

CL/SuperSession for z/OS



Best Practices Migrating to CL/SuperSession for z/OS

Version 2.1

CL/SuperSession for z/OS



Best Practices Migrating to CL/SuperSession for z/OS

Version 2.1

Note

Before using this information and the product it supports, read the general information under “Notices” on page 45.

Second edition (September 2017)

This edition applies to Version 2, Release 1 of IBM CL/SuperSession for z/OS (program number 5601-B28), and to all subsequent releases and modifications until otherwise indicated in new editions. Make sure you are using the correct edition for the level of the product.

© Copyright IBM Corporation 2016, 2017.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Figures	v	Preparing to migrate data into security systems . . .	18
Tables	vii	Migrating data into security systems	19
Preface	ix	System definitions	21
About this document	ix	Application definitions	21
Intended audience	ix	Pool definition in KLS\$VSMS	21
Accessibility	ix	Global and individual profiles	23
How to send your comments to IBM	ix	Chapter 5. General recommendations	25
Chapter 1. Introduction	1	User data and variables	25
Migration assistance	1	General VTAM suggestions	25
Chapter 2. Session management and migration concepts	3	ACBs	26
Organizational requirements	4	Administration tasks	26
Technical requirements	4	Trigger usage	26
Chapter 3. Preparing for migration	5	Manual activities	27
Organizational preparation	5	Chapter 6. Migration summary	31
Technical preparation	7	Appendix. Migration checklists	33
Auditing and control requirements	12	Project workflow checklist	33
Developing the migration plan	12	Customer checklist - general questions	38
Chapter 4. Migration tooling	13	Exit checklist	41
The migration process	14	Bibliography	43
First stage scripts	14	Related Publications	43
Manual update	15	IBM Publications	43
Second stage scripts	15	How to get IBM Publications	43
Manual tasks	16	Notices	45
Working with the spreadsheet	16	Trademarks	46
Function Groups and Groups	17	Index	47

Figures

1. Groups of people affected by migration, and influences	5	9. Example PERMIT statements for RACF	20
2. Overview of migration tooling	14	10. Example CONNECT statements for RACF	20
3. Sample spreadsheet data following session manager database extract	16	11. Example Application definition statement generated by second stage scripts	21
4. Continuation of sample spreadsheet data	17	12. Example VSM DEF definition statement generated by second stage scripts	22
5. Overview of user connections to Function Groups and Groups.	17	13. A sample Initial Dialog to VPSSCUA application.	28
6. Example ADDGROUP command using Universal Groups.	18	14. A sample Termination Dialog for a CICS application.	28
7. Example resource definitions for RACF	20	15. RACF macro required if an own class is to be used	29
8. Example ADD GROUP statements for RACF	20		

Tables

1.	First recommended naming convention	17	5.	Example exits from IBM NetView Access Services	41
2.	Second recommended naming convention	18	6.	Example exits from IBM Session Manager for z/OS.	42
3.	Key items to address for the migration project workflow	33			
4.	Customer checklist with general questions	39			

Preface

About this document

This best practices guide discusses the concepts and tasks which must be understood and applied when migrating from an existing session management product to IBM® CL/SuperSession for z/OS®.

The objectives of this best practices guide are:

- To describe the major activities that take place during migration to CL/SuperSession
- To provide a checklist for use when creating and executing a successful migration plan
- To study specific aspects of the migration activities in detail

Intended audience

This best practices guide is intended for architects, systems programmers, analysts and programmers wanting to understand the performance characteristics of CL/SuperSession.

The information is not intended as the specification of any programming interface that is provided by CL/SuperSession. It is assumed that the reader is familiar with the concepts and operation of CL/SuperSession.

Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. The major accessibility features in CL/SuperSession enable users to:

- Use assistive technologies such as screen readers and screen magnifier software. Consult the assistive technology documentation for specific information when using it to access z/OS interfaces.
- Customize display attributes such as color, contrast, and font size.
- Operate specific or equivalent features using only the keyboard.

You can perform most tasks required to set up and run CL/SuperSession using a 3270 emulator logged on to TSO.

IBM Personal Communications for Windows provides 3270 emulation with accessibility features for people with disabilities. You can use this product to provide the accessibility features you need.

How to send your comments to IBM

Your feedback is important in helping us to provide accurate, high-quality information. If you have comments about this information or any other IBM CL/SuperSession for z/OS documentation, contact us in one of these ways:

- Use the Online Readers' Comments Form at www.ibm.com/software/awdtools/rcf/.

- Send your comments to the following address: comments@us.ibm.com. Be sure to include the name of the document, the publication number, the version of IBM CL/SuperSession for z/OS, and, if applicable, the specific location (for example, the page number or section heading) of the text that you are commenting on. When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way that IBM believes appropriate without incurring any obligation to you.

Chapter 1. Introduction

CL/SuperSession is a VTAM® application which is reachable by 3270 Terminal or by Terminal emulation such as IBM Personal Communications, offering secure and user-friendly access to multiple z/OS systems from a single 3270 terminal or equivalent. CL/SuperSession provides users with a single secure sign on, and allows controlled access to all their applications from multiple concurrent virtual sessions, using a wide range of tools and facilities.

The terminal session can simultaneously connect to and navigate between multiple VTAM applications. CL/SuperSession delivers greater user productivity, enhanced system usage, increased security, reduced costs, passphrase support, profile and user clean-up tools, and improved communication.

The information in this best practices guide discusses migration to IBM CL/SuperSession for z/OS V2.1 and later. Any version-specific information is clearly identified.

Migration assistance

IBM consultants are available to aid your migration to IBM CL/SuperSession for z/OS, as discussed in the body of this best practices guide. They can provide guidance on how best to use the available tools, given specific requirements.

IBM consultants are highly skilled people with broad architectural knowledge, deep technical experience, and who may access lead developers and product support personnel across IBM. The teams provide worldwide assistance and support for IBM products to customers and partners alike, through a rich portfolio of bespoke and customized services offerings.

By engaging IBM consultants, you not only have access to some of the deepest product consulting and training skills in the world, you also have a team dedicated to help ensure the success of your migration to CL/SuperSession. IBM Consultants have the resources and knowledge to help reduce the time required to complete the migration effort.

To learn how this team may assist you to migrate to IBM CL/SuperSession for z/OS efficiently, please access:

<http://www.ibm.com/services>

Chapter 2. Session management and migration concepts

In this section, we discuss session management, and explain why an organization may need to migrate to IBM CL/SuperSession for z/OS.

A modern business environment requires users to move smoothly and quickly between tasks, switching between applications frequently throughout the day. The applications may be installed on their local workstation, or on remote systems. For example, a user may be processing a customer order. They may need to check the customer details and current credit status, ensure that sufficient items are in stock, and schedule final delivery. Each of these tasks could be carried out using a different application. As the user moves between the various tasks and applications, they must keep track of the complete package of work.

A modern enterprise will have many applications running on multiple IBM z/OS systems. These applications are accessed through sessions from a variety of systems, including personal computer workstations and dedicated 3270 terminals. Application access must be planned and managed to allow easy and secure use.

Session management tools provide transparent access to business applications using multiple, concurrent virtual sessions. These sessions may be selected by a small number of keystrokes from a single, password-protected menu. This enables users to access the information they need quickly and easily, without having to know where the applications are actually installed.

CL/Supersession is one of the leading session management products, providing secure and reliable access to multiple applications from a single 3270 terminal session. CL/SuperSession provides flexible user access to applications and the ability to create customized menus for users and groups, logon screens, network news and more. HAO support allows two or more copies of CL/SuperSession to maintain synchronized profile databases. Multi-point HelpDesk provides enhanced, single-point control for multiple copies of CL/SuperSession.

Other session management products have been available and deployed by businesses, including some from IBM such as IBM Session Manager for z/OS, and Tivoli® NetView® Access Services. In each case, the tools had specific strengths which matched the customer requirements at the time. As business requirements change and develop, the existing tool may become less suitable.

There may be other reasons for migration. Customers may believe that the current solution is not cost-effective, or that projected costs make the product too expensive to maintain. Support issues may be a problem, particularly if the support is not available in the customer's natural language.

The business is faced with the problem of migrating to a new product which provides the essential new performance and functional benefits, while fulfilling the original requirements. This migration has to be performed in a complex distributed environment, where mission-critical tasks are performed by large-numbers of users. Access to the applications through sessions cannot be interrupted at any time.

Planning a migration to CL/SuperSession is a very important process. It must take account of technical and organizational requirements. It has specific steps where migration tooling will be helpful or even essential.

Organizational requirements

The success of the migration plan depends directly on the attention given to organizational requirements. System users and customers should be consulted. Their representatives should participate in the migration planning tasks. The purpose is to ensure that user and customer requirements are addressed in the plan. For users, this will help ensure smooth migration of the everyday tasks they perform with the new system.

Formal business controls affect the migration process. Internal processes and external obligations such as legal compliance must be maintained before, during and after the migration. An important example is ensuring audit compliance throughout. Regional variations may complicate migration within a global enterprise.

Customers must not be adversely affected by the migration process. The migration plan must check to ensure that any impact prevented, or at least minimized and clearly notified in advance.

Existing business processes may depend on the existing session management tools; these must be identified and reviewed to determine how they will be affected by the new systems.

Any large organization will have different and sometimes conflicting views and expectation for the migration process. If groups within the organization cannot agree on the details, the migration is unlikely to succeed. Sometimes, a department or even an individual may block a migration. It is essential that executive-level support for the migration be confirmed, to help resolve any questions that may arise.

Technical requirements

A successful migration plan will prepare for new functionality and provide support for the existing capabilities. This means addressing technical administration requirements, to ensure that technical and support staff are fully involved and prepared for migration, and are trained in the new system configuration.

A migration is also an opportunity to review the existing infrastructure. Old components may not be required any more, or could be implemented more effectively in the new system.

An essential technical consideration is to ensure that any customized applications are identified and assessed. These may depend on the current session manager product. The successful migration plan must ensure these dependencies are identified and addressed.

Having discussed the migration concepts, in the next section we describe the preparation tasks.

Chapter 3. Preparing for migration

In this section, we outline the major preparation activities for migration to IBM CL/SuperSession for z/OS. We consider organizational, technical and auditing issues. We discuss the process of creating a migration project plan, highlighting areas of special importance which may differ from other project plans.

