction.

SUPERVISOR ASSISTANT IS FREE

SuperVisor Assistant is FREE
SuperVisor External File

SuperVisor Assistant is FREE

SuperVisor marketing distributes SuperVisor with a free 30 day trial period. Use this time to use SuperVisor to interact with system find out what you can learn and what can do with SuperVisor as compared to having to always write you tools from scratch.

Within SuperVisor are core functions that you can continue to use after the trial period is over. We call this section of the program SuperVisor Assistant. Access this section of SuperVisor using option 3 “SuperVisor Assistant” on the SuperVisor Main Menu.

```
MAIN                               SuperVisor5(tm) Main Menu                               System: SMI5
Select one of the following.
SuperVisor Assistant                               Authorized
                                                    *FREE
    27. External File
    28. SuperVisor *LIST
    29. Table Service
    30. Tools and Utilities
SuperVisor Configuration
    33. Edit authorization code
    34. Configure Remote Location
    35. Edit system values
    36. Select commands by name
                                                    Bottom
Selection
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs    F16=Main menu        F18=Work with output
F21=Print menu help
```

Look to the Authorized special value on the right side of the screen to determine if the proper authorization code for Free Access to SuperVisor Assistant has been entered.

If it does not say *FREE, contact Software Management, Inc. at www.smisupervisor.com to request the FREE access code.

As you explore the different Assistant menus, the individual commands are identified. Use these commands as you would anywhere an iSeries command can be executed. On the command line, in Remote Explorer, or from within CL programs you write.

- SuperVisor *LIST enables a user repeat a series of AS/400 commands based upon a list of names.
- SuperVisor External Files enable a user to repeat a series of AS/400 commands based upon an outfile.
- Table Service enables developers and operators to create and maintain table databases without programming.
- Edit SuperVisor System Values is used to customize SuperVisor.

SuperVisor Assistant is FREE SuperVisor External File

- Work with Commands lists SuperVisor commands.

SuperVisor Assistant Menu (ASSISTANT)

```
ASSISTANT                               SuperVisor Assistant                               System: SMI5
Select one of the following.

Programming Assistance

    3. Retrieve Information
    4. Window API tools
    5. SuperVisor Utilities

Operations Assistance

    9. Manage Output Queues

More...

Selection
===>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs    F16=Main menu         F18=Work with output
F21=Print menu help
```

SuperVisor Assistant is a set of utilities that enable users to automate AS/400 operations. The options on this menu direct the user to the sub-menus for these automation tools and to SuperVisor's support options. A brief explanation of each option is provided below.

- Convert Information Format convert information from one format to another.
- Retrieve Information
- Window API tools enable a programmer to create pop-up windows with a minimum of programming.
- SuperVisor Utilities provides some handy utilities for the user's session.
- Manage Output Queues enables a user to manage, archive and restore output queues.
- The Support Menu provides options for obtaining technical support for SuperVisor.

As with all SuperVisor functions the facilities in SuperVisor Assistant are command driven, meaning the user can run these utilities by selecting a menu option, by entering a command, or by including the command in a CL program.

SuperVisor Assistant is FREE
SuperVisor External File

SuperVisor External File Menu (EXTF)

```
EXTF                               SuperVisor External File Menu                               System: SMI

Select one of the following.

Manage External File

    3. Create External File          CRTEXTF
    4. Edit External File            EDTEXTF
    5. Delete External File          DLTEXTF
    6. Copy External File            CPYEXTF
    7. Match Two External Files      MATEXTF
    8. Execute Command using External File EXCCMDEXTF

More...

Selection
====>
F3=Exit      F5=Next Menu- *SELECT  F6=Display Message  F10=Command entry
F12=Cancel  F14=Work with jobs  F16=Main menu      F18=Work with output
F21=Print menu help
```

Managing SuperVisor External Files

A SuperVisor external file is an outfile that contains information about objects. SuperVisor provides commands that enable an operator or programmer to create, maintain and use external files. External files can be used to repeat any AS/400 command for each record in an external file. The Edit External File (EDTEXTF) command enables a user to execute user options such as IBM PDM commands.

```
EXTF                               SuperVisor External File Menu                               System: SMI

Select one of the following.

Manage External File

    3. Create External File          CRTEXTF
    4. Edit External File            EDTEXTF
    5. Delete External File          DLTEXTF
    6. Copy External File            CPYEXTF
    7. Match Two External Files      MATEXTF
    8. Execute Command using External File EXCCMDEXTF

More...

Selection
====> 3
F3=Exit      F5=Next Menu- *SELECT  F6=Display Message  F10=Command entry
F12=Cancel  F14=Work with jobs  F16=Main menu      F18=Work with output
F21=Print menu help
```

Option 3 prompts the SuperVisor Create External File command. As with most SuperVisor menus, all of the commands on the External File Menu prompt SuperVisor commands. These commands can be entered on a command line or included in CL programs.

SuperVisor Assistant is FREE

SuperVisor External File

```
                Create External File (CRTEXTF)

Type choices, press Enter.

Output External File . . . . . MYEXTF      Name, *NONE
  Library name: . . . . . MYLIB          Name
Member option . . . . . *REPLACE        *REPLACE, *ADD
Based on format . . . . . LSTOBJD       LSTOBJD, LSTAUT
Text description . . . . . spo file objects

                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

This command creates an empty external file, MYEXTF, in MYLIB.

Note that there are two types of external files, LSTOBJD and LSTAUT. For most situations the LSTOBJD file format is used. LSTAUT is an external file format for working with object authorities. The LSTOBJD format is chosen for this example.

```
EXTF                SuperVisor External File Menu                System: SMI

Select one of the following.

Manage External File

   3. Create External File                CRTEXTF
   4. Edit External File                  EDTEXTF
   5. Delete External File                DLTEXTF
   6. Copy External File                  CPYEXTF
   7. Match Two External Files            MATEXTF
   8. Execute Command using External File EXCCMDEXTF

                                                    More...

Selection
===> 4
F3=Exit   F5=Next Menu- *SELECT   F6=Display Message   F10=Command entry
F12=Cancel F14=Work with jobs     F16=Main menu       F18=Work with output
F21=Print menu help
```

Option 4 can be used to view and edit the created external file.

SuperVisor Assistant is FREE

SuperVisor External File

```
                                Edit External File (EDTEXTF)

Type choices, press Enter.

External File . . . . . MYEXTF      Name, *NONE
Library name: . . . . . MYLIB       Name
Data refresh . . . . . *NO          *NO, *YES

                                                                Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

The command is prompted. This command actually creates a copy of the external file in QTEMP. So that after editing the external file can be updated or not (like SEU).

```
EDTEXTF          Edit External File

                                Position . . . .
Object . . . . . MYEXTF      spo file objects
Library . . . . . MYLIB

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source  4=Remove entry
5=Display program file 6=Print                14=Compile ...
Object      Object
Opt Name    Library    Type      Attribute  Description

F3=Exit      F4=Prompt      F6=Add      F5=Refresh
F10=Command entry  F11=Next view  F23=More options  F24=More keys
```

This external file is empty, because nothing has been added yet. Most SuperVisor commands can produce or extend external files based upon an analysis. F6 prompts for the SuperVisor List Object Description command, which can be used to add records to the External File.

SuperVisor Assistant is FREE SuperVisor External File

```

List Object Description: (LSTOBJD)

Type choices, press Enter.

Objects--Generic* or *ALL: . . . *ALL      Name, generic*, *ALL, *SELECT
Object Library name: . . . . . SPOOBJ      Name, generic*, *USRLIBL...
Object type: . . . . . *FILE              Character value, *ALL...
Sub-Type (Files/Programs): . . . *PHY      Character value, *ALL...
From create date QDATFMT . . . . 000000   Date
To create date QDATFMT . . . . 999999   Date, 999999
From last changed date QDATFMT . . 000000   Date
To last change date QDATFMT . . . 999999   Date, 999999
From last used date QDATFMT . . . 000000   Date
To last used date QDATFMT . . . 999999   Date, 999999
Check Observability . . . . . *NO          *NO, *YES
RISC conversion required . . . . *ALL      *NO, *YES, *ALL
Include ILE source . . . . . *NO          *NO, *YES
Select Using EXCQRYS command: . . *NO        Name, *YES, *NO, *SELECT
Retain SQU Statement: . . . . . *TEMP      *TEMP, *PERM
Check Source File: . . . . . *NO          *NO, *YES

More...

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys

```

In this case all physical files in SPOOBJ are selected. Notice that the LSTOBJD command has many parameters to pinpoint which objects should be added. The LSTOBJD command can also be run from a command line or executed from a CL program.

```

EDTEXTF          Edit External File

                                Position . . . .
Object . . . . . MYEXTF      spo file objects
Library . . . . . MYLIB

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source  4=Remove entry
5=Display program file 6=Print              14=Compile ...

Object      Object
Opt  Name    Library   Type      Attribute   Description
---  ---    ---      ---      ---        ---
PAA16EW1    SPOOBJ    *FILE     PF          Totals Work File for bu
PAA23CWF    SPOOBJ    *FILE     PF          PO Summary Report Work
PDSCP00     SPOOBJ    *FILE     PF          Purchase Order Item Des
PDTLP00     SPOOBJ    *FILE     PF          Purchase Order Detail F
PFLDR00     SPOOBJ    *FILE     PF          Purchase Order System F
PHDRP00     SPOOBJ    *FILE     PF          Purchase Order Header F
PNMEP00     SPOOBJ    *FILE     PF          Purchase Order Name and

Bottom

F3=Exit      F4=Prompt      F6=Add      F5=Refresh
F10=Command entry  F11=Next view  F23=More options  F24=More keys

```

Since LSTOBJD was run from this display, SuperVisor returns to the edit display. Now the external file contains a record for each physical file in the SPOOBJ library.

SuperVisor Assistant is FREE SuperVisor External File

```

EDTEXTF          Edit External File
                                     Position . . . . .
Object . . . . . MYEXTF          spo file objects
Library . . . . . MYLIB

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source      4=Remove entry
5=Display program file 6=Print              14=Compile ...

Opt  Object      Object      Source      Source      Source      Source Last
     Name        Library     File        Library     Member     Chg Date
-----
     PAA16EW1    SPOOBJ     QDDSSRC    SPOSRC     PAA16EW1   91/12/06
     PAA23CWF    SPOOBJ     QDDSSRC    SPOSRC     PAA23CWF   91/12/06
     PDSCP00     SPOOBJ     QDDSSRC    SPOSRC     PDSCP00    91/12/06
     PDTLP00     SPOOBJ     QDDSSRC    SPOSRC     PDTLP00    91/12/06
     PFLDR00     SPOOBJ     QDDSSRC    SPOSRC     PFLDR00    91/12/06
     1  PHDRP00   SPOOBJ     QDDSSRC    SPOSRC     PHDRP00    91/12/06
     PNMEP00     SPOOBJ     QDDSSRC    SPOSRC     PNMEP00    91/12/06

                                     Bottom

F3=Exit          F4=Prompt        F6=Add          F5=Refresh
F10=Command entry F11=Next view    F23=More options F24=More keys

```

External file records contain a lot of information about the objects they represent. Pressing F11 cycles through various views of this object information.

Notice the Options listed in the center panel of the display. F23 can be used to cycle through the options included with SuperVisor. The options execute commands and SuperVisor passes object information from the external file record to the command. For example Option 1 executes the Work Object Using PDM command.

```

                                     Work with Objects Using PDM          SMI
Library . . . . . SPOOBJ          Position to . . . . .
                                     Position to type . . . . .

Type options, press Enter.
2=Change          3=Copy          4=Delete        5=Display        7=Rename
8=Display description 9=Save          10=Restore      11=Move ...

Opt  Object      Type      Attribute      Text
-----
     PHDRP00     *FILE     PF-DTA        Purchase Order Header File

                                     Bottom

Parameters or command
====>
F3=Exit          F4=Prompt        F5=Refresh      F6=Create
F9=Retrieve      F10=Command entry F23=More options F24=More keys
This is a subsetted list.                                     +

```

In this case an IBM command is executed and SuperVisor passed the parameters for selecting the library and object.

SuperVisor Assistant is FREE SuperVisor External File

```

EDTEXTF          Edit External File
                                     Position . . . . .
Object . . . . . MYEXTF          spo file objects
Library . . . . . MYLIB

Type options, press Enter.
1=Work object PDM      2=Work member PDM      3=Copy source      4=Remove entry
5=Display program file 6=Print              14=Compile ...

  Object   Object   Source   Source   Source   Source Last
Opt  Name   Library  File     Library  Member   Chg Date
PAA16EW1  SPOOBJ  QDDSSRC SPOSRC  PAA16EW1 91/12/06
PAA23CWF  SPOOBJ  QDDSSRC SPOSRC  PAA23CWF 91/12/06
PDSCP00   SPOOBJ  QDDSSRC SPOSRC  PDSCP00   91/12/06
PDTLP00   SPOOBJ  QDDSSRC SPOSRC  PDTLP00   91/12/06
PFLDR00   SPOOBJ  QDDSSRC SPOSRC  PFLDR00   91/12/06
PHDRP00   SPOOBJ  QDDSSRC SPOSRC  PHDRP00   91/12/06
PNMEP00   SPOOBJ  QDDSSRC SPOSRC  PNMEP00   91/12/06

                                     Bottom

F3=Exit          F4=Prompt        F6=Add           F5=Refresh
F10=Command entry F11=Next view    F23=More options F24=More keys
  
```

After executing the option SuperVisor returns to the edit display.

F3 exits the edit program and displays the exit options pop up. Option 1 saves any changes made to the external file during the session. Option 2 leaves the external file as it was before the session began.

```

EXTF          SuperVisor External File Menu
                                     System: SMI

Select one of the following.

Manage External File

  3. Create External File           CRTEXTF
  4. Edit External File             EDTEXTF
  5. Delete External File           DLTEXTF
  6. Copy External File             CPYEXTF
  7. Match Two External Files       MATEXTF
  8. Execute Command using External File EXCCMDEXTF

                                     More...

Selection
====> 8
F3=Exit      F5=Next Menu- *SELECT    F6=Display Message  F10=Command entry
F12=Cancel   F14=Work with jobs      F16=Main menu       F18=Work with output
F21=Print menu help
File QTEMP/EOBJIWF Member EOBJIWF contains 7 records

                                     +
  
```

Option 8 from the menu enables an operator or programmer to repeat a command for every record in the external file.

SuperVisor Assistant is FREE SuperVisor External File

```

Execute User Option (EXCCMDEXTF)

Type choices, press Enter.

External (LSTOBJD) file . . . . MYEXTF      Name
  Library name . . . . . MYLIB      Name, *LIBL
Command String (256) . . . . . CPYF

Prompt Command . . . . . *NO          *YES, *NO
Print Log Report (*YES, *NO): . *YES      *NO, *YES

Bottom

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

A prompt for the EXCCMDEXTF is presented. This command can also be entered from a command line or included in a CL program, in which case it would not be prompted.

In this example the copy file command is to be executed for each object represented by records in MYEXTF in MYLIB.

F4 can be used to prompt the command to be repeated.

```

Copy File (CPYF)                                Level: 2

Type choices, press Enter.

From file . . . . . UNAME      Name
  Library . . . . . ULIB      Name, *LIBL, *CURLIB
To file . . . . . UNAME      Name, *PRINT
  Library . . . . . MYLIB     Name, *LIBL, *CURLIB
From member . . . . . *FIRST  Name, generic*, *FIRST, *ALL
To member or label . . . . . *FIRST  Name, *FIRST, *FROMMBR
Replace or add records . . . . . *NONE *NONE, *ADD, *REPLACE...
Create file . . . . . *NO      *NO, *YES
Print format . . . . . *CHAR   *CHAR, *HEX

Bottom

F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display   F24=More keys
```

SuperVisor prompts the command. To repeat the command SuperVisor has to substitute data from the external file to command parameters. In this example the “from file” name/library and the “to file” name. The EXCCMDEXTF command uses the following values to substitute command parameters.

SuperVisor Assistant is FREE SuperVisor External File

Value	Description
UNAME	Object name
ULIB	Object library
UTYPE	Object type
UATTR	Object attribute
USRCF	Object Source File Reference
USRCL	Object Source Library Reference
USRCM	Object Source Member Reference
UTEXT	Object Text

In the above command UNAME and ULIB are used. In this example the external file only contained records representing objects in the SPOOBJ library. The copy file command will copy these objects to MYLIB, however, the external file could have contained records from several libraries.

```
Execute User Option (EXCCMDEXTF)

Type choices, press Enter.

External (LSTOBJD) file . . . . > MYEXTF      Name
Library name . . . . . > MYLIB      Name, *LIBL
Command String (256) . . . . . > CPYF FROMFILE (ULIB/UNAME) TOFILE (MYLIB/UNAME
)

Prompt Command . . . . . *NO          *YES, *NO
Print Log Report (*YES, *NO): . *YES    *NO, *YES

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

After entering the prompted command parameters, SuperVisor returns to the EXCCMDEXTF command. Pressing Enter copies the files involved.

SuperVisor *LIST (LIST)

```
LIST                               SuperVisor *LIST                               System: SMI

Select one of the following.

Manage SuperVisor *LIST

    3. Create a *LIST                ADDLIST
    4. Edit a *LIST                  EDTLIST
    5. Remove *LIST                  RMVLIST
    6. Merge Two *LISTs             MRGLIST
    7. Match Two *LISTs             MATLIST
    8. Copy *LIST                   CPYLIST
    9. Identify *LIST               IDLIST
   10. Audit *LIST                  AUDLIST
   11. Work with commands using *LIST  WRKCMDLST
   12. Execute Command using *LIST    EXCCMDLST

Selection
====> 3
F3=Exit      F5=Next Menu- *SELECT  F6=Display Message  F10=Command entry
F12=Cancel  F14=Work with jobs      F16=Main menu       F18=Work with output
F21=Print menu help
```

Managing SuperVisor *LIST

A SuperVisor *LIST is a list of names. *LIST's are used in SuperVisor commands to filter items processed or to repeat an operation for each item on a *LIST. Many SuperVisor analyses also optionally create or add entries to *LIST's.

- Create a *LIST
- Add *LIST Entries
- Remove *LIST Entries
- Remove a *LIST

*LIST's can also be used to automate user-defined processes. For example, SuperVisor provides commands that will execute any valid AS/400 command for each item on a *LIST.

