SAFR - Scalable Architecture for Financial Reporting
PDF from SAFR Information Center 4.15.00

Overviews
SAFR Workbench 4.15.000

SAFR - Scalable Architecture for Financial Reporting
PDF from SAFR Information Center 4.15.00

Overviews
SAFR Workbench 4.15.000

Note
Before using this information and the product it supports, read the information in "Appendix: Notices" on page 193.

Fourth Edition

This edition applies to version 4, release 15, modification 000 of SAFR Workbench (part of SAFR product number 6949-17P) and to all subsequent releases and modifications until otherwise indicated in new editions.

Contents

Chapter 1. Overviews

SAFR overview - START HERE

01 Summary of this topic

The sections in this topic are as follows:

10 What is SAFR?

SAFR stands for **Scalable Architecture for Financial Reporting.**

SAFR is pronounced "SAF-er". Rhymes with "matter".

What does Scalable Architecture for Financial Reporting mean?

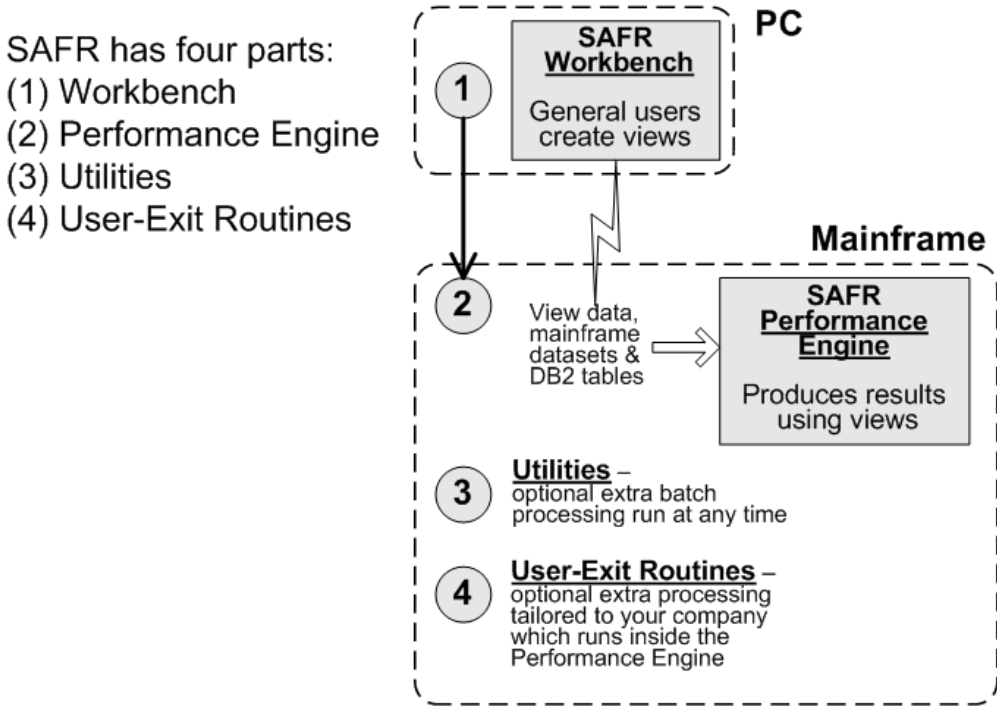
The word **scalable** means that as more demands are made for financial data and reporting, the existing financial processing can produce the results without major redesigning or rebuilding of the financial systems.

Financial reporting systems typically display what are called "dis-economies of scale". This means that as a company requires more summaries of specific financial issues, the complexity and cost of creating the computing processes is larger than expected. With SAFR, it does not matter how much larger your company grows or how complex the financial reporting, SAFR remains a fast, efficient and accurate method to generate all your financial reports.

There are two main benefits from using SAFR:

- **Easy access to financial details.** Traditional financial systems produce the financial reports, but if you want to question the reasons behind one of the figures, this takes long and costly manual investigation. With SAFR, the details behind the financial figure are retained, so finding the reasons behind a figure is fast and easy. This is because SAFR does not "throw away" the detailed transactions like a traditional financial system does.
- **Faster processing.** Is your company unable to obtain reports from a data warehouse because the processing takes too long? Is your overnight batch window becoming too short to finish all processing required? Do SQL queries take too long to produce results? These are all situations in which SAFR can provide answers much faster.

20 SAFR software



SAFR consists of four parts:

1. **SAFR Workbench.** This PC software creates and modifies views. For more, see topic "**Workbench (WE) overview**". This topic is elsewhere in this PDF - see the table of contents.
2. **SAFR Performance Engine (PE).** This mainframe software produces the results of SAFR using views. The Performance Engine runs under the operating system **z/OS** as a batch job using JCL. For more, see topic "**Performance Engine (PM) overview**". This topic is elsewhere in this PDF - see the table of contents.
3. **Utilities.** These are mainframe batch programs that deliver extra processing. These steps can be run anytime.
4. **User-Exit Routines.** Also called "Exits". These are mainframe programs that are tailored to your company. User-Exit Routines run inside the Performance Engine. For more, see topic "**User-Exit Routines overview**". This topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Common Key Buffers overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is a large cardinality file?

Cardinality refers to how many records in a file. For example, if a customer file has 20 million records, then the customer key must allow for at least 20 million unique key values. Large cardinality files create special processing problems, which are addressed by Common Key Buffers.

20 What is a Common Key Buffer?

This is a method to improve SAFR performance when two or more sorted logical files share the same key (or part of a key). Common Key Buffers are especially useful for processing large cardinality files.

The best way to understand is with an example. Consider an insurance company situation where the data files are as follows:

- **Claim** File, with a primary key of Claim Number. This describes who is making a claim and other broad details.
- **Claim Vehicle** File, with a primary key of Claim Number and a Vehicle Number. This describes the vehicles that are involved in the claim.
- **Claim Result** File, with a primary key of Claim Number. This describes whether the claimant is at fault and other details.

An example of the data of the three files is below. The Common Key Buffer shows all the records for Claim Number 124. The number of Claim Vehicle records is variable, with at least one per claim.

				Common Key Buffer		
Claim File	(Key)	Claim Num	123	**124**	125	126
		Claimant ID	987AB	135CD	246EF	357GH
		▪				
Claim Vehicle File	(Key)	Claim Num	123	**124**	**124**	125
	(Key)	Vehicle Num	1	1	2	1
		Vehicle type	SEDAN	TRUCK	SUV	PICKUP
		▪				
Claim Result File	(Key)	Claim Num	123	**124**	125	126
		At fault?	YES	NO	UNKNOWN	NO
		▪				

40 Example: Complex search

SCENARIO: the US Department of Homeland Security receives foreign intelligence reports that a suspect has gained entry to the USA in the last two years. All that is known about the suspect is some information the suspect told to a friend who was overheard bragging about it.

The information is about a strange coincidence when the suspect arrived in the US:

- The flight number was the same as the day of the month of the flight.
- The flight number was the same as the seat number for the suspect.

- The flight number was exactly the same as how many minutes the flight was late.

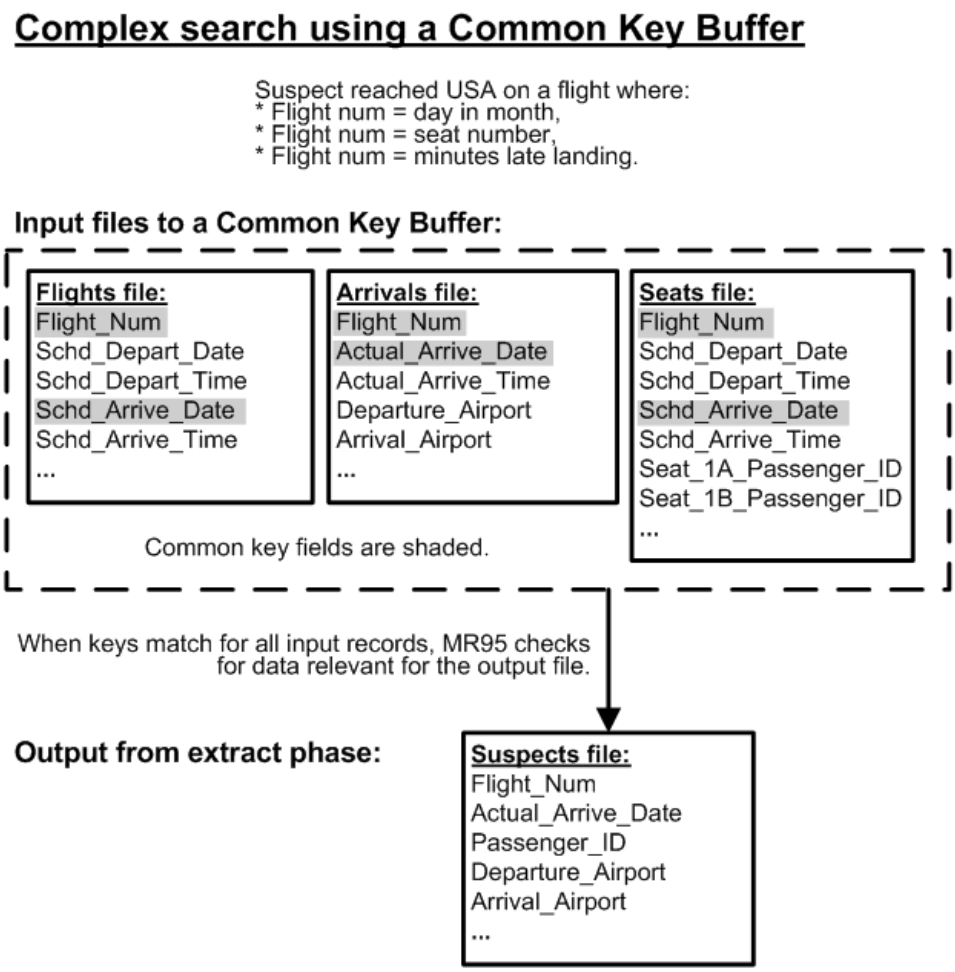
The challenge is to go through all the flights into the US in the last two years and search for all flights into the USA and find passengers that fit this pattern.

Below is way to use the extract phase to find the solution.

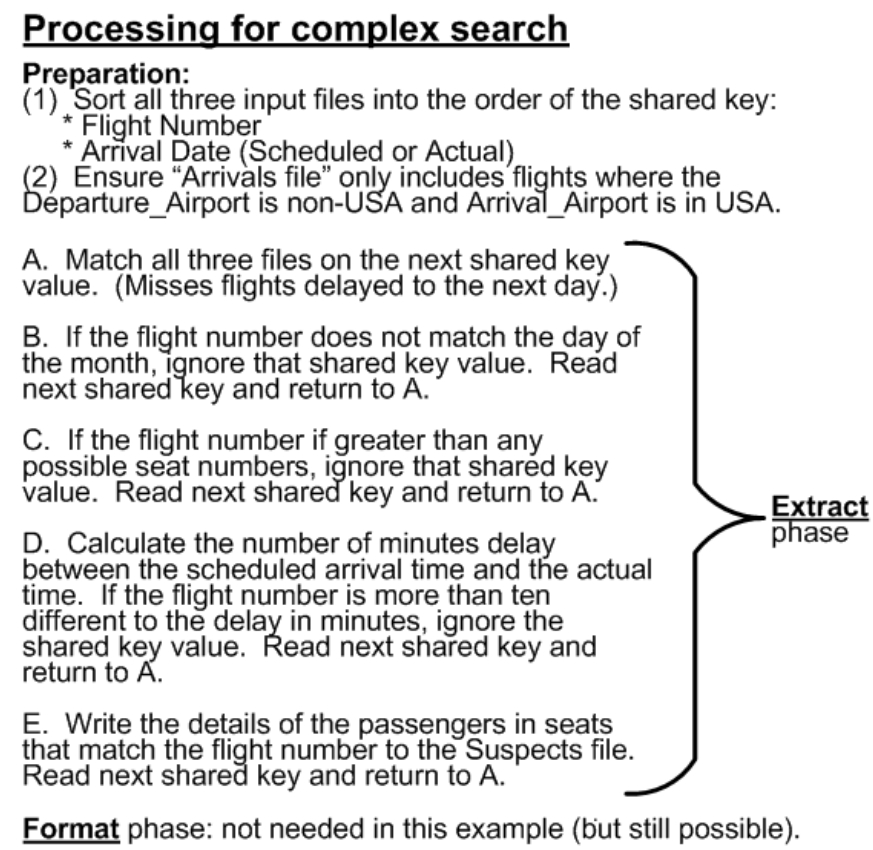
The solution reads three files:

- Flights file - to provide the scheduled arrival time for a flight (to calculate the delay in landing),
- Arrivals file - to determine if the flight was into the USA and provide actual arrival time for a flight (to calculate the delay in landing),
- Seats file - to provide passenger data for seats (to find the passengers sitting in seats with the same number as the flight number).

These three files are read in a technique called a "**Common Key Buffer**". This means that the three files share part of a key - in this case a flight number and the arrival date. The Common Key Buffer coordinates the read of all three files, so that the program MR95 can see the same key value in all three files. For example, flight 123 on a certain date is a common key to all three files. When MR95 can see matching records then MR95 can determine if any passengers meet the search requirements, and write the relevant data to a Suspects file.