Organizational preparation

Migration will affect several groups within the organization. There may be many technical reasons for the migration, but the use of a session management product to perform tasks is a business-driven decision. Changing the session management product will therefore affect the way in which the business imperatives are delivered.

Figure 1 shows some of the groups affected, and provides examples of the influences on those groups.

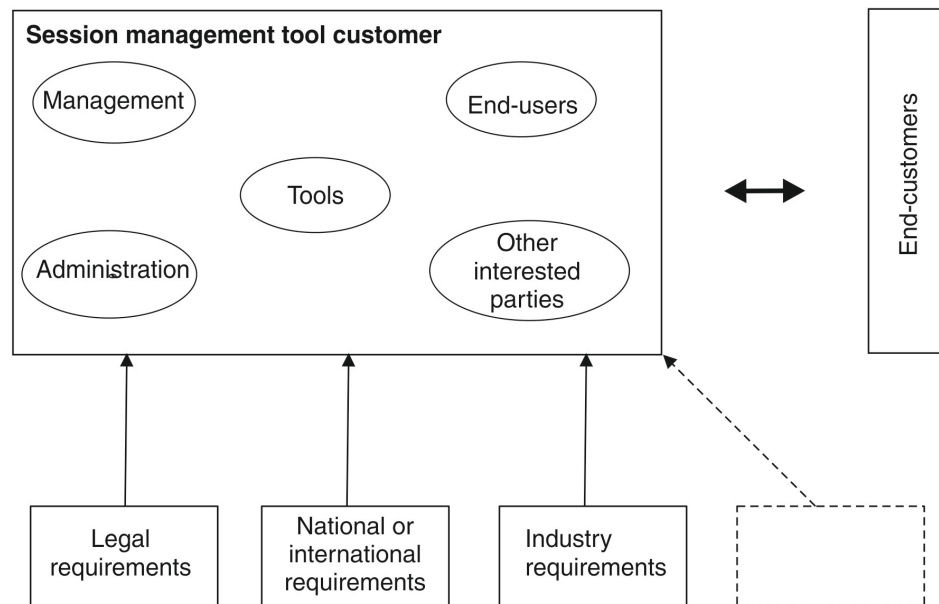


Figure 1. Groups of people affected by migration, and influences

A report is required for each group, describing how the migration to IBM CL/SuperSession for z/OS will affect them, and how the effects will be addressed. There are three possible outcome scenarios:

- There is no significant organizational impact arising from the migration to CL/SuperSession. The report document will note this as a low impact consideration.
- There is some organizational impact arising from the migration. The impact could be in terms of people or teams affected, or in terms of specific business processes or objectives. This impact must be analyzed more closely. In particular, if CL/SuperSession requires a task to be performed in a way that may conflict with established or mandated operational procedures, a check must be made to

ensure that a change to the procedures is acceptable and can be implemented and approved within the proposed time scale of the migration.

- There is significant organizational impact arising from the migration. One or more teams or major business processes will be severely constrained or compromised as a result of the intended migration. In this case, a review must take place to determine what action to take. It may be possible to modify the existing process so that the objectives that required the original facility might be fulfilled in another way.

Note: In general, most migration projects do not seriously and adversely affect business processes. If preparation work reveals a problem for business processes, this indicates a serious organizational exposure which must be remedied as a matter of priority.

The outcome of this analysis should be an organizational analysis document structured by group, detailing their business drivers and objectives, how the current product facilities are used in addressing these objectives, and how they will be implemented or replaced using CL/SuperSession.

Realistically, it is possible that early work indicates migration is not advisable. It must be remembered that such an outcome is still extremely valuable, and an indication of professionalism from the migrating team. Continuing with a migration in such circumstances increases the likelihood of project failure.

End-user considerations

End-users could be greatly affected by a migration, although they are rarely the main reason for migrating. An end-user is defined here as someone at any level within the organization, who uses the session manager product as a significant part of their normal work.

A good way to evaluate migration impact is to invite end-users or their representatives to describe the tasks they perform on a regular or occasional basis. Depending on the size of the organization, it may be appropriate to have meetings directly with a representative sample of the end-users, or with formal representatives such as labor-unions or work councils. In some countries, legislation may exist that make formal consultation with employee groups mandatory.

Note: Any difficulty identifying representative end-users indicates more significant organizational problems that could impact the migration project. These indications should be used to factor in additional time for checking organizational acceptance during pilot testing and introduction of the production system.

When considering end-user requirements, it is helpful to have a general set of starter questions. These are extended and supplemented as answers are obtained from various sources. Examples include:

- What aspects of the current daily task require use of a session manager?
- How is the current session manager product used?
- Show me how you perform some example tasks?
- How do these tasks meet the business requirements?
- In what ways could your task be made easier?

As the answers are collated, a picture is built up of the way in which end-users currently use the session manager, and how they may be affected by migration to the new tool. Identification of ways to improve the current task is extremely important; as well as influencing the design of the final system, it helps ensure that the new system will be adopted enthusiastically.

Normally, many dependencies will be identified. A lack of dependencies suggests that more detail is required before proceeding with the migration. Each end-user dependency must be analyzed thoroughly, and the implications for the organization as a whole evaluated.

End-customers

Figure 1 on page 5 identifies another important group who may also be affected by the migration. The end-customers are the consumers of the services or products offered by the migrating organization. It is possible that end-customer feedback is a major reason for considering the migration. An example might be where the migrating organization provides a service, such as a disaster recovery support. Here, the end-customer is a service consumer. The customer will have specific requirements. These may include that a particular session management tool be provided as part of the service. This may force the service provider organization to include support for the new session management tool, whether by clean-room provision or migration as part of disaster recovery response reviews.

There should be no adverse impact on the end-customers as a result of the migration. However, given that a benefit of migration is to improve overall task performance, a useful way of measuring the success of the migration would be through end-customer feedback. During the migration project, a pilot program will run to help identify any large-scale migration issues to address before operational deployment. It is possible that some end-customer groups may be prepared to help with this pilot, by contributing some realistic scenarios to the pilot.

If the migrating organization has a user-group of its own, it is helpful to consider inviting representatives into some of the migration discussions. This may identify scenarios that are very suitable for the pilot program. Some end-customers may be approaching a critical time in their business year, where a reduced service caused by a pilot or migration program could be unwelcome.

Keeping end-customers informed and aware of organizational changes is a good way to promote and maintain good relations. At a time of potentially significant organizational change, this is especially important.

Technical preparation

There are many technical aspects to consider when preparing a migration to CL/SuperSession. Some differences may occur in the way the two products implement a particular concept, such as generic resources. Other considerations arise from the opportunities provided by using CL/SuperSession; a new function or feature could make an existing way of doing things unnecessary.

As with group considerations, a report will be produced for each technical feature. The report describes technical differences between the current session management product and CL/SuperSession. There are three possible scenarios:

- There is a perfect match between the current session management product and CL/SuperSession. The report document will note this correspondence.

- There is some match or overlap between the current session management product and CL/SuperSession. This must be analyzed more closely. In particular, if CL/SuperSession provides more functionality than the current product, a check must be made to ensure that the extra capability does not conflict with the required functionality. There may be an opportunity to remove other features that are no longer required.
- There is no match between the current session management product and CL/SuperSession. In this case, a review must take place to determine what action to take. It may be possible to modify the existing process so that the objectives that required the original facility might be fulfilled in another way. Alternatively, it may be possible to remove the need for the facility.

The outcome of this analysis should be a technical analysis document structured by features, detailing each of the current product facilities used, and how they will be implemented or replaced using CL/SuperSession.

Note: The technical analysis document must not include another vendor's copyright-protected material, unless permitted by an applicable license from the copyright owner.

Using the list of existing features to structure the document helps ensure that current functionality is maintained in the migrated system.

The document should emphasize the business purpose that drives the technical requirements. This helps to determine whether the facility should continue to be used, given the additional capabilities of CL/SuperSession. For example, a business requirement that needed two or more facilities within the current session management product might be replaced by a single, different facility within CL/SuperSession.

We will now discuss some technical considerations in more detail.

Generic resource support

A front end tool such as CL/SuperSession has to have a high availability. The best way to achieve this is through the use of generic resources. IBM session management products support generic resources, and can run within a parallel sysplex configuration. This means that end-users will get their session back if the physical session was broken, or if they are disconnected. The session is disconnected and routed by VTAM to a different instance using the CLSDEST path feature.

It is important to understand current use of generic resources before migration, to ensure that there are no incompatibilities with the resulting system after migration.

System- and end-user administration

The administration facilities provided by the current session management tool should be examined in two ways:

- By reviewing existing process documentation and observing use patterns. This will provide significant detail about the commonly used facilities, but limited information about the rarely-used facilities.
- By reviewing the product documentation for the current session management tool. This will help ensure that all facilities are identified, but will provide little information about the importance or frequency-of-use for the facilities.

It is also helpful to take into account the longer-term strategy for session management tasks. For example, some tasks may increasingly be performed by end-users rather than administrators. The objective is to distinguish between system-administration and end-user requirements. In meeting the objective, the best approach is to provide the minimum possible authority for individual end-users.

As the facilities are identified and categorized for the current session management product, a one-to-one correspondence for each of the capabilities should be established with CL/SuperSession.

Exits

Some session management products were implemented directly in machine assembler code, rather than being developed using a high-level language. A characteristic of these assembler-based products is that they have a large number of exits. They are often heavily-customized by using the exits to support particular tasks or applications.

Each of the exits used must be identified and evaluated. For established systems, a problem may arise if the specialists who originally installed or configured the product have moved on to other roles or even left the organization. Some organizations have systems and applications that have not been reviewed or modified for a long time.

As with the administration facilities, each of the exits identified and used should be evaluated, to determine what task they perform. Each is assessed to identify its business objective. Once this chain of requirements is established, a document is created, detailing which tasks are dependent on current exits. The document must then describe how the tasks will be implemented using corresponding CL/SuperSession capabilities.

Note: The document detailing exits must not include another vendor's copyright-protected material, unless permitted by an applicable license from the copyright owner.

Use of exits in a configuration can produce a high-level of dependency on the specific session management product. As the number of exits used increases, so does the task of identifying, designing and implementing alternatives. The analysis of exits used should be monitored carefully; in particular watch for indications that suggest the cost of attempting to migrate the exits would exceed the cost of re-implementing the functionality using native capabilities within the new session manager.