SuperVisor Assistant is FREE

SuperVisor *LIST

```
LIST                               SuperVisor *LIST                               System: SMI

Select one of the following.

Manage SuperVisor *LIST

    3. Create a *LIST                ADDLIST
    4. Edit a *LIST                  EDTLIST
    5. Remove *LIST                  RMVLIST
    6. Merge Two *LISTs              MRGLIST
    7. Match Two *LISTs              MATLIST
    8. Copy *LIST                    CPYLIST
    9. Identify *LIST                IDLIST
   10. Audit *LIST                   AUDLIST
   11. Work with commands using *LIST WRKCMDLST
   12. Execute Command using *LIST   EXCCMDLST

More...

Selection
====> 3
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs       F16=Main menu         F18=Work with output
F21=Print menu help
```

Create a *LIST

To create a select Option 3 from the menu to execute the ADDLIST command.

```
                                Add SuperVISOR *LIST: (ADDLIST)

Type choices, press Enter.

List Name: . . . . . mylist      Name
List Name Description: . . . . . programs I have checked

                                                                Bottom
F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

Like many SuperVisor menu options, this option prompts a SuperVisor command. This means the command may also be entered from a command line or used in a CL program. The command prompt enables the user to name the *LIST and enter a text description.

SuperVisor Assistant is FREE

SuperVisor *LIST

```
LIST                               SuperVisor *LIST                               System: SMI

Select one of the following.

Manage SuperVisor *LIST

    3. Create a *LIST                ADDLIST
    4. Edit a *LIST                  EDTLIST
    5. Remove *LIST                  RMVLIST
    6. Merge Two *LISTs              MRGLIST
    7. Match Two *LISTs              MATLIST
    8. Copy *LIST                    CPYLIST
    9. Identify *LIST                IDLIST
   10. Audit *LIST                   AUDLIST
   11. Work with commands using *LIST WRKCMDLST
   12. Execute Command using *LIST   EXCCMDLST

More...

Selection
====> 4
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs        F16=Main menu         F18=Work with output
F21=Print menu help
```

Add entries to a SuperVisor *LIST

Option 4 is selected to enter some items on the *LIST.

```
                                Edit SuperVISOR *LIST: (EDTLIST)

Type choices, press Enter.

List Name: . . . . . *SELECT      Name, *SELECT
List Name Description: . . . . . *SAME

                                Bottom
F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

The command prompt is presented. The list name could be entered here. However, *SELECT is entered. *SELECT is a common key word in SuperVisor which displays a list of the items used in the command. In this case *SELECT will display *LIST's that have been created.

SuperVisor Assistant is FREE SuperVisor *LIST

```
UMBRA                MANAGEMENT - (L) Edit SuperVISOR *LIST                7/07/00

Select one, press enter
 1=Select
Opt Entry            Description
 1 MYLIST            programs I have checked

F3=Exit                                                    Bottom
```

Only 1 has been created. It is selected. Press enter continues to the Edit SuperVisor *LIST screen.

```
EDTLIST             MANAGEMENT - (L) Edit SuperVISOR *LIST                7/07/00
MYLIST              programs I have checked
Type changes, press enter
 4=Delete  8=Generic* 9=*Generic*      Position to . . .
Opt Entry   Opt Entry   Opt Entry   Opt Entry
*ADD*
PAA02A
PAA03A
PAA05B
PAA06A
PAA08A
PCA01A

F3=Exit  F5=Refresh  F13=Repeat  F14=Continue  F17=Top  F18=Bottom
                                                    Bottom
```

Use the *ADD* line to place new entries onto the list. SuperVisor *LIST entries are just 10 character names without attributes.

Entries on a *LIST are unique. It is not possible to have duplicate *LIST entries. This feature of a SuperVisor *LIST can prove useful in many situations.

SuperVisor Assistant is FREE

SuperVisor *LIST

```
LIST                               SuperVisor *LIST                               System: SMI

Select one of the following.

Manage SuperVisor *LIST

    3. Create a *LIST                ADDLIST
    4. Edit a *LIST                  EDTLIST
    5. Remove *LIST                  RMVLIST
    6. Merge Two *LISTs             MRGLIST
    7. Match Two *LISTs             MATLIST
    8. Copy *LIST                   CPYLIST
    9. Identify *LIST               IDLIST
   10. Audit *LIST                  AUDLIST
   11. Work with commands using *LIST WRKCMDLST
   12. Execute Command using *LIST  EXCCMDLST

More...

Selection
====> 4
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs       F16=Main menu         F18=Work with output
F21=Print menu help
```

The EDTLIST command will also create a *LIST if it is not present.

```
                                Edit SuperVISOR *LIST: (EDTLIST)

Type choices, press Enter.

List Name: . . . . . fredlist      Name, *SELECT
List Name Description: . . . . . programs Fred has checked

Bottom

F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

A new list name and description are entered at the command prompt.

SuperVisor Assistant is FREE SuperVisor *LIST

```
                Edit SuperVISOR *LIST: (EDTLIST)

Type choices, press Enter.

List Name: . . . . . fredlist      Name, *SELECT
L ..... ckd
: SuperVisor *LIST FREDLIST not found      :
: Select one of the following, press enter. :
: 1 1. Create SuperVisor *LIST             :
: 2. Select existing SuperVisor *LIST      :
: 3. Cancel                                 :
: F12=Cancel                               :
:                                           :
:.....:

                                                    Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

SuperVisor presents a pop up display, enabling the user to optionally create the *LIST.

```
EDTLIST      MANAGEMENT - (L) Edit SuperVISOR *LIST      7/07/00
FREDLIST     programs Fred has checked
Type changes, press enter      Position to . . .
4=Delete 8=Generic* 9=*Generic*
Opt Entry      Opt Entry      Opt Entry      Opt Entry
*ADD*
PAA013C
PAA014C
PAA015C
PAA016C

                                                    Bottom
F3=Exit  F5=Refresh  F13=Repeat  F14=Continue  F17=Top  F18=Bottom
```

Items are entered on this *LIST.

SuperVisor Assistant is FREE SuperVisor *LIST

```
LIST                               SuperVisor *LIST                               System: SMI

Select one of the following.

Manage SuperVisor *LIST

    3. Create a *LIST                ADDLIST
    4. Edit a *LIST                  EDTLIST
    5. Remove *LIST                  RMVLIST
    6. Merge Two *LISTs              MRGLIST
    7. Match Two *LISTs             MATLIST
    8. Copy *LIST                    CPYLIST
    9. Identify *LIST                IDLIST
   10. Audit *LIST                  AUDLIST
   11. Work with commands using *LIST WRKCMDLST
   12. Execute Command using *LIST  EXCCMDLST

More...

Selection
====> 6
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs      F16=Main menu         F18=Work with output
F21=Print menu help
```

SuperVisor provides commands such as match, merge and copy *LIST's. In this example the *LIST's will be merged.

```
                                Merge Two SuperVISOR *LISTS (MRGLIST)

Type choices, press Enter.

LIST1 is to be merged: . . . . . fredlist      Name
with *LIST2: . . . . . mylist                 Name
Remove *LIST1 After Merge: . . . *NO          *YES, *NO

Bottom

F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

The command prompt is presented. List 1 is added to List 2.

SuperVisor Assistant is FREE SuperVisor *LIST

```
EDTLIST      MANAGEMENT - (L) Edit SuperVISOR *LIST          7/07/00
MYLIST      programs I have checked
Type changes, press enter          Position to . . .
4=Delete 8=Generic* 9=*Generic*
Opt Entry      Opt Entry      Opt Entry      Opt Entry
*ADD*
PAA013C
PAA014C
PAA015C
PAA016C
PAA02A
PAA03A
PAA05B
PAA06A
PAA08A
PCA01A

Bottom
F3=Exit  F5=Refresh  F13=Repeat  F14=Continue  F17=Top  F18=Bottom
```

MYLIST now contains both sets of names. Duplicate entries are removed, because a SuperVisor *LIST cannot have duplicates.

Work with commands using SuperVisor *LIST

```
LIST          SuperVisor *LIST          System: SMI
Select one of the following.
Manage SuperVisor *LIST
3. Create a *LIST          ADDLIST
4. Edit a *LIST           EDTLIST
5. Remove *LIST           RMVLIST
6. Merge Two *LISTs      MRGLIST
7. Match Two *LISTs      MATLIST
8. Copy *LIST            CPYLIST
9. Identify *LIST        IDLIST
10. Audit *LIST          AUDLIST
11. Work with commands using *LIST  WRKCMDLST
12. Execute Command using *LIST     EXCCMDLST
More...
Selection
===> 11
F3=Exit  F5=Next Menu- *SELECT  F6=Display Message  F10=Command entry
F12=Cancel F14=Work with jobs  F16=Main menu      F18=Work with output
F21=Print menu help
```

A *LIST can be used to repeat an entry for each item in the *LIST. Option 11 is used to enter a command that will be repeated for each item on the *LIST.

SuperVisor Assistant is FREE SuperVisor *LIST

```

Work with Command using *LIST (WRKCMDLST)
Type choices, press Enter.
List Name: . . . . . *SELECT      Name, *SELECT
List Name Description: . . . . . *SAME
Remove List After Processing: . *NO          *YES, *NO
Print Log Report (*YES, *NO): . *YES        *NO, *YES
Command to Execute . . . . .

Printer Output Queue: . . . . .      Name
Library name: . . . . . *LIBL        Name, *LIBL
Job Desc. for Submit Job: . . . *USRPRF   Name, *USRPRF
Library name: . . . . . *LIBL        Name, *LIBL

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

```

The *SELECT default is used to display a list of existing SuperVisor *LIST's. It is not necessary to remember the name of every *LIST you create.

The Command line, is used to provide default values to with "Work with", is optional. The same is true regarding the job description. When using the "Work with" screen and use submit processing to batch, this is the job description that process will use.

```

UMBRA      MANAGEMENT - (L) Work with Commands Using *LIST      7/19/00
Select one, press enter
1=Select
Opt Entry      Description
ALLRPG        all programs created from 010199 to 060100
FREDLIST      programs Fred has checked
1 PAORD#      programs I have checked

Bottom
F3=Exit

```

Since *SELECT (the default) was entered in the original command. SuperVisor asks for the name of the *LIST to be used. The *LIST's displayed depends on what exists at the time. Select SuperVisor *LIST PAORD#

SuperVisor Assistant is FREE SuperVisor *LIST

```

WRKCMDLST          Work with commands using list
PAORD#            MEMBERS USING PAORD#
Type changes, press enter

Command:

2=Execute, 4=Delete, 6=Submit          Position to . . .
Opt List Entry  Opt List Entry  Opt List Entry  Opt List Entry

$INZPA          PAA05D
@SFDS           PAA06A
GENPO           PAA14C
PAA01B          PAA15C
PAA02A          PAA16C
PAA03A          PAA23C
PAA05C          PBA01A

                                                    Bottom
F3=Exit        F4=Prompt Command    F5=Refresh     F9=Retrieve    F10=Command Entry
F13=Repeat     F14=WRKSEMJOB      F16=Prompt Execute  F17=Top
F18=Bottom     F20=Change Defaults F22=Halt on error

```

The screen uses the top portion to input the commands you wish to execute. The lower portion includes the SuperVisor *LIST entries. Options and command keys provide different functions.

The powerful thing about a SuperVisor *LIST entry is that it has no attributes. It is a simple list of names. In this example, these entries were added to the list as output from the Scan Source. Each was a source member in the file SPOSRC/QRPGLESRC that contains the search strings.

Once on a *LIST, we can apply them in different ways supplying different types of attributes.

```

WRKCMDLST          Work with commands using list
PAORD#            MEMBERS USING PAORD#
Type changes, press enter

Command: CHKOBJ OBJ(SPODEV/QRPGLESRC) OBJTYPE(*FILE) MBR(UNAME)

2=Execute, 4=Delete, 6=Submit          Position to . . .
Opt List Entry  Opt List Entry  Opt List Entry  Opt List Entry

2 $INZPA          2 PAA05D
2 @SFDS           2 PAA06A
2 GENPO           2 PAA14C
2 PAA01B          2 PAA15C
2 PAA02A          2 PAA16C
2 PAA03A          2 PAA23C
2 PAA05C          2 PBA01A

                                                    Bottom
F3=Exit        F4=Prompt Command    F5=Refresh     F9=Retrieve    F10=Command Entry
F13=Repeat     F14=WRKSEMJOB      F16=Prompt Execute  F17=Top
F18=Bottom     F20=Change Defaults F22=Halt on error

```

Execute the CHKOBJ command to see if these same members are found in the file SPODEV/QRPGLESRC.

SuperVisor Assistant is FREE
SuperVisor *LIST

```
WRKCMDLST          Work with commands using list
PAORD#            MEMBERS USING PAORD#
Type changes, press enter

Command: CHKOBJ OBJ(SPOOBJ/UNAME) OBJTYPE(*MODULE)

2=Execute, 4=Delete, 6=Submit          Position to . . .
Opt List Entry  Opt List Entry  Opt List Entry  Opt List Entry

$INZPA          2 PAA05D
@SFDS           2 PAA06A
GENPO           2 PAA14C
2 PAA01B        2 PAA15C
2 PAA02A        2 PAA16C
2 PAA03A        2 PAA23C
2 PAA05C        2 PBA01A

Bottom
F3=Exit      F4=Prompt Command  F5=Refresh  F9=Retrieve F10=Command Entry
F13=Repeat   F14=WRKSEMJOB      F16=Prompt Execute  F17=Top
F18=Bottom   F20=Change Defaults F22=Halt on error
```

Executing this command finds out if each of these source members exist as a module in library SPOOBJ.

```
WRKCMDLST          Work with commands using list
PAORD#            MEMBERS USING PAORD#
Type changes, press enter

Command: SAVOBJ OBJ(ULIST) LIB(SPOBJ) DEV(*SAVF) SAVF(MYLIB/MYSAVF)

2=Execute, 4=Delete, 6=Submit          Position to . . .
Opt List Entry  Opt List Entry  Opt List Entry  Opt List Entry

$INZPA          2 PAA05D
@SFDS           2 PAA06A
GENPO           2 PAA14C
2 PAA01B        2 PAA15C
2 PAA02A        2 PAA16C
2 PAA03A        2 PAA23C
2 PAA05C        2 PBA01A

Bottom
F3=Exit      F4=Prompt Command  F5=Refresh  F9=Retrieve F10=Command Entry
F13=Repeat   F14=WRKSEMJOB      F16=Prompt Execute  F17=Top
F18=Bottom   F20=Change Defaults F22=Halt on error
```

Executing this command will save the objects in the library SPOOBJ, that have the same name. Both modules and programs.

SuperVisor Assistant is FREE
SuperVisor *LIST

Execute commands using a SuperVisor *LIST

```
LIST                               SuperVisor *LIST                               System: SMI

Select one of the following.

Manage SuperVisor *LIST

    3. Create a *LIST                ADDLIST
    4. Edit a *LIST                  EDTLIST
    5. Remove *LIST                  RMVLIST
    6. Merge Two *LISTs             MRGLIST
    7. Match Two *LISTs             MATLIST
    8. Copy *LIST                   CPYLIST
    9. Identify *LIST               IDLIST
   10. Audit *LIST                 AUDLIST
   11. Work with commands using *LIST WRKCMDLST
   12. Execute Command using *LIST  EXCCMDLST

Selection
====> 11
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs       F16=Main menu         F18=Work with output
F21=Print menu help
```

Option 12 “Executing commands using SuperVisor *LIST”, is similar to the “Work with”, but there is no interaction.

```
Execute CMD Driven by *LIST (EXCCMDLST)

Type choices, press Enter.

Command to Execute . . . . . > CHKOBJ OBJ(SPOOBJ/UNAME) OBJTYPE(*PGM)

SuperVisor *LIST name . . . . . paord#           Name, *SELECT
List Name Description: . . . . . *BYPASS

Remove List After Processing: . *NO           *YES, *NO
Print Log Report (*YES, *NO): . *YES         *NO, *YES
Printer Output Queue: . . . . . Name
Library name: . . . . . *LIBL           Name, *LIBL

Bottom
F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

This command executes the CHKOBJ command one time for each list entry on SuperVisor *LIST PAORD#. The list entry is substituted for the literal “UNAME”.

SuperVisor Assistant is FREE SuperVisor *LIST

Executing the command again, replaces all list entries for the literal "ULIST" and executes the command one time.

```
Execute CMD Driven by *LIST (EXCCMDLST)

Type choices, press Enter.

Command to Execute . . . . . > SAVOBJ OBJ(ULIB) LIB(SPOOBJ) DEV(*SAVF) OBJT
YPE(*ALL) SAVF(MYLIB/MYSAVF)

SuperVisor *LIST name . . . . . > PAORD#           Name, *SELECT
List Name Description: . . . . . *BYPASS

Remove List After Processing: . *NO             *YES, *NO
Print Log Report (*YES, *NO): . *YES           *NO, *YES
Printer Output Queue: . . . . . Name
Library name: . . . . . *LIBL           Name, *LIBL

Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

*Note: The special value *BYPASS instructs the command to execute the command, without operator intervention. Remove this default and the SuperVisor *LIST will display before executing the command, providing the opportunity to view the list entries prior to executing the command.*

SuperVisor Assistant is FREE
Retrieve Information

Retrieve Information

```
RTVMENU                      Retrieve Information                      System: SMI

Select one of the following.

Retrieve Command Language (CL) Source

    3. Retrieve Job Schedule Entry Source      RTVSCDESRC
    4. Retrieve Menu Source                    RTVMNUSRC
    5. Retrieve Command File Source           RTVCMDFSRC
    6. Retrieve Binding Directory             RTVBNDDIR
    7. Retrieve File Trigger Source           RTVTRGSRC
    8. Retrieve File Journal Source           RTVJRNSRC
    9. Retrieve Directory Entry Source        RTVDIRSRC
   10. Retrieve Subsystem Description Source  RTVSBDSRC
   11. Retrieve System Information Source     RTVSYISRC

Execute in CL programs only

   15. Retrieve Physical File Attributes      RTVPFA
   16. Retrieve Logical File Attributes       RTVLFA
   17. Retrieve File Trigger Attributes       RTVFTRGA
   18. Retrieve Record Format Attributes      RTVRFMTA
   19. Retrieve File Journal Attributes       RTVFJRNA
   20. Retrieve Display File Attributes       RTVDSPPFA
   21. Retrieve Printer File Attributes       RTVPRTFA
   22. Retrieve Printer Attributes 2         RTVPRTFA2
   23. Retrieve Field Attributes             RTVFLDA

   27. Retrieve Program Attributes           RTVPGMA
   28. Retrieve Module Attributes            RTVMODA
   29. Retrieve Service Program Attributes    RTVSRVA
   30. Retrieve Program SQL Attributes        RTVSQLA

   32. Retrieve ILE Program Activation Group  RTVPACTG

   39. Retrieve Command Attributes           RTVCMDA
   40. Retrieve Library Size                 RTVLIBSIZ
   41. Retrieve Date Description             RTNDD
   42. Retrieve Number Edited                RTVNUMEDT
   43. Retrieve Job Description              RTVJOB
   44. Return Query Statement String (5000)  RTNQYS5
   45. Retrieve Active (signed on) user     RTVACTU
   46. Retrieve Object Lock                  RTVOBJLCK
   47. Retrieve Spool File Number            RTVSPLNBR

   52. Retrieve User Space Attributes        RTVUSRSA
   53. Retrieve Data Queue Attributes        RTVDTAQA
   54. Retrieve Data Area Attributes         RTVDTAAA

   58. Return Member Information             RTNMBRI
   59. Return Object Information             RTNOBJI

Selection                                                                Bottom
====>
F3=Exit      F5=Next Menu- *SELECT      F6=Display Message  F10=Command entry
F12=Cancel   F14=Work with jobs             F16=Main menu       F18=Work with output
F21=Print menu help
```

Use the commands provided here to easily access iSeries information, which is usually, requires a great deal of work to obtain.

Note: These commands can be executed only in CL programs. Each command returns information in CL fields that can then use the information being provided.

The first group of commands is used to process specific types of objects and create source in a CL program that can then be used to better manage those objects. The second group is commands that can only be used

SuperVisor Assistant is FREE Retrieve Information

in CL programs. Using these commands will greatly simplify the programming effort usually needed to obtain and use this information.

SuperVisor Assistant is FREE
Window Menu

Window Menu (WINDOW)

```
WINDOW                               Window Menu                               System: SMI

Select one of the following.