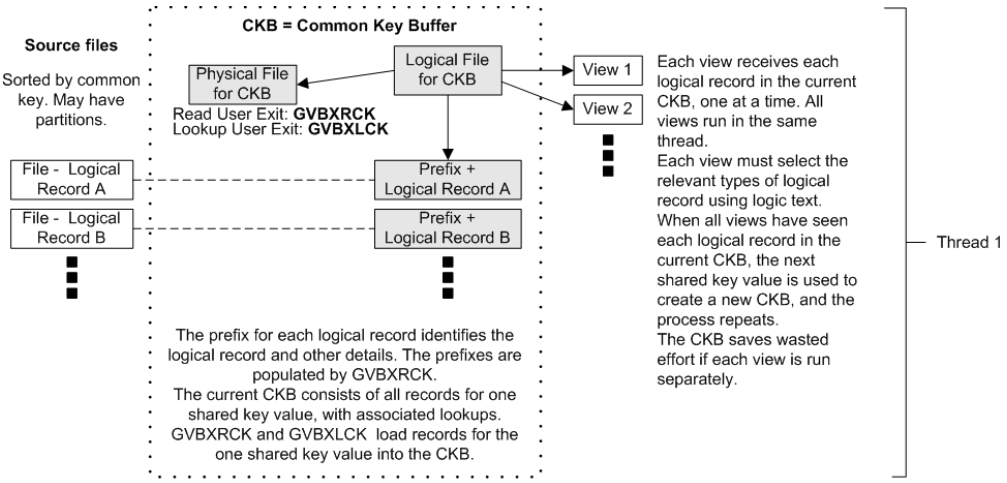
Below are the processing steps for this complex search. The diagram shows the extract phase and format phase for this complex search.



60 Components of a Common Key Buffer

In general, there are at least two source files (each file may have multiple partitions). Each source file is sorted into the common key order.

The following are the minimum components to make up a Common Key Buffer:



80 Strengths and weaknesses

The **strengths** of a common key buffer are:

- Can handle large cardinality files.
- Allows high performance lookups because searches are limited to the records in the Common Key Buffer (rather than to entire files).
- Loads into memory on a minimum number of records at a time.
- Can handle complex groups of logical records.
- Saves indexed reads (by using sequential reads).
- Can be used in combination with pipes.
- Can be used in combination with tokens.

The **weaknesses** of a common key buffer are:

- Calls user exit routines, which have an overhead (whereas pipes can avoid this overhead).
- All reader views are in the same thread as the Common Key Buffer. This can be avoided if one reader view writes to a Pipe, which is read in a different thread.
- Assumes that the processing for each value of the common key is independent of any other value of the common key.

Overall, a common key buffer is best used when there are one or more large cardinality files, or when pipes or tokens are not practical.

See also topics "**Pipes overview**" and "**Tokens overview**". These topics are elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

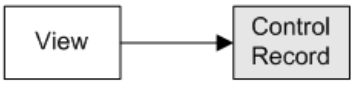
Control records overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is a control record?

A control record contains data that affects financial reporting in a view.

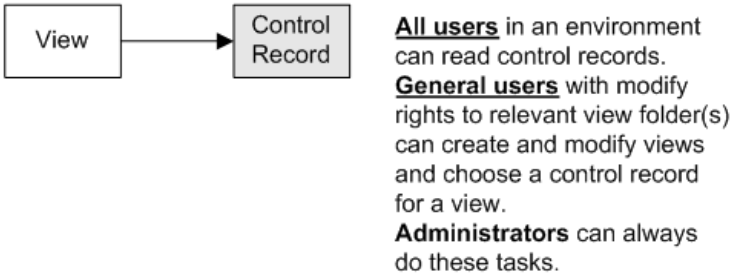


A control record can be used in multiple views. An environment must have at least one control record, and can have many control records (one for each variation of financial reporting required).

The **most important** fields in a control record are:

Field	Definition
Maximum Extract File Number	The maximum number of extract files that can be created in the Extract Phase. Must be greater than zero and up to nine digits.
First Fiscal Month	The calendar month in which the fiscal year starts. A number in the range one to 12, representing the months January to December. (Optional.)
Beginning Period	A month numbered from zero to 11. This is an alternative to First Fiscal Month which is from one to 12. (Optional.)
Ending Period	A month numbered from zero to 11. Must be larger than Beginning Period, if set. (Optional.)

20 How do I use a control record?



Select a control record in **views** you are working on. To learn more about views, read these topics:

- **Views overview**
- **Views - advanced overview**
- **Creating views**
- **Modifying views**

The overview topics are elsewhere in this PDF - see the table of contents. To find the other topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I know which control record to use?

All users can read control records, to see the values. In the **Navigator**, click on "**Control Records**", and in the **Metadata List** double click on any control record.

See your system or environment administrator also if the environment needs an extra control record.

50 How do I create or modify a control record?

Only system and environment administrators can create or modify control records - see these topics:

- **Creating control records**
- **Modifying control records**

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

For a complete discussion of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

90 How do I delete a control record?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Environments overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is an environment?

An environment is a logical collection of data within the SAFR Workbench.

The actual environment data is stored in a SAFR Database. The data refers to some mainframe data in your company.

System administrators define environments. There may be separate environments for Accounts, Sales, Stock Inventory and so on.

All users of the SAFR Workbench login to one environment per session.

A system administrator logs into one environment per session while still retaining some access to all environments at all times.

General users and environment administrators log into one environment per session, and can only access that environment during that session.

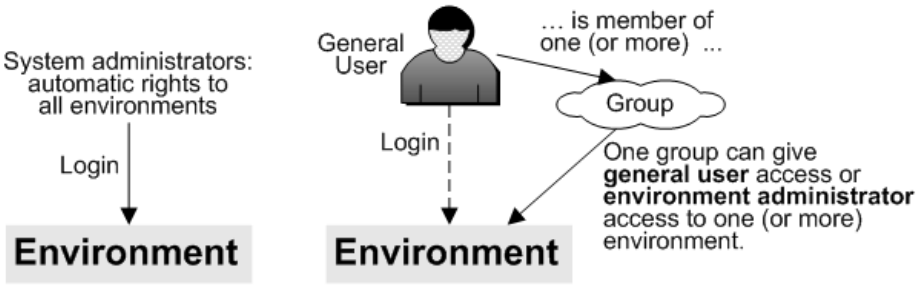
When already logged into the workbench, to change environment or change group for the same environment a user clicks **File -> Return to login...**

Separate environments provide these benefits for your company:

- **Provide access controls** for sensitive data in your company. The most sensitive data can be in an environment with the most restrictive access controls.
- **Simplify the job of general users** in that environment. For example, general users in the Sales environment are not distracted by other mainframe data and can quickly find work done by other users in that environment.

20 How do I use an environment?

All users choose an environment during **login** to the SAFR Workbench. During login, a general user must choose a group (if a choice is available). An environment administrator must choose the group that provides the administrator access rights.



General Users choose an environment during login in order to get work done. The general user must be a member of a group that has access to that environment.

System administrators and environment administrators choose an environment during login to prepare that environment for general users.

For more about login, read topic "**Logging into the SAFR Workbench**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

For a complete discussion of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

30 How do I know which environment to use?

If you are a ...	**then ...**
General User	See your system administrator, who can specify an environment to use. It is possible the system administrator may also give a choice of **groups** to use with that environment.
System or environment administrator	Speak to other administrators. All administrators must agree on the environments and associated groups for your company.

For more, see topics "**Environment - advanced overview**" and "**Groups overview**". These topics are elsewhere in this PDF - see the table of contents.

40 What are the components in an environment?

An environment can contain components of these types:

- Control records,
- Global fields,
- Groups,
- Logical files,
- Logical records,
- Lookup paths,
- Physical files,
- Users,
- User-exit routines,
- Views,
- View folders.

For more on what these components are, see the **overview** for each component.

There is no limit to the number of examples of each type of component in an environment. For example, an environment may have at least ten examples of each of the above types.

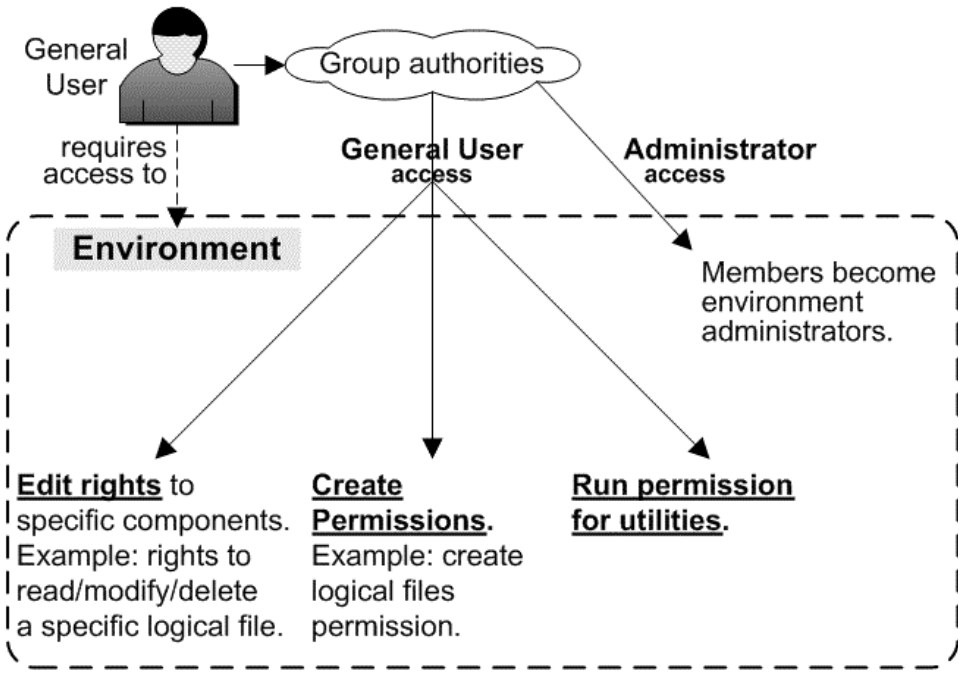
Environments and their components are examples of metadata - see topic "**Metadata overview**". All these overviews are elsewhere in this PDF - see the table of contents.

50 Environments and groups

Every environment must be associated with at least one **group**. A general user or environment administrator gain **access to an environment** by becoming a **member of a group**.

Groups have authorities to access one or more environments. Group authorities for each environment can be any of the following:

- **Administrator** access authorities. This allows members of that group to be environment administrators.
- **General user** access authorities, as follows:
  - Edit rights to specific components. Example: rights to read/modify/delete a specific logical file.
  - Create permissions for types of components Example: create logical files permission.
  - Run permission for utilities.



One group may have only **read access** to all components in an environment and no create rights. Another group may have **more rights** to individual components. User rights in an environment are controlled by the group selected during login to that environment.

For more, see topics "**Environments - advanced overview**" and "**Groups overview**". These topics are elsewhere in this PDF - see the table of contents.

For a complete discussion of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Environments - advanced overview

This topic assumes you are familiar with the topic "**Environments overview**". That topic is elsewhere in this PDF - see the table of contents.

01 Summary of this topic

This topic covers the following:

10 Group permissions to create types of components

Group members receive create permissions for types of components.

The group create permissions possible are as follows:

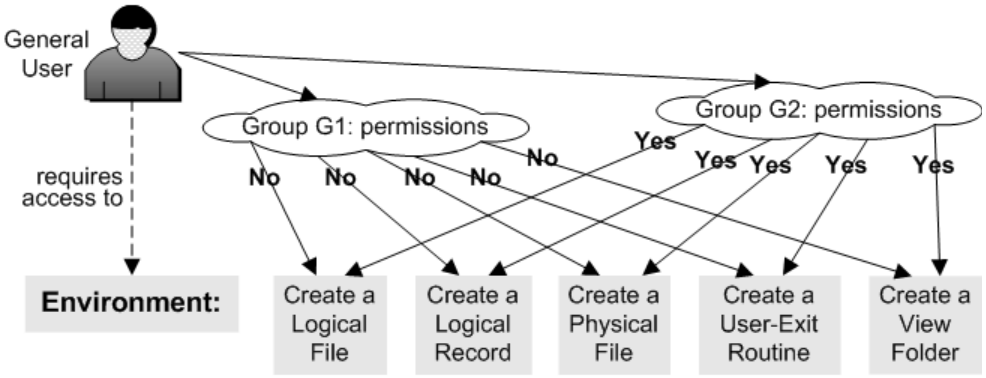
- Create Logical Files,
- Create Logical Records,
- Create Physical Files,
- Create User-Exit Routines,
- Create View Folders.

The above permissions can be applied to a general user. Administrators in an environment always have these permissions.

Notice there is no "create" right for control records and global fields - these are only created by administrators. General users can make use of existing control records and global fields, but cannot create, modify or delete them.

Different groups have different authority, and this can give a choice to a general user. For example:

- **Group G1** is for regular reporting, so this group cannot create any new components. This applies to general users who login to this environment using group G1.
- **Group G2** is for update of data, so this group can create any new components required. This applies to general users who login to this environment using group G2.



In the above example, the general user can **change rights to create items** in the environment by the **choice of group** during login to that environment.

20 Group permissions to run a utility

Group members can receive a permission to run utilities.

The utilities available in the SAFR Workbench are:

- **Batch Activate Lookup Paths.** This utility checks lookup paths are ready to use in an environment, and if possible sets the status to "active". For more, see topic "**Lookup Paths overview**". That topic is elsewhere in this PDF - see the table of contents.
- **Batch Activate Views.** This utility checks views are ready to run in the SAFR Performance Engine, and if possible sets the status to "active". For more, see topic "**Views overview**". That topic is elsewhere in this PDF - see the table of contents.
- **Migration Utility.** This utility copies selected metadata from a source environment to target environment in the same SAFR Database. For more, see topic "**Migrate metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

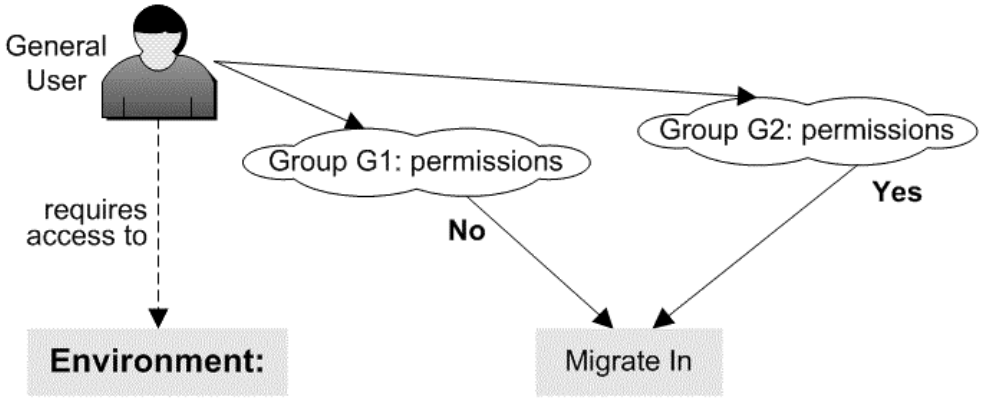
There is one group run permission:

- **Migrate In**. This provides access to all three utilities:
  - **Migration Utility** where the target environment is this environment ,
  - **Batch Activate Lookup Paths** and **Batch Activate Views** in this environment.