Virtual terminal assignment and masking

Existing session management products can have complex terminal masking and assignment tools. These are used to provide stronger terminal security. Virtual terminal masking may also be used for terminal security. For example, to perform a high-security task, an application might require that virtual terminal 01 must be used. This means that the user would have to access that virtual terminal to perform the task.

The existing Virtual terminal masking rules must be maintained during migration. Otherwise, the application imposing the rules would have to be changed, which would be a very complex task.

When preparing for migration, analyze all of the virtual terminal masking rules currently in use to ensure they will be maintained. Check all the system, application profile, and user level terminal requirements. Existing exits may also be used within scripts, and must also be checked for this reason. ACBs may also be reserved, and these reservations must be maintained.

Features and products

Some applications work in parallel with the session manager. Examples include compression tools such as IBM's OTTO, BMC's ULTRAOPT or CA-NETCompress. It is important to analyze how these applications are currently used. Next, consider whether there is an alternative and more cost-effective way of achieving the same result.

Note: It is important to identify all the applications that are part of the information flow through the session manager. Failure to do so could result in missing some important process implications and cost savings.

Another example is printer support. CL/SuperSession does support virtual printers. It is also possible to use other IBM products that provide corresponding capability, for example NETSPOOL, which is a centralized part of the z/Series INFOPRINT server.

Emulation scripts

The usage model for session management in general assumes that users will be working and interacting using intelligent 'green-screen' terminals, such as the IBM 3270.

In practice, most users today have a personal computer or workstation on their desk rather than a 3270 or similar dedicated device. The workstation will have emulation software installed, which makes it appear as a 3270 to the session manager. In addition to the emulation software, the workstation may also provide additional capabilities.

For example, a simple click on a menu within the emulation software may translate into an otherwise complex series of commands that would normally be entered manually in sequence, such as performing a login. Using an emulation script enables tasks to be performed more quickly, and with less likelihood of error.

Note: The session management administrator has no control over the use of such scripts or enhanced emulation facilities. This is because the session manager will be unable to distinguish between a command sequence coming from a real 3270 or an emulated equivalent.

IBM consultants have encountered systems where entire applications had been constructed using these scripts. For example, some scripts were created for call-center scenarios, and are intended to speed the process of entering data and responding appropriately.

Note: While customized applications and scripts give great flexibility to the end-user, they can be extremely complex to analyze and migrate. Significant additional time should be allowed for migrating complex scripts.

Emulation scripts and customized versions of the emulation software will normally be included as part of the standard set-up for a new user, for example when a new

employee joins the team. The setup tasks for all new users should be inspected to identify the enhancements or configuration changes made to emulation software. Any standard scripts should be identified. It is advisable to get information from the support staff who create and maintain these scripts and configuration settings.

Each script or customization must be analyzed to identify any screen dependencies. For example, some scripts may expect that a specific terminal emulator is being used, or that certain terminal components will appear at precise locations on the screen. These locations and terminal components may change when migrating to CL/SuperSession. For each script, identify the screen or terminal dependencies. Using the resulting list, determine the revised values that should be used in the script.

Note: It is possible that an emulation script is no longer actively supported or maintained within the organization. In this case, review the script with end-users, and identify the business function performed. Next, determine whether the script needs to be migrated. If so, design and develop an alternative implementation. This will be thoroughly tested and validated as part of the pilot work later in the migration plan, as discussed in “Pilot” on page 12.

Customized applications using the session manager

Some session management products can be enhanced or extended by customized applications. These use the base product to enable rapid execution of existing tasks, or combine tasks for new business purposes. An example might be the creation of a 'hotkey sequence' that will carry out specific transactions given a standard set of entries.

As with the “Emulation scripts” on page 10, customized applications will normally be part of the standard set-up and configuration provided for new users, and so should be well-documented. The configuration information for the applications should provide sufficient detail to identify session management dependencies. Information should also be obtained from the development or support team who maintain these customized applications. These dependencies should be documented, and corresponding CL/SuperSession equivalents identified.

The current session management product may be required for separate applications. These could be applications included with the session manager, or which make use of components, facilities, exits or other aspects of the session manager. Other applications may be developed or obtained independently, and specify the session manager product as a prerequisite. Examples of these applications include BOS-Help and IBM Office Vision.

It is important to be aware of regional-specific tools and requirements. As with the other technical preparations, the process is to identify and analyze what the customer does, starting with an examination of the standard applications and configurations provided during a new-user setup. Once the business contribution of the specific application is understood, another solution may be possible that does not depend on the original session manager functionality.

Auditing and control requirements

The majority of tasks and actions performed by end-users and administrators should be auditable, to ensure compliance with legal and other requirements. For example, the organization may record business performance information. This data could be used to improve system response times. Similarly, some business processes may be subject to external inspection, for example to ensure compliance with pharmaceutical standards.

The existing session management facility may capture and record on-going control information. The recording may use standard features within the tool, or a combination of scripts or other applications.

The specific audit requirements must be identified. Normally, this is straightforward to achieve with the help of the staff responsible for audit conformance. These requirements are compared with the facilities provided by CL/SuperSession. Some data may be collected, albeit through a modified collection process. Other data may require new or modified applications or tools. These should be identified, and obtained or implemented.

Note: When discussing audit requirements, do not confuse them with other requirements which should normally be addressed using security and access controls.

Developing the migration plan

The flow of the migration project will include phases and stages that are very similar to most other project management plans. The migration plan will include activities such as initialization, organizational analysis, development and testing, education and skills transfer for affected teams.

Project workflow

A useful set of checklists is provided in “Migration checklists,” on page 33. These should be completed with help from the customer. The checklists will help increase the amount of information available for creating the migration plan.

In particular, “Project workflow” helps with identifying the effort required to perform typical migration project tasks. For example, the effort required to re-write or modify scripts will vary significantly between projects, but an estimate is still required to help size and schedule the project.

Pilot

An important special case within the migration plan is the pilot phase. This is where a complete team or department uses the new session manager product to perform all their normal tasks. The majority of unusual tasks should also be performed, where possible. Good candidate departments to join the pilot would be those which have standard or unusual requirements. For example, a department which has no special customizations would be good for establishing a baseline within the Pilot. Departments with many customizations as described in “Technical preparation” on page 7, would also be important for the pilot, as they will help ensure the required customizations are migrated correctly.

In the next section, we consider some of the technical steps performed during migration, and suggest tooling that will help.

Chapter 4. Migration tooling

In this section, we describe how to migrate information into an IBM CL/SuperSession for z/OS environment. We assume that information will also be migrated for configuration in a security environment such as RACF®.

The basic process is shown in Figure 2 on page 14, and starts with the information available from the existing session manager. This will normally be held in a database, and can be extracted or reported as a text file. The text file is processed through a set of analysis tools, and converted into a spreadsheet format for easier analysis and review. Tooling generates command statements with data from the spreadsheet. The statements populate the CL/SuperSession and security environments.

For example, resource class data held within a security system helps create dynamic Applists within CL/SuperSession. This data must be obtained from the current system.

Database structures used by session managers vary. The migrating project team will require help from the system administrators to understand the current database structure. Analysis tools are adapted to recognize the data available from the current database.

Note: Where possible, the migrated data should be stored in an independent database. This will ease subsequent upgrades or migration tasks. For example, using the RACF database or similar is a good choice, as all session managers may use of the resource class facility and its features.

CL/SuperSession Session Management Workflow Data Migration - a toolkit -

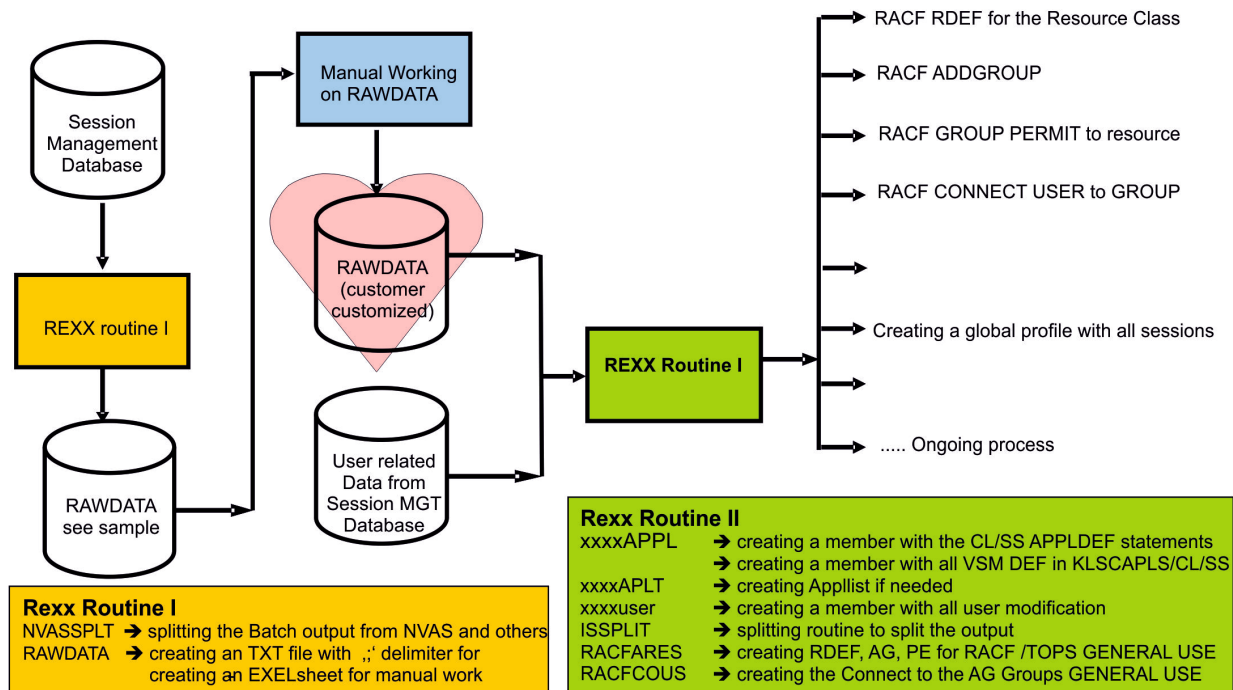


Figure 2. Overview of migration tooling

The migration process

The process is shown in Figure 2. There are two sets of scripts. The first stage scripts prepare the extracted data for use within a spreadsheet. The spreadsheet is analyzed and checked. The second stage scripts use the data to prepare command statements which populate the new session manager and security environments.