    1. Window Display API                WINDSP
    2. Window Option Menu API            WINOPT
    3. Window Selection API              WINRTN
    4. Status Message Window            WINSTS

    6. Dynamic Window (Access by Key)    DYNWINKEY
    7. Dynamic Window (Access by RRN)    DYNWINRRN

    9. Window Location System Value      WINCTL
   10. Window Frame Default System Value CHGFRMDFT

   12. Work with Window Commands        SUPERCMD

Selection
====>
F3=Exit      F5=Next Menu- *SELECT    F6=Display Message  F10=Command entry
F12=Cancel   F14=Work with jobs             F16=Main menu       F18=Work with output
F21=Print menu help
```

Working with Window API's

The SuperVisor window API consists of commands that display pop-up windows. These commands can be used in user CL and HHL programs. The commands activate the pop-up window and wait for a response. The contents of the pop-up window are passed to the command using parameters. The parameters can contain sting data or a file name. When a file name is used, fields from the specified file are displayed.

The WINCTL and CHGFRMDFT set SuperVisor's values for window location and window frame characters. The WINDSP, WINOPT, WINRTN, DYNWINKEY, DYNWINRRN commands present various types of pop-up windows.

Three window sizes are supported, small, medium and large. The SIZE parameter specifies which size to use. The default *DFT, will automatically choose the size based on the longest the line being displayed. Otherwise, the size to be displayed in accordance with the following:

- *SMALL - for up to 24 characters
- *MEDIUM - for up to 40 characters
- *LARGE - for up to 66 characters

The location of windows can be specified in terms of the horizontal position (left or right) and vertical position (top or bottom). The Window Location System Value (WINCTL) command specifies the screen positions that will be used when these parameters are entered. There are three sets of coordinates, those used by small, medium and large windows.

There are 5 types of window displays:

1. Window Display (WINDSP) - displays a window.
2. Window Option (WINOPT) - displays a window and allows the user to enter a 1-character option field. The selected option is returned.
3. Window Selection (WINRTN) - displays a window and allows the user to enter up to 6 1-character selection fields. The selected fields are returned.
4. Dynamic Window by Key (DYNWINKEY) - displays a record. The record is accessed by key.
5. Dynamic Window by RRN (DYNWINRRN) - displays a record. The record is accessed by relative record number.

SuperVisor Assistant is FREE
Table Service

Table Service

Store and organize important system values and decision tables.

SuperVisor Assistant is FREE

Table Service

TABLE	Table Service Commands	System: SMI
Select one of the following.		
1. Edit Table Service	EDTCDEFLE	
2. Print Table Service	PRTCDEFLE	
3. Display Table Service	INQDEFLE	
5. Duplicate Table Service Data Base	DUPCDEFLE	
6. Move Table Service Data Base	MOVCDEFLE	
7. Organize Table Service Data Base	RGZCDEFLE	
9. Print Table Service Data Base	PRTTBLDB	
10. Print Table Service Manual	PRTXTFLE	
12. Work with Table Service Commands	SUPERCMD	
Manage Table Service Entries in CL Programs		
15. Return System Entry	RTNSYSE	
16. Return Category Entry	RTNCATE	
17. Return Table Entry	RTNTBLE	
18. Return Code Entry	RTNCDEE	
20. Write System Entry	WRTSYSE	
21. Write Category Entry	WRTCATE	
22. Write Table Entry	WRTBLE	
23. Write Code Entry	WRTCDEE	
27. Copy System Entry	CPYSYSE	
28. Copy Category Entry	CPYCATE	
29. Copy Table Entry	CPYTBLE	
30. Copy Code Entry	CPYCDEE	
32. Delete System Entry	DLTSYSE	
33. Delete Category Entry	DLTCATE	
34. Delete Table Entry	DLTBLE	
35. Delete Code Entry	DLTCDEE	
39. Retrive System Entry	RTVSYSE	
40. Retrieve Category Entry	RTVCATE	
41. Retrieve Table Entry	RTVTBLE	
42. Retrieve Code Entry	RTVCDEE	
44. Retrieve System Entry from Window	RTVSYSWIN	
45. Retrieve Category Entry from Window	RTVCATWIN	
46. Retrieve Table Entry from Window	RTVTBLWIN	
47. Retrieve Code Entry from Window	RTVCDEWIN	
51. Input Next System Entry	INPNXTSYS	
52. Input Next Category Entry	INPNXTCAT	
53. Input Next Table Entry	INPNXTTBL	
54. Input Next Code Entry	INPNXTCDE	
56. Return Query Statement from System	RTNQRYSYS	
57. Return Query Statement from Category	RTNQRYCAT	
58. Return Query Statement from Table	RTNQRYTBL	
59. Return Query Statement from Code	RTNQRYCDE	
63. Display Systems in Window	DSPSYSWIN	
64. Display Categories in Window	DSPCATWIN	
65. Display Tables in Window	DSPTBLWIN	
66. Display Codes in Window	DSPCDEWIN	
68. Edit Code Comments	EDTCDECMT	
69. Display Code Comments	DSPCDECMT	
70. Ask Using Code Comments	ASKCDECMT	
71. Execute Menu using Codes	EXCCDEMNU	
72. Input Next Code Comment	INPNXTCDC	
73. Get Code Comment	GETCDEC	

SuperVisor Assistant is FREE Table Service

Table Service is an application that enables programmers and operators to create maintain and use a database without programming. Built in functions enable operators and programmers to perform common tasks, such as display pop-up windows or print reports using SuperVisor commands, rather than having to write custom code. User written programs can also access the Table Service database in calculations and call SuperVisor commands.

Relational Data Base 3rd form normalization

Any data base design using 3rd normalization can take advantage of these tables to provide descriptions normally removed from files conforming to this data base standard.

This tool is provided distributed as part of SuperVisor at no charge. Feel free to integrate the Table Service capabilities within your own applications, the same way Table Service is integrated into SuperVisor itself.

What is Table Service for

The first reason to use Table Service is for your system values and processing parameters. Data that is usually left out of the system design that becomes necessary as the development of the system takes place. On the iSeries, Table Service is where you would put the WEB cookies. Decision tables used to control program processing. All the different information now being stored in data area, user spaces and undocumented files.

Database Design

The table service is designed as a look up table. Key information is used to return related information and controls.

Detail Record or Row

The Table Service database is ideal for applications requiring tables or arrays. Each record contains the following data elements:

Description	Length	Attributes
Full Description:	24	-character description field.
Short Description:	11	character description field.
Relational Code:	10	character field often used to access another entry (like an alternating table).

Extended Factors

In addition each record contains four sets of numeric fields. These fields are referred to as “extended factors” in Table Service documentation and help text. These are the Decimal Factors, Quantity Factors, Count Factors, and Dollar Amount Factors. The idea behind these factors is that the table service database can be used to store numeric data in the most commonly used numeric formats.

Each factor has three associated data elements, description, factor 1, factor 2, and limit. These are outlined below. The factor one and factor two fields are often used to represent a range. The limit field is often used to apply a data entry rule. Formats of the numeric fields is shown below:

Field Group	Group Description
-------------	-------------------

SuperVisor Assistant is FREE Table Service

Field Group	Group Description
Decimal Factors:	A group of fields often used as percentages. There is a 24-character description field, and three 5-digit numeric fields, each with 5 positions to the right of the decimal.
Quantity Factors:	A group of fields often used to represent quantities. There is a 24-character description field, and three 7-digit numeric fields.
Count Factors:	A group of fields often used to represent counts. There is a 24-character description field, and three 5-digit numeric fields.
Dollar Amount Factors:	A group of fields often used to represent dollars. There is a 24-character description field, and three 9-digit, 2-decimal, numeric fields.

Title Fields

Title fields are used as columnar headings and prompts on displays and reports. Since SuperVisor Table Service is a generic solution, field titles are soft coded and can be modified by the maintenance programs. As will be demonstrated later in this section, title fields are also used to activate or deactivate field usage in one level of the hierarchy.

Comment Fields

Each record in the database can also have associated comment records. Comment records can also be entered at each level. Comment records are sometimes used to store help related to each table service entry. SuperVisor manages the link between the comment fields and the table entries. Up to 9999 comments may be associated with each database record. SuperVisor commands can be used to display or print comments for a specific database record.

The Hierarchy

The Table Service database is organized into a four level hierarchy, the system level, category level, table level, and code level. With each level providing the same combination of field elements. A 10-character key field defines each level.

Table Service Levels

System Level

Category Level

Table Level

Codes Level

Table Service Data Base

System File

USYSP00

Category File

UCATP00

Table File

UTBLP00

Code File

UCDEP00

SuperVisor Assistant is FREE

Table Service

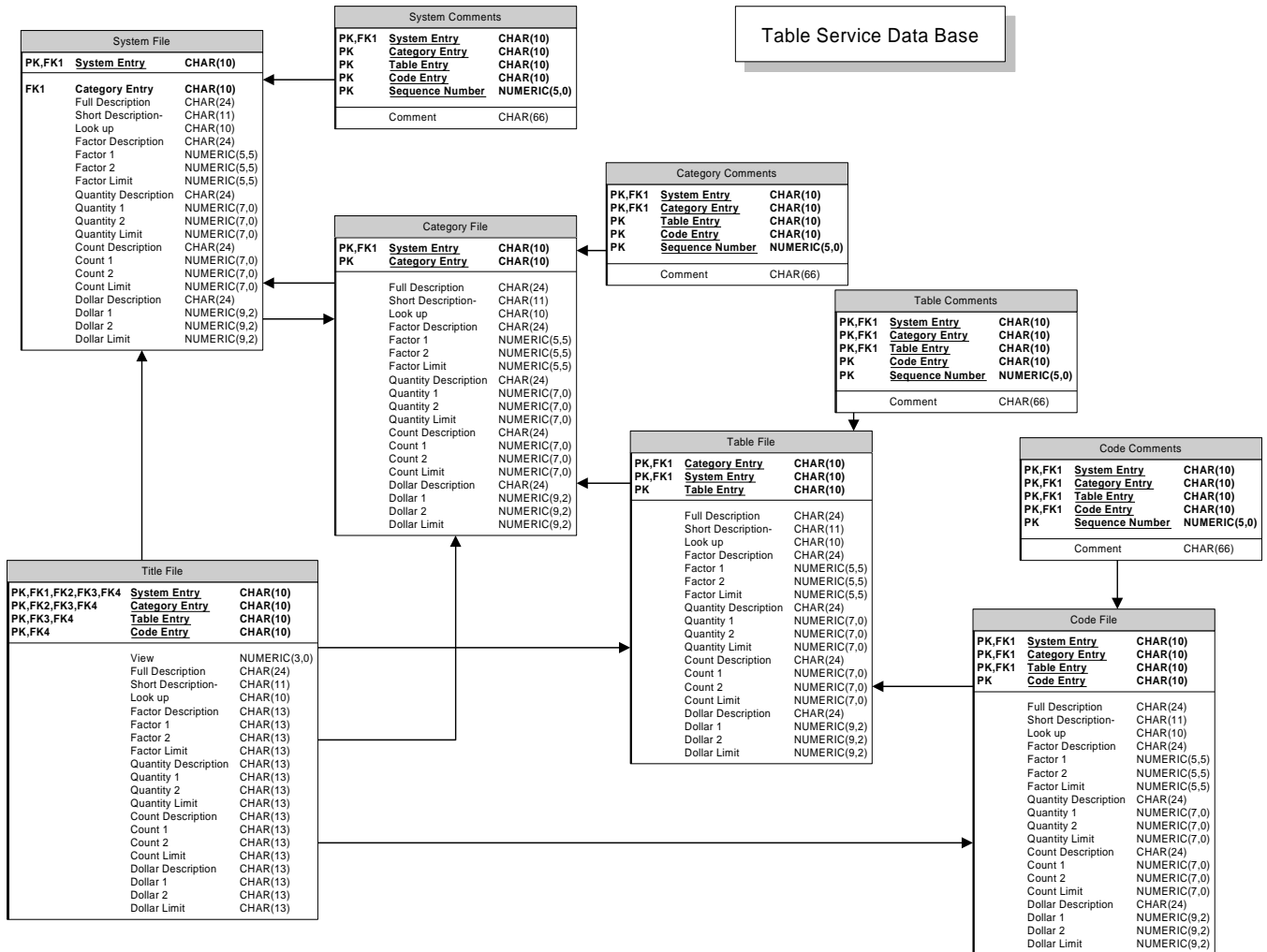


Table Service Data Diagram

Use this data base as you would an Excel spread sheet. It has the ability to store and organize your information with the ability to describe what the information is and how to use it as part of the data being stored.

There is no right way to use this data base. Use what works for you All records contain all of the data fields. So, it is not necessary to divide all tables into all four levels. For example, a state table might have but three levels, system (cross application), category (state table), and code level (state abbreviation). In this example, the category level was not used.

Create your own instance of the Table Service Database

One copy of the Table Service database resides in the SuperVisor library. SuperVisor uses this copy to store data used internally by SuperVisor.

Note: Your data must stored in your data base library. There is no other way to ensure that your Table Service Data is saved, restored, and mirrored they way it must be, when it is being used as a part of your application. The programs used to provide access to the Table Service data remains in SuperVisor, but that is all.

SuperVisor Assistant is FREE Table Service

```

TABLE                               Table Service Commands                               System: SMI

Select one of the following.

      1. Edit Table Service                EDTCDEFLE
      2. Print Table Service              PRTCDEFLE
      3. Display Table Service            INQCDEFLE

      5. Duplicate Table Service Data Base  DUPCDEFLE
      6. Move Table Service Data Base     MOVCDEFLE
      7. Organize Table Service Data Base  RGZCDEFLE

      9. Print Table Service Data Base    PRTTBldb
     10. Print Table Service Manual       PRTTxFLE

     12. Work with Table Service Commands  SUPERCMD

More...

Selection
====> 5
F3=Exit   F5=Next Menu- *SELECT   F6=Display Message   F10=Command entry
F12=Cancel F14=Work with jobs     F16=Main menu        F18=Work with output
F21=Print menu help

```

Option 5 from the Table Service Menu is used to create a duplicate Table Service database.

```

Duplicate Code Files (DUPCDEFLE)

Type choices, press Enter.

Duplicate Files From Library: .  supervisor  Name
Duplicate Files to Library: . .  mylib       Name
Duplicate Data (*YES or *No): .  *NO         *YES, *NO
Printer Output Queue: . . . . . Name
Library name: . . . . . *LIBL      Name, *LIBL
Job Desc. for Submit Job: . . . > SUPERVISOR Name, *USRPRF
Library name: . . . . . *LIBL      Name, *LIBL

Bottom

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys

```

Option 5 prompts the duplicate code file command, which duplicates a table service database. In the above example the database is copied from SuperVisor to MYLIB. Note that the Duplicate Data is set to *NO to eliminate the tables used internally by SuperVisor.

SuperVisor Assistant is FREE Table Service

```

                                Edit Code File (EDTCDEFLE)

Type choices, press Enter.

System Level Code: . . . . . Character value
Category Level Code: . . . . . Character value
Table Level Code: . . . . . Character value
Direct to Code Display: . . . . *YES          *YES, *NO
Protect Entries (*YES, *NO): . . *NO          *YES, *NO
TABLE SERVICE File Override: . . *LIBL       Name, *LIBL
Remote location . . . . . *LOCAL         Character value
Screen Title (40): . . . . . *NONE

Edit exit program . . . . . *NONE          Name, *NONE, EDTCDERPG...
Library name . . . . . *LIBL             Name, *LIBL

                                                                Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
  
```

The option presents the above prompt. The table service database is organized into a 4 level hierarchy, system, category, table and code. The command enables the user to specify a level. For this example the entire database will be viewed, so nothing is entered in system, category, or table levels. Pressing Enter displays all of the “systems” in the library.

```

USYSL01          TABLE SERVICE - Table Level Maintenance          7/13/03
SUPERVISOR                                           1 of 5

Type options, press enter
1=Select  3=Copy  4=Delete  5=Edit extended factors  7=Edit comments
9=Edit column heading
Opt System      Full Description          Short Desc      X-Ref
ADD
  SUPERVISOR    SuperVisor System Values      SuperVisor
1  XAPPL        Cross Application Tables      X-App1

F3=Exit  F12=Return  F5=Refresh  F11=Next view  F16=Alt view
F18=Position
  
```

One of the ideas behind table service is that all of the tables for an application may be stored in one place. In this example the system is XAPPL is the Cross-Application Tables.

Note: Notice the upper left corner of the screen. Each Table Service screen identifies the file name and library information being displayed is stored. In this example library SUPERVISOR file USYSL01.

SuperVisor Assistant is FREE

Table Service

```
UCATL01          TABLE SERVICE - Table Level Maintenance          7/13/03
SUPERVISOR      System      XAPPL      Cross Application Tables    1 of 5

Type options, press enter
1=Select 3=Copy 4=Delete 5=Edit extended factors 7=Edit comments
9=Edit column heading
Opt Code      Full Description      Short Desc      X-Ref
ADD
  CDEMENU      Code Menu Example
1  MISC        Miscellaneous          Misc

F3=Exit F12=Return F5=Refresh F11=Next view F16=Alt view
F18=Position
```

This display shows all of the categories in the XAPPL system. The MISC category is chosen.