The above run permission can be applied to a general user in one or more environments. Administrators in an environment always have this run permission.

Different groups have different authority, and this can give a choice to a general user. For example:

- **Group G1** is for users who run established views and cannot migrate any metadata. Users in this group cannot run any utilities.
- **Group G2** is for users who can migrate metadata from other environments. Users in this group can run all three utilities.



In the above example, the general user can **change rights to run utilities** in the environment by the **choice of group** during login to that environment.

30 Groups and edit rights to specific components

Group members receive edit rights to specific components of metadata.

The specific components must be of the following types:

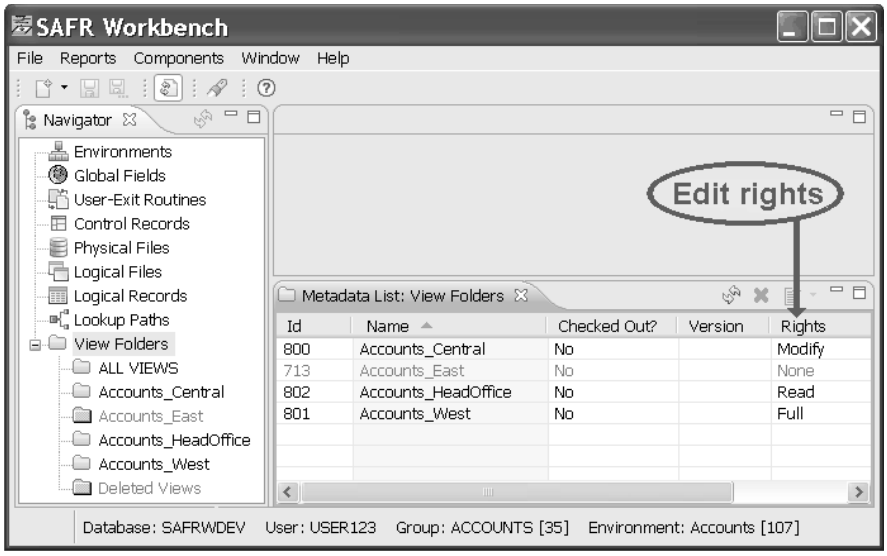
- Logical File,
- Logical Record,
- Physical File,
- User-Exit Routine,
- View Folder.

The edit rights possible are:

- **No rights at all**.
- **Read** right which allows both display and usage of an item. For example, a user needs the read right to a logical record in order to refer to that logical record in a view.
- **Modify** right which implies Read as well.
- **Delete** right which implies the Modify and Read rights as well. This right is also called "**Full**" rights.

The above edit rights for a specific component can be applied to a general user. Administrators always have full rights to all components.

Edit rights can be seen in column "**Rights**" in the **Metadata List**. For example, if you click on "**View Folders**" in the **Navigator**, the Metadata List may appear as follows:

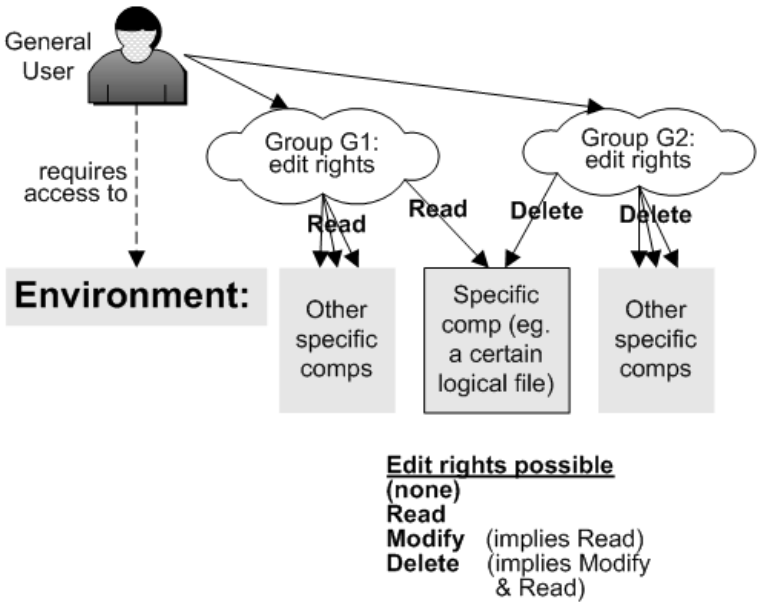


A group can have different edit rights to each individual component.

Different groups have different authority, and this can give a choice to a general user. For example:

For example:

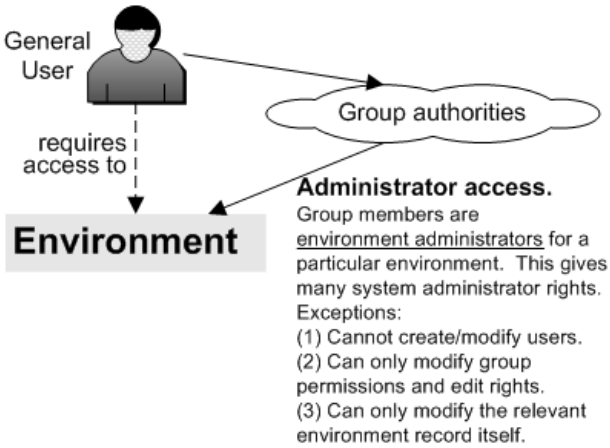
- **Group G1** is for reporting, so this group has the read right to a specific logical file.
- **Group G2** is for update of data, so this group has the delete right (which is all rights) to a specific logical file.



In the above example, the general user can **change access to specific components** in the environment by the **choice of group** during login to that environment.

40 Groups and administrator access

A group can provide a user with **administrator access to an environment**. This means all members of that group become **environment administrators** in that environment.



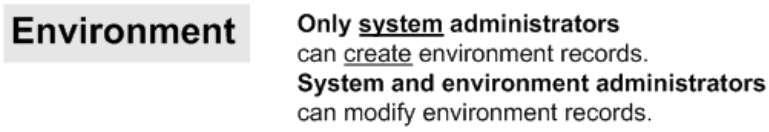
Environment and system administrators have all the create permissions, run permission and edit rights that are possible.

An environment administrator and has almost the same rights as a system administrator. The difference is that an **environment** administrator:

- Cannot create, read, modify or delete users.
- Can only modify group permissions and rights. Cannot give administrator access. Cannot change group membership. Cannot create or delete groups.
- Can only modify the environment record itself for the relevant environment. Cannot create or delete environments. Cannot modify environment records where there is no environment administrator access.

50 How do I create or modify an environment?

This section covers the environment record itself, which names the environment.



System administrators can **create, read, modify and delete** environment records.

Environment administrators can **modify** the environment record.

For help with create or modify of environment records, read these topics:

- "**Creating environments**",
- "**Modifying environments**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

For information about creating and modifying different components in an environment (like logical records), see the overview for that type of component.

60 Copying or deleting metadata

Before performing the above tasks, you must be aware of any metadata item that exists in multiple environments - see the next section.

To copy or delete metadata inside one environment, or to delete the environment itself, see these topics:

- "**Copying metadata**",
- "**Deleting metadata**",
- "**Clear environment**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

65 Environment Checker Report

This report shows all environments for a particular a metadata item name. This is useful when planning to update or delete a metadata item, because the same action may be necessary for the same item name in other environments.

All users can generate this report, using one of these topics:

- FAQ topic "**How do I generate an Environment Checker Report?**"
- Task "**Finding a metadata item name in all environments**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

70 Minimum components for an environment

Every environment requires these minimum components:

- **One control record**. Control records are created and modified only by administrators. Users of an environment always have read access to all control records for that environment.
- **One view folder**.
- **One group that has access to the environment**.

The SAFR Workbench ensures that these minimum components exist for every environment at all times. For example, during create of a new environment, administrators must nominate names for each of these three minimum components.

80 Environment Security Report

All users can access this report. To see a report that lists the groups in one or more environments, use one of these topics:

- FAQ topic "**How do I generate an Environment Security Report?**"
- Task "**Generating reports on metadata**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

For a complete discussion of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

90 How to delete an environment

Environments can only be deleted if all metadata in that environment has been cleared.

See task "**Clear environment**". This topic is elsewhere in this PDF - see the table of contents.

Only a system administrator can delete an environment, with two conditions. Firstly, the environment must contain no metadata, and secondly the system administrator cannot be currently logged into that environment.

95 The best way to use environments

Environments aim to achieve the following goals:

- **Accuracy:** provide only the appropriate data for users for their work. Too much data risks confusion and unintended update of data. Too little data stops general users from doing their work.
- **Flexibility:** allow some extra environments for unusual work situations that require unusual data access for a short period of time.
- **Simplicity:** fewer environments means less environments for administrators to maintain.

These goals are often in competition: for example to achieve goal of accuracy perfectly may require a separate environment for each user which contradicts the goal of simplicity. Your company must find the appropriate choice of environments that balances the above goals.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Export metadata overview

01 Summary of this topic

The sections in this topic are as follows:

02 Knowledge you need first

This topic assumes you are familiar with these topics:

- "**Metadata overview**",
- "**Metadata - advanced overview**".
- "**WE Security overview**".

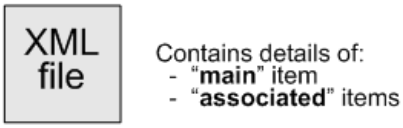
These topics are elsewhere in this PDF - see the table of contents.

03 What is export of metadata?

Export means copying a selected metadata item into an XML file.

The selected metadata item can be called the "**main**" item. Many "**main**" items can be exported at one time, and each "**main**" item is exported to a separate XML file. The exported XML file consists of the following:

- Details of the "**main**" item.
- Details of metadata associated with the "main" item. These can be called "**associated**" items. There could be zero, one or many of these.



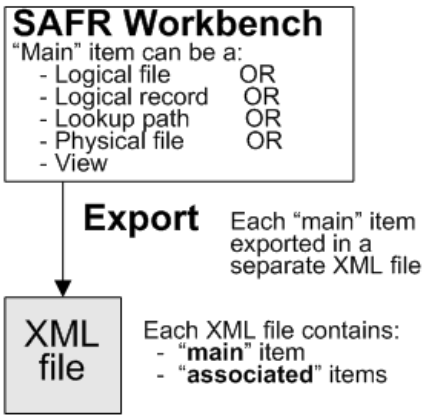
Only certain types of metadata can be a "**main**" item in an export:

- Physical file,
- Logical file,
- Logical record,
- Lookup path,
- View.

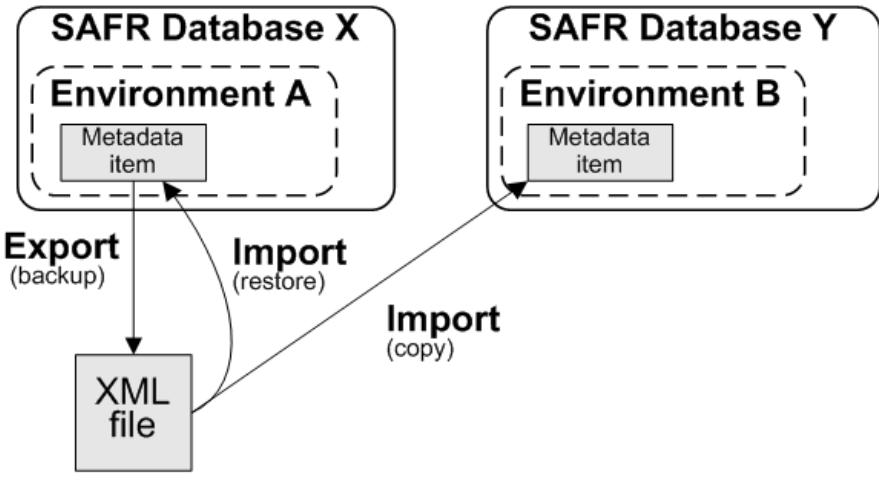
The table below shows the possible "associated" items:

"Main" item	**Possible "associated" items**
Physical file	User-exit routine
Logical file	Physical file, user-exit routine
Logical record	Logical file, physical file, user-exit routine
Lookup path	Logical record, logical file, physical file, user-exit routine
View	Control record, lookup path, logical record, logical file, physical file, user-exit routine

Notice that a control record cannot be a "main" item in an export, but a control record can be an "associated" item for backup of a view. To copy a control record for one environment to another, you must copy a view that contains that control record.



04 Why is export useful?



There are two reasons to export metadata into XML files:

- Restore from a backup of a metadata item.
- Copy an exported metadata item into a target environment. The target environment may be in the same database or a different database.

All backup, restore and copy options are outlined in help topic "**Metadata - advanced overview**". That topic is elsewhere in this PDF - see the table of contents.

10 Perform an export

Do the following:

1. Ensure you know the "main" item(s) to export in the relevant environment and database.
2. Choose a location for the XML files - see section “20 Location for storing XML files.”
3. Perform the export using topic "**Exporting metadata**" in the Administration Guide. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
4. Perform section “30 Move XML file to a permanent location” on page 23.