This process makes it easier to revisit earlier steps in the migration. For example, if the generated command statements require modification, but the data is known to be correct, it is only necessary to modify the second stage scripts.

We will now examine the migration process steps in more detail.

First stage scripts

These scripts analyze the data extracted from the existing session manager databases. The scripts are normally very specific to the current session manager and the way it is currently configured.¹

The data processed by the scripts will normally include the following details:

1. Dependent on the current session manager product installation, a different set of migration toolkits are utilized by IBM consultants..

- System Application Profiles These will include key data such as the system internal name, the primary LU name, ACB Type, Pseudo Terminal Prefix, the maximum number of sessions, as the Logon and Logoff scripts which are needed for the application profiles.
- Group Application Profiles These will include key data such as the application name and description, the system internal name, the jump key, as the Logon and Logoff scripts which are needed for the application profiles.
- Bind User Data Profiles These will include key data such as the system internal name, date and time of the last update.

Note: The data processed by the scripts is usually a CSV text file. This file may be analyzed using any tools capable of parsing text files, such as REXX scripts. IBM service consultants have designed and implemented various tools to help with this type of task.

The result of first stage analysis should be a CSV text file² format. This is the first stage raw data shown in Figure 2 on page 14, and is suitable for importing into a spreadsheet tool.

Manual update

This task is where the data from the current session manager is consolidated, checked, and if necessary changed. The details of this task are discussed in “Working with the spreadsheet” on page 16.

Second stage scripts

These scripts take the data from the spreadsheet, and create sets of system commands.

The second stage scripts generate statements to perform the following tasks:

- Creating application definition statements (KLSCAPLS)
- Creating ranges for virtual terminals (KLS\$VSMS)
- Creating static applists if needed (KLSAPLST)
- The global profile is created during initialization of the RLSNAM DB and will include the information from KLSCAPLS.
- Creating RACF statements to define resource classes, add groups, permit groups to access resources, and connect users to groups and the back off commands to delete the definitions if wrong.
- Several CL/SuperSession dialogs available which creating the User definitions directly in the VSAM-Cluster at migration time. The input for these run are the output of the Usermod scripts to gather the modifications together and splitting them into the 3 following user related profiles.
 - Userid.USER.COMMON.PROFILE
 - Userid.USER.SESSION.PROFILE
 - Userid.USER.TRIGGER.PROFILE

As with the first stage scripts, any tooling which can parse and process CSV data exported from a spreadsheet may be used.

2. CSV = Comma-Separated Values: A text, which is normally comma separated. In this case the fields are separated by ;

Note: IBM service consultants can assist with creating and using the scripts for migrating data into CL/SuperSession, subject to information being available under applicable copyright ownership and licenses.

Manual tasks

Some local customizations of the existing session manager configuration cannot be migrated with the help of automated tools. For example, login scripts and the exits used are often product- and user-specific. While it may be possible to create tooling for migrating these local customizations, the effort required is probably not justified. The customizations must be inspected manually, and modified or written again, either by the migration team or the customers themselves.

Note: The information required to inspect, understand and migrate an existing configuration and any local customizations or extensions may be subject to copyright ownership and licensing terms. Before commencing a migration project, it is important to identify and understand any constraints of this kind which may limit the work carried out by consultants, whether from IBM or elsewhere.

Working with the spreadsheet

The databases used by the current session manager will contain a number of major fields. Reports or extracts from the database will normally provide much of the following data:

- Session ID (SESSIONID) (current, old, new Session ID fields)
- Application ID (APPLID)
- Profile
- Start at sign-on
- Logmode
- Start script reference
- Stop script reference
- Maximum sessions
- Terminal pool or masking information
- User data
- Selection or PF key
- Owner and Superior Group
- Other description details like GROUP and ORDER to ordering the appls on the Menu.

The data is analyzed by the first stage scripts to produce a 'raw data' spreadsheet. It contains essential information that will be useful to all project team members.

Example spreadsheet data can be seen in Figure 3, and continued in Figure 4 on page 17.

Function	Group	Seoid	Profile	Old Sessid	New Seoid	Applid	Start at signon	Logmode	Start script	Stop script
#SMPROD		CICSPROD	NN	CICSPROD	CICSPROD	CICSP	Y	NN	CICS	CICSOFF
	#SMCTST	CICSTEST	NN	CICSTEST	CICSTEST	CICST	N	NN	CICS	
#SMPROD		TS001	NN	TS001	TS001	TS001	N	NN	TSO	
	#BMTT80	TS002	NN	TS002	TS002	TS002	N	NN		

Figure 3. Sample spreadsheet data following session manager database extract

Start script	Stop script	Max Sess	Description	Pool	Owner	Superior Owner	Userdata	Selection key
CICS	CICSOFF	5000	CICS Prod	SMVP----	IBMOWN	IBMSGRP		1
CICS		10	CICS Test	SMVT----	IBMOWN	IBMSGRP		2
TSO		100	TSO Prod	SM	IBMOWN	IBMSGRP		3
		20	TSO Test	SM	IBMOWN	IBMSGRP	UID=UID.OPW=PWD	4

Figure 4. Continuation of sample spreadsheet data

Function Groups and Groups

It is recommended that two extra fields are added to the spreadsheet. These are the Function Group and the Group. These fields make it easier to create the security information that will be used later in the migration process, as described in “Migrating data into security systems” on page 19.

Users may be connected in any of three ways, as shown in Figure 5:

1. Function Group A user is connected to a Function Group which is permitted to access more than one resource in a resource class.
2. Group A user is connected to a Group which is permitted access to a single resource in a resource class.
3. Direct permission When the user is given specific access to a resource, but not as part of Function Group or Group membership.

Note: This type of direct permission is not very convenient. It requires a database refresh before the permission becomes active for the user.

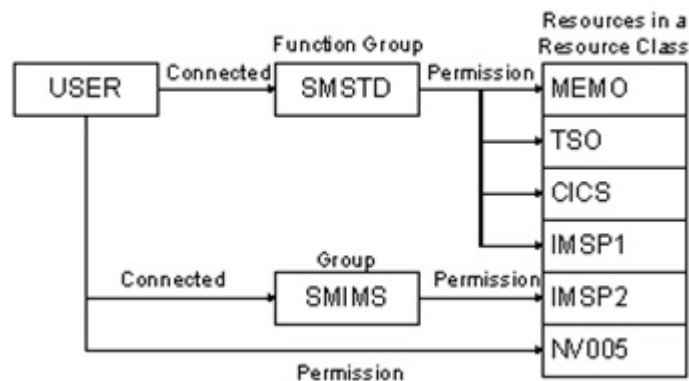


Figure 5. Overview of user connections to Function Groups and Groups

It is also helpful to have a naming convention for the Group names. Two possible conventions are suggested in the following, but mostly the customers already have naming conventions:

Scenario one

The eight-character name is defined as shown in Table 1.

Table 1. First recommended naming convention

Character position	Recommendation
1	# or \$ character

Table 1. First recommended naming convention (continued)

Character position	Recommendation
2	A single letter. Use P for Production, T for Test, or F for Function.
3 to 6 inclusive	Four letters for the session ID
7 to 8 inclusive	A two-digit sequence number

Example names created using this convention include #PTSOA01, #PTSOA10, #TTSOZ01 and #TFTSO00.

Scenario two

The eight-character name is defined as shown in Table 2.

Table 2. Second recommended naming convention

Character position	Recommendation
1	# or \$ character, or a special non-numeric sign
2 to 6 inclusive	Five letters for the session ID. If the session ID name is too long, it may be abbreviated. If the session ID name is shorter than five letters, fill the remaining space with X characters.
7 to 8 inclusive	A two-digit sequence number

Preparing to migrate data into security systems

Before migrating data into the security system, a number of details should be considered.

1. Ensure that the security database has sufficient free space. The spreadsheet review should indicate the quantity of data going into the database and the space required. A more precise figure can be determined by running the second stage scripts and analyzing the exact statements generated. If the database is not large enough, the RACF administrator must increase the database size before proceeding.
2. Currently, the maximum number of users who may be connected to a group is 5957. If the migration would require more users in the group, it is possible to define and use a Universal Group within RACF. If this method is used, the second stage scripts will honor the Parameter and within the ADDGROUP the 'UNIVERSAL' parameter is added to the statement as shown in Figure 6.

```
TSOEXEC ADDGROUP #ICL/SSUNIV OWNER(IBMOWN) +
SUPGROUP(IBMGRP) UNIVERSAL
```

Figure 6. Example ADDGROUP command using Universal Groups.

3. The migration scripts may produce large numbers of CONNECT statements for end users. The exact number will depend on how many sessions are defined in the current session manager, and how many Function Groups are created.
4. Each CONNECT statement takes a certain amount of time to update the RACF database. This time will increase for heavily-loaded systems. It may take a long time to run all the command statements generated by the second stage scripts.

Migrating data into security systems

If the current session manager already working with a session list which is dynamically created by RACF the CL/SS can use that and nothing need to be done. Only when the profiles in that resource class are longer than 8 bytes than the Resource class need to be maintained or created new based on a copy of the current one.

The spreadsheet created by the first stage scripts contains useful material for security administration. For example, with details of the Function Group or Group, the Owner and the Superior Group, dynamic Applists may be configured.

The spreadsheet must be evaluated by the security administrators, to decide which connections will be required in the security system for end users.

To implement the 'dynamic Applist' for CL/SuperSession, the system administrator has to run through the following steps³:

- Creating a member in RLSPARM such as DYNAPLST with the value VGWAPLST EXTERNAL=\$SMAPPL where \$SMAPPL is the Resource Class (SAMPLE) which is used to defined the profiles.
- The member DYNAPLST needs to be implemented into the Control Point Definitions and the usage of the NAM DB with the following definitions to enable Password validation against RACF and to check the applications to create a dynamic Applist for the user

```
DEFAULT DSNAME(vsam-dataset name)      -
      RACF                               -
      CLASSES=dynap1st                   -
      NODB                                -
      DISP=SHR                            -
MSG4= 'You are not defined in RACF '     -
MSG8= 'the entered password is not valid' -
MSG12='your password has been expired''  -
MSG16='the password is not valid.'       -
MSG24='too many not valid tries. Your id is revoked.'
```

In this case the resource Class \$SMAPPL is used to check the applications whether the user has the right to get that on the menu or not.