```
UTBLL01          TABLE SERVICE - Table Level Maintenance          7/13/03
SUPERVISOR      System      XAPPL      Cross Application Tables    1 of 5
                Category      MISC        Miscellaneous

Type options, press enter
1=Select 3=Copy 4=Delete 5= Edit extended factors 7=Edit comments
9=Edit column heading 21=Print Table
Opt Table      Full Description      Short Desc      X-Ref
ADD
  ADMIN        Administration Users
  BRANCH       Branch Codes          Branch
1  MONTH       Months of the Year    Month
  STATE        State Table           State Table

F3=Exit F12=Return F5=Refresh F11=Next view F16=Alt view
F18=Position
```

This display shows all of the tables in the miscellaneous category. Let's look at the sample Month Table.

SuperVisor Assistant is FREE Table Service

```

UCDEL01          TABLE SERVICE - Code Level Maintenance          7/13/03
SUPERVISOR      System      XAPPL      Cross Application Tables    1 of 2
                Category    MISC      Miscellaneous
                Table      MONTH     Months of the Year

Type options, press enter
3=Copy  4=Delete  5=Extended factors  7=Comments

Opt Month      Full Description          Short Desc
ADD
  01          January              Jan.
  5 02          Feburary              Feb.
  03          March                March
  04          April                April
  05          May                  May
  06          June                 June
  07          July                 July
  08          August              Aug.
  09          September           Sept.
  10          October             Oct.
                                           +

F3=Exit  F5=Refresh  F12=Return  F21=Print  F16=Alt heading
F11=Next View
  
```

This display shows all of the entries or “codes” in the month table. A code, Description and Short Description may be captured for each table entry. In addition table service can store additional data for each code in the table. SuperVisor refers to this additional data as Extended Factors. Option 5 will display the extended factors.

```

UCDEL01          TABLE SERVICE - Extended Factors          7/13/03

Type changes, press enter

Enter the default view and column headings to be used when this table
is displayed. System: XAPPL      Category: MISC      Table: MONTH

      Month      Full Description          Short Desc  X-Ref
      02          Feburary              Feb.

Dec/Desc          Qty/Desc          Number of Days in Mth 00
Dec/Factor1      .00000          Num. of Days          28
Dec/Factor2      .00000          Qty/Factor2           0
Limit            .00000          Limit                 0

Count/Desc          Amt/Desc
Count/Factor1        0          Amt/Factor1           .00
Count/Factor2        0          Amt/Factor2           .00
Limit                0          Limit                 .00

F3=Exit  F12=Return
  
```

In this example one quantity factor is used to represent the number of days in each month.

SuperVisor Assistant is FREE Table Service

```

UCDEL01          TABLE SERVICE - Extended Factors          7/10/00

Type changes, press enter

Enter the default view and column headings to be used when this table
is displayed. System: XAPPL          Category: MISC          Table: MONTH

          Code          Full Description          Short Desc  X-Ref
          02          Feburary          Feb.

Dec/Desc          Qty/Desc          Number of Days in Mth 00
Dec/Factor1 .00000          Num. of Days          28
Dec/Factor2 .00000          Qty/Factor2          0
Dec/Limit .00000          Qty/Limit          0

Count/Desc          Amt/Desc
Count/Factor1          0          Amt/Factor1          .00
Count/Factor2          0          Amt/Factor2          .00
Count/Limit          0          Amt/Limit          .00

F3=Exit  F12=Return
  
```

We will change the Description Field and Qty/Factor1 (shown as "Num of Days). The prompting fields, such as Qty/Factor1 can also be updated.

Define your own column headings

```

UTBLL01          TABLE SERVICE - Table Level Maintenance  7/13/03
SUPERVISOR          System          XAPPL          Cross Application Tables  1 of 5
                   Category        MISC          Miscellaneous

Type options, press enter
1=Select 3=Copy 4=Delete 5= Edit extended factors 7=Edit comments
9=Edit column heading 21=Print Table
Opt Table          Full Description          Short Desc  X-Ref
ADD
  ADMIN          Adminstration Users
  9 BRANCH          Branch Codes          Branch
  MONTH          Months of the Year          Month
  STATE          State Table          State Table

F3=Exit  F12=Return  F5=Refresh  F11=Next view  F16=Alt view
F18=Position
  
```

Option 9 can be used to update columnar headings and activate or deactivate extended factors.

SuperVisor Assistant is FREE

Table Service

Control the number of available views

```
UT@001          TABLE SERVICE - Edit Column Heading          7/10/00
Type changes, press enter
Use this screen to enter column headings for this table and to set the
default view. Blank column heading to mask view from screen.
System: XAPPL          Category: MISC          Table: MONTH

View 1:          Default View: 001 (1-5)
Code             Full Description             Short Desc
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxx  xxxxxxxxxxxx

View 2:
Code             Full Description
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxx

View 3:
Code             Full Description             Num. of Days
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxx             0

View 4:
Code             Full Description
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxx

View 5:
Code             Full Description
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxx

F3=Exit   F12=Return
```

This display shows the columnar headings associated with the code level of the MONTH table, in the MISC category in the XAPPL system. Note that the columnar heading refer to one level down in the hierarchy.

The term “View” refers to the elements that will be displayed when the user presses F11. F11 displays the next view, providing there are column headings for the data elements represented by the view. In other words, when a field has column headings the field is active and used in the database. When the column headings are blank, the data elements are not active. The Table Service data elements displayed by each view are as follows:

- View 1 is the initial default display and contains the Full Description, Short Description, and Relational Code fields.
- View 2 contains the column headings for Decimal Factors.
- View 3 contains the column headings for Quantity Factors.
- View 4 contains the column headings for Count Factors
- View 5 contains the column headings for Dollar Amount Factors

In the above example columnar headings are present for the Full Description and Short Description, which are shown in View 1. There is no column heading for the relational factor in View 1.

View 2, 4, and 5 are blank, and therefore will not be displayed when F11 is pressed.

View 3, Quantity Factors is displayed, but only Qty/Factor 1 is displayed. The columnar heading for Qty/Factor 1 is “Num. Of Days”.

Lets look at another example.

SuperVisor Assistant is FREE Table Service

```

UTBLL01          TABLE SERVICE - Table Level Maintenance          7/13/03
SUPERVISOR      System      XAPPL      Cross Application Tables    1 of 5
                Category    MISC      Miscellaneous

Type options, press enter
1=Select  3=Copy  4=Delete  5= Edit extended factors  7=Edit comments
9=Edit column heading  21=Print Table
Opt Table      Full Description          Short Desc      X-Ref
ADD
  ADMIN        Administration Users
  BRANCH       Branch Codes              Branch
  MONTH        Months of the Year          Month
9  STATE       State Table                State Table

F3=Exit  F12=Return  F5=Refresh  F11=Next view  F16=Alt view
F18=Position

```

The state table is selected.

```

UT@001          TABLE SERVICE - Edit Column Heading          7/10/00
Type changes, press enter
Use this screen to enter column headings for this table and to set the
default view. Blank column heading to mask view from screen.
System: XAPPL      Category: MISC      Table: STATE

View 1:          Default View: 001 (1-5)
State            Full Description          Short Desc      Branch
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx  xxxxxxxxxxxxxx  XXXXXXXXXXXXX
View 2:
State            Full Description
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
View 3:
State            Full Description          From Zip        To Zip
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx          0              9,999,999
View 4:
State            Full Description
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
View 5:
State            Full Description
XXXXXXXXXX      xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

F3=Exit  F12=Return

```

In this example View 1 has column headings for the Full Description, Short Description, and Relational Code fields. The Column heading for the Relational Code has been changed to “Branch”

View 2, 4, and 5 are inactive. View 3, Quantity Factors is used. The column headings for Qty/Factor 1 and Qty/Factor 2 have been changed to “From Zip” and “To Zip”.

Displaying Table Service In a Window

SuperVisor menu options enable an operator to set up and maintain a Table Service database without programming. As with most menu options, these options prompt a command. So, the command could have also been entered from a command line or entered as an instruction in a CL program.

SuperVisor Assistant is FREE Table Service

Rolling down the Table Service menu shows some additional commands that may be used in the same way.

```

TABLE                               Table Service Commands                               System: SMI

Select one of the following.

Using Table Service Information in CL Programs

    63. Display Systems in Window           DSPSYSWIN
    64. Display Categories in Window        DSPCATWIN
    65. Display Tables in Window            DSPTBLWIN
    66. Display Codes in Window             DSPCDEWIN

    68. Edit Code Comments                  EDTCDECMT
    69. Display Code Comments               DSPCDECMT
    70. Ask Using Code Comments             ASKCDECMT
    71. Execute Menu using Codes           EXCCDEMNU

                                                                 Bottom

Selection
====> 66
F3=Exit   F5=Next Menu- *SELECT   F6=Display Message   F10=Command entry
F12=Cancel F14=Work with jobs     F16=Main menu        F18=Work with output
F21=Print menu help

TABLE
  
```

The above menu page shows some display options. Option 30 is selected to display codes in a window.

```

                                Display Codes in a Window (DSPCDEWIN)

Type choices, press Enter.

System Level Entry: . . . . . xappl           Character value
Category Level Entry: . . . . . misc          Character value
Table Level Entry: . . . . . month           Character value
Display on Left or Right: . . . R             L, R
Display Top/Bottom/Full: . . . B             T, B, F
TABLE SERVICE File Override: . . *LIBL       Name, *LIBL

                                                                 Bottom

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
  
```

The command prompt is presented showing the display options. The month table is selected from the samples.

SuperVisor Assistant is FREE Table Service

```

TABLE                               Table Service Commands                               System: SMI

Select one of the following.

    1. Edit Table Service              EDTCDEFLE
    2. Print Table Service             PRTCDEFLE
    3. Display Table Service           INQCDEFLE

    5. Duplicate Table Service Data Base  DUPCDEFLE
    6. Move Table Service Data Base     MOVCDEFLE
    7. Organize Table Service Data Base  RGZCDEFLE

    9. Print Table Service Data Base    PRTTBldb
   10. Print Table Service Manual       PRTTxFLE

   12. Work with Table Service Commands  SUPERCMD

                                                    More...

.....
:                               Command                               :
:                               :                                     :
:   ==> supercmd table                                     :
:   F4=Prompt  F9=Retrieve  F12=Cancel                     :
:                                                         :
:.....

```

SuperVisor will display a list of additional Table Service commands

```

SUPERCMD                               SuperVISOR Commands

Type Option, Press Enter              Position . . . . .
1=Select  5=Display

Opt Command      Text Description

  ASKCDEcMT      TABLE SERVICE - Ask Code Comments
  CPYcATE        TABLE SERVICE - Copy Category Entry
  CPYcDEE        TABLE SERVICE - Copy Code Entry
  CPYsYSE        TABLE SERVICE - Copy System Entry
  CPYtBLE        TABLE SERVICE - Copy Table Entry
  CVTCDElST      TABLE SERVICE - Convert Code to List
  DLTCATE        TABLE SERVICE - Delete Category Entry
  DLTCDEE        TABLE SERVICE - Delete Code Entry
  DLTSYSE        TABLE SERVICE - Delete System Entry
  DLtTBLE        TABLE SERVICE - Delete Table Entry
  DSPcATWIN      TABLE SERVICE - Display a Categories in a Window
  DSPcDEcMT      TABLE SERVICE - Display Code Comments
  DSPcDEWIN      TABLE SERVICE - Display Codes in a Window

                                                    More...

F3=Exit  F10=Command entry  F11=Text sequence  F12=Return

```

RPG Example

To see an example of how to code an RPG program that exits to table service, see SINGLE in the SAMPLES source file in the SUPERVISOR library.

This program uses the table service files and then calls Table Service API commands to show many different ways the Table Service data can be made available and used. Most without any actual High level language programming like RPG.

Output Queue Management (OUTQMAIN)

```
OUTQMAIN                Manage Output Queues Main Menu                System: SMI

Select one of the following.

    1. Save/Restore Output Queue
    2. Copy output queue to source file
    3. Output Queue Management

Selection
====>
F3=Exit      F5=Next Menu- *SELECT    F6=Display Message  F10=Command entry
F12=Cancel  F14=Work with jobs                 F16=Main menu       F18=Work with output
F21=Print menu help
```

SuperVisor Output Queue Management enables programmers and operators to save, restore, manipulate and convert spooled print files and entire output queues.

The save/restore utilities enable the user to archive spool files or an entire output queue. Archived spool files can be reprinted or viewed electronically.

The copy output queue to a source file menu option enables the user to copy spool files or output queues to a source file. This facility is primarily used to create documentation.

The Output Queue Management menu option enables the user to send spool files to another output queue or to another AS/400. This sub-menu also enables a user to copy spool files to a database file.

SuperVisor Assistant is FREE

Output Queue Management

Archiving Spool Files

```
OUTQMAIN          Manage Output Queues Main Menu          System: SMI

Select one of the following.

    1. Save/Restore Output Queue
    2. Copy output queue to source file
    3. Output Queue Management

Selection
===> 1
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs       F16=Main menu         F18=Work with output
F21=Print menu help
```

To access output queue archiving utilities select option 1 from the Output Queue Management main menu.

SuperVisor Assistant is FREE

Output Queue Management

Create an Archive File

```
OUTQ01                Save/Restore Output Queue                System: SMI5

Select one of the following.

Manage Archive File

    3. Create Archive File                CRTARCF
    4. Delete Archive File                DLTARCF

Archive Reports

    8. Archive Entire Output Queue        ARCOUTQ
    9. Archive Reports by User            ARCUSRSPL
   10. Select Report to Archive by Output Que DSPOQ
   11. Select Report to Archive by User    DSPSPLE
   12. Archive an Individual Report        ARCSPLF

Generate New Report using Archive

   15. Generate New Reports from Archive   GENSPLE
   16. Select Report to Generate from Archive GENSPLE

Selection                                                    Bottom
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs        F16=Main menu         F18=Work with output
F21=Print menu help
```

This brings forward the save/restore menu. Select Option 3 to create an archive file.

```
                Create Archive OUTFILE (CRTARCF)

Type choices, press Enter.

Archive Output File Name: . . .    arc00629    Name
Library name: . . . . .            myarchive    Name
Output File Text: . . . . .        Archived Spool Files from 06/29/00

Maximum Report Width: . . . . .    198        Number

                                                    Bottom
F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

SuperVisor output queue archive files are multi-member database files. Each file member corresponds to one spool file. In the above example the file is named arc00629 in library MYARCHIVE. The maximum report width defines the record length of the archive file.

SuperVisor Assistant is FREE

Output Queue Management

Archive an Output Queue

```
OUTQ01                Save/Restore Output Queue                System: SMI5
Select one of the following.

Manage Archive File

    3. Create Archive File                CRTARCF
    4. Delete Archive File                DLTARCF

Archive Reports

    8. Archive Entire Output Queue        ARCOUTQ
    9. Archive Reports by User            ARCUSRSPL
   10. Select Report to Archive by Output Que DSPOQ
   11. Select Report to Archive by User    DSPSPLEFLE
   12. Archive an Individual Report        ARCSPLF

Generate New Report using Archive

   15. Generate New Reports from Archive  GENSPLEF
   16. Select Report to Generate from Archive GENSPLEF

Selection                                                    Bottom
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs        F16=Main menu         F18=Work with output
F21=Print menu help
```

To archive an entire output queue, select option 10. This causes the Display Output Queue command to be prompted.

```
Display Output Queue: (DSPOQ)

Type choices, press Enter.

Output Queue Name: . . . . . SMI                Name
Library name: . . . . . *LIBL                Name, *LIBL
Allow Parm Change (*YES,*NO): . *YES            *YES, *NO
Archive Output File Name: . . .                Name
Library name: . . . . .                Name
Output File Text: . . . . .

                                                    Bottom
F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
```

The output queue in this example is SMI. The user can choose any output queue. The name/library of the archive can be entered in this command. However, for this example no archive file is entered.

SuperVisor Assistant is FREE Output Queue Management

The command display's the spool files in the specified queue.

```

DSPOQ                               SuperVISOR(tm)                6/29/00    6:26:05
Type changes, press enter
3=Hold 5=Display 6=Release 8=Attributes A=Archive T=Copy to Text

File      Nbr  Job Name  User      Nbr      Page  Sts  Cpy  Formtype
TOBJI     0001 TOBJI     SMI       068266   0098 RDY  01  *STD
TOBJI     0001 TOBJI     SMI       068282   0098 RDY  01  *STD
LODSPFA   0001 LODSPFA   SMI       068283   0007 RDY  01  *STD
RMVBNDDIRB 0001 RMVBNDDIRB SMI       068284   0005 RDY  01  *STD
UPRINTER  0001 LODSPF   SMI       068296   0001 RDY  01  *STD
UPRINTER  0003 LODSPF   SMI       068296   0001 RDY  01  *STD
UPRINTER  0004 LODSPF   SMI       068296   0001 RDY  01  *STD
UPRINTER  0005 LODSPF   SMI       068296   0007 RDY  01  *STD
SXREF3012 0006 LODSPF   SMI       068296   0003 RDY  01  *STD
SXREF3052 0007 LODSPF   SMI       068296   0003 RDY  01  *STD
UPRINTER  0008 LODSPF   SMI       068296   0004 RDY  01  *STD
LODSPFA   0001 LODSPFA   SMI       068298   0007 RDY  01  *STD
UPRINTER  0001 DSP04   SMI       068297   0001 RDY  01  *STD
UPRINTER  0002 DSP04   SMI       068297   0001 RDY  01  *STD
UPRINTER  0004 DSP04   SMI       068297   0001 RDY  01  *STD      +

F5=Redisplay  F9=Change Parm
  
```

The display resembles the IBM work with output queue command, except that spool files may be archived or copied to a text file. (The text file is actually a source file.)