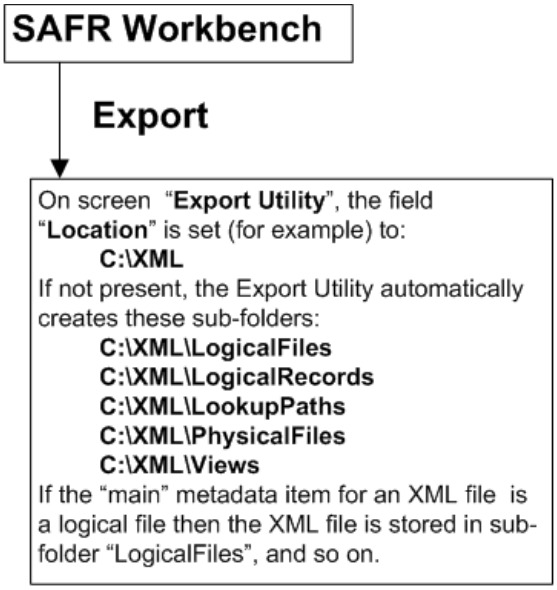
20 Location for storing XML files

On the "Export Utility" screen, specify a value for the field **"Location"**, for example "**C:\XML**".

If not already present, the Export Utility creates these sub-folders:

- LogicalFiles
- LogicalRecords
- LookupPaths
- PhysicalFiles
- Views

If the "main" item for an XML file is a **logical file**, then the XML file is stored in the sub-folder **LogicalFiles**, and so on.



There are two ways to choose a location for XML files:

- **EITHER** specify the permanent location for the XML files every time you export,
- **OR** specify a "holding" folder for the XML files. After export is complete, move or copy the XML files to a permanent location.

Permanent locations for backups can be organized for later reference, for example:

- **C:\Backups\Environments\Env_A\YYMMDD\Views**
- **C:\Backups\Selected_Items\Env_A\Views\YYMMDD**

where "**Env_A**" is an environment name and "**YYMMDD**" is a date.

30 Move XML file to a permanent location

This section is optional. Ensure you have read section "20 Location for storing XML files" on page 22.

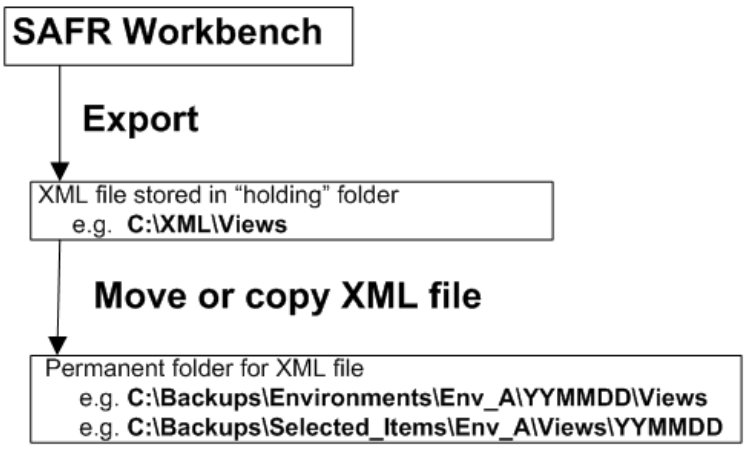
When the export is complete, consider the folder where the XML file is stored. If this folder is an appropriate permanent location for the XML file, then no further action is necessary.

You may take action if that folder is a 'holding' folder, for example **C:\XML**. Export creates an XML file in a sub-folder, for example **C:\XML\Views**.

The possible action is to move or copy the XML file to a permanent backup folder. Examples of a permanent backup folder are:

- **C:\Backups\Environments\Env_A\YYMMDD\Views**
- **C:\Backups\Selected_Items\Env_A\Views\YYMMDD**

where "**Env_A**" is an environment name and "**YYMMDD**" is a date .



40 Example of an XML file

An example of part of an XML file is below. The example is for a logical file called "**Stock_1**":

Stock_1[1345].xml - Notepad

File Edit Format View Help

```
<LogicalFile-1345>
    <LogicalFile>
        <Record>
            <ENVIRONID>154</ENVIRONID>
            <FILEID>1345</FILEID>
            <FILENAME>Stock_1</FILENAME>
            <EFFDATE></EFFDATE>
            <TERMDATE></TERMDATE>
            <COMMENTS></COMMENTS>
            <CREATEDTIMESTAMP>2010-12-08 06:16:46.262221</CREATEDTIMESTAMP>
            <CREATEDUSERID>ADMIN</CREATEDUSERID>
            <LASTMODTIMESTAMP>2010-12-08 06:16:46.262221</LASTMODTIMESTAMP>
            <LASTMODUSERID>ADMIN</LASTMODUSERID>
        </Record>
    </LogicalFile>
    <File-Partition>
        <Record>
            <ENVIRONID>154</ENVIRONID>
            <XFILEPARTITIONID>14363</XFILEPARTITIONID>
            <CHILDPARTITIONID>8402</CHILDPARTITIONID>
            <PARTSEQNBR>1</PARTSEQNBR>
            <PARENTFILEID>1345</PARENTFILEID>
            <CHILDFILEID>0</CHILDFILEID>
            <CHILDTYPE>0</CHILDTYPE>
            <CREATEDTIMESTAMP>2010-12-08 06:16:46.625069</CREATEDTIMESTAMP>
            <CREATEDUSERID>ADMIN</CREATEDUSERID>
            <LASTMODTIMESTAMP>2010-12-08 06:16:46.625069</LASTMODTIMESTAMP>
            <LASTMODUSERID>ADMIN</LASTMODUSERID>
        </Record>
    </File-Partition>
    <PhysicalFile>
        <Record>
            <ENVIRONID>154</ENVIRONID>
            <PARTITIONID>8402</PARTITIONID>
            <PARTITIONNAME>Stock_Cat_1</PARTITIONNAME>
            <FILETYPECD>DISK</FILETYPECD>
```

In the example above the "**main**" item is a logical file, and at least one physical file is an "**associated**" item.

50 Structure of XML files

See topic "**XML structure for metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

60 Backup an entire environment

You must have the appropriate rights to backup an entire environment. System and environment administrators always have the appropriate rights to do this.

General users can backup an entire environment if the **group selected during login** has at least **Read** rights for the "main" and "associated" items. Note that to export a view a general user must have read rights to the view folder. See your system administrator if you require more rights.

To backup an entire environment, do the following:

1. Choose the location for the XML files - see section “20 Location for storing XML files” on page 22.
2. In the workbench under **Administration** or **Components**, click **Export...**
3. Select an **Environment** from the drop down box. System administrators can choose from all environments. Environment administrators can only choose the environment selected during login. General users can choose from the environments that the **group selected during login** has access to.
4. Select a **Component Type** from the drop down box. This must be one of these types:
   - Physical file,
   - Logical file,
   - Logical record,
   - Lookup path,
   - View.
5. Wait a moment. The table of **Component(s)** is populated. System and environment administrators can see all the components of the selected type in that environment. For general users, the only components listed are where the group has read or modify or delete rights.
6. For component types of **Logical Record, Lookup Path and View**, click the button for **Both**.
7. Click **Select All**.
8. If required, modify the **Location** field. If you type a new folder name that does not exist, this is created when you perform the next step. Alternatively, click **Browse** to select an existing folder. When you perform the next step, the folder you specify in **Location** is automatically given a sub-folder named for the Component Type. That sub-folder is where the XML file is stored.
9. Click **Export**. If this button is grey, check you have completed the previous steps.
10. A value in the **Result** column of "**Pass**" indicates that the export is complete. This means that the folder for **Location** now has a sub-folder for the **Component Type** and an XML file exported for each item exported. Each XML file in that sub-folder is named **componentName[id[.xml**.

The **Result column** shows if any errors affected the export of the XML file for each component. The errors may be caused by the general user not having read rights to a component. Errors may also be caused by a system input/output error. **Click on the row for a component** to see any error messages in the **Errors section** on the right.

For errors or messages on this screen, see topic "**Export Utility errors**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

11. Return to step 4 and select another component type. Do not proceed to step 12 until export is complete for all the component types.
12. When all backups are complete, use Windows Explorer to go to the **folder given in the Location field.** This folder has sub-folders for each component type. Each XML file has name **componentName[id[.xml** in a sub-folder. If required, perform section “30 Move XML file to a permanent location” on page 23.

70 Comparison: export and import for WW and WE

This online help is provided for the SAFR Workbench called **WE (Workbench Eclipse)**.

The export and import functions were provided in an older version of the SAFR Workbench called **WW (Windows Workbench).**

At a high level, the export and import in WW and WE are similar but not compatible. For example, an XML file created by an export in WW cannot be imported into WE.

A **comparison of the export functions of WW and WE** is as follows:

- WE provides export for only these metadata types:
  - Physical file,
  - Logical file,
  - Logical record,
  - Lookup path,
  - View.
- XML files exported from WE have a different format for date/time stamp compared to XML files exported from WW. The formats are as follows:
  - WW format in the USA is: **MM/DD/YYYY HH:MM:SS AMPM** (at midnight the time part is omitted)
  - WW format in some other countries is: **DD/MM/YYYY HH:MM:SS AMPM** (at midnight the time part is omitted)
  - WE format is: **YYYY-MM-DD HH:MM:SS** (24 hour clock)
- WE does not request selecting a SAFR database at the start of the export function. WE export uses the database for your current session.
- WE provides a choice of location for the XML file that is generated by the export.

A **comparison of the import functions of WW and WE** is as follows:

- WE import performs more checks on the XML files and their relationship with existing metadata in the target environment.
- WE import is more consistent and reliable.

- WE import ensures that serious potential problems are blocked and the import cannot proceed.
- WE import provides a warning for minor potential problems, and gives a choice to proceed or cancel the import.
- Overall, the WE import function preserves the integrity of the metadata in the target environment after import is complete.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

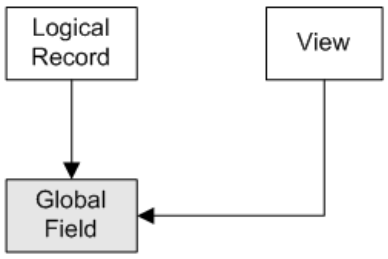
Global fields overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is a global field?

A global field is a **field used in logical records and views**. Global fields belong to the environment that contains them.



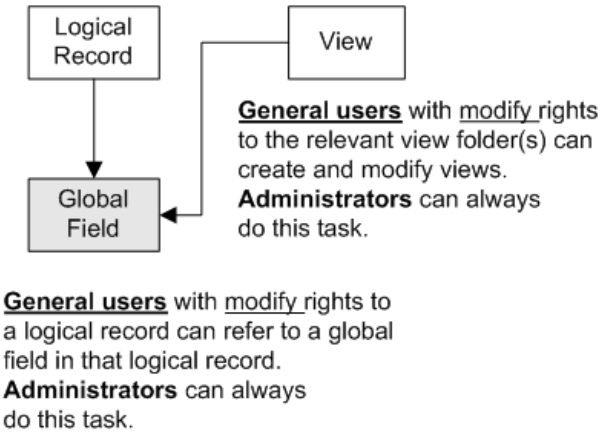
A global field is defined once and used many times. This saves time and avoids consistency problems in logical records and views.

For more introduction see these overviews:

- "**Logical records overview**",
- "**Views overview**",
- "**Views - advanced overview**".

These topics are elsewhere in this PDF - see the table of contents.

20 How do I use a global field?



Use a global field in logical records and views.

A general user can use a global field in a logical record if the general user has at least the modify right to the logical record.

A general user can use a global field in a view if the general users has at least the modify right to the relevant view folder that contains the view.

Administrators can always use Global Fields.

For more on how to use a global field, see these topics:

- "**Creating logical records**",
- "**Modifying logical records**",
- "**Creating views**",
- "**Modifying views**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

System administrators and environment administrators can always **use global fields to create or modify fields in a logical record**.

General users can **use global fields to create or modify fields in a logical record** if the group selected during login has the following authorities:

- **Create Logical Record** permission in the relevant environment (when creating a new logical record).
- **Modify or Delete rights to the relevant logical record** in that environment (if modifying an existing logical record).

For more on these authorities, see topics "**Groups overview**" , "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

Once the group has the above authorities, users in that group can **use global fields to create or modify fields in a logical record** by using the tasks below, which are **administrator tasks:**

- "**Creating logical records**",
- "**Modifying logical records**".

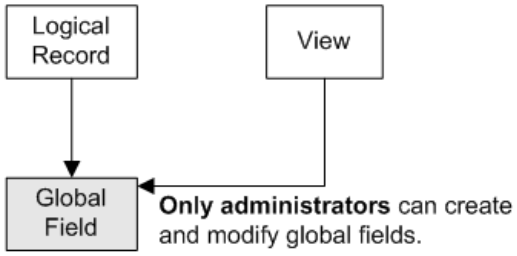
To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I know which global field to use?

All users can read global fields, to see the values. In the **Navigator**, click on "**Global Fields**", and in the **Metadata List** double click on any global field.

See your system or environment administrator also if the environment needs an extra global field.

50 How do I create or modify a global field?



Only system or environment administrators can create or modify global fields - see these topics:

- "**Creating global fields**",
- "**Modifying global fields**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

For a complete discussion of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

90 How do I delete a global field?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Groups overview

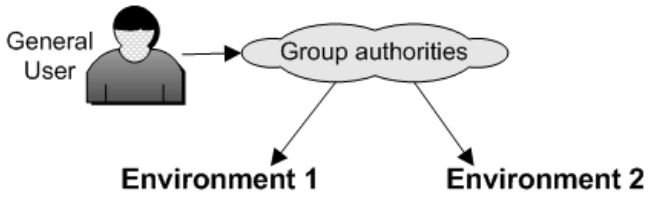
01 Summary of this topic

The sections in this topic are as follows:

10 What is a group?

A group is a collection of users in the SAFR Workbench.