Whether the resource will be checked by Sessid or VTAM applid can be set by the Global Profile within the ADMIN Platform.

Some general notes on the CL/SuperSession Admins:

- The first user who is logging on to CL/SuperSession will be the ADMIN and he can give other userids the ADMIN rights to maintain others
- There is no Check of the Admins against the RACF whether they are Admin or not. This has to be handled in CL/SuperSession itself for User ADMIN and for the Helpdesk
- Be careful who you give the CUA Operator and VPSS Operator, only Admins should have these functions. For the example data contained in the spreadsheet shown in Figure 3 on page 16 and Figure 4 on page 17, the second stage scripts should generate commands to populate the RACF security systems.

3. The detailed documentation will be found in 'Basic configuration Guide' chapter 5

The example resource definitions are shown in Figure 7.

```
TSOEXEC RDEFINE FACILITY CICSPROD UACC(NONE) OWNER(IBMOWN)      +
DATA('CL/SUPERSESSION FOR Z/OS: CICS PROD') +
  AUDIT(FAILURES(READ))
TSOEXEC RDEFINE FACILITY CICSTEST UACC(NONE) OWNER(IBMOWN)      +
DATA('CL/ SUPERSESSION FOR Z/OS: CICS TEST') +
  AUDIT(FAILURES(READ))
TSOEXEC RDEFINE FACILITY TS001 UACC(NONE) OWNER(IBMOWN)        +
DATA('CL/ SUPERSESSION FOR Z/OS: TSO PROD') +
  AUDIT(FAILURES(READ))
TSOEXEC RDEFINE FACILITY TS002 UACC(NONE) OWNER(IBMOWN)        +
DATA('CL/ SUPERSESSION FOR Z/OS: TSO TEST') +
  AUDIT(FAILURES(READ))
```

Figure 7. Example resource definitions for RACF

The example ADD GROUP statements are shown in Figure 8

Note: If RACF Universal Groups are being used, the statements must be modified as described in “Preparing to migrate data into security systems” on page 18.

```
TSOEXEC ADDGROUP #CL/SSPROD OWNER(IBMOWN)      +
  SUPGROUP(IBMGRP) TERMUACC      +
DATA('CL/SUPERSESSION FOR Z/OS PROFIL FOR +
  Group: function Group #CL/SSPROD')
TSOEXEC ADDGROUP #CL/SSCTST OWNER(IBMOWN)      +
  SUPGROUP(IBMGRP) TERMUACC      +
DATA('CL/ SUPERSESSION FOR Z /OS PROFIL FOR +
  Group: CICS TEST')
TSOEXEC ADDGROUP #CL/SSTTSO OWNER(IBMOWN)      +
  SUPGROUP(IBMGRP) TERMUACC      +
DATA('CL/ SUPERSESSION FOR Z /OS PROFIL FOR +
  Group: TSO TEST')
```

Figure 8. Example ADD GROUP statements for RACF

The example PERMIT statements to give a group access are shown in Figure 9

```
PERMIT CICSPROD CLASS(FACILITY) ACCESS(READ) +
  ID(#CL/SSPROD)
PERMIT TS001 CLASS(FACILITY) ACCESS(READ) +
  ID(#CL/SSPROD)
PERMIT CICSTEST CLASS(FACILITY) ACCESS(READ) +
  ID(#CL/SSCTST)
PERMIT TS002 CLASS(FACILITY) ACCESS(READ) +
  ID(#CL/SSTTSO)
```

Figure 9. Example PERMIT statements for RACF

Note: After all Groups have been defined and PERMITs issued, the security database must be refreshed. For RACF, the command would be:

```
SETROPTS RACLIST(FACILITY) REFRESH
```

The example statements to connect users with groups are shown in Figure 10

```
CO USERA GROUP(#CL/SSPROD) OWNER(IBMOWN)
CO USERA GROUP(#CL/SSCTST) OWNER(IBMOWN)
CO USERA GROUP(#CL/SSTTSO) OWNER(IBMOWN)
CO USERB GROUP(#CL/SSPROD) OWNER(IBMOWN)
```

Figure 10. Example CONNECT statements for RACF

System definitions

The second stage scripts produce statements for creating CL/SuperSession system definitions. These include application definitions, ranges and if needed static Appllists when the dynamic Appllist is not supported which was described in “Migrating data into security systems” on page 19. The following subchapter will explain that in more detail.

Application definitions

The information available in the spreadsheet should be sufficient to create all the application statements required for the CL/SuperSession configuration.

Using the sample data shown in the spreadsheet shown in Figure 3 on page 16 and Figure 4 on page 17, the second stage scripts would produce application definition statements similar to the one shown in Figure 11.

```
APPLDEF HELPDESK -
      DEST=CL/SSHLP -
      DESC='CL/SS HELPDESK/MONITOR' -
      POOL=VIRT3270 -
      INITDLG=KLHLOGON -
      USERDATA='&VSSUSER,&VIGPHPW' -
      MULTSESS=YES -
      ORDER=9999 -
      GROUP=910
```

Figure 11. Example Application definition statement generated by second stage scripts

Additional parameters may be use to customize the migration. These are described in more detail in the IBM CL/SuperSession for z/OS Customization Guide V2.1. Examples include:

If the installation is configured to work with user data, add the parameter like these:

```
USERDATA= 'UID=&VSSUSER. OPW=&VIGPHPW.'
USERDATA='&VSSUSER,&VIGPHPW'
```

- If the pool is empty and MULTSESS=NO then a CLOSEDST PATH will be opened
- If the pool is empty and MULTSESS=YES a single session is opened
- A parallel ACB will be used when the POOL will have a Parallel VSM DEF in.
- Pool can be used in several ways. If a Pool is a *, a dynamic pooling will be done by a INIT scripts. To do masking like changing first character of the physical terminal and so on. This is done in the KLS\$VSMS member by editing the NODE PARM

Pool definition in KLS\$VSMS

The Ranges will be created in a different way, as several possibilities are available to maintain the KLS\$VSMS statements. As session manager like CL/SuperSession has their own RANGE statements it could be taken over 1:1 into the KLS\$VSMS.

If special bits has to be maintained like Masking up front or at the end of the virtual terminals the KLS\$VSMS and its VSM DEF parameter need to be used. It is explained in detail in the IBM CL/SuperSession for z/OS Customization Guide V2.1 in Chapter 3 Virtual Terminal Pools.

Here are some samples to get an idea how to maintain the KLS\$VSMS definition:

Using the sample data shown in the spreadsheet shown in Figure 3 on page 16 and Figure 4 on page 17, the second stage scripts would produce statements similar to the one shown in Figure 12.

```
VSM DEF VIRTURNS  VTP00000      LOGMODE(&DEFLMODE) DEFER PARALLEL
VSM DEF VIRTDED   VT000001 TH(10 LOGMODE(&DEFLMODE) DEFER DEDICATE
VSM DEF VIRTURNS  VTPASS11 TH(70) LOGMODE(&DEFLMODE) DEFER PASS
VSM DEF VIRTURNS  VTV00001 TH(70) LOGMODE(&DEFLMODE) DEFER
VSM DEF VIRTURNS  VT000002      LOGMODE(&DEFLMODE) DEFER TIMEOUT -
                                NODE='V&SYSEDIT((1,7)&SYSTEM)'
VSM DEF VIRTURNS  VT000003      LOGMODE(&DEFLMODE) DEFER TIMEOUT
VSM DEF VIRTURNS  VT000004      LOGMODE(&DEFLMODE) DEFER TIMEOUT
VSM DEF VIRTURNS  ABC00002 TH(70) LOGMODE(&DEFLMODE) DEFER TIMEOUT -
                                NODE='&SYSEDIT((0,3)&SYSTEM).???'
VSM DEF VIRTURNS  XYZ0002  TH(70) LOGMODE(&DEFLMODE) DEFER TIMEOUT
VSM DEF DUMMY     DUMMY          LOGMODE(&DEFLMODE) DEFER
                                NODE='&VIRTTERM' .....
```

Figure 12. Example VSM DEF definition statement generated by second stage scripts

In the above samples are several VSM Definitions how to define a virtual terminal pool. In General it can be summarized

- DEFER ==> an ACB which is used for several different VTAM application but not the same for the same or different Userid's
- PARALLEL ==> several sessions to the same APPLID for different or equal USER like TSO, NETVIEW. Not used much
- PASS ==> one session at one time over one virtual terminal
- TIMEOUT ==> is the value to timeout the session (is a sessiontime value) If different timeouts needed additional pools with the same terminals can be defined.
- NODE ==> this value can be used to create specific terminals based on the Physical terminal and or specific Literals If NODE is set to a specific variable (in this case &VIRTTERM) this has to be generated by an initial dialog like the following⁴

Note: Previous migration projects found that fewer than 10% of customers use virtual terminal masking. Most customers assign virtual terminals using range statements, with the remainder using one-to-one mapping between physical and virtual terminals.

If specific ACB mappings are required, these can be implemented using an on or more different INIT Dialogs which are called during Signon time to create the terminal and add it by a VSM DEF statement to a pool. The terminals are attached to the user and will be used during session request.

Note: Customer-specific scripts may also be written for specific ACB scenarios.

4. Will be found in CL/SuperSession Documentation: Customization Guide Chapter 3. A sample init script in an easy form could be like this. The complexity is strongly dependent on the customer's requirements.

```
)OPTIONS LEVEL (1) )
INIT
SET VIRTTERM '&SUBSTR
('&SYSTEM',4,4)'
COMMAND('AS '*SYSVLG*' VSM DEF
DUMMY &VIRTTERM DEFER')
```

Global and individual profiles

In order to migrate data into CL/SuperSession, a global common profile definition must be prepared first. A default is created if no GLOBAL.**.PROFILE is present in the VSAM cluster **.RLSTDB. The smallest possible number of profiles should be defined. This helps avoid duplicated data, and makes it easier to construct dynamic Appllist. Established session management systems tend to have large numbers of profiles. This is because users have specific requirements that may be complex and change frequently. Without a clear policy for creating profiles, or a regular review and removal of incorrect profiles, it is often easier to create a new profile definition when required. The result is many profiles, often no longer valid.

As CL/SuperSession supporting a Group and/or one static application list per user, it might be needed to think about the group profile and moving the changes into a group profile which will be addressed to the user. In the meantime we saw that the most of the users having several different profiles but the common parameter coming always coming from the Global profile.