Since an archive file was not named in the initial command display F9 must be pressed to name the archive file.

```

DSPOQZ                               SuperVISOR(tm)                6/29/00

Archive Spool File Parameter (ARCOUTQ):
Archive Output File Name:      OUTFLE  ARC00629
  Library Name:                MYARCHIVE
Output File Text:              TEXT     Archive From 062900
Cancel Spool File After Copy:  CNLSPL  *NO
Job Desc. For Submit Job:      JOBD
  Library Name:                *LIBL

SPLIT Spool File for Printing on the Front/Back of the Paper (SPLITSPL):
Number of Cover Pages:        COVER  0

Copy Spool File to Source File (CPYQUESRC):
Output Source File Name:      SRCFLE
  Library Name:                *LIBL
Output Source File Member:     MBR     *FIRST
Cancel Spool File After Copy:  CNLSPL  *NO
Maximum # of Records to Copy:  MAXRCD  9999999
Type of Source Data:          PURPOSE  *NONE

F 3 - Exit                      Enter - Continue
  
```

SuperVISor Assistant is FREE Output Queue Management

From this display the parameters can be specified for the display's options. In this case the archive file name, library and description are entered.

```

DSPOQ                               SuperVISOR(tm)                6/29/00    6:29:15
Type changes, press enter
3=Hold 5=Display 6=Release 8=Attributes A=Archive T=Copy to Text

File      Nbr  Job Name  User      Nbr      Page  Sts  Cpy  Formtype
A TOBJI   0001 TOBJI     SMI       068266   0098 RDY  01  *STD
TOBJI    0001 TOBJI     SMI       068282   0098 RDY  01  *STD
A LODSPFA 0001 LODSPFA  SMI       068283   0007 RDY  01  *STD
RMVBNDDIRB 0001 RMVBNDDIRB SMI       068284   0005 RDY  01  *STD
UPRINTER  0001 LODSPFF  SMI       068296   0001 RDY  01  *STD
UPRINTER  0003 LODSPFF  SMI       068296   0001 RDY  01  *STD
UPRINTER  0004 LODSPFF  SMI       068296   0001 RDY  01  *STD
UPRINTER  0005 LODSPFF  SMI       068296   0007 RDY  01  *STD
SXREF3012 0006 LODSPFF  SMI       068296   0003 RDY  01  *STD
SXREF3052 0007 LODSPFF  SMI       068296   0003 RDY  01  *STD
UPRINTER  0008 LODSPFF  SMI       068296   0004 RDY  01  *STD
LODSPFA   0001 LODSPFA  SMI       068298   0007 RDY  01  *STD
UPRINTER  0001 DSP04  SMI       068297   0001 RDY  01  *STD
UPRINTER  0002 DSP04  SMI       068297   0001 RDY  01  *STD
UPRINTER  0004 DSP04  SMI       068297   0001 RDY  01  *STD      +

F5=Redisplay  F9=Change Parm
  
```

Now the archive option may be used from the display.

Archive a User's Spool Files

```

OUTQ01                               Save/Restore Output Queue                System: SMI5

Select one of the following.

Manage Archive File

    3. Create Archive File                CRTARCF
    4. Delete Archive File                DLTARCF

Archive Reports

    8. Archive Entire Output Queue        ARCOUTQ
    9. Archive Reports by User            ARCUSRSPLE
   10. Select Report to Archive by Output Que DSPOQ
   11. Select Report to Archive by User    DSPSPLEFLE
   12. Archive an Individual Report        ARCSPLF

Generate New Report using Archive

   15. Generate New Reports from Archive   GENSPLEF
   16. Select Report to Generate from Archive GENSPLEF

Bottom

Selection
====>
F3=Exit      F5=Next Menu- *SELECT    F6=Display Message  F10=Command entry
F12=Cancel  F14=Work with jobs        F16=Main menu       F18=Work with output
F21=Print menu help
  
```

SuperVisor Assistant is FREE

Output Queue Management

Option 11 can be used to view a list of the user's spool files.

```
Display Spool File: (DSPSPLFLE)

Type choices, press Enter.

Select Spool File for User: . . *CURRENT      Name, *CURRENT, *ALL
Allow Parm Change (*YES,*NO): . *YES        *YES, *NO
Archive Output File Name: . . . arc063000     Name
  Library name: . . . . . myarchive          Name
Output File Text: . . . . . archive from 063000

Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Option 11 brings forward the Display Spool File Command. Any user name can be entered. In this example *CURRENT is entered for the user entering the command. This time the archive file name and library are entered.

```
DSPSPLFLE                               SuperVISOR(tm)                               6/29/00   6:33:44
Type changes, press enter
3=Hold 5=Display 6=Release 8=Attributes A=Archive T=Copy to Text

File      Nbr  Job Name  User      Nbr      Page  Sts  Cpy  Formtype
A QPSUPRTF 0002 QTIRMTDSP ITS      039074   0002 RDY   01 *STD
UPRINTER  0001 XBLDXDB  ITS      063740   0003 RDY   01 *STD
UPRINTER  0002 XBLDXDB  ITS      063740   0003 RDY   01 *STD
UPRINT1   0001 QTIRMTDSP ITS      065021   0001 RDY   01 *STD
UPRINTER  0001 XBLDXDB  ITS      065768   0001 RDY   01 *STD

F5=Redisplay   F9=Change Parm
```

A list of the users spool files is presented. The archive option can be used to archive a specific spool file because the archive file was named in the initial command prompt.

Generate a Spool File from an Archive

SuperVisor Assistant is FREE Output Queue Management

```

OUTQ01                Save/Restore Output Queue                System: SMI5

Select one of the following.

Manage Archive File

    3. Create Archive File                CRTARCF
    4. Delete Archive File                DLTARCF

Archive Reports

    8. Archive Entire Output Queue        ARCOUTQ
    9. Archive Reports by User            ARCUSRSPL
   10. Select Report to Archive by Output Que DSPOQ
   11. Select Report to Archive by User    DSPSPLFLE
   12. Archive an Individual Report        ARCSPLF

Generate New Report using Archive

   15. Generate New Reports from Archive   GENSPLF
   16. Select Report to Generate from Archive GENSPLF

                                                    Bottom

Selection
====>
F3=Exit    F5=Next Menu- *SELECT    F6=Display Message    F10=Command entry
F12=Cancel F14=Work with jobs      F16=Main menu        F18=Work with output
F21=Print menu help
  
```

Option 16 can be used to generate a spool file from an archive.

```

                                Generate Spool File: (GENSPLF)

Type choices, press Enter.

Archive Output File Name: . . . arc00629    Name
Library name: . . . . . myarchive          Name
Spool File Member Select: . . . > *SELECT  Name, *FIRST, *SELECT, *ALL
Delete OUTFLE After Gen: . . . *NO         *YES, *NO
Prompt OVRPRTF Command: . . . *NO         *YES, *NO
Split for Front/Back Printing: . . . *NO   *YES, *NO
Number of Cover Pages: . . . 0             1, 2, 3, 4, 5, 6, 7, 8, 9, 0
Printer Output Queue: . . . . . qprint2    Name
Library name: . . . . . *LIBL              Name, *LIBL
Job Desc. for Submit Job: . . .           Name, *USRPRF
Library name: . . . . . *LIBL              Name, *LIBL

                                                    Bottom

F3=Exit    F4=Prompt    F5=Refresh    F12=Cancel    F13=How to use this display
F24=More keys
  
```

The SuperVisor GENSPLF command is prompted.

The first parameter names the archive and library. The Spool File Member is used to specify which spool file to retrieve. *SELECT can be used as shown here to present a list of available spool files. The output queue to receive the restored archive is shown as qprint2.

SuperVisor Assistant is FREE Output Queue Management

```
USPLA          MANAGEMENT - (Q) Generate Spool File/Archive File          6/29/00

Type options, press enter
1=Select 5=Display
Opt Member      Description
 1 S001810001    Archive From 062900 068266/SMI/TOBJI/0001
  S001810002    Archive From 062900 068283/SMI/LODSPFA/0001

F3=Exit

Bottom
```

Since *SELECT was specified, the above display is presented and a spool file has been selected for restoration to qprint2.

What is an Output Queue Archive File?

An archive out file is a multi-member data base file. Each spool file is copied into a separate member within the file. Member names are assigned in the Archive using the following convention:

- SBBBBBCCCC - where:
- S - is the literal "S"
- B - is the Julian date the member was added to the file
- C - is a sequential number

The Archive File name is used to name an Attribute Member. The Attribute Member retains a record for each spool file in the Archive and each record contains the spool file attributes for the corresponding spool file.

All Archive files are logged into the SuperVisor Archive Outfile Log (UARCP01).

Archive Job Logs

As a way to demystify this process, a quick example using job logs is used. In this example we will:

- Archive all the job logs that exist in the QEZJOBLOG output queue.
- Look at the archive file using WRKMBRPDM
- Generate a new job log from the archive.

Execute the archive

SuperVisor Assistant is FREE Output Queue Management

```
Selection or command
==> ARCOUTQ OUTQ(QUEZJOBLOG) OUTFLE(MYLIB/LOG060121) OUTTXT('job logs saved J
anuary 21, 2006')
```

View data in the archive

```
Work with Members Using PDM SMI5
```

```
File . . . . . LOG060121
Library . . . . . MYLIB          Position to . . . . .
```

```
Type options, press Enter.
3=Copy    4=Delete    5=Display    7=Rename    8=Display description
9=Save    13=Change text 18=Change using DFU    25=Find string ...
```

Opt	Member	Date	Text				
	ARC0000001	01/21/06	QPJOBLOG	0001 QS9SACOL	QSRVAGT	021210	2
	ARC0000002	01/21/06	QPJOBLOG	0001 SUPERSAV	QIJS	021224	3
	ARC0000003	01/21/06	QPJOBLOG	0001 QS9SACOL	QSRVAGT	021260	2
	ARC0000004	01/21/06	QPJOBLOG	0001 SUPERSAV	QIJS	021274	3
	ARC0000005	01/21/06	QPJOBLOG	0001 QS9SACOL	QSRVAGT	021310	2
	ARC0000006	01/21/06	QPJOBLOG	0001 SUPERSAV	QIJS	021324	3
	ARC0000007	01/21/06	QPJOBLOG	0001 QS9SACOL	QSRVAGT	021360	2
	ARC0000008	01/21/06	QPJOBLOG	0001 SUPERSAV	QIJS	021374	3

More...

```
Parameters or command
==>
F3=Exit    F4=Prompt    F5=Refresh    F6=Create
F9=Retrieve F10=Command entry F23=More options F24=More keys
```

Each report is saved as a separate member in the file. Text Description includes important information from the spool file attributes.

```
Display Physical File Member
```

```
File . . . . . : LOG060121          Library . . . . . : MYLIB
Member . . . . . : ARC0000001      Record . . . . . : 3
Control . . . . . : +1             Column . . . . . : 36
Find . . . . .
```

```
....4....+....5....+....6....+....7....+....8....+....9....+....0....+....1....
QSJINV          Library . . . . . : QSYS
```

SEV	DATE	TIME	FROM PGM	LIBRARY	INST	TO PGM
00	01/15/06	22:49:00.154880	QWTPPIPP	QSYS	061C	*EXT

```
Message . . . . : Job 021210/QSRVAGT/QS9SACOL started on 01/15/06 at
22:49:00 in subsystem QSYSWRK in QSYS. Job entered system on 01/15/06 at
22:49:00.
00 01/15/06 22:49:00.155160 QWTPCRJA    QSYS      0108    *EXT
Message . . . . : Job 021210/QSRVAGT/QS9SACOL submitted.
Cause . . . . . : Job 021210/QSRVAGT/QS9SACOL submitted to job queue QSJI
in QSYS from job 008182/QSYS/QJOBSCD. Job 021210/QSRVAGT/QS9SACOL was
started using the Submit Job (SBMJOB) command with the following job
attributes: JOBPTY(5) OUTPTY(5) PRTTXT() RTGDTA(RUNPTY50) SYSLIBL(SMISYS
QSYS      QSYS2    QHLPYSYS    QUSRSYS) CURLIB(*CRTDFT) INLLIBL(QGPL
QTEMP) LOG(4 00 *NOLIST) LOGCLPGM(*NO) INQMSGRPY(*RQD)
OUTQ(QSRVAGT/QS9SRVAGT) PRIDEV(PRT01) HOLD(*NO) DATE(*SYSVAL) SWS(0000000
More...
```

```
F3=Exit    F12=Cancel    F19=Left    F20=Right    F24=More keys
```

SuperVisor Assistant is FREE

Output Queue Management

Job logs are text, *SCS type reports. You can use all the tools available to PDM or SuperVisor to interrogate the information to meet your research needs.

Generate a new report

Use the Generate Spool File (GENSPLF) to reproduce reports using information in the archive.

```
Work with Members Using PDM                               SMI5

File . . . . . LOG060121
Library . . . . MYLIB                               Position to . . . . .

Type options, press Enter.
 3=Copy    4=Delete    5=Display    7=Rename    8=Display description
 9=Save    13=Change text 18=Change using DFU    25=Find string ...

Opt Member      Date      Text
ARC0000001 01/21/06 QPJOBLOG 0001 QS9SACOL QSRVAGT 021210 2
ARC0000002 01/21/06 QPJOBLOG 0001 SUPERSAV QIJS    021224 3
ARC0000003 01/21/06 QPJOBLOG 0001 QS9SACOL QSRVAGT 021260 2
ARC0000004 01/21/06 QPJOBLOG 0001 SUPERSAV QIJS    021274 3
ARC0000005 01/21/06 QPJOBLOG 0001 QS9SACOL QSRVAGT 021310 2
ARC0000006 01/21/06 QPJOBLOG 0001 SUPERSAV QIJS    021324 3
ARC0000007 01/21/06 QPJOBLOG 0001 QS9SACOL QSRVAGT 021360 2
ARC0000008 01/21/06 QPJOBLOG 0001 SUPERSAV QIJS    021374 3
More...

Parameters or command
===> GENSPLF OUTFILE(MYLIB/LOG060121) MBR(ARC0000001)
F3=Exit      F4=Prompt      F5=Refresh      F6=Create
F9=Retrieve   F10=Command entry F23=More options F24=More keys
```

Executing the command, then use WRKJOB command to view the new report.

```
Work with Job Spooled Files

Job:  QPADEV0003    User:  SMI          Number:  021493

Type options, press Enter.
 1=Send  2=Change  3=Hold  4=Delete  5=Display  6=Release  7=Messages
 8=Attributes  9=Work with printing status

Opt File          Device or      User Data  Status  Total  Current  Copies
 5  QPJOBLOG      SMI          QS9SACOL  RDY     2       1       1

Bottom

Parameters for options 1, 2, 3 or command
===>
F3=Exit  F10=View 3  F11=View 2  F12=Cancel  F22=Printers  F24=More keys
```

This time the report is created as part of the new job. It can be printed, displayed as you would any iSeries report

SuperVisor Assistant is FREE Output Queue Management

SuperVisor Archive supports all report types

This example used job logs as an example, but the using archive format2, all iSeries reports can be archived. This includes *AFP and *IPDS reports.

SuperVisor Assistant is FREE
Utility Menu

Utility Menu (UTILITY)

UTILITY	Utility Commands	System: SMI	
Select one of the following.			
1. Convert Break Message	CVTBRKMSG		
2. Print Program Queue Messages	PRTPGMMMSG		
4. Copy Data From Save File	CPYFRMSAVF		
5. Copy Data to Save File	CPYTOSAVF		
7. Display Date and Time	WHEN		
8. Record Count	COUNT		
10. Spelling Aid	SPELLAID		
11. Check word spelling	SPELLCHK		
System management utilities			
15. Add Job Description Library Entry	ADDJOBLE		
16. Remove Job Description Library Entry	RMVJOBLE		
18. Verify Save/Restore Tape	VFYSAVTAP		
19. Restore library using SuperVisor *LIST	RSTLIBLST		
Copy Source File Members to Transfer File			
23. Copy Source to Transfer File	CPYSRCTFR		
24. Copy Transfer File to Source	CPYTFRSRC		
Valid in CL programs only			
27. Add Days to Date	ADDDATE		
28. Retrieve Number Edited	RTVNUMEDT		
29. Bubble Messages to Previous Program	BUBBLEMSG		
32. Set Program or File Active	SETACTIVE		
33. Test if Program or File is Active	IFACTIVE		
35. Save Job Library List	SAVJOBLIBL		
36. Restore Job Library List	RSTJOBLIBL		
SuperVisor Tutor Library Support			
39. Restore Tutor Libraries	RSTTUTOR		
40. Delete SuperVisor Tutor Library	DLTTUTOR		
More Utilities			
44. Display External Messages	DSPEXTMSG		
Selection		Bottom	
==>			
F3=Exit	F5=Next Menu- *SELECT	F6=Display Message	F10=Command entry
F12=Cancel	F14=Work with jobs	F16=Main menu	F18=Work with output
F21=Print menu help			

SuperVisor Assistant is FREE Utility Menu

Time Saving Tools

SuperVisor Utilities provides some handy utilities for the user's session. As with most SuperVisor menus, all of the commands on this menu prompt SuperVisor commands. These commands can be entered on a command line or included in CL programs.

1. Convert Break Message (CVTBRKMSG) causes break messages from the current menu to be converted to non-break messages, enabling the user to view messages as jobs end rather than being interrupted by a break message.
2. Print Program Queue Messages (PRTPGMMSG) causes messages for the session to be moved to a spool file. It is a simple way to produce audit reports for CL programs, in format much easier than having to review the job log.
4. Copy Data From Save File (CPYFRMSAVF) restores data from a disk resident save file.
5. Copy Data to Save File (CPYTOSAVF) copies data to a disk resident save file.
7. Display Date and Time (WHEN) displays the system date and time.
8. Record Count (COUNT) counts the records in a file.
11. Spelling aid receives up to 100 character text and performs a spell check. If a word is misspelled, a window is displayed with choices for the correct spelling. Choose from the list and the corrected character string is returned.
15. Add Job Description Library Entry, works similar to the ADDLIBLE command, but changes the specified job description.
16. Remove Job Description Library Entry, removes the specified library from the job description.
18. Verify Save/Restore Tape goes beyond display tape to verify information on the tape can still be read. This command will read the entire tape beginning to end. If an I/O error is encountered you are notified of the problem.
19. Restore library using SuperVisor *LIST. This command will step through a Save/Restore tape stopping at each library. It checks to see if the library is on the SuperVisor *LIST. If found on the *LIST, the library is restored.
23. Copy Source to Transfer File. Use this command to process all members of a source file, and output a data base file with all the members compressed into the single member. Separator records are placed into the data to identify the beginning and end of each source member. No the source is in a single member it is easy to download or FTP to another system.
24. Copy Transfer File to Source, will read the file created using the "Copy Source to Transfer File" and output a source file distributing the source to rebuild the original member list.
27. Add Days to Date. Use this command in a CL program to add or subtract a number of days from the specified date.
28. Retrieve Number Edited.. Use this command in a CL program. It receives a numeric number and returns a character field with the specified editing inserted.
29. Bubble Messages to Previous Program. Move program messages from one program in the invocation stack to another.
32. Set Program or File Active, Places a allocation on an object, generally a program to show that the program is being executed.
33. Test if Program or File is Active, Use in the CL program to control program actions if the specified object is being used.
35. Save Job Library List, Places the library list for the current job into file located in the QTEMP library.

SuperVisor Assistant is FREE Utility Menu

- 36. Restore Job Library List. Access the library list, saved using “Save Job Library List” and changes the job to the values of the saved library list.
- 39. Restore Tutor Libraries. Throughout the user guide you see references to the Simple Purchase Order System. This system really exists. Use this command to restore the libraries that are SPO application.
- 40. Delete SuperVisor Tutor Library. Use this command to delete the libraries that make up the Simple Purchase Order application.
- 44. Display External Messages. This command displays the external messages associated to the current job.

**SUPERVISOR
SYSTEM VALUES**

SuperVisor System Values

```
SYSVAL                SuperVisor System Values                System: SMI5

Select one of the following.