The function of the group is to **provide access to one or more environments** associated with the group. The group can have **different access to each environment** associated with the group. All the users in that group get the access authorities of that group.



System administrators define groups to be used by general users and environment administrators.

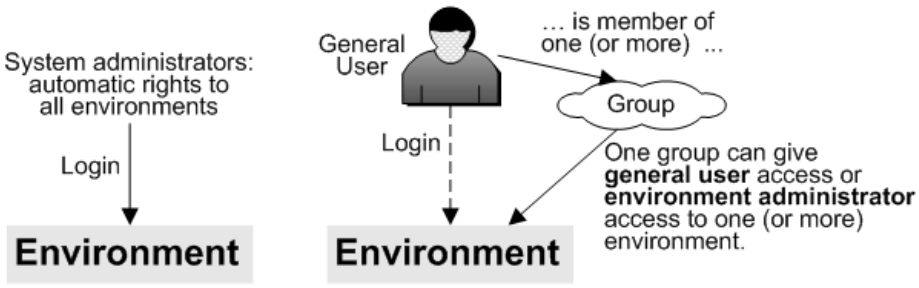
General users login using a group. The group gives users access to the environment for that session (and may make the general user an environment administrator). If a user needs to change group for that environment, then that user clicks **File -> Return to login...** to specify a new group for the same environment. All work for general users is done using one group and one environment per session.

Groups allow your company to achieve these benefits:

- **Apply access authorities once to a group** instead of configuring each member of the group individually for each environment. This saves time in applying access authorities and solving problems with access.
- When setup, a general user can have a **choice of group**, which allows the user to **change their access to an environment** depending on their work situation. For example, a user may normally work with only read access to the physical files in an environment, and occasionally the user changes group to obtain the authority to modify those physical files (for example to remove a duplicate transaction). Most users are likely to have no choice of group during login.

20 How do I use a group?

General users select an environment when they **login** to the SAFR Workbench, and the user **may have a choice of group** for that environment. General users become environment administrators only when a group provides the necessary access authority. Only a system administrator can change group authorities so that members of a group are environment administrators.



System administrators are not affected by groups and so the group field on the login screen is always grey.

For more about login, see topic "**Logging into the SAFR Workbench**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I know which group to use?

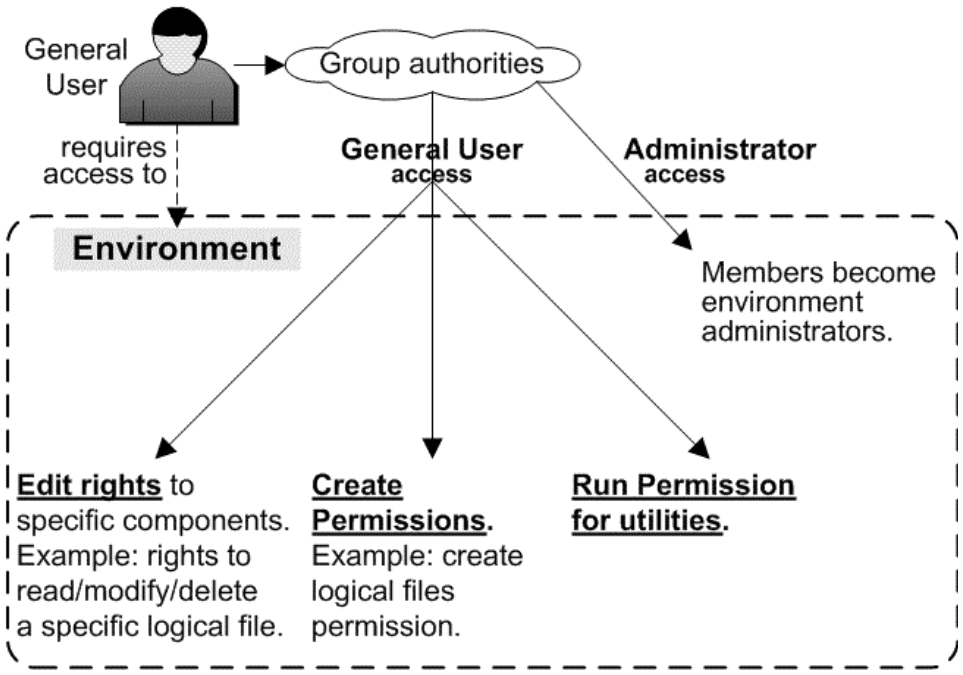
If you are a ...	**then ...**
General User	If you have a **choice of group during login**, see your system administrator. Otherwise use the only group given to you.
Environment Administrator	You must use the group given to you, because that is the group that provides the environment administrator access to you.
System Administrator	Groups do not affect system administrators during login.

50 Group authorities and environments

Every environment must be associated with at least one **group**. A general user or environment administrator gain **access to an environment** by becoming a **member of a group**.

Groups have authorities to access one or more environments. Group authorities for each environment can be any of the following:

- **Administrator** access authorities. This allows members of that group to be environment administrators.
- **General user** access authorities, as follows:
  - Edit rights to specific components. Example: rights to read/modify/delete a specific logical file.
  - Create permissions for types of components Example: create logical files permission.
  - Run permission for utilities.



One group may have only **read access** to all components in an environment and no create rights. Another group may have **more rights** to individual components. User rights in an environment are controlled by the group selected during login to that environment.

For more, see topics "**Groups - advanced overview**" and "**Environments - advanced overview**". These topics are elsewhere in this PDF - see the table of contents.

For a complete discussion of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Groups - advanced overview

This topic assumes you are familiar with the topic "**Groups overview**". That topic is elsewhere in this PDF - see the table of contents.

01 Summary of this topic

This topic covers the following:

10 Group permissions to create types of components

Group members receive create permissions for types of components.

The group create permissions possible are as follows:

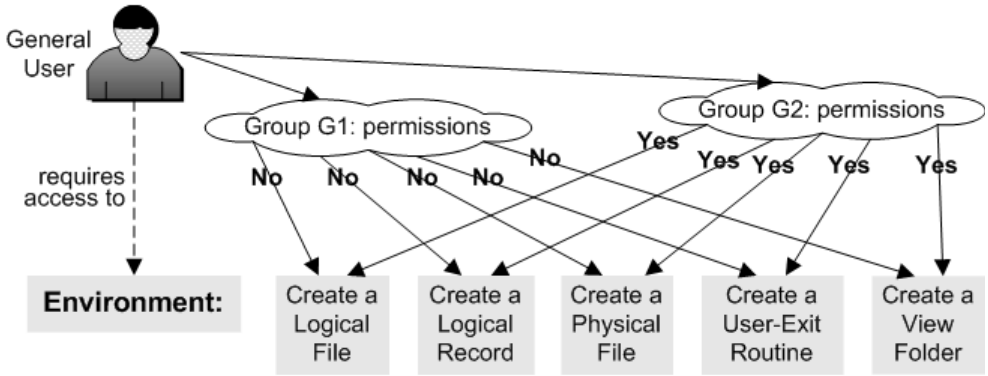
- Create Logical Files,
- Create Logical Records,
- Create Physical Files,
- Create User-Exit Routines,
- Create View Folders.

The above permissions can be applied to a general user in one or more environments. Administrators in an environment always have these permissions.

Notice there is no "create" right for control records and global fields - these are only created by administrators. General users can make use of existing control records and global fields, but cannot create, modify or delete them.

Different groups have different authority, and this can give a choice to a general user. For example:

- **Group G1** is for regular reporting, so this group cannot create any new components. This applies to general users who login to this environment using group G1.
- **Group G2** is for update of data, so this group can create any new components required. This applies to general users who login to this environment using group G2.



In the above example, the general user can **change rights to create items** in the environment by the **choice of group** during login to that environment.

20 Group permissions to run a utility

Group members can receive a permission to run utilities.

The utilities available in the SAFR Workbench are:

- **Batch Activate Lookup Paths.** This utility checks lookup paths are ready to use in an environment, and if possible sets the status to "active". For more, see topic "**Lookup Paths overview**". That topic is elsewhere in this PDF - see the table of contents.
- **Batch Activate Views.** This utility checks views are ready to run in the SAFR Performance Engine, and if possible sets the status to "active". For more, see topic "**Views overview**". That topic is elsewhere in this PDF - see the table of contents.
- **Migration Utility.** This utility copies selected metadata from a source environment to target environment in the same SAFR Database. For more, see topic "**Migrate metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

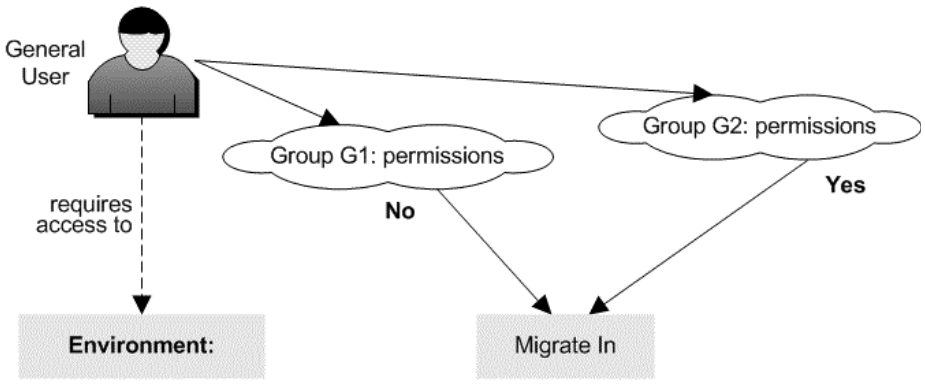
There is one group run permission:

- **Migrate In**. This provides access to all three utilities:
  - **Migration Utility** where the target environment is this environment ,
  - **Batch Activate Lookup Paths** and **Batch Activate Views** in this environment.

The above run permission can be applied to a general user in one more environments. Administrators in an environment always have this run permission.

Different groups have different authority, and this can give a choice to a general user. For example:

- **Group G1** is for users who run established views and cannot migrate any metadata. Users in this group cannot run any utilities.
- **Group G2** is for users who can migrate metadata from other environments. Users in this group can run all three utilities.



In the above example, the general user can **change rights to run utilities** in the environment by the **choice of group** during login to that environment.

30 Groups and edit rights to specific components

Group members receive edit rights to specific components of metadata.

The specific components must be of the following types:

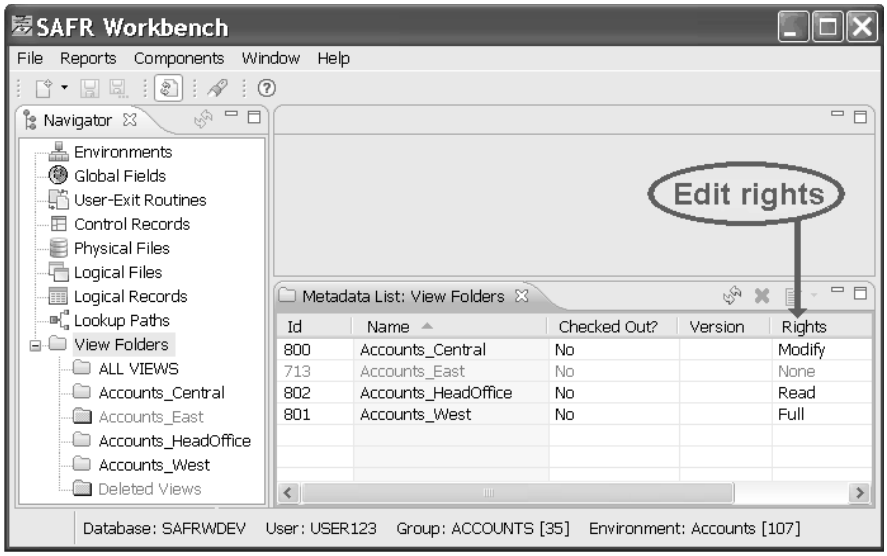
- Logical File,
- Logical Record,
- Physical File,
- User-Exit Routine,
- View Folder.

The edit rights possible are:

- **No rights at all**.
- **Read** right which allows both display and usage of an item. For example, a user needs the read right to a logical record in order to refer to that logical record in a view.
- **Modify** right which implies Read as well.
- **Delete** right which implies the Modify and Read rights as well. This right is also called "**Full**" rights.

The above edit rights for a specific component can be applied to a general userin one or more environments. Administrators always have full rights to all components.

Edit rights can be seen in column "**Rights**" in the **Metadata List**. For example, if you click on "**View Folders**" in the **Navigator**, the Metadata List may appear as follows:

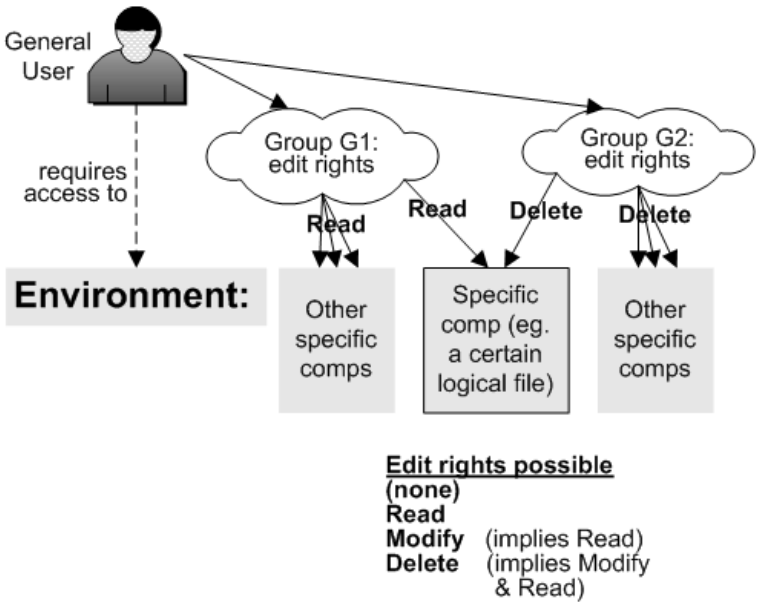


A group can have different edit rights to each individual component.