As a best practice over the years was the implementation of the dynamic menu which are managed by RACF resource class and a group concept how to address which user is getting which session.

There are migration routines used by the IBM technical service consultants to migrate the User into userid.USER.**.PROFILES during Migration time. During migration data is split into the Common, session and trigger related information and will be inserted by a SSPL Dialog into the VSAM Cluster where the differences between Global and User related Profiles are stored like

- PFkeys per session
- Different Menu and Jump keys than in the global profile
- Language of the users if different to the Global profile
- Different Session related variables like INITIAL and TERMINATION Dialogs Description and others
- Also extension variables were created to an existing table for each user which having a 'userid.USER.SESSION.PROFILE' table to honor the session relevant Data, which are stored in specific variables in the current installed session manager. This could be a second userid, or a start transaction. There will be 5 variables available stored in VUPPARM and separated by a ':'

The GLOBAL.COMMON.PROFILE including all the system parameter and KLSCAPLS/KLGCAPLS Application definitions. There is no need to create a Global profile from the spreadsheet.

In this section, we have discussed technical aspects of migrating data from the current session manager, into a form suitable for use with CL/SuperSession. In the next section, we will outline more general recommendations that will help with the migration project.

Chapter 5. General recommendations

In this section, we make some recommendations that will be of help in most migration projects. The detail contained in these recommendations is based on documented information from successful migrations.

User data and variables

When a user is logged in to the system, data about them and the work they are doing is stored in several different locations and variables. These data items need to be identified to ensure that no information is lost. For example, variables from the old session manager must be identified to establish where they are required and what tasks they support. These tasks are then considered for migration into the new system.

CL/SuperSession supports a scripting language, SSPL (Structured Session Procedure Language).

In general the user as also system definitions will be held in tables, which are stored in a VSAM Cluster. All the user definitions and the system related information are stored in the RLSTDB, some others are in the RLSNAM DB, but less.

Per ADD or MOD command 'extended variables' can be added as a column to the original tables at user, group and/or global profile table. These extended fields will be managed by CL/SuperSession for z/OS to use them after a TBGET, but not maintained. Changes on these 'extension variables' has to be done by the customer by changing the Admin dialogs or writing their own admin dialogs.⁵

Variables can be created by session or local. They also can be defined system wide. This is declared in the INIT sections of the Dialogs.

Further information on the mechanisms used to create and manipulate SSPL variables may be found in the IBM CL/SuperSession for z/OS Dialog Language Reference Manual. There are also all available predefined variables listed.

General VTAM suggestions

When designing the migrated application configuration, it is easier to use generic resources for distributing workload and tasks.

If generic resources are already installed and used, identify whether the installation uses WLM or VTAM based routing. With the major node GRPREFS it is possible to define the Rules for the generic Resource instead of using the ISTEXCGR to maintain VTAM by Exit. The GRPREFS is reloadable. In general the Generic Resource is used in that way that the user will be on the system where he was routed by SYSPLEX Distributor or by dynamically VIPA. Than GRPREFS need to get the PARM LOCAPPL = YES and all the others set to NO.

5. Extension variables are columns which can be added to an existing table or created when a table is created.

CLSSGR	VBUILD	TYPE=GRPREFS	
xxxxxxx	GRPREF	GREXIT=NO,	Default NO
		WLM=NO,	Default YES
		LOCAPPL=NO,	Default YES
		LOCLU=NO,	Default YES
		PASSOLU=NO	Default NO

Generally, if all user requests come into the system from a single entry point, it makes sense to leave them where they are. The alternative is to distribute the requests, but this introduces the risk that later requests will be switched to a different system.

ACBs

Existing applications will already have a VTAM applname within the current session manager. To use the main ACB during movement make it is easier to re-use this name within CL/SuperSession for z/OS for the end users. And another main reason is to make roll-back simpler if problems should occur during deployment.

Once the migrated system is up and running, it is possible to change the names as a separate change activity.

Administration tasks

This was discussed in “System- and end-user administration” on page 8. In general, it is advisable to minimize the administration options available to an end-user.

Another point for the Admins is that the Administration rights not coming from RACF. It has to be set over the ADMIN popup in the action bar on the main menu if the user is ADMIN.

There is also an Admin within the Helpdesk who can maintain the other Helpdesk Admins.

Always the first user is the main Admin User and Helpdesk Admin.

Trigger usage

Different to other session manager products, CL/SuperSession for z/OS offers a real variable possibility to maintain a Trigger with an AID key. Means not only one Command character can be possible it depends on how the user wants to have it maintained. He can use phrases, with a maximum Length of 8 and an AID key. In the following box some samples are explained.

/	PF7	KLSGOTO TS01	<== jump into TS01 and activate
#	PF7	KLSGOTO TS02	<== jump into TS02 and activate
/w	PF12		<== jump into the Menu
#m	PF12		<== jump into the Menu
#j	PF12	KLSNEXTS	<== jump into next active Session
/admin	ENTER	KLSADMM	<== calling the ADMIN function from the action bar
/selfadm	ENTER	KLSOPTS	<== calling the selfadmin from the action bar
/trigger	ENTER	KLSTRG1	<== calling the Trigger admin from the Action bar
#admin	ENTER	ADMIND00	<== calling a self written Admin platform

The trigger phrase and the AID can only be entered in an active session. The trigger is normally not callable from the Menu.

Manual activities

It is likely that script modifications will be needed. This can become a major migration task, and plenty of time should be allowed within the project plan. Identifying the affected scripts is included as a task in the customer checklist presented in “Customer checklist - general questions” on page 38. Here we consider some detailed examples.

INIT Dialog at signon time of the user

It might be needed that several things need to be done for the user during signon time to CL/SuperSession. One reason we addressed already is the assignment of a special virtual terminal which has to be used for this user or user range during logon time.

Normally KLHLOGON is called as an Initial Dialog that the user is logged on to the internal Helpdesk. Also it is valid to check special sessions and inactivating them during signon time that the user is not taking the session with when for example Generic Resource is used and user should use the application which reside on the same system.

Initial and Termination Dialog

Each Initial and Termination dialog must be analyzed. For each script, identify the data required, and any cursor position details. Figure 13 on page 28 shows an example LOGON script for a NetView application, and includes sample statements that would need to be analyzed for migration to CL/SuperSession.

Some scripts may vary according to a second userid. The ID information may be stored in various ways:

- By hard-coding into the LOGON script
- Writing a specific PDSE member to store the userid
- Storing a second userid in the USER statement of the PDSE
- Storing them in an extension variable which is an additional column to an existing table entry.

The following sample should illustrate what do to in an Initial Dialog. CL/SuperSession now supports password phrases and the Dialog must be written in that way to honor password phrases as only a password.

The steps are the following:

- Declaring the variables the dialog need
- Check that userid, password and/or passphrase is not empty
- Wait for screen with VSSWAIT
- If the string 'Press ENTER to continue' is found
 - Hitting enter and wait
 - Hitting HOME and
 - Enter USERID and password/passphrase
 - And hitting enter

- Returning the user to the first screen after validation successful otherwise he will get a message that the userid password/passphrase might be wrong

```

)OPTIONS LEVEL(1)
)declare
  retry   scope(local)
  pswd    scope(local)                * PASSPHRASE
  vssuser scope(session)
  vsspswd scope(session)
  vigphrs scope(session)             * PASSPHRASE
)prologue
/* if userid=terminal and vsspswd is empty and passphrase is empty return */
if ((fold('&vssuser') = &system) and ('&vsspswd' = '')) and ('&vigphrs' = '') return
set retry 5
VSSWAIT(&sysparm 10 1)
while (&retry) do
  if not (VSSFIND(&sysparm,'Press ENTER to continue' '' NOCASE)) do
    VSSKEY( &sysparm,'ENTER')
    VSSWAIT(&sysparm,5)
    VSSKEY( &sysparm,'HOME')
    VSSTYPE(&sysparm,'&vssuser')
    VSSKEY( &sysparm,'TAB')
    If &vigphrs                                /* PASSPHRASE */
      set pswd '&vigphrs'                      /* PASSPHRASE */
    else                                        /* PASSPHRASE */
      set pswd '&vsspswd'                      /* PASSPHRASE */
    VSSTYPE(&sysparm,'&encdec('&pswd'))        /* PASSPHRASE */
    VSSKEY( &sysparm,'ENTER')
    VSSWAIT(&sysparm,5)
    return
  end
  set retry &retry-1
  VSSWAIT(&sysparm,2,2)
end
/* */

```

Figure 13. A sample Initial Dialog to VPSSCUA application

A Termination Dialog is similar to an Initial Dialog. The difference is that a Termination Dialog only runs when the user terminate the session, or when a time-out occurs or entering the EXIT function to inactivate all sessions. Other scenarios, such as disconnect, do not cause the Termination Dialog to run. A sample Termination Dialog is shown in Figure 14.

```

)OPTIONS LEVEL(1)
)prologue
  VSSKEY(&SYSPARM 'CLEAR')                /* clear screen*/
  VSSKEY(&SYSPARM 'CLEAR')                /* clear screen*/
  VSSKEY(&SYSPARM 'CLEAR')                /* clear screen*/
  VSSTYPE(&SYSPARM 'CESF LOGOFF')         /* logoff transaction */
  VSSKEY(&SYSPARM 'ENTER')                /* give it to application */
  log('Logoff finished')
return

```

Figure 14. A sample Termination Dialog for a CICS application

Samples which are in the *.*.SKLSPNLS stored, other available on the desk of the service consultants:

```

KLSONCIC   Initial Dialog to CICS
KLSONCUA   Initial Dialog to CUA platform of CL/SS
KLSONENG   Initial Dialog to the Engine

```


KLSONOM	Initial Dialog to CUA
KLSONTSA	Initial Dialog to TSO without Passphrase
KLSONTSO	Initial Dialog to TSO with Passphrase/Password
KLSONVM	Initial dialog to VM

Panel modifications

From the end-user point-of-view, it would seem that the new system should be similar in appearance and function to the old system, for example in presentation of sign-on and menu panels. However, property rights such as copyright ownership and licensing terms may make it impossible to create panels similar in appearance to those in the previous session manager. Retaining the previous design may also make it difficult to release some of the benefits of migrating to CL/SuperSession.