    1. Edit System Size                EDTSYSSIZ
    2. Edit default file editor        SUPFED
    3. Edit SuperVisor Test default profile TSTEDTDFI
    4. Edit User Options                EDTUSROPT
    5. Edit Exit on Enter               EDTEXIT
    6. Edit Main Menu Attention Menu    SUPERVISOR
    7. Edit Remote Location             EDTRMTLOC

Selection
===>
F3=Exit      F5=Next Menu- *SELECT    F6=Display Message  F10=Command entry
F12=Cancel  F14=Work with jobs        F16=Main menu       F18=Work with output
F21=Print menu help
```

SuperVisor includes different information to provide defaults or control processing of different SuperVisor commands and processes. Use this menu to display or change this information.

1 EDTSYSSIZ - Edit System Size

```
Edit System Size (EDTSYSSIZ)
No parameters to show; press Enter to run, F3 to exit.
```

General Description

This command is used to enter or change the SuperVisor System Value SYSSIZ. The system size is a data area that stores the amount of auxiliary storage on the AS/400.

SuperVisor uses this value to calculate percentage of disk spaced used by an object, group of objects, or library.

SuperVisor System Values

The following screen will be displayed.

```

EDTSYSSIZ          Edit System Auxiliary Storage Size          9/19/93
                                                            0:12:08

Type changes, press enter

Enter the Total System Auxiliary Storage Size. This is then used
as a SuperVISOR system value. This information can be found in the
upper right hand corner of the IBM Work with System Status (WRKSYSSTS)
display. Enter the Storage System in megabytes, including the trailing
zeros.

System Size:                0

F3=Exit    F5=WRKSYSSTS
    
```

The Auxiliary Storage Information is found in the upper right hand corner of the IBM Work with System Status display. From the EDTSYSSIZ display, press F5.

This displays the IBM Work with System Status display.

```

5738SS1 V2R1M0 910524          SMI          03/02/94    08:39:42

Display Device . . . . . : PT00A2
User . . . . . : CJHOLKE

                Work with System Status          SMI
                                                03/02/94    08:39:42
% CPU used . . . . . : 50.8    Auxiliary Storage:
Elapsed time . . . . . : 00:25:23    System . . . . . : 1802 M
Jobs in system . . . . . : 222    % used . . . . . : 83.9627
% addresses used:    Total . . . . . : 1802 M
  Permanent . . . . . : 4.116    Current unprotected used : 247 M
  Temporary . . . . . : 4.511    Maximum unprotect 279 M

Type pool size and activity level changes, press Enter.

System   Pool   Reserved   Max   ----DB----   ---Non-DB---
Pool   Size (K)   Size (K)   Active   Fault   Pages   Fault   Pages
  1     4500     2404     +++     .0     .0     .1     .9
  2     6780         0         4     3.7   78.2   .2     1.3
  3      200         0         3     .0     .0     .0     .2
  4     9000         0         8     .7     8.0   2.4   11.7

Command                                          Bottom
===>
F3=Exit   F4=Prompt          F5=Refresh   F9=Retrieve   F10=Restart
F11=Display transition data   F12=Cancel   F24=More keys
    
```

Write down the total disk space available on your system as it appears on this screen (in megabytes). You will need to enter this information when you return to the EDTSYSSIZ display.

SuperVisor System Values

From the WRKSYSSTS screen, press the command key:

F3=Exit

You should now be returned to the EDTSYSSIZ screen.

```
EDTSYSSIZ          Edit System Auxiliary Storage Size          9/19/93
                                                           0:12:08

Type changes, press enter

Enter the Total System Auxiliary Storage Size. This is then used
as a SuperVISOR system value. This information can be found in the
upper right hand corner of the IBM Work with System Status (WRKSYSSTS)
display. Enter the Storage System in megabytes, including the trailing
zeros.

System Size:                1802

F3=Exit   F5=WRKSYSSTS
```

Enter the system size value (in megabytes) you wrote down from the WRKSYSSTS screen. SuperVisor will convert this number to bytes by adding trailing zeros.

2. Edit Default File Editor (SUPFED)

```
          Edit File Editor System Value (SUPFED)

F3=Exit   F5=Refresh   F12=Cancel   F13=How to use this display   F24=More keys

No parameters to show; press Enter to run, F3 to exit.
```

General Description

SuperVisor provides screens that provide the ability to select a file to be viewed or edited. This command provides the ability to enter the name of the file editor of choice at that should be used at your installation.

The IBM Update Data (UPDDTA) command is provided as a default. It appears as follows.

```
? UPDDTA
```

When used SuperVisor will append the name of the file and library to create the complete command.

```
? UPDDTA  &LIB/&FILE
```

SuperVisor System Values

The command prompt will be displayed, if the file edit option is selected, and command key 4 (prompt) is pressed.

Any valid file edit command can be used, so long as the first parameter received by the command is a qualified file name.

4 EDTUSROPT - Edit User Options

```
                                Edit User Options (EDTUSROPT)
F3=Exit   F5=Refresh   F12=Cancel   F13=How to use this display   F24=More keys

No parameters to show; press Enter to run, F3 to exit.
```

General Description

This command is used to enter, change, or delete options from the SuperVisor user option file. This command is also available using F18 from many of the SuperVisor displays.

SuperVisor user options provide a capability similar to the IBM user options available in PDM, with the following additional capabilities:

Multiple Commands: Up to 999 commands can be executed under a single option.

Continue with errors: If an error is encountered, you have a choice to continue with the next command, or stop and return to the calling screen.

Hold Code: A hold code will prevent the command from being executed without having to delete it. This is useful when executing multiple commands under a single option.

Prompt Text: Up to 5-prompted text fields (UKEY1-UKEY5) can be specified in a command. The first time the option is run, a prompt window is displayed. UKEY1-UKEY5 retained and not prompted for subsequent runs of the same option. The UKEY1-UKEY5 fields are reset when one of the following occurs: 1) A new user session is started (sign-on), 2) The option is changed, or 3) the user selects Option 5 (refresh) from the calling display.

Substitution: The below information from a SuperVisor display can be substituted into the command.

SuperVisor System Values

	Substitution
Value	Description
UNAME	Object name
ULIB	Object library
UTYPE	Object type
UATTR	Object attribute
USRCF	Object Source File Reference
USRCL	Object Source Library Reference
USRCM	Object Source Member Reference
UTEXT	Object Text
UKEY1	Prompted Field
UKEY2	Prompted Field
UKEY3	Prompted Field
UKEY4	Prompted Field
UKEY5	Prompted Field

Sample Display

Following is the Edit User Options work display.

```

                                EDTUSROPT          Edit User Options

Type changes, press enter                               Position to . . . .
  1=Add 2=Change 4=Delete 5=Display

Opt Option  Seq  Hold  Command
  1      A1
  C1      000      /* Copy/Replace Object */ CRMDUPG OBJ(ULIB/UNAME)
  DM      000      /*** Display Messages ***/ dspmsg
  EA      000      /* Edit Object Authority */ EDTOBJAUT OBJ(ULIB/UNA
  E1      000      /* 0. Copy source member */ CPYSRCG FRMFLE(USRCL/U
  E1      001      /* 1. Copy object source */ CRMDUPG OBJ(ULIB/UNAM
  E1      002      /* 2. Update source location */ CHGOBJSF OBJ(ULIB/
  SA      000      /* Compare Source File */ CMPSRCF SRCFLE1(USRCM/UK
  SB      000      /* Indent RPG Source */ INDENTRPG MBR(USRCM) SRCFL
  SC      000      /* Change Object Source Info */ CHGOBJSF OBJ(ukey1
  SD      000      /* Display Member Information */ DSPMBRI FILE(ULIB
  SE      000      /* Display Object Information */ DSPOBJI OBJ(ULIB/
  SF      000      /* Display Dependant Logical Files */ DSPDLF FILE(
                                                More...

F3=Cancel  F5=Refresh  F17=Top  F18=Bottom  F21=Print

```

To add an additional option enter a 1 in the option field and an option code. Press "enter" to display the option definition screen.

SuperVisor System Values

```

EDTUSROPT      Edit User Options                                ADD

Type changes, press enter

Option . . . . .:  A1

Sequence . . . . .  000
Hold Code . . . . .      Y=Yes
Continue with errors . .      Y=Yes

Command . . . . .  CRTDUPOBJ OBJ(UNAME) FROMLIB(ULIB) OBJTYPE(utype
) TOLIB(U1KEY)

Substitution Values                                Prompt text
UNAME = Object Name          USRCF = Source File      UKEY1 = Target Library
ULIB = Object Library        USRCL = Source Library  UKEY2 = _____
UTYPE = Object Type          USRCM - Source Member  UKEY3 = _____
UATTR = Object Attribute                                           UKEY4 = _____
                                                                    UKEY5 = _____

F3=Exit  F4=Prompt  F12=Cancel
    
```

Parameters

Title	Description
Sequence	The sequence code identifies the order that duplicate options entries will execute.
Hold Code	Y = prevents the option from being executed. Hold commands can be retained in the file, but will be bypassed if selected.
Continue With Errors	Y = continues when an error occurs. Blank halts on errors.

```

EDTUSROPT      Edit User Options                                ADD

Type changes, press enter

Option . . . . .:  A1

Sequence . . . . .  000
Hold Code . . . . .      Y=Yes
Continue with errors . .      Y=Yes

Command . . . . .  CRTDUPOBJ OBJ(UNAME) FROMLIB(ULIB) OBJTYPE(utype
) TOLIB(U1KEY)

Substitution Values                                Prompt text
UNAME = Object Name          USRCF = Source File      UKEY1 = Target Library
ULIB = Object Library        USRCL = Source Library  UKEY2 = _____
UTYPE = Object Type          USRCM - Source Member  UKEY3 = _____
UATTR = Object Attribute                                           UKEY4 = _____
                                                                    UKEY5 = _____

F3=Exit  F4=Prompt  F12=Cancel
'UTYPE   ' not valid for parameter OBJTYPE.                                +
    
```

Note: F4 prompts for parameters. In some cases, prompting verifies the validity of parameters. Substitution keywords often cause a warning message to be displayed. If the substituted value will be correct, override the warning message by pressing enter. The command will be accepted and stored as entered.

5. EDTEXT - Edit Exit List on Enter

```
                Edit Exit list on enter (EDTEXT)

Type choices, press Enter.

Exit list on enter . . . . . EXIT          *YES

F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys                                     Bottom
```

General Description

This value specifies how SuperVisor displays will terminate when no data is entered.

Parameter

EXIT

*NO exits by pressing enter. *YES requires F3 to exit.

6. Edit Main Menu Attention Menu

General Description

Edit SuperVisor Attention Key Menu Use this option to edit the SuperVisor Attention Key Menu. The Menu is made up of three parts.

• Function	• Description
• Opt Code	• Controls the sequence the options will be displayed.
• Full Description	• This information is a brief description of the menu function.
• Short Description	• This column is not used.
• Look up	• Enter the name of the command or program to execute when the option is selected.

5 SUPFED - Edit File Editor System Value

SuperVisor System Values

```
Edit File Editor System Value (SUPFED)
F3=Exit  F5=Refresh  F12=Cancel  F13=How to use this display  F24=More keys

No parameters to show; press Enter to run, F3 to exit.
```

General Description

This command specifies the name of the file editor to be used.

```
SUPFED          Edit File Editor System Value

Type changes, press enter

Enter the name of the file edit utility command that should be
used when file edit option has been selected.

File Editor.:  ? UPDDTA

F3=Exit
```

The IBM Update Data (UPDDTA) command is the default.

When used in SuperVisor the file and library are appended to the value specified.

```
? UPDDTA &LIB/&FILE
```

6 TSTEDTDF - Edit SuperVisor Test Default Profile

```
          Edit Test Default Test Profile (TSTEDTDF)

Type choices, press Enter.

Test default test profile . . . '9999'      Character value

                                                                 Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

General Description

This command updates name of the default test profile used in SuperVisor Test. The default profile is a container for storing a complete list of files involved in testing and copy/filter specifications for these files. A default profile is not required. However, its use makes SuperVisor test easier to use when a large numbers of test profiles are involved.

7 Edit Remote Location

The most significant change made in this new release deals with the ability SuperVisor has to process information on remote systems and return the results to the local system. In today's modern iSeries networks, the use of multiple iSeries systems or Logical Partitions (Lpars) is no longer the exception. At the very least, you see development separated from production, with multiple instances of applications running on 1 or more production systems.

Using Remote Location, SuperVisor will submit processing requests to that remote iSeries then return the results to the local system. Different combinations of SuperVisor commands with multiple iSeries systems or Lpars simplifies system management for both business applications and different systems management tools.

SuperVisor Remote Location Central

There are two things that make SuperVisor Remote Location important

- Easy of use
- Ease of access

Easy to use

Your iSeries network configuration is build around a development system, with one or more production environment deployments. The extended network may include one or more High Availability systems supporting the different production environments.

Using SuperVisor Remote Location you can perform analysis or retrieve information from any of your distributed systems and return the results to wherever you are logged on.

Ease of access

SuperVisor uses Distributed Data Management “DDM” to access remote systems or logical partitions. This means that you can use existing communication configuration. There is no need to configure anything special.

By default, SuperVisor Remote Location uses the traditional DDM over SNA to access remote destinations. If you are configured to use Send Network File (SNDNETF) command, you are ready to use SuperVisor Remote Location.

If your configuration is different than this, it is a simple matter to create a SuperVisor Remote Location ID capable of accessing remote systems using any combination of DDM parameters.

What is a SuperVisor Remote Location

SuperVisor Remote Locate is not a communication configuration, but is instead simply the parameters you provide that will describe to SuperVisor how to create DDM files. SuperVisor commands will then retrieve this information and then use it to access each remote location defined by you.

You can create as many different SuperVisor Remote Locations as you like. It is possible to create multiple Remote Locations providing different access to a single system, but most often you will create a single SuperVisor Remote Location for each remote system or LPAR you wish to locate.

In Summary

When executing SuperVisor commands with Remote Location capability, each command will create temporary DDM files in QTEMP which will be used by the command to access information and programs at the specified remote system.

The Remote Location name is used to access the SuperVisor Remote Location table and return the parameters needed to successfully create a DDM link to the remote system.

Once the SuperVisor no longer needs that DDM link, the temporary DDM file is deleted.

If the Remote Location name specified on the SuperVisor command is not found on the SuperVisor Remote Location Table, SuperVisor will create the DDM file as DDM over SNA. If the DDM file fails to return any information, any errors are noted in the job log, but command processing will continue.

SuperVisor System Values

Keeping the definition of the Remote Location separated from the SuperVisor commands keeps both interfaces simple. Create as many different Remote Locations as you need. Later when using SuperVisor commands, you only need to specify the Remote Location name.

**SUPERVISOR
COMMON INTERFACES**

Understanding SQU, *LIST, and *EXTF

Before we get into the details of the different areas of support provided by SuperVisor SDM, this section will describe some important structures used within SuperVisor commands. These structures referred to over and over again as you review this User Guide and the SuperVisor Command Reference Manuals.

Explaining these features now should help you understand the information being explained in other areas of the SuperVisor documentation. It is also important to understand what these structures are in that this is what makes SuperVisor different than other tools available for the iSeries.

Advanced Structures within the SuperVisor program

Because of the architecture of SuperVisor commands, it is possible to perform complex analyses of AS/400 applications software. The results of a SuperVisor analysis can be used to perform a task. This is because of three common features of most SuperVisor commands. These are SuperVisor Structured Query Utility (SQU), SuperVisor *LIST, and the SuperVisor External File (*EXTF).

SQU is used to select objects to be processed by a command. SQU makes nearly all of the object data stored by OS/400 available. This means that almost any object data can be used to select objects.

A *LIST is a list of items, such as, programs, files, or field names. Most SuperVisor commands can create a list and can use a list to drive other commands.

An external file (*EXTF) is an outfile. Most SuperVisor commands can create an outfile. Typically an outfile contains a list of qualified names (name/library/type) and can be used to drive commands.

Structured Query Utility (SQU)

Many SuperVisor commands can use SQU to select and sort objects to be processed by the command. SQU makes nearly all of the object data stored by OS/400 available. This means that almost any object data can be used to select objects.

SQU is a function in SuperVisor that builds open query file statements. SuperVisor stores the SQU query statements internally. So, the same SQU statement can be used repeatedly to perform similar analyses.

Commands that can use SQU have two common parameters, SQU and SQUKEEP.

SMI SuperVisor Remote Location

```

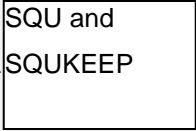
List Record Format Attr (LSTRFMTA)

Type choices, press Enter.

File, Generic* or *ALL: . . . . FILE
Object Library name: . . . .
File attribute . . . . . FILEATR      *PFLF
Record format name: . . . . . FORMAT  *ALL
                                + for more values
*LIST library search . . . . . LISTLIB *LIBL
System . . . . . SYSTEM               *LCL
Remote location name . . . . . RMTLOCNAM *LOCAL
Select Using EXCQRYS command: . SQU    *NO
Retain SQU Statement: . . . . . SQUKEEP ← *TEMP
Include carbon-copy files . . . CARBONCPY *NO
Print record format info. . . . RPT1     *YES
Print object information . . . . RPT2     *NO
Select Data for list: . . . . . SELECT  *FILE
List Name: . . . . . LIST              *NONE
List Name Description: . . . . . LSTDSC  *SAME

Printer Output Queue: . . . . . OUTQUE
Library name: . . . . .                *LIBL
Job Desc. for Submit Job: . . . . . JOB
Library name: . . . . .                *LIBL

```



The SQU Parameter

This parameter specifies whether and how SQU specifications will be entered.

The name of an existing SQU specification can be entered.

*NO specifies that SQU will not be used by the command.

*YES specifies that SQU will be used and that new SQU specifications will be entered interactively.

*SELECT specifies that existing SQU specifications will be used, and that a list of exiting specifications will be displayed.

The SQUKEEP parameter

This parameter specifies whether the SQU statement will be retained.

*PERM retains the specifications, *TEMP deletes them when the command is finished.

SQU Sort Specifications

SQU presents two sets of specifications, sort specifications and selection specifications. The upper panel of the display shows the possible sort fields. The cursor may be positioned in the upper panel and the roll keys may be used to view all of the available sort fields.

The lower panel of the display is for entering the sort specifications.