Different groups have different authority, and this can give a choice to a general user. For example:

- **Group G1** is for reporting, so this group has the read right to a specific logical file.

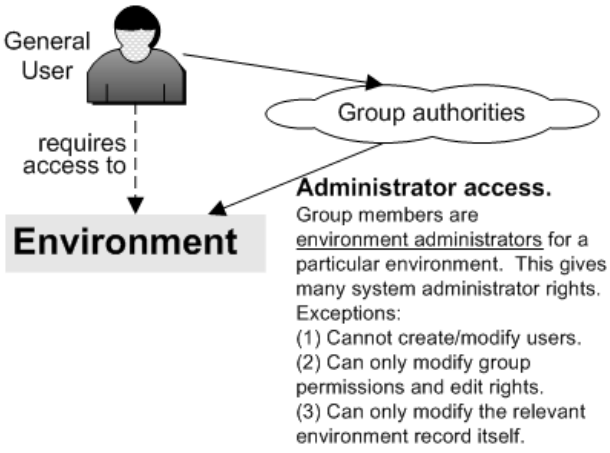
- **Group G2** is for update of data, so this group has the delete right (which is all rights) to a specific logical file.



In the above example, the general user can **change access to specific components** in the environment by the **choice of group** during login to that environment.

40 Groups can allow administration rights

A group can have **administration rights to an environment**. This means all users in that group become **environment administrators** in that environment.



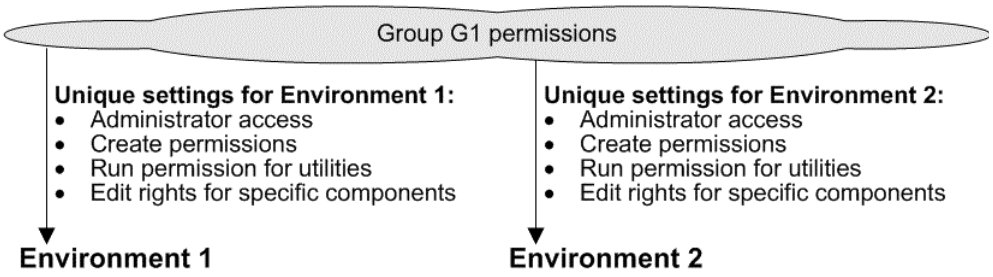
Environment and system administrators have all the create permissions, run permission and edit rights that are possible.

An environment administrator and has almost the same rights as a system administrator. The difference is that an **environment** administrator:

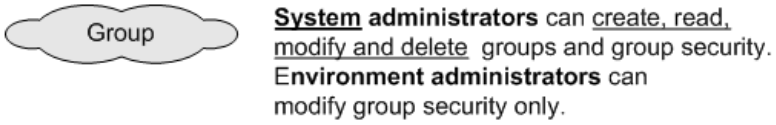
- Cannot create, read, modify or delete users.
- Can only modify group permissions and rights. Cannot give administrator access. Cannot change group membership. Cannot create or delete groups.
- Can only modify the environment record itself for the relevant environment. Cannot create or delete environments. Cannot modify environment records where there is no environment administrator access.

50 Groups have unique settings for each associated environment

All the above permissions apply separately to each environment associated with the group. For example, if a group G1 is associated with two environments all the above permissions can be set uniquely for each environment.



60 How do I create or modify a group?



System administrators can do the following tasks:

- "**Creating groups**",
- "**Modifying groups**",
- "**Modifying group membership**",
- "**Modifying group permissions by group**",
- "**Modifying group permissions by environment**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

Environment administrators can do the following task in the appropriate environment:

- "**Modifying group permissions by environment**".

To find this topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

70 How do I delete a group?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

80 Environment Security Report

All users can run this report. To see a report that lists the groups in one or more environments, use one of these topics:

- FAQ topic "**How do I generate an Environment Security Report?**"
- Task "**Generating reports**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

For a complete discussion of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

90 The best way to use groups

Groups aim to achieve the following goals:

- **Accuracy:** provide only the appropriate access for users to the relevant environments. Too much access risks unintended update of environments. Too little access stops general users from doing their work.
- **Flexibility:** allow some extra groups for unusual work situations that require unusual access for a short period of time.
- **Simplicity:** fewer groups means less groups to maintain.

These goals are often in competition: for example to achieve goal of accuracy perfectly may require a separate group for each user which contradicts the goal of simplicity. Your company must find the appropriate choice of groups that balances the above goals. The next heading describes a suggestion.

95 Suggestion on groups for a large company using SAFR

If your company has a large number of environments and a large number of users, the following suggestion describes the groups to create:

- Create a group for the system administrators - for example **YourCo_System_Administrators**
- There are many environments, such as Market_Research and Data_Warehouse. Call the environment name **"eee"**. Create two groups per environment:
  - **eee_Environment_Administrators** group - for example **Market_Research_Environment_Administrators**
  - **eee_Users** group - for example **Market_Research_Users**
- Group **eee_Environment_Administrators** has **administration rights** to that environment only. It is recommended that this group has only one or two members. **This delegates administration of this environment** to the members of this group.
- Group **eee_Users** has less rights, such as only read rights to all components of that environment. This group has all other general users for this environment.
- **Create extra groups as required** where some members of group eee_Users need more than the minimum rights. These extra groups are created by the system

administrators (and not by members of the eee_Environment_Administrators group, who do not have the right to create or modify groups).

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Import metadata overview

01 Summary of this topic

You need to read this topic because:

- An import can be blocked by data problems in certain situations.
- An import can generate error messages that require a choice of action.
- An import can delete or modify data that you still require such as related items of metadata. Sometimes this occurs without a warning message.

This topic explains how to fix and prevent import problems. It is recommended you do not attempt an import before you read this topic.

Below is a guide to the sections you need to read:

1. An introduction is covered in sections:
   - "02 Knowledge you need first" on page 40
   - "03 What is import of metadata?" on page 40
   - "04 Why is import useful?" on page 41
2. The basics of import are explained in sections:
   - "10 Are ID numbers important to your company?" on page 42
   - "11 Critical issues when ID numbers are important" on page 42
   - "12 Critical issues when ID numbers are NOT important" on page 43
3. The basics of ID numbers are explained in sections:
   - "20 Ranges of ID numbers and 'missing' ID numbers" on page 45
   - "21 Delete also creates 'missing' ID numbers" on page 46
   - "22 Import can increase a range by 1" on page 46
   - "23 How to find most 'missing' ID numbers" on page 47
4. Ideal and non-ideal situations offer a guide to how to plan and prepare for import. The situations for import are:
   - "30 Ideal 'All New' - ID numbers important" on page 48
   - "31 Ideal 'All Update' - ID numbers important" on page 49
   - "33 Ideal 'All New' - ID numbers NOT important" on page 50
   - "34 Ideal 'All Update' - ID numbers NOT important" on page 51
   - "36 Non-ideal situations" on page 53
   - "38 Message about replace of existing items" on page 53
   - "39 Message about name duplications" on page 54
5. Prepare for import to prevent possible problems. If none of the ideals are relevant, then prepare for a non-ideal situation. Choose from these sections:
   - "40 Prepare for ideal 'All New' - ID numbers important" on page 56
   - "41 Prepare for ideal 'All Update' - ID numbers important" on page 57
   - "44 Prepare for ideal 'All New' - ID numbers NOT important" on page 60

- “45 Prepare for ideal 'All Update' - ID numbers NOT important” on page 61
- “49 Prepare for a non-ideal situation” on page 64

6. The actual import happens in section:
   - “50 Perform the import” on page 67
7. Problem descriptions allow you to plan to avoid problems or take action when the problems occur. See these sections:
   - “60 Possible data problems during import” on page 67
   - “61 Possible problem A - import ID out of range” on page 69
   - “62 Fix problem A” on page 69
   - “63 Prevent problem A” on page 70
   - “64 Possible problem B - import ID and name in different items” on page 70
   - “65 Fix problem B” on page 71
   - “66 Prevent problem B” on page 72
   - “67 Possible problem C - import name exists and ID not found” on page 72
   - “68 Fix problem C” on page 73
   - “69 Prevent problem C” on page 74
   - “70 Possible problem D - delete of relationships” on page 74
   - “71 Fix problem D” on page 75
   - “72 Prevent problem D” on page 76
8. Background information is described in these sections:
   - “90 Structure of XML files” on page 76
   - “91 Does an ID number exist in an environment?” on page 76
   - “92 Is an ID number 'missing' in an environment?” on page 77
   - “93 How to find a range of ID numbers” on page 77
   - “94 Does a name already exist in an environment?” on page 78
   - “95 Comparison of export and import functions in WW and WE” on page 78
   - “97 Import history in log file” on page 79
   - “100 Need more on this page?” on page 79

02 Knowledge you need first

This topic assumes you are familiar with these topics:

- "**Export metadata overview**",
- "**WE Security overview**".

These topics are elsewhere in this PDF - see the table of contents.

03 What is import of metadata?

Import means data in an XML file creates or updates metadata in a target environment.

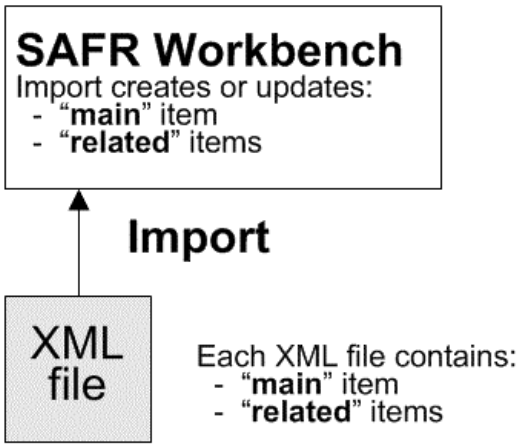
As for export, an XML file has a "main" item and potentially some "related" items". The "main" metadata types are:

- Physical file,
- Logical file,
- Logical record,
- Lookup path,

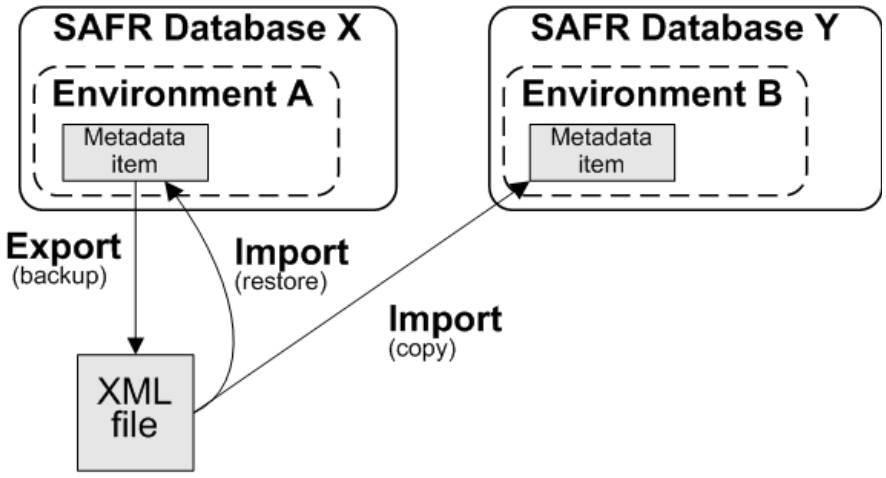
- View.

The table below shows the possible "related" items:

"Main" item	Possible "related" items
Physical file	User-exit routine
Logical file	Physical file, user-exit routine
Logical record	Logical file, physical file, user-exit routine
Lookup path	Logical record, logical file, physical file, user-exit routine
View	Control record, lookup path, logical record, logical file, physical file, user-exit routine



04 Why is import useful?



There are two reasons to import metadata from XML files:

- Restore from a backup of a metadata item.

- Copy an exported metadata item into a target environment. The target environment may be in the same database or a different database.

All backup, restore and copy options are outlined in help topic "**Metadata - advanced overview**". That topic is elsewhere in this PDF - see the table of contents.

10 Are ID numbers important to your company?

The ID number of a metadata item is useful to identify that metadata item in the workbench.

The ID number does not change the results produced by SAFR in the Performance Engine. For example, if you prepare a view with name **Monthly_Update** and ID number **1234**, then the results of that view are identical if the same view data has ID number **5678**. Once the view begins processing by the Performance Engine, the view ID number only affects some output DD file names in the JCL and otherwise the ID number of metadata is irrelevant to the results from the Performance Engine.

The ID numbers may be important to your company if the ID number is part of audit requirements, for example. It is also possible that your company considers the ID numbers are not important data. Sometimes the decision about ID numbers depends on the data. For example, ID numbers may be important for production data but not important for development data.

You must consult management in your company to decide if ID numbers are important, because this decision affects the appropriate import processing in your company. This is discussed in the next two sections below.

11 Critical issues when ID numbers are important

Ensure you have read section “10 Are ID numbers important to your company?.”

This section applies when your company considers ID numbers are important.