We would recommend thinking carefully about what user interface migration may be technically appropriate and legally acceptable. It is likely that the technical work required to migrate panels could be very complex. It may be more effective to consider a fresh approach, applying best practice in user interface design while incorporating the specific business objectives of the new session manager implementation.

General security suggestions

If an own class has to be used, the macro shown in Figure 15 must be entered into the z/OS system parameters. Nowadays it is possible to create the class dynamically, in the past an IPL was needed.

```

ICHRFRTB CLASS=##,ACTION=RACF
$ ICHERCDE CLASS=##SMAPPL,
    ID=149,
    MAXLNTH=08,
    FIRST=ALPHA,
    OTHER=ANY,
    OPER=NO,
    RACLIST=ALLOWED,
    DFTUACC=NONE,
    POSIT=32
PUNCH '    ORDER  ##'

```

Figure 15. RACF macro required if an own class is to be used

Note: The FIRST and OTHER parameters in the macro shown in Figure 15 are installation-dependent.

Note: Documented migration projects have typically required between 35,000 and 60,000 connect statements when moving the administration of the sessions into a resource class. These may take some time to process all this commands

Another point which needs to be addressed is the security of the Commands which can be entered in the CUA platform, at the Engine operator and/or by a Modify command into the STC. There is an exit available which can be used to only allow the *MASTER* operator to enter the SHUTDOWN command. This exit is stored in *.*.SKLVSAMP(KLVNAMX). If needed there could be add additional commands. The Modify command normally is protected by the modify command in RACF. This can also be used to protect from off-site.

Having made general recommendations for migrating to CL/SuperSession, the next section summarizes the key points for a successful migration project.

Chapter 6. Migration summary

In this migration guide, we have presented some best practices to help migrate a system to use IBM CL/Supersession for z/OS.

The aim has been to make the migration as smooth as possible. Some customers may be concerned that certain migration tasks are too difficult. However, by applying the general principles presented in this migration guide, many of the migration challenges are easier to overcome.

The migration tasks should normally followed in the order shown in “Project workflow” on page 12. This should help ensure that any non-standard aspects are correctly identified and addressed early in the project.

Information required for migration may be subject to copyright ownership and licensing terms. These may limit the detail available about the current system, and constrain the options for creating the new system. It is important to identify these constraints early in the migration project.

A common challenge in a migration is to complete the task successfully in the available time. A decision may have been taken to terminate an existing contract, making the cut-over date to the new system immovable. Following the recommendations in this migration guide will help produce more accurate estimates of the time required for complete migration.

A good rule of thumb is that each person-day of external consulting services would also require two person-days from internal staff, because of the organizational knowledge required. The internal people will have to handle much of the internal processes, such as scheduling meetings, organizing pilot participation, and so on.

An initial estimate of 30 days external consulting, coupled with 60 days of internal support, is a good first step in calculating the overall cost of the project. However, this estimate is very dependent on the complexity and scale of the migration.

Finally, the financial incentives for performing the migration could be significant. Removing unnecessary software licenses, and consolidating existing systems, will all help to reduce overall costs, and speed up the Return on Investment (ROI). In some cases, the cost savings following migration could be recovered in two years or less, depending on the complexity of the migration project.

Appendix. Migration checklists

When preparing the major tasks for a migration to CL/SuperSession, it is essential to ensure that the major aspects described in this migration guide are addressed. This appendix presents several checklists which may be used to ensure that these details are identified and processed correctly.

Note: Some of the details requested in the checklists may be subject to copyright ownership or licensing terms. It may not be possible for IBM or other consultants working on the migration project to have access to some of the information. When preparing a migration project, it is advisable to identify beforehand any materials, data or other resources where access cannot be authorized, and be prepared to discuss acceptable alternatives.

Project workflow checklist

The project workflow checklist shown in Table 3 is used to identify the major project stages, and the information required at each of these stages. Although the items in the checklist are presented in a recommended sequence which is typical for most projects, the order is not compulsory in practice.

The person-days column indicates the number of days required for one skilled person to perform the task, in a typical migration project.

Note: The person-day figures are estimates only, and should not be treated as absolute values applicable to any migration project.

The sum of person-day figures does not give an estimate of the overall project run-time. If migration is carried out without external assistance, the internal staff will probably require more person-days to perform each task.

Table 3. Key items to address for the migration project workflow

Step	Activity		Person-days
1	Project start. Identify the internal project leader. Identify the external project leader. Identify all other project members. Issue all project members with role and example task definitions.		Varies
2	Prepare and perform a 'kickoff' meeting. This is very important, as it will help consolidate the project team and influence the viability and success of the entire project within the organization.		3

Table 3. Key items to address for the migration project workflow (continued)

Step	Activity		Person-days
3	Identify and agree the departments and people who will participate in the migration pilot.		1
4	Identify the staff responsible for performing system test on the migrated system.		1
5	Identify and agree the 'go-live' production date.		1
6	Plan and agree all change management controls and processes.		Varies
7	If ACB masking will be required, and the standard used by CL/SuperSession does not match the customer requirements, identify and agree a recommended solution.		Varies
8	Perform ACB analysis, creating new major node members, preferably using an ACB model such as VT*.		1
9	Identify and analyze the use and implementation of softkey or other applications within the current session manager. This work may be constrained by copyright ownership or licensing terms.		Varies, but can be expected to take a significant amount of time.

Table 3. Key items to address for the migration project workflow (continued)

Step	Activity		Person-days
10	Identify and analyze the use and implementation of dialogs, scripts and exits in the current session manager. Map these requirements into the equivalents available within CL/SuperSession. This work may be constrained by copyright ownership or licensing terms.		Varies, but can be expected to take a significant amount of time.
11	Prepare security system (RACF) definitions for the test and production systems. Create RDEF, ADDGROUP and PERMIT definitions. This is discussed at length in Chapter 4, "Migration tooling," on page 13..		10
12	Create application and session definitions using the data reviewed by the customer. This is discussed at length in Chapter 4, "Migration tooling," on page 13.		1
13	If required, create or modify any application logon routines.		Varies
14	If printer support is provided in the current session manager, and is still required, create the corresponding facility as described in "Features and products" on page 10.		Varies

Table 3. Key items to address for the migration project workflow (continued)

Step	Activity		Person-days
15	Identify education materials for the help-desk and operations teams. Examples include information about logon, CL/SuperSession commands, sending and receiving messages, System administration and how to work with the helpdesk.		2
16	Develop and deliver a user-guide and short reference pack. The project cannot successfully complete unless this material is available for end-users. Ensure that the material is original, and does not draw on any information that is subject to copyright ownership or licensing terms.		5
17	Modify or add applications into system areas such as PROCLIB, VTAMLST, AUTOMATION and Scheduling.		2
18	It is recommended to keep the name for the main application used in the previous session manager product, unless there is a very good reason to change it. This is because no changes will be made to the Communications Server configurations. This makes a rollback easier if required, because a restart of the STC will make the old session manager available again.		Varies

Table 3. Key items to address for the migration project workflow (continued)

Step	Activity		Person-days
19	Milestone I: Create a prototype of the new session manager installation. Milestone II: After testing, install the prototype into a production system as a subsystem for use as the pilot.		
20	Advise the pilot team that the system is ready for use. Prepare the pilot system for use as the production system. Migrate RACF Data for specific pilot USERIDs. Create a change record in the ITIL process and request approval from all interested parties to commence the pilot stage.		1
21	Run the pilot system		5-10
22	Migrate all users into the RACF system.		1
23	Milestone III: Complete the consolidation of all data. Milestone IV: Successful completion of all pilot testing.		
24	Prepare any generic resources.		Varies
25	Milestone V: Ready for production use.		

Table 3. Key items to address for the migration project workflow (continued)

Step	Activity		Person-days
26	'Go-live' production date Create a checklist to confirm who will do what activities, and when. For example, checking that RACF migration of session data has completed successfully, confirming that all automation mechanisms are present and operational, and that the rollback system is still in place. All tests should complete successfully. Seek formal approval from all interested parties to make the new production system active. Change the ITIL process to activate the new session manager.		4
27	Final milestone: End of project.		
28	Project wrap-up. Review the work done, and identify lessons learned.		Varies

Customer checklist - general questions

The customer checklist shown in Table 4 on page 39 is used to collect important information about the current session management implementation. These details are used to help size and scope the migration plan. They also influence the design of any customization required for CL/SuperSession. For example, what PF key sequences are defined, and how are they used?

For each question in the checklist, consider:

- Is the information available under the product licensing terms? Your migration consultant may not have access to license terms and conditions for the existing session management product, so please check this.
- Should the specific feature or facility be migrated? If no migration will be performed, what is the impact on current users and business processes?
- Is there an exact correspondence available in CL/SuperSession?
- How many person-days would it take to provide the feature or facility within CL/SuperSession?

Table 4. Customer checklist with general questions

Question	Comment
General questions	
Which session management product is currently installed? Exactly which release and when will the license been expired	Also identify any official patches or modifications installed.
How many users are defined in total?	
How many users would be active at the same time?	Provide typical numbers, and a peak number.
How many users would be disconnected at the same time?	Provide typical numbers, and a peak number.
How many sessions are defined?	This number is often underestimated.
How many sessions are active at the same time?	
How many SM-STC are running?	Are they dependent or independent STC's?
How many session managers are running on the same LPAR?	This is important to know for designing the usage model.
How much storage is used above and below the line?	This provides an early indication of the storage required in the migrated system.
Is the VSAM cluster or PDS(E) are shared?	
Is the session manager running as a Generic Resources?	
Does the session manager run with Telnet 3270 server or with LU2 connections?	This affects how the CL/Supersession installation will be configured.
If IMSTM is used, is terminal security provided by IMSTM or by different means?	Will the users need specific terminals for their work or can they use any terminal?
What security system is used?	ACF2, RACF, or other?
Is a user handbook, or training guide available for end-users?	Obtain a copy of all information provided to end-users.
Features and functions	
Is any session manager mail facility used?	If so, what is the facility used for?
Is compression enabled for inbound or outbound traffic?	
What printer facilities are used? What virtual printer facilities are used?	
What INITSCRIPTs, ENDSCRIPTs, LOGON and LOGOFF scripts are defined?	For each script, provide: Script name Description and Purpose Contact details for script owner Decision on whether migration is required Factors that influenced the decision
Are there any product-specific interface requirements?	For example, a product interface may require a connection between CICS® and IMS™ or IMS and TSO.
What data is normally entered into the session manager database using automated batch jobs?	
The end-user view of the session manager	
What modifications can an end-user make?	Session parameters, key selections, command characters, and so on.