SMI SuperVisor Remote Location

```

EXCQRYS                               Sort Specification Selection          5/30/96
1 Field Name Type Length Dec Description
ODDCEN      A         1      Display century
ODDDAT      A         6      Display date: format- Job date format
ODDTIM      A         6      Display time
ODLBNM      A        10      Library
ODOBNM      A        10      Object
ODOBTP      A         8      Object type
                                     Possible Sort
                                     Fields
                                     More...

Type changes, press enter
1=Add 4=Delete
Opt Key Field ..... Selected Fields .....

- ODOBTP      Ascending
  ODCYY       Ascending
  ODCMM       Ascending
  ODCDD       Ascending
                                     Bottom

F3=Exit  F14=Continue

```

The option column (Opt) instructs the display to add or delete a sort field. Pressing enter enters the option and refreshes the display.

Command 14 presents the selection display.

SQU Selection Specifications

```

EXCQRYS                               List Object Description          5/30/96
1 Field Name Type Length Dec Description
ODDCEN      A         1      Display century
ODDDAT      A         6      Display date: format- Job date format
ODDTIM      A         6      Display time
ODLBNM      A        10      Library
ODOBNM      A        10      Object
ODOBTP      A         8      Object type
                                     More...

Type changes, press enter
1=Add 2=Change 3=Copy 4=Delete
Opt  Oper Field Name Rel  Selection Criteria
AND
IF   ODCRTU      EQ   JOEPGMR
                                     Selection Logic.
                                     Bottom

F3=Exit  F4=Fold/Unfold  F14=Continue

```

The lower panel of this display is for entering the selection specifications.

OPT Column

The Option (Opt) column instructs the display to (1) Add, (2) Change, (3) Copy or (4) Delete a specification line. Pressing enter enters the option.

Oper Column

The Operation (Oper) column specifies the relationship of the specification line to other specifications. The first operation must be IF. AND and OR may be used to add additional conditions.

Field Name Column

The Field Name column specifies the field that will be used in the operation.

Rel Column

The Relationship column (Rel) specifies the relationship between the field name and the selection criteria expression. Objects will be selected by the command when this relationship resolves to “true.” Possible entries are:

```
GT -- Greater Than
LT -- Less Than
EQ -- Equal To
GE -- Greater Than or Equal To
LE -- Less Than or Equal To
NE -- Not Equal To
NG -- Not Greater Than
NL -- Not Less Than
CT -- Contains Within
```

Selection Criteria Column

The selection criteria column is used to enter an expression that is resolved and compared to the entry in the field name column.

Expressions are similar to expressions specified for CL. An expression can contain values, field names, operators, and functions. Parentheses control the order of evaluation.

For a complete description of functions and expressions, see the OPNQRYF command in IBM's documentation.

Examples

Example 1

To select physical and logical files that are damaged the selection specifications

```
_ IF_ ODOBDM_____ EQ %value("1" "2")
_ AND_ ODOBAT_____ EQ PHY
_ OR_ ODOBDM_____ EQ %value("1" "2")
_ AND_ ODOBAT_____ EQ LGL
```

Example 2

To select objects which contain the word 'INVENTORY' in the text description.

```
_ IF_ ODOBTX_____ CT INVENTORY_____
```

Example 3

To select commands that have had the command defaults changed using the IBM Change Command Default (CHGCMDDFT) command.

```
_ IF_ ODAPAR_____ EQ CHGDFT_____
```

Example 4

To select objects that using defined fields. In this example the field RSTDATE is created. Then only objects restored on or after the March 23, 2001 are selected.

SMI SuperVisor Remote Location

```
DEFN RSTDATE_ = (odrcen || odryy || odrmm || odrdd) _____  
IF_ RSTDATE_ GE 1010325 _____
```

SuperVisor LIST (*LIST)

A *LIST is a list of names without attributes. A SuperVisor *List can be named to receive the results of an analysis performed by a SuperVisor command, or be used as input to control the processing of another SuperVisor command. Because a *LIST has no attributes, it can represent any kind of relationship.

Once a *LIST has been created it can then be used to control SuperVisor commands. It can be incorporated into CL programs, and can be used as a driver to execute any AS/400 command. There is no practical limit to the number of list that can be retained. So lists can be established for a myriad of purposes.

A list can be used to filter items processed or to repeat functions for each item on the *LIST.

What is a SuperVisor *LIST

When using SuperVisor *LIST, it is not necessary to know what it actually is. The SuperVisor commands handle all the low level actions for you. All you need to do is execute the SuperVisor commands create, use and manage SuperVisor *LIST's.

With that said, we have found that people want to know what it is under the covers. So to take out the mystery, a Personal SuperVisor *LIST is a physical file named ULSTP00. When you add a SuperVisor *LIST, you add a member to that file. The file ULSTP00 is an externally defined file with a single 10 character field.. To aid the processing there is also a logical view, ULSTL01 and a log file named ULSTP01.

Until you begin to use SuperVisor *LIST there is not much to see, because it is the SuperVisor programming that makes this really work for you. But to see more of the internals SuperVisor *LIST, you can feel free to execute the following command.

```
LSTDBD FLE(SUPERVISOR/ULSTP00)
```

This command will print a series of reports showing all there is to know about the internal workings of SuperVisor *LIST.

To really understand what you can do with SuperVisor *LIST, please review SuperVisor Assistant, which includes an entire menu of commands used to support and manipulate SuperVisor *LIST.

Managing a SuperVisor *LIST

A SuperVisor *LIST is managed in ways similar to a User Defined Option file used by PDM (Program Development Manager).

IBM provides a User Defined Option file QAUOOPT in the library QGPL. SuperVisor provides the SuperVisor *LIST, file (ULSTP00) in library SUPERVISOR.

You create a personal option file, by creating a copy of the file QAUOOPT into your personal library and then changing the PDM defaults to use the file in your library.

You create a personal SuperVisor *LIST using the CRTPERLST command to create a copy of the SuperVisor *LIST files in your personal library. You then place your personal library before SuperVisor in the jobs library list to access your personal SuperVisor *LIST.

Using a SuperVisor *LIST

For example, the command shown below will print a report showing only query programs that use one of the files on the *LIST MONTHEND. In this example the *LIST causes the command to omit queries that do not refer to files on the *LIST.

SMI SuperVisor Remote Location

```
XREFQRY QRY(*LIBL/*ALL) FILE(*LIST/MONTHEND)
```

A *LIST can also be used to repeat a function. As an example, the following command prints a program flow chart for each program in the *LIST MYLLIST.

```
LSTPGMFLW PGM(*LIST/MYLIST) TYPE(*PGM) LISTLIB(*ALLUSR)
```

The SuperVisor commands EXCCMDLST and WRKCMDLST enable allow any AS/400 command to be executed using a *LIST.

When the command accepts a list of items the keyword ULIST refers to the entire SuperVisor *LIST. For example, the following command only saves the file members in the SAVMEM *LIST for the UTMANUAL file.

```
EXCCMDLST CMD+  
savobj obj(UTMANUAL) lib(supervisor) +  
dev(tap01) filembr((UTMANUAL (ULIST))) +  
list(SAVMEM) lstdsc(*bypass)
```

A command can also be repeated for every item on a list. When the command uses each name on a *LIST the keyword UNAME refers to the list entry. The following command, for example, deletes each library named in the OLDLIB *LIST.

```
EXCCMDLST CMD(DLTLIB LIB(UNAME)) LIST(OLDLIB)
```

As in the previous examples, most SuperVisor commands can use a *LIST to select which items are processed. Most SuperVisor commands can also produce a *LIST as a product of an analysis. In the following example, the command compares source files and creates a *LIST of the source members that are different.

```
CMPSRCF SRCFLE1(*ALL/PRODSRC/QRPGSRC) +  
SRCFLE2(*ALL/PRODSRC9/QRPGSRC1) +  
LIST(NOCHANGE) LSTDSC('Version 8 to 9 +  
with no changes')
```

Since most SuperVisor commands can create a *LIST from an analysis, and then use the *LIST in another command, *LIST enables programmers to perform complex analyses with groups of items.

****LIST in SuperVisor Commands***

Generally, the SuperVisor commands can be controlled by a *LIST accept *LIST in the library portion of the object or member name parameter. These commands interpret the object or member parameter as a *LIST name. When the list name is not known the *SELECT keyword will prompt the user with a display showing existing *LIST's.

Commands that create a *LIST usually have a LIST (*LIST name) and LSTDSC (List Description) parameter. If the command can create different types of lists, a SELECT parameter determines which names are used to populate the *LIST.

If a *LIST already exists the *SAME keyword retains the existing list description, eliminating the need to re-enter. If the command creates a new *LIST the *BYPASS parameter prevents the command from prompting for a *LIST description. Ordinarily *BYPASS is used when the command is in a CL program.

****LIST Files and Personal *LISTS***

SuperVisor *LIST's are stored in the SuperVisor library in the ULSTP00, USTP01, and ULSTL01 files.

List entries can be up to 10 positions long.

ULSTP01 is A *LIST log file. This file contains one record for each *LIST. This file is used when *SELECT is entered for commands that read *LIST's

ULSTP00 contains *LIST entries. Each *LIST occupies a separate member. ULSTL01 is a logical file over ULSTP00.

A personal *LIST is one stored in a different library. However, in order to access a personal *LIST the library containing the personal list must appear before SuperVisor in the library list.

The Create Personal List (CRTPERLST) creates a personal *LIST in any library.

*Note: If SUPERVISOR is on the System Library List a personal *LIST cannot be created or accessed.*

External Files (*EXTF)

A SuperVisor external file is the IBM outfile created by the IBM supplied Display Object Description (DSPOBJD) command. SuperVisor has enhanced this output file by including a logical file. To the SuperVisor External File, is actually 2 files not just 1.

The reason we call these files External Files, because they are more than just output files. It is true that SuperVisor commands can output to an External File, but an External File can also be used as input to a SuperVisor command.

This makes the information in the file, external to the command processing.

External File Name Limitation

The SuperVisor External File (*EXTF) is actually a small data base made up of 2 files. A physical file and a logical view.

Limit Names to 8 characters

The physical file is the IBM Display Object Description (DSPOBJD) output file, Version V3R2. The logical file is created with an access path and all date fields subdivided into separate month, day and year fields.

Because of the need to create a logical file, the name of an External is limited to 8 characters and the logical file named is derived by adding the number 1 to the end of the name.

As an example, when the following command is executed.

```
CRTEXTF EXTF(QTEMP/WRKFLE) BASED_ON(LSTOBJD) TEXT('Work External File')
```

The following two files are created

- WRKFLE
- WRKFLE1

You can see this by executing the SuperVisor Display Dependent Logical Files (DSPDLF) command.

```
dspdlf qtemp/wrkfle
```

SMI SuperVisor Remote Location

```
DSPDLF          Display Dependent Logical Files

File . . . . . WRKFLE          Position . . . .
Library . . . . QTEMP          Work External File

Type options, press Enter.
1=Work object PDM      2=Work member PDM   3=Copy source
5=Display full file desc 6=Print          14=Compile ...

Opt  File      Library  S/O Format  Uniq Key fields
WRKFLE  QTEMP   N  QLIDOBJD
WRKFLE1 QTEMP   N  QLIDOBJD  N  ODOBNM.ODLBNM.ODOBTP.ODOBAT

Bottom

F3=Exit      F4=Prompt      F5=Refresh      F10=Command entry
F11=Next view F13=Repeat     F23=More options F24=More keys
```

Name cannot end using the number 1

A second limitation is that the name of the External File cannot end with the number 1. Since the logical file will be created using this name, using it for the External File will create duplicate name conflicts.

Note: Instead it is recommend that letters be used such as "a,b,c" be used to create a group files with similar names.

Managing SuperVisor External File

External Files are manage in ways similar to the way IBM output files are managed. By default, commands used to create External Files use the library SUPEREXTF as the default library where the file will be placed. Library SUPEREXTF is a default, not a requirement. Feel free to place External Files in any library you choose, that meets your development requirements.

Using SuperVisor External Files

Most SuperVisor analysis commands can optionally produce an outfile representing the analysis completed. And an external file can drive most SuperVisor analysis commands. For example, to analyze all the job descriptions used since 01/01/97, the below commands might be used. .

```
LSTOBJD OBJ(*ALLUSR/*ALL) TYPE(*JOBDD) FRUDAT(010197) +
OUTF(ACTJOBDD) OUTT('Active Job +
Descriptions') /* Place the names of all +
JOBDD's used since 01/01/97 in the file +
ACTJOBDD. */

ANZJOBDD JOBDESC(SUPEREXTF/ACTJOBDD) TYPE(*EXTF) /* +
Process the file ACTJOBDD and analyze the +
job descriptions found there. */
```

The first command analyzes the object type *JOBDD for usage after 01/01/97. The second prints a job description report driven by the external file.

SMI SuperVisor Remote Location

External files can also be used as a driver to repeat a command for every entry in the external file. The EXCCMDEXTF command executes an embedded command string using for each entry in the external file.

In the example below the external file, ILSSRC is created by the LSTOBJD command. A record is created for every for source file that begins with "ILS." The EXCCMEXTF repeats the SAVOBJ command for each source file. UNAME and ULIB values representing the object name and library from the external file.

```
LSTOBJD OBJ(ILS*/*ALL) TYPE(*FILE) SUBTYP(*SRCPF) +
OUTF(ILSSRC) OUTL(SUPEREXTF) OUTT('Source +
Files for ILS system') /* Find all source +
files in any library that begins with the +
letters ILS. */

EXCCMDEXTF EXTF(SUPEREXTF/ILSSRC) USRCMD(SAVOBJ +
OBJ(UNAME) LIB(ULIB) DEV(TAP01) +
OBJTYPE(*FILE) ENDOPT(*LEAVE)) LOG(*NO) +
/* Process the file ILSSRC and save each +
source file to tape. */

CHKTAP DEV(TAP01) ENDOPT(*UNLOAD) /* Unload tape */
DLTEXTF EXTF(SUPEREXTF/ILSSRC) /* Delete external +
file */
```

External Files in SuperVisor Commands

Generally, the SuperVisor commands can be controlled by an external file accept *EXTF in the TYPE or SUBTYPE parameters. These commands interpret the object library parameter as the external file/library. Commands that create an external file have OUTF, OUTL, and OUTT parameters, giving the file name, library, and description.

Commands that perform substitution use the following special values to represent data from an external file field.

Value	Field Name	Description
UNAME	ODOBNM	Object name
ULIB	ODLBNM	Object library
UTYPE	ODOBTP	Object type
UATTR	ODOBAT	Object attribute
USRCF	ODSRCF	Object Source File Reference
USRCL	ODSRCL	Object Source Library Reference
USRCM	ODSRCM	Object Source Member Reference
UTEXT	ODOBTX	Object Text

SMI SuperVisor Remote Location

External File Objects

By default, external files are stored in the SuperVisor library SUPEREXTF.

Object Description external files are created using the model OBJDP00 in the library SUPERVISOR. OBJDP00 has the logical file OBJDP01 that is keyed by object name, object library, type and attribute.

For a complete description of this file, enter the following command.

```
LSTDBD      FILE (SUPERVISOR/OBJDP00)  SUBTYP (*PF)
```

Object Authority external files are created from the model LSTAUTA in the library SUPERVISOR.

External File Details

Now we have an External File, lets look at the file itself and see what we have.

First use the SuperVisor Display Dependent Logical File (DSPDLF) command to see the data base relations.

```
Type command, press Enter.  
==> dspdlf superextf/spoobj
```

The following screen shows the two files, including the keyed logical file.

```
DSPDLF          Display Dependent Logical Files  
  
File . . . . . SPOOBJ          Position . . . . .  
Library . . . . SUPEREXTF      Library SPOOBJ objects  
  
Type options, press Enter.  
1=Work object PDM      2=Work member PDM      3=Copy source  
5=Display full file desc 6=Print          14=Compile ...  
  
Opt  File      Library  S/O  Format  Uniq  Key fields  
     SPOOBJ    SUPEREXTF  N    QLIDOBJD  
5    SPOOBJ1   SUPEREXTF  N    QLIDOBJD  N  ODOBNM.ODLBNM.ODOBTP.ODOBAT  
  
Bottom  
  
F3=Exit      F4=Prompt      F5=Refresh      F10=Command entry  
F11=Next view  F13=Repeat     F23=More options F24=More keys
```

Select the logical file to view the detailed full file field descriptions.

SMI SuperVisor Remote Location

```

DSPFFFD          Display Full File Field Description

File . . . . . SPOOBJ1      Library SPOOBJ objects
Library . . . . SUPEREXTF   Created by SuperVISOR(tm) system
                          Position . . . .

Type options, press Enter.
1=Record format  2=Keys      5=Select/Omit   7=Join   6=Print field usage
22=Write list entry
Format          Field
Opt Name        Name          Position Text
QLIDOBJD       ODDCEN         1      Display century
QLIDOBJD       ODDDAT         2      Display date: format- Job date format
QLIDOBJD       ODDTIM         8      Display time
QLIDOBJD       ODLBNM        14     Library
QLIDOBJD       ODOBNM        24     Object
QLIDOBJD       ODOBTP        34     Object type
QLIDOBJD       ODOBAT        42     Object attribute
QLIDOBJD       ODOBFR        52     Storage freed: 0-not freed,1-freed
                                          More...

F3=Exit          F4=Prompt          F5=Refresh          F10=Command entry
F11=Next view    F12=Cancel          F23=More options   F24=More keys
  
```

Roll to the next page shows where the create date has been separated into individual date fields.

```

DSPFFFD          Display Full File Field Description

File . . . . . SPOOBJ1      Library SPOOBJ objects
Library . . . . SUPEREXTF   Created by SuperVISOR(tm) system
                          Position . . . .

Type options, press Enter.
1=Record format  2=Keys      5=Select/Omit   7=Join   6=Print field usage
22=Write list entry
Format          Field
Opt Name        Name          Position Text
QLIDOBJD       ODOBSZ         53     Object size
QLIDOBJD       ODOBTX         59     Text description
QLIDOBJD       ODOBLLK        109    Object locked: 0-not locked,1-locked
QLIDOBJD       ODOBDM        110    Object damaged: 0-not damaged,1-full,
QLIDOBJD       ODDCEN        111    Creation century
QLIDOBJD       ODCDAT        112    Creation date: format- mmddy
QLIDOBJD       ODCMM         118    Create Month
QLIDOBJD       ODCDD         120    Create Day
                                          More...

F3=Exit          F4=Prompt          F5=Refresh          F10=Command entry
F11=Next view    F12=Cancel          F23=More options   F24=More keys
  
```

All date fields in the file have had this change made, to allow for easier sorting and record selection by date, when compared to the IBM supplied version of this output file.

More on building External Files

Where appropriate, SuperVisor commands support output to an External File. There are many reasons for this. The iSeries computer system is an object based system. Except for the IFS, everything is stored as an object. An object description file is the perfect format for iSeries system analysis and management.

Is this possible?

Because the records in an External File, represent objects, External Files allow for analysis which would other wise be impossible.

In today's data center, you will almost always see an iSeries network, a development system supporting one or more production systems. For complete analysis of a problem, you need information from multiple computers. The following example will illustrate this.