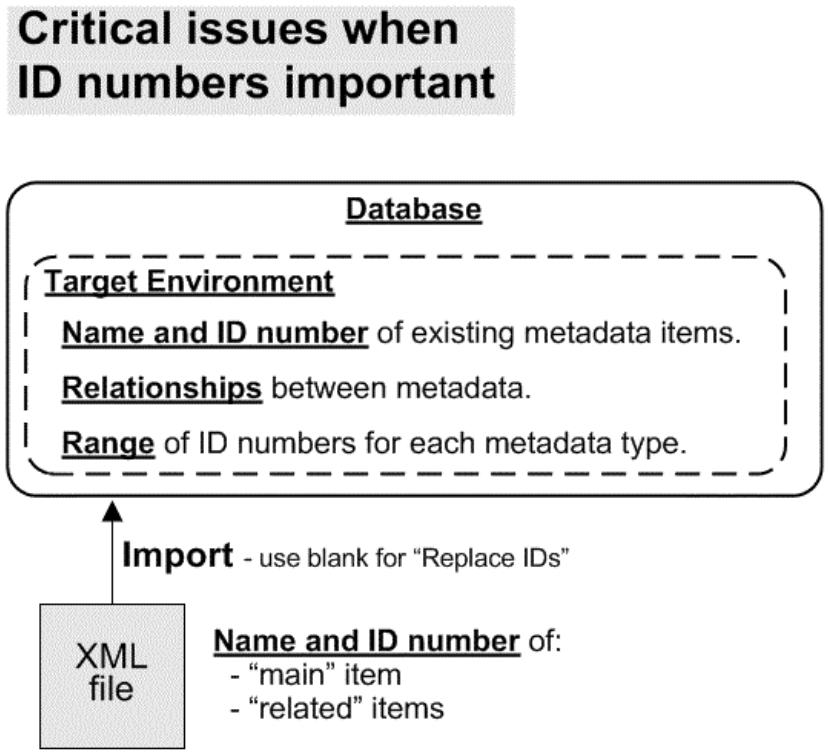
Critical issues in the SAFR Workbench:

- Target environment
- Database for the target environment
- Name and ID number of existing metadata items
- Related items of metadata.
- Ranges of ID numbers for each metadata type in the target environment. This is discussed more in section “20 Ranges of ID numbers and 'missing' ID numbers” on page 45.

Critical issues in the XML file:

- Name and ID number of the "**main** " item.
- Name and ID number of each "**related** " item.

When performing the import always use **blank** for field "**Replace IDs**". The field "Replace IDs" is part of the Import Utility screen. A value of blank means that the exact ID number in the XML file is used in the target environment.



ID numbers make an import process more complex to perform. Before importing, the relevant sections to read are:

- "20 Ranges of ID numbers and 'missing' ID numbers" on page 45
- "21 Delete also creates 'missing' ID numbers" on page 46
- "22 Import can increase a range by 1" on page 46
- "23 How to find most 'missing' ID numbers" on page 47
- "30 Ideal 'All New' - ID numbers important" on page 48
- "31 Ideal 'All Update' - ID numbers important" on page 49
- "38 Message about replace of existing items" on page 53
- "40 Prepare for ideal 'All New' - ID numbers important" on page 56
- "41 Prepare for ideal 'All Update' - ID numbers important" on page 57
- "49 Prepare for a non-ideal situation" on page 64

12 Critical issues when ID numbers are NOT important

Ensure you have read section "10 Are ID numbers important to your company?" on page 42.

This section applies when your company considers ID numbers are NOT important.

Critical issues in the SAFR Workbench:

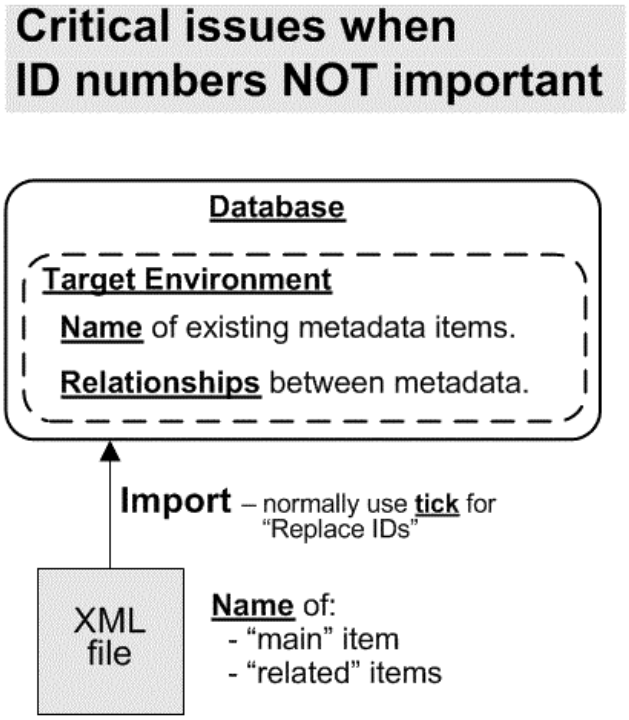
- Target environment

- Database for the target environment
- Name of existing metadata items
- Related items of metadata.

Critical issues in the XML file:

- Name of the "**main**" item.
- Name of each "**related** " item.

When performing the import use **tick** for field "**Replace IDs**" in most situations. The field "Replace IDs" is part of the Import Utility screen. A value of tick means imported items are given a new ID number in the target environment. In some situations such as restore from a backup it is appropriate to use **blank** for "Replace IDs" because the import ensures the existing data is updated.



Before importing, the relevant sections to read are:

- "33 Ideal 'All New' - ID numbers NOT important" on page 50
- "34 Ideal 'All Update' - ID numbers NOT important" on page 51
- "38 Message about replace of existing items" on page 53
- "39 Message about name duplications" on page 54
- "44 Prepare for ideal 'All New' - ID numbers NOT important" on page 60
- "45 Prepare for ideal 'All Update' - ID numbers NOT important" on page 61
- "49 Prepare for a non-ideal situation" on page 64

20 Ranges of ID numbers and 'missing' ID numbers

This section is relevant only when ID numbers are important to your company, as discussed in section “10 Are ID numbers important to your company?” on page 42.

A SAFR **database** can contain many **environments**. The current database and environment are always shown at the bottom of the screen in the SAFR Workbench.

Every SAFR database has a range of ID numbers for each metadata type. The range for each metadata type applies to all the environments inside that database.

The range is always between 1 and the largest ID number for that metadata type in that database.

A new item always increases the range for that metadata type by 1 . Notice how creating an item in one environment increases the range for all environments in that database.

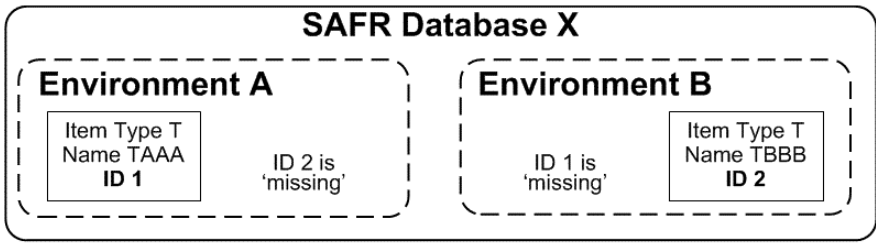
Let's look at an example. Consider a metadata type T in a **SAFR Database X** with **Environment A** and **Environment B**. Metadata type T does not exist - it only represents a metadata type.

In this example, a user in Environment A creates an item of type T. The new item has name TAAA and ID number 1. Later, a user in Environment B creates another item of type T. This second item has name TBBB and ID number 2.

Notice that both environments have the same range for metadata type "T" - from 1 to 2.

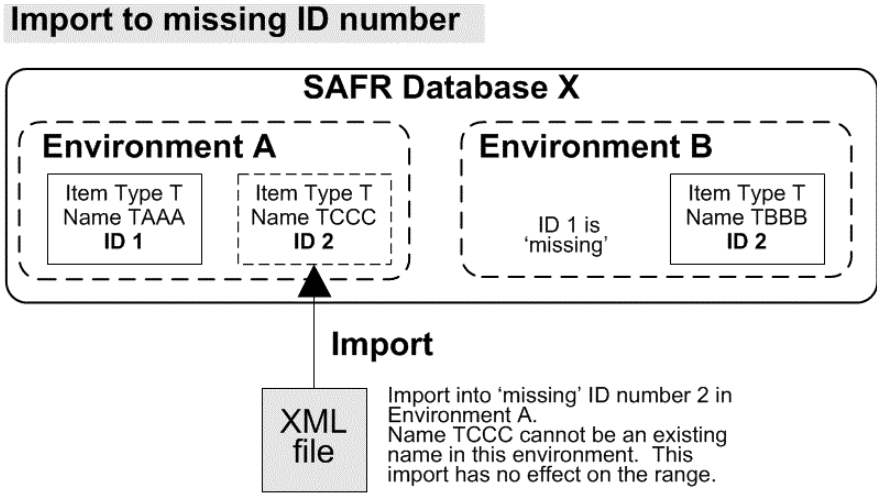
Notice that Environment A has a **'missing' ID number** of 2. In the same way, Environment B has a 'missing' ID number of 1. A diagram of this example is below.

Ranges and missing ID numbers



These 'missing' ID numbers are normal. You can import into 'missing' ID numbers in any environment, without affecting the range.

For example, in the above situation, a user in Environment A can import a "main" item of type T, using name TCCC and ID number 2. The result is below:



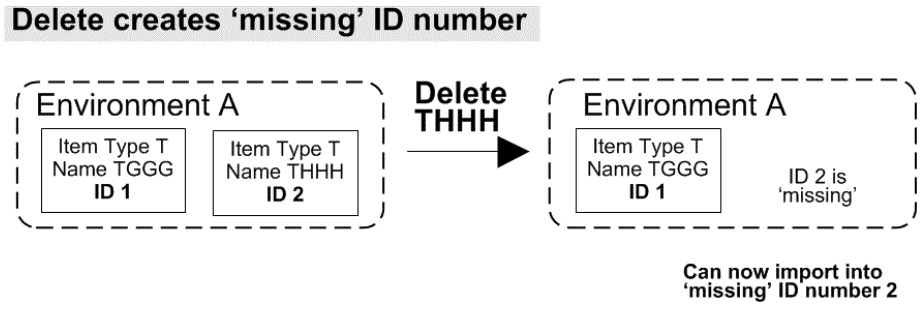
Almost all environments have 'missing' ID numbers.

21 Delete also creates 'missing' ID numbers

This section is relevant only when ID numbers are important to your company, as discussed in section "10 Are ID numbers important to your company?" on page 42.

In the previous section, the 'missing' ID numbers are the result of creating items in different environments in the same database.

A delete of metadata also creates 'missing' ID numbers. In the example below, if item THHH with ID number 2 is deleted, then Environment A now has 'missing' ID number 2.



22 Import can increase a range by 1

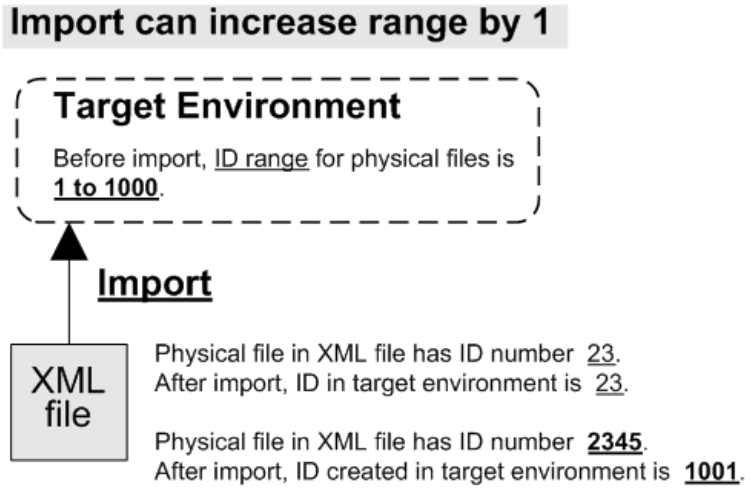
This section is relevant only when ID numbers are important to your company, as discussed in section "10 Are ID numbers important to your company?" on page 42.

If an import has an ID number larger than the range, then an **import increases the range by 1.**

For example, physical files may have a range of 1 to 1000 in a database. Consider an import into a target environment in that database.

If an XML file has a physical file of ID 23, then this ID is in the range. If the target environment has a 'missing' ID 23, then the import creates a physical file of ID 23. If the target environment already has a physical file of ID 23, then the import updates this record. In both cases, the ID after import is unchanged at 23.

If an XML file has a physical file of ID **2345**, then this ID is out of the range. An import creates an ID number of **1001**.



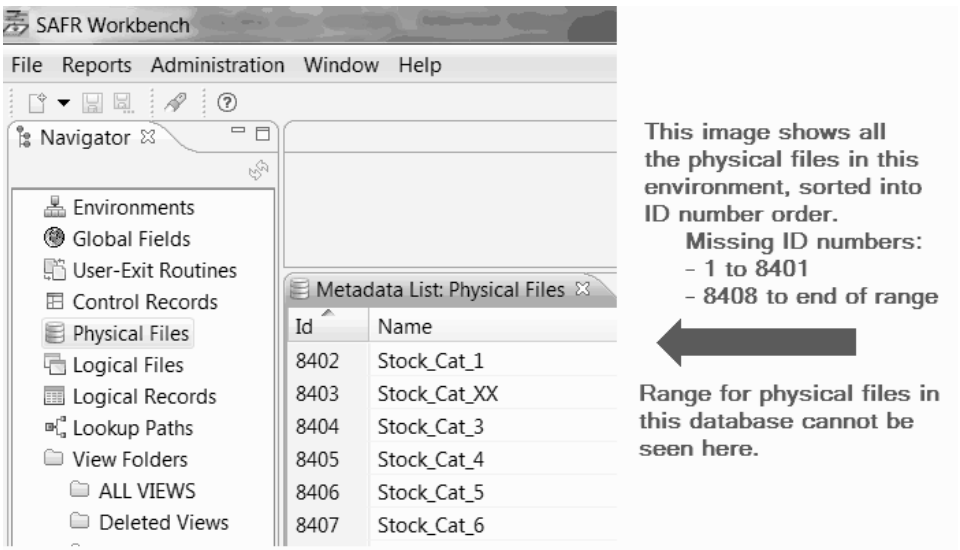
Notice that after an import the ID number of a metadata item may not be the same as in the XML file. This may be a problem for your company, for example for audit results. This issue is further discussed in section “61 Possible problem A - import ID out of range” on page 69.