Table 4. Customer checklist with general questions (continued)

Question	Comment
What sessions may be listed on a screen for a user, and what are their descriptions?	
Numeric selections may be presented on the screen?	
PF key selections may be presented on the screen?	
Dynamic menu options appear, controlled by the security product?	
Static menu or dynamic Menu over Security System like RACF, CA-TOP SECRET or CA-ACF2	
What language support is required?	
Are there any specific command characters or keys?	
What time-outs have been set?	Include all sessions and users.
Can more than one USERID be in use at session logon time (over SESSDATA1 to 5 or Pn parameter or ACLuid in CA-TPX)?	
Is multiple logon possible with the same USERID?	
Is terminal transfer possible?	
Are there any other panel items not already mentioned?	
ACB selection	
Is selection LIFO, FIFO or using pools?	
Are ACBs kept dynamically or statically?	
Are ACBs connected using an exit?	
What kind of masking is used?	
Exit implementations	
What exits are defined?	Complete the exit details using the checklist provided in Table 5 on page 41.
Mailing within the system	
Can messages be sent to All?	
Can messages be sent to applications?	
Can messages be sent to sessions?	
Can messages be sent to active users?	
Are distribution lists provided?	
Can messages interrupt the current session?	
How are messages stored?	
Customized macros	
Which key sequences run macros within the emulation?	
How are any reports created?	

Table 4. Customer checklist with general questions (continued)

Question	Comment
Are any web to host connections created using macros?	
Location specifics	
Are there any national, international, or corporate conventions or requirements that apply?	These will normally include corporate standards or guidelines, or legal requirements.
Any other factors	
Are there any other session management configuration or usage characteristics not already reported?	

Exit checklist

The current session manager may use exits to perform various tasks. Identify all the exits used, and the function they perform. Table 5 shows example exits from IBM NetView Access Services (NVAS), and outlines the information required for migration purposes.

For each exit in the current session manager, decide:

- Is the information available under the product copyright ownership and licensing terms?
- Is the exit is still required in the CL/SuperSession configuration
- What functionality is required
- What functionality is no longer required

Table 5. Example exits from IBM NetView Access Services

Exit name	Description
EMSEAPAN or similar exit which is invoked during automatic logon or logoff.	Automatic logon and logoff exit This exit identifies where the user is in the automatic logon or logoff sequence.
EMSEATTE or a similar exit which deals with session or end-user time-outs.	Inactivity Time-out exit This exit allows sessions at terminals displaying the NetView Access Services Logon panel to be de-allocated after a given time. The exit can be used to cause de-allocation of a terminal session immediately, or to disable idle time checking for a terminal session.
EMSEDTEX or a similar exit which deals with date and time.	Date and Time exit This exit is called whenever a date and time field is being prepared for display. It contains sample coding for the US date format (MM/DD/YY) as well as for other formats. The exit allows the date and time to be displayed in whichever format is required, for example DD/MM/YY or MM/DD/YY.

Table 6. Example exits from IBM Session Manager for z/OS

Exit name	Description
EXIT11/ISZEXT11	Logon Exit to analyse the BIND data before showing SIGNON or changing the PANEL
EXIT21/ISZEXT21	SIGNon validation exit normally used for USERID/PASSWORD/PHRASE check and for the dynamic Menu creation
EXIT22/ISZEXT22	SSIGNon completion exit. Normally used to create Dynamic menu and to show the user what he is able to select on the menu.
EXIT29/ISZEXT29	SIGNOFF exit less used
EXIT31/ISZEXT31	Used for Passticket generation and putting Data together to call the Passticket generator routine. Called after the INIT script.
EXIT33/ISZEXT33	Normally used to write SMF Records
EXIT39/ISZEXT39	Normally used to write SMF Records
EXIT01, EXIT05, EXIT06, EXIT08, EXIT09, EXIT25, EXIT26, EXIT29, EXIT35 and EXIT36	Less used and need to be evaluated if it was implemented

Bibliography

Related Publications

The publications listed in the section are considered particularly suitable for a more detailed discussion of the topics covered in this migration guide.

IBM Publications

For information on obtaining these publications, see “How to get IBM Publications.”

- IBM CL/SuperSession for z/OS License Information V2.1
- IBM CL/SuperSession for z/OS Program Directory V2.1
- IBM CL/SuperSession for z/OS Readme V2.1
- IBM CL/SuperSession for z/OS Customization Guide V2.1
- IBM CL/SuperSession for z/OS User’s Guide V2.1
- IBM CL/SuperSession for z/OS Configuration Guide V2.1
- IBM CL/SuperSession for z/OS Problem Determination Guide V2.1
- IBM CL/SuperSession for z/OS Operator’s Guide V2.1
- IBM CL/SuperSession for z/OS Messages V2.1
- IBM CL/SuperSession for z/OS SSPL Programming Guide V2.1
- IBM CL/SuperSession for z/OS SSPL Reference Manual V2.1

How to get IBM Publications

You can obtain IBM publications from the IBM Publications Center at the following Web site:

<https://www-01.ibm.com/software/htp/cics/class/library/>

Notices

This information was developed for products and services offered in the U.S.A.

IBM® may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

The IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785, U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact: IBM UK Limited, Department 88013, Legal, 4NW, 76/78 Upper Ground, London, SE1 9PZ, England. Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

notices

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Programming License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at Copyright and trademark information at www.ibm.com/legal/copytrade.shtml.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a Registered Trade Mark of AXELOS Limited.

Index

A

- ACBs 26
- added fields 17
- additional parameters 21
- administration facilities
 - comparing 8
- administration rights 26
- AID keys 26
- application definition
 - statements 21
- application definition statements 15
- application profiles 14
- applists
 - dynamic 19, 23
 - static 21
- applname
 - VTAM 26
- audit requirements 12
- availability
 - need for 8

B

- bibliography 43
- business controls 4
- business processes 4

C

- checklists
 - customer 38
 - exit 41
 - migration 33
 - project workflow 33
- commands
 - system 15
- compatibility 7
- compression tools 10
- connect statements 18
 - number of 29
- connection
 - direct permission 17
 - function group 17
 - group 17
- constraints
 - migration 31
- consultants
 - migration 1
- consultation 4
- control requirements 12
- controls
 - business 4
- customer
 - checklists 38
- customized applications
 - dependencies 11
 - migrating 11

D

- data
 - raw 16
 - updating 15
 - user 25
- data profiles 14
- database
 - reports 16
- dependencies
 - session management 11
- dialog
 - INITIAL 23, 27
 - TERMINATION 23, 27
- direct permission connection 17
- dynamic applists 19, 23
- dynamic session list 19

E

- emulation scripts 10
- end-customer considerations 7
- end-user considerations 6
- estimates
 - migration 31
- evaluation
 - spreadsheet 19
- exit
 - checklists 41
- exits 9
- extended variables 25

F

- function group connection 17

G

- general recommendations 25
- generic resource
 - rules 25
- generic resources 8
- global profile 15
- global profiles 23
- group connection 17
 - naming conventions 17
- groups affected by migration 5

I

- IBM Session Manager 3
- individual profiles 23
- initial Dialog 27
- INITIAL dialog 23, 27

K

- KLHLOGON 27

- KL\$VSMS

- pool definition in 21

M

- manual migration 16, 27
- masking terminals 9
- maximum number of users 18
- migrating customized applications 11
- migration
 - assistance 1
 - checklists 33
 - concepts 3
 - constraints 31
 - effects on end-customer 7
 - effects on end-users 6
 - estimates 31
 - groups affected 5
 - manual 16, 27
 - preparing for 5
 - process 14
 - reasons for 3
 - scripts 14
 - security system data 18, 19
 - summary 31
 - tooling 13
- modifying panels 29

N

- naming conventions
 - group connection 17
- notices 45

O

- organizational preparation 5
- organizational requirements 4

P

- panels
 - modifying 29
- parameters
 - additional 21
- password phrases 27
- PF keys
 - session-related 23
- pool 21
- pool definition
 - in KL\$VSMS 21
- preparations
 - for migration 5
 - organizational 5
 - technical 7
- privileges
 - administration 26
- process
 - migration 14

- processes
 - business 4
- profile
 - global 15
 - user-related 15
- profiles
 - application 14
 - data 14
 - global 23
 - individual 23
- project workflow
 - checklists 33
- publications 43

R

- RACF 18, 19, 26
 - resource definitions 19
- RACF statements 15
- ranges
 - virtual terminal 15
- raw data 16
- recommendations
 - general 25
 - VTAM 25
- reports
 - database 16
- requirements
 - audit 12
 - control 12
 - technical 4
- resource definitions
 - RACF 19
- REXX 14
- rights
 - administration 26
- rules for the generic resource 25

S

- scripts
 - emulation 10
 - migration 14
- second userid 27
- security 29
- security system data
 - migration 18, 19
- session list
 - dynamic 19
- session management 3
 - dependencies 11
- session-related
 - PF keys 23
- session-related variables 23
- signon
 - user 27
- spreadsheet 14
- spreadsheet evaluation 19
- spreadsheets 16
 - added fields 17
- SSPL 25
- statements
 - application definition 15, 21
 - connect 18
 - number of 29
 - RACF 15

- statements (*continued*)
 - system definition 21
- static applists 21
- structured session procedure
 - language 25
- superior group 19
- system commands 15
- system definition
 - statements 21
- Systems Network Architecture (SNA) 3

T

- task analysis 6, 7
- TCP/IP 3
- technical preparation 7
- technical requirements 4
- terminal assignment 9
- TERMINATION dialog 23, 27
- Tivoli NetView Access 3
- tooling
 - migration 13
- tools
 - compression 10
- trademarks 46
- triggers
 - maintaining 26

U

- universal group 18
- updating
 - data 15
- user
 - signon 27
- user data 25
- user variables 25
- user-related profiles 15
- userid
 - second 27
- users
 - maximum number of 18

V

- variables
 - extended 25
 - session-related 23
 - user 25
- Virtual Telecommunication Access Method (VTAM) 3
- virtual terminal
 - ranges 15
- virtual terminal assignment and masking 9
- VTAM applname 26
- VTAM recommendations 25



Printed in USA

SC27-9020-01