Task:

The project at hand is to locate any programs in the library SPOOBJ on the development system where the source member is not found. You need to report the number of programs and what is the impact. If the program is no longer used in production, it will have little or no impact.

Issue1:

The system was created using CRTBNDxxx commands. Using the IBM Display Object Description (DSPOBJ) command does not include source references.

Issue2:

Program usage is not on the development iSeries, but on the production boxes. You need to process objects in the production system, then look for source code on the development system.

Resolution:

Using SuperVisor commands and the External File, this analysis can be performed without any programming required.

The analysis will need to print only active programs which the program source reference is not found.

All processing will be done on the development system

- Step1 Retrieve object descriptions to an External File from production system, for all programs used since the first of the year, or if they were created in the last 30 days.
- Step2 Create an External File with all programs in the development system, if the source file/member for the program is not found.
- Step3 Print an exception report

SMI SuperVisor Remote Location

Step 1:

Use the Convert Object to External File (CVROBJEXTF) command to retrieve the object description for all active programs from the production iSeres. *Note: Assume today is 7/29/2004,*

```
Type command, press Enter.  
==> CVROBJEXTF RMTOBJ(PROD01/SPOOBJ/*ALL) RMTTYPE(*PGM) RMTTCDATE(070104) RMTF  
RUDAT(010104) OUTF(SPOPROD) OUTL(SUPEREXTF) OUTMROPT(*REPLACE) OUTT('Active Pro  
duction Programs')
```

Step 2:

Use the List Object Description (LSTOBJD) command to process all programs in the library SPOOBJ on the development system, if the source reference is not found.

```
Type command, press Enter.  
==> LSTOBJD OBJ(SPOOBJ/*ALL) TYPE(*PGM) RPT1(*NO) CHKSRC(*YES) OUTF(SPODEV) OU  
TL(SUPEREXTF) OUTMROPT(*REPLACE) OUTT('Development Programs missing source')
```

The List Object Description (LSTOBJD) command checks to see if the program being processed was created using a CRTBNDxxx command. If true, the program information is automatically processed to a deeper level, to locate the source code referenced used to create the program.

ILE programs created from multiple modules is also supported.

Step 3:

Produce a listing of active programs with missing source references.

```
Type command, press Enter.  
==> CMPOBJD OBJ(SUPEREXTF/SPODEV SUPEREXTF/SPOPROD) TYPE(*EXTF) MISSING(*NONE)  
OUTF(PRODSS) OUTL(SUPEREXTF) OUTT('Active Production Program with Source Issu  
es')
```

There are two important parameters that make this command work.

First is the special value, *EXTF used as the object type "TYPE". This instructs SuperVisor commands that the object being processed is an External File. The command will process records in the file, as if they were objects.

SMI SuperVisor Remote Location

Second is the Unique Item Specification “MISSING” parameter. Using the special value *NONE instructs the command to select only object descriptions that appear more than once. To qualify, there must be both a development program with the missing source and active program from production.

Note: If the External File does not exist, the SuperVisor command will create it. It is not necessary to create the External File fist. However, you will find the creating it first then using the file does make the programs easier to follow.

Back to basics

The SuperVisor External File is one of the most used structures in the SuperVisor system. Almost all commands support this format in one way or another.

In addition, there is a group of commands written for nothing other than to support this file type.

To review these commands and explore there capabilities, enter the following command.

```
Type command, press Enter.  
==> supervisor extf
```

SMI SuperVisor Remote Location

Will display the SuperVisor External File menu.

```
EXTF                               SuperVisor External File Menu                               System: SMI

Select one of the following.

Manage External File

    3. Create External File           CRTEXTF
    4. Edit External File             EDTEXTF
    5. Delete External File          DLTEXTF
    6. Copy External File             CPYEXTF
    7. Match Two External Files       MATEXTF
    8. Execute Command using External File EXCCMDEXTF

    10. Print External File           LSTOBJD

Manage External File Entries

    15. Write External File Entry     WRTEXTFE
    16. Delete External File Entry    DLTEXTFE
    17. Change External File Entry    CHGEXTFE
    18. Check External File Entry     CHKEXTFE

Convert Data to External File

    27. Convert Based on File to External File CVTBOFEXTF
    28. Convert Dependant File to External File CVTDBREXTF
    29. Convert Member list to External File CVTMBREXTF
    30. Convert Binding Directory to *EXTF CVTBNDXTF
    31. Convert SuperVisor *LIST to *EXTF CVTLSTEXTF
    32. Convert Object Description to *EXTF CVTOBJEXTF
    33. Convert Object Lock to *EXTF CVTLCKEXTF

                                                                 Bottom

Selection
====>
F3=Exit      F5=Next Menu- *SELECT    F6=Display Message  F10=Command entry
F12=Cancel  F14=Work with jobs             F16=Main menu       F18=Work with output
F21=Print menu help
```

SuperVisor Remote Location

The most significant change made in the SuperVisor V5R1M14 release deals with the ability SuperVisor has to process information on remote systems and return the results to the local system. In today's modern iSeries networks, the use of multiple iSeries systems or Logical Partitions (Lpars) is no longer the exception. At the very least, you see development separated from production, with multiple instances of applications running on 1 or more production systems.

Note: This step is necessary only when using SuperVisor on multiple iSeries or Logical Partitions.

Using Remote Location, SuperVisor will submit processing requests to that remote iSeries then return the results to the local system. Different combinations of SuperVisor commands with multiple iSeries systems or Lpars simplifies system management for both business applications and different systems management tools.

SuperVisor Remote Location Central

There are two things that make SuperVisor Remote Location important

- Easy of use
- Ease of access

Easy to use

Your iSeries network configuration is build around a development system, with one or more production environment deployments. The extended network may include one or more High Availability systems supporting the different production environments.

Using SuperVisor Remote Location you can perform analysis or retrieve information from any of your distributed systems and return the results to wherever you are logged on.

Ease of access

SuperVisor uses Distributed Data Management "DDM" to access remote systems or logical partitions. This means that you can use existing communication configuration. There is no need to configure anything special.

By default, SuperVisor Remote Location uses the traditional DDM over SNA to access remote destinations. If you are configured to use Send Network File (SNDNETF) command, you are ready to use SuperVisor Remote Location.

If your configuration is different than this, it is a simple matter to create a SuperVisor Remote Location ID capable of accessing remote systems using any combination of DDM parameters.

SMI SuperVisor Remote Location

What is a SuperVisor Remote Location

SuperVisor Remote Locate is not a communication configuration, but is instead simply the parameters you provide that will describe to SuperVisor how to create DDM files. SuperVisor commands will then retrieve this information and then use it to access each remote location defined by you.

You can create as many different SuperVisor Remote Locations as you like. It is possible to create multiple Remote Locations providing different access to a single system, but most often you will create a single SuperVisor Remote Location for each remote system or LPAR you wish to locate.

In Summary

When executing SuperVisor commands with Remote Location capability, each command will create temporary DDM files in QTEMP which will be used by the command to access information and programs at the specified remote system.

The Remote Location name is used to access the SuperVisor Remote Location table and return the parameters needed to successfully create a DDM link to the remote system.

Once the SuperVisor no longer needs that DDM link, the temporary DDM file is deleted.

If the Remote Location name specified on the SuperVisor command is not found on the SuperVisor Remote Location Table, SuperVisor will create the DDM file as DDM over SNA. If the DDM file fails to return any information, any errors are noted in the job log, but command processing will continue.

Keeping the definition of the Remote Location separated from the SuperVisor commands keeps both interfaces simple. Create as many different Remote Locations as you need. Later when using SuperVisor commands, you only need to specify the Remote Location name.

Edit Remote Location (EDTRMTLOC) Command

Use this command to create, change or delete SuperVisor Remote Location entries. There is no limit to the number of entries or attributes found on the different SuperVisor Remote Locations.

Execute the following command to access the Edit Remote Location main screen”

```
Selection or command  
====> edtrmtloc
```

This will display the Edit Remote Locations main menu.

SMI SuperVisor Remote Location

```
RMT010                                Edit with Remote Locations                11/21/05
                                          16:30:13

Type choice, press enter.

1. Add Remote Location
2. Change Remote Location
4. Delete Remote Location
5. Verify Remote Location

Remote Location:                        Description:

F3=Exit  F4=Prompt  F6=Add remote location  F7=Change remote location
F10=Command line  F23=Delete remote location
```

New locations can be created by pressing F6 “Add remote location”.

```
RMT011                                Edit with Remote Locations                11/21/05
SUPERV5                                Add Project                               16:33:23

Type changes, press enter.

Remote
Location: PROD01      Description: Production 01

No Change description as needed, press enter to continue
to DDM file specification

F3=Exit  F12=Cancel  F10=Command Line
```

The first part of the Remote Location definition, is the name and description. Key your information then press enter to continue.

SMI SuperVisor Remote Location

```
RMT011                      Edit with Remote Locations          11/20/05
SUPERV5                      Add Project                        16:37:26

Type changes, press enter.

.....
:  PROD01 Confirm Add      :
:  Select one of the following, press enter. :
:    1. Continue          :
:    2. Cancel            :
:  F3=Exit                :
:                          :
:.....

F3=Exit  F12=Cancel  F10=Command Line
```

Confirm the creation of the Remote Location, then press enter again to view the Create DDM file prompts.

```
                          Create DDM File (CRTDDMF)

Type choices, press Enter.

Remote location:
Name or address . . . . .

Type . . . . . *SNA      *SNA, *IP

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys
```

At this point, enter the keywords you need to access the remote location using Distributed Data Management (DDM). To access the remote host SMI5 using TCP/IP make the following changes.

SMI SuperVisor Remote Location

```
                Create DDM File (CRTDDMF)

Type choices, press Enter.

Remote location:
  Name or address . . . . . smi5

Type . . . . . *ip          *SNA, *IP

                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display   F24=More keys
```

```
                Create DDM File (CRTDDMF)

Type choices, press Enter.

Remote location:
  Name or address . . . . . > smi5

Type . . . . . > *IP          *SNA, *IP
Text 'description' . . . . . Software Management, Inc. i5

                                                    Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display   F24=More keys
```

We are using the iSeries command prompt, so this works as it does for all iSeries commands. Press enter again. This completes input for the CRTDDMF command.

SMI SuperVisor Remote Location

```
RMT010                                Edit with Remote Locations                                11/20/05
                                                                                               16:42:57

Type choice, press enter.

1.  Add Remote Location
2.  Change Remote Location
4.  Delete Remote Location
5.  Verify Remote Location

Remote Location:  PROD01      Description:  Production 01

F3=Exit  F4=Prompt  F6=Add remote location  F7=Change remote location
F10=Command line  F23=Delete remote location
Remote Location PROD01 verified
Remote Location PROD01 added
```

The Remote Location PROD01 is now created. In addition, this connection was created and verified. If problems existed with the DDM attributes, you will be notified right away.

To explore this in more detail select PROD01 using option 2 “Change Remote Location”, then press enter.

```
RMT011                                Edit with Remote Locations                                11/20/05
SUPERV5                                Change Remote Location                                16:45:59

Type changes, press enter.

Remote
Location:  PROD01      Description:  Production 01

F3=Exit  F12=Cancel  F10=Command Line
```

Press enter to continue

SMI SuperVisor Remote Location

```
RMT011                               Edit with Remote Locations          11/20/05
SUPERV5                               Change Remote Location             16:45:59

Type changes, press enter.

.....
:   PROD01 Confirm Change              :
:   Select one of the following, press enter. :
:   1 1. Continue                      :
:   2. Cancel                          :
:   F3=Exit                            :
:                                       :
:.....

F3=Exit  F12=Cancel  F10=Command Line
```

Press enter again to provide information confirmation.

```
                                Create DDM File (CRTDDMF)

Type choices, press Enter.

Remote location:
Name or address . . . . . > smi5

Type . . . . . > *IP                *SNA, *IP
Text 'description' . . . . . > 'Software Management, Inc. i5'

                                Bottom
F3=Exit  F4=Prompt  F5=Refresh  F10=Additional parameters  F12=Cancel
F13=How to use this display  F24=More keys
```

At this point, you see the DDM attributes for Remote Location PROD01. If the connection required additional DDM attributes they can also be entered. To view additional parameters press F10.

SMI SuperVisor Remote Location

```
                Create DDM File (CRTDDMF)

Type choices, press Enter.

Remote location:
  Name or address . . . . . > smi5

  Type . . . . . > *IP          *SNA, *IP
  Text 'description' . . . . . > 'Software Management, Inc. i5'

                Additional Parameters

Device:
  APPC device description . . . *LOC          Name, *LOC
  Local location . . . . . *LOC          Name, *LOC, *NETATR
  Mode . . . . . *NETATR          Name, *NETATR
  Remote network identifier . . . *LOC          Name, *LOC, *NETATR, *NONE
                                          More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

This displays the additional parameters valid for *IP connections. Use the roll key to view more keywords on the next page.

```
                Create DDM File (CRTDDMF)

Type choices, press Enter.

Port number . . . . . *DRDA          *DRDA, 1-65535
Access method:
  Remote file attribute . . . . *RMTFILE          *RMTFILE, *COMBINED...
  Local access method . . . . . *BOTH, *RANDOM, *SEQUENTIAL
Share open data path . . . . . *NO          *NO, *YES
Protected conversation . . . . . *NO          *NO, *YES
Record format level check . . . *RMTFILE          *RMTFILE, *NO
Authority . . . . . *LIBCRTAUT          Name, *LIBCRTAUT, *ALL...
Replace file . . . . . *YES          *YES, *NO

                                          Bottom
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Summary

The SuperVisor Remote Location is designed to be a flexible, yet simple interface to allow processing to take place across multiple iSeries systems in an easy to use and understand way.

Remote Location Commands

Following is a list of the SuperVisor commands which include the Remote Location parameter.

SMI SuperVisor Remote Location

Command	Description
CMPAUTL	ANALYSIS - Compare Authorization List Objects
CMPDTAA	ANALYSIS - Compare Data Area Attributes
CMPDTAQA	ANALYSIS - Compare Data Queue Attributes
CMPFFLD	ANALYSIS - Compare File Fields
CMPJOB	ANALYSIS - Compare Job Description
CMPKFLD	ANALYSIS - Compare Key Fields
CMPMBRL	ANALYSIS - Compare Member List
CMPMODPGM	ANALYSIS - Compare Module Programs
CMPOBJD	ANALYSIS - Compare Object Description
CMPPGMA	ANALYSIS - Compare Program Attributes
CMPPGMFLE	ANALYSIS - Compare Program Files
CMPPGMMOD	ANALYSIS - Compare Program Modules
CMPPGMREF	ANALYSIS - Compare Program References
CMPPTFI	ANALYSIS - Compare PTF Information
CMPRFLD	ANALYSIS - Compare Ref.Fields
CMPRFMT	ANALYSIS - Compare Record Format
CMPSBSD	ANALYSIS - Compare Subsystem Description
CMPSFLD	ANALYSIS - Compare Select/Omit Fields
CMPSFWRSC	ANALYSIS - Compare Software Resources
CMPSRVA	ANALYSIS - Compare Service Program Attributes
CMPSYSI	ANALYSIS - Compare System Information
CMPUSRP	ANALYSIS - Compare User Profiles
CMPUSRSA	ANALYSIS - Compare User Space Attributes
CPYCATE	TABLE SERVICE - Copy Category Entry
CPYCDEE	TABLE SERVICE - Copy Code Entry
CPYLIST	MANAGEMENT - Copy list
CPYSYSE	TABLE SERVICE - Copy System Entry
CPYTBLE	TABLE SERVICE - Copy Table Entry
CVTACTS	ANALYSIS - Convert Active Subsystem
CVTAUTL	ANALYSIS - Convert Authorization List Objects
CVTDTAA	ANALYSIS - Convert Data Area Attributes
CVTDTAD	ANALYSIS - Convert Data Area Data
CVTDTAQA	ANALYSIS - Convert Data Queue Attributes
CVTIBMJS	ANALYSIS - Convert IBM Job Schedule Jobs
CVTJOBAUD	ANALYSIS - Convert Job Audit Data

SMI SuperVisor Remote Location

Command	Description
CVTJOBBD	ANALYSIS - Convert Job Description
CVTJOBI	MANAGEMENT - Convert Job Information
CVTOBJEXTF	ANALYSIS - Convert Object Description
CVTOBJLCK	ANALYSIS - Convert Object Lock
CVTOQD	ANALYSIS - Convert Output Queue Data
CVTOUTQD	ANALYSIS - Convert Output Queue Description
CVTPGMA	ANALYSIS - Convert Program Attributes
CVTPGMFLE	ANALYSIS - Convert Program Files
CVTPGMMOD	ANALYSIS - Convert Program Modules
CVTPGMREF	ANALYSIS - Convert Program References
CVTPTFI	ANALYSIS - Convert PTF Information
CVTRBTJS	ANALYSIS - Convert ROBOT Job Scheduler Jobs
CVTSBSD	ANALYSIS - Convert Subsystem Description
CVTSRVA	ANALYSIS - Convert Service Program Attributes
CVTSUBPRC	ANALYSIS - Convert Subprocedure references
CVTSYSI	ANALYSIS - Convert System Information
CVTUSRSA	ANALYSIS - Convert User Space Attributes
EDTCDEFLE	TABLE SERVICE - Edit Code File
EDTRMTLOC	ANALYSIS - Edit Remote Location DDM Attributes
INQCDEFLE	TABLE SERVICE - Inquiry Code File
LSTDUPOBJ	ANALYSIS - List Duplicate Objects
LSTLVLCHK	ANALYSIS - List Level Check Programs
LSTMISOBJ	ANALYSIS - List Missing Objects
LSTRFMTA	ANALYSIS - List Record Format Attributes
PRTCDEFLE	TABLE SERVICE - Print Code File Manual
RTNCATE	TABLE SERVICE - Return Category Entry
RTNCDEE	TABLE SERVICE - Return Code Entry
RTNSYSE	TABLE SERVICE - Return System Entry
RTNTBLE	TABLE SERVICE - Return Table Entry
RTVCATE	TABLE SERVICE - Retrieve Category Entry
RTVCATWIN	TABLE SERVICE - Retrieve Category in a Window
RTVCDEE	TABLE SERVICE - Retrieve Code Entry
RTVCDEWIN	TABLE SERVICE - Retrieve Code Through a Window
RTVCMDFSRC	ANALYSIS - Retrieve Command File Source
RTVSYSE	TABLE SERVICE - Retrieve System Entry

SMI SuperVisor Remote Location

Command	Description
RTVSYSWIN	TABLE SERVICE - Retrieve System in a Window
RTVTBLE	TABLE SERVICE - Retrieve Table Entry
RTVTBLWIN	TABLE SERVICE - Retrieve Table in a Window