23 How to find most 'missing' ID numbers

This section is relevant only when ID numbers are important to your company, as discussed in section “10 Are ID numbers important to your company?” on page 42.

When the Metadata List is sorted into ID number order, most 'missing' ID numbers are clearly visible. Do the following:

1. Log into the Workbench using the relevant environment.
2. In the Navigator, **click on the relevant metadata type**.
3. The Metadata List displays a list of existing metadata items of that type. **Click on the heading "Id"** to sort the list into descending order for ID number. If you click this heading multiple times, the list varies between descending and ascending order for ID number.
4. If every ID number exists, you will see a complete list of numbers: 1, 2, 3, 4 etc. Any gaps indicate 'missing' ID numbers. An example is below.



To find all the 'missing' ID numbers, see section “93 How to find a range of ID numbers” on page 77. Once you know the range, you can determine if there are any 'missing' ID number between the last ID number displayed above and the limit of the range. For example, if the range in the above image is 8420, then all the numbers from 8408 to 8420 are 'missing' in this environment.

30 Ideal 'All New' - ID numbers important

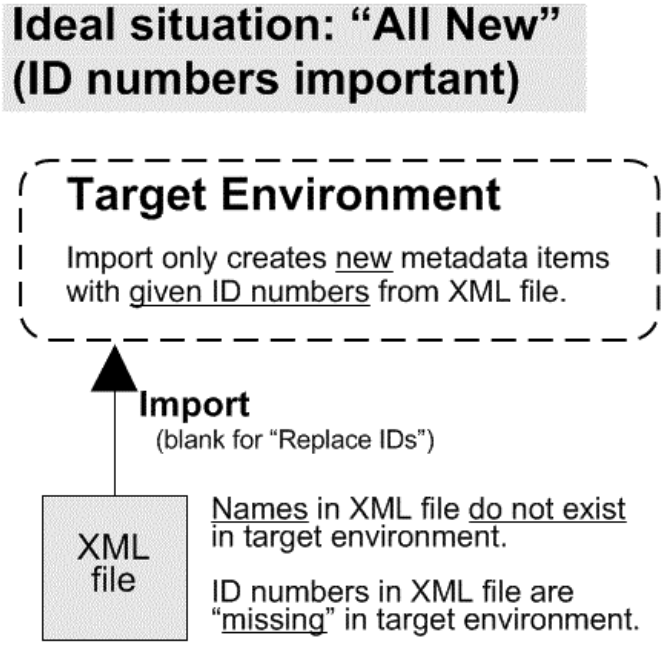
This section is relevant only when ID numbers are important to your company, as discussed in section “10 Are ID numbers important to your company?” on page 42.

In this ideal the following is true:

- Names of every item in the XML file do not exist in the target environment.
- ID numbers of every item in the XML file are 'missing' ID numbers in the target environment.

In this ideal, an **import with blank for "Replace IDs"** creates only new metadata items with the exact ID number given in the XML file. The blank in the "Replace IDs" field means the import attempts to use the values for ID number given in the XML file.

This is useful because the existing metadata in the target environment is untouched by the import.



The above ideal is not mandatory for import. If it is possible and appropriate, it is recommended to adjust your data to conform to this ideal.

To use this ideal, see section “40 Prepare for ideal 'All New' - ID numbers important” on page 56.

31 Ideal 'All Update' - ID numbers important

This section is relevant only when ID numbers are important to your company, as discussed in section “10 Are ID numbers important to your company?” on page 42.

In this ideal the following is true:

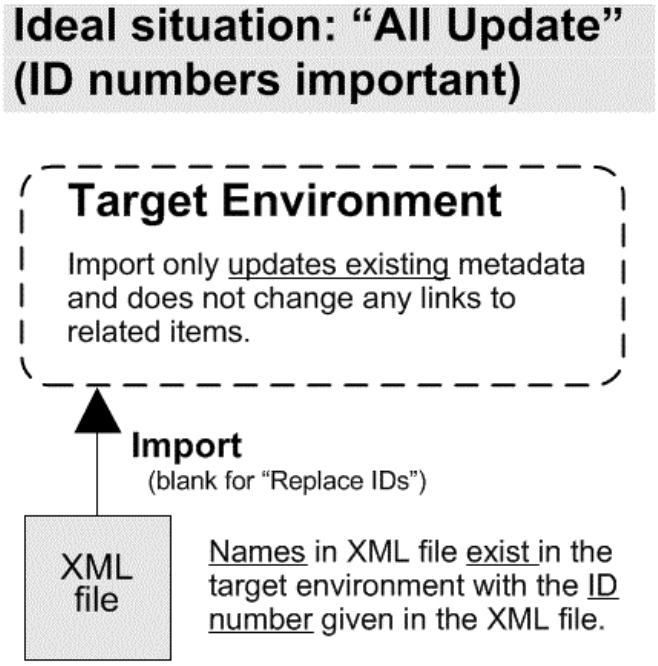
- Every combination of name and ID in every record in the XML file already exists in a metadata item in the target environment,
- The related items are the same when comparing the main items in the XML file with the same main items in the target environment.

In this ideal, an**import with blank for "Replace IDs"** only updates existing metadata items. The blank in the "Replace IDs" field means the import attempts to use the values for ID number given in the XML file.

This import may change some metadata items to status "inactive", which is normal when data is updated.

This type of import is useful because the changes to the target environment are limited. This type of import generates a standard message given in section “38 Message about replace of existing items” on page 53.

This ideal often applies when restoring from a backup.



This ideal needs to be used with care: the list of "related" items for the main items in the XML file must be accurate when compared to the same items in the target environment (unless the differences are acceptable). If any existing related metadata itemsin the target environment are not in the XML file, then these related items are deleted during the import. For more on this, see section “70 Possible problem D - delete of relationships” on page 74.

To use this ideal, read these sections:

- “38 Message about replace of existing items” on page 53,
- “41 Prepare for ideal 'All Update' - ID numbers important” on page 57.

33 Ideal 'All New' - ID numbers NOT important

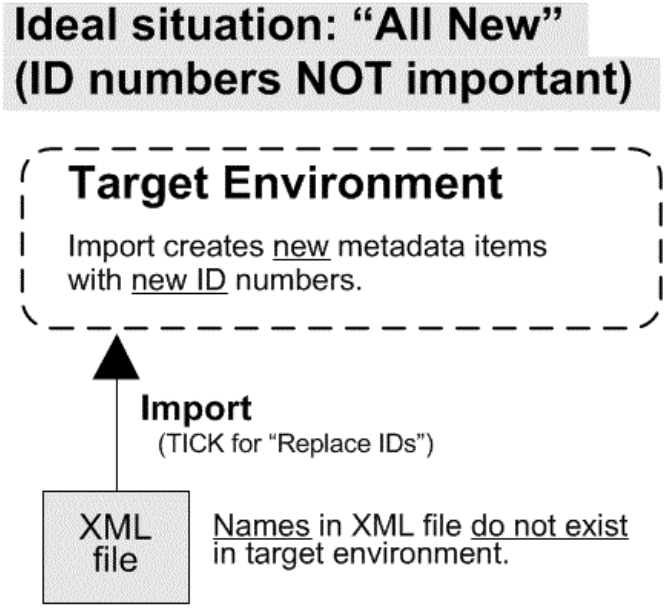
This section is relevant only when ID numbers are not important to your company, as discussed in section “10 Are ID numbers important to your company?” on page 42.

In this ideal the following is true:

- Names of every item in the XML file do not exist in the target environment.

In this ideal, an **import with a tick for "Replace IDs"** creates only new metadata items with new ID numbers. The tick in the "Replace IDs" field means the import creates new ID numbers in the target environment that are unlikely to be the same as the ID numbers given in the XML file.

This is ideal because the existing metadata in the target environment is untouched by the import. If it is appropriate to adjust your data to conform to this ideal, then this import is a safe process for your existing data.



The above ideal is not mandatory for import. If it is possible and appropriate, it is recommended to adjust your data to conform to this ideal.

To use this ideal, see section “44 Prepare for ideal 'All New' - ID numbers NOT important” on page 60.

34 Ideal 'All Update' - ID numbers NOT important

This section is relevant only when ID numbers are not important to your company, as discussed in section “10 Are ID numbers important to your company?” on page 42.

This ideal contains a conflict. The ideal of "All Update" implies that only existing records are updated, which implies the ID number is important for every record updated. This conflicts with the choice of your company that ID numbers are not important.

The solution to the conflict is to prepare and perform this ideal the same way as for when ID numbers are important, even though your company chooses otherwise. This ensures that the ideal is performed correctly and is still useful for your company. So this ideal is effectively exactly the same as section “31 Ideal 'All Update' - ID numbers important” on page 49.

In this ideal the following is true:

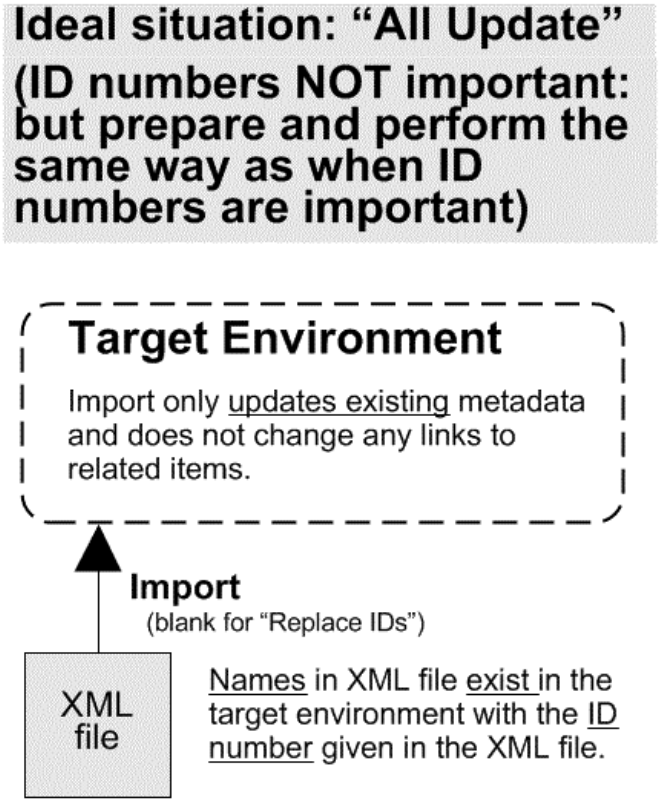
- Every combination of name and ID in every record in the XML file already exists in a metadata item in the target environment, Even though the ID numbers are not important, the ID numbers in the XML file must match the target environment exactly.
- The related items are the same when comparing the main items in the XML file with the same main items in the target environment.

In this ideal, the **import with blank for "Replace IDs"** only updates existing metadata items. The blank in the "Replace IDs" field means the import attempts to use the values for ID number given in the XML file. This is consistent with the "All Update" ideal as discussed above.

This import may change some metadata items to status "inactive", which is normal when data is updated.

This type of import is useful because the changes to the target environment are limited. This type of import generates a standard message given in section “38 Message about replace of existing items” on page 53.

This ideal often applies when restoring from a backup.



This ideal needs to be used with care: the list of "related" items must be accurate. If any existing related metadata items are not in the XML file, then these related

items are deleted during the import. For more on this, see section “70 Possible problem D - delete of relationships” on page 74.

To use this ideal, read these sections:

- “38 Message about replace of existing items”
- “45 Prepare for ideal 'All Update' - ID numbers NOT important” on page 61

36 Non-ideal situations

Almost all situations for import are **non-ideal** - which means the import includes both adding new items and updating existing items in the target environment.

A non-ideal situation is a normal way to import metadata. With correct preparation a non-ideal situation produces useful results.

A non-ideal situation may happen when you believe the data is in an ideal situation. Often you can discover the data is in a non-ideal situation when one of these things happens:

- A surprise **fail** result for an import.
- A surprise in **data displayed** in messages during the import. The relevant messages are discussed in these sections:
  - “38 Message about replace of existing items”
  - “39 Message about name duplications” on page 54

If you are surprised by data in the above messages, make a note of the data in the messages and press **Cancel** to stop the import, and review the import you are attempting.

In a non-ideal situation, an **import can use either tick or blank for "Replace IDs"**. The best value for "Replace IDs" depends on your company and the situation for your data.

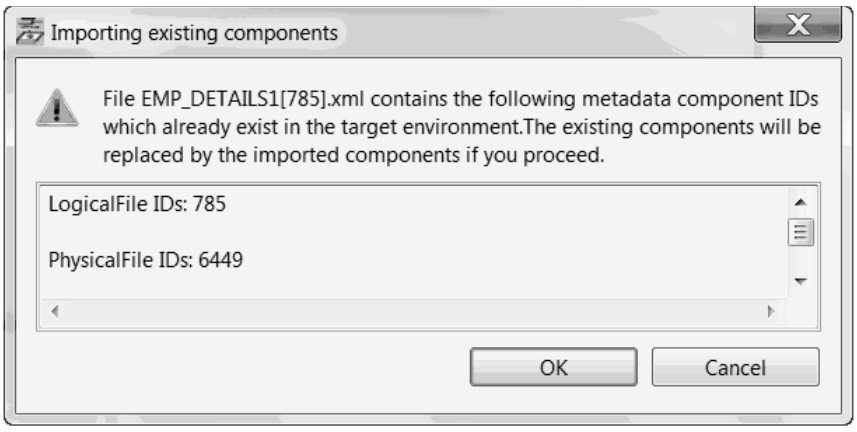
For non-ideal situations, read these sections:

- “38 Message about replace of existing items”
- “39 Message about name duplications” on page 54
- “49 Prepare for a non-ideal situation” on page 64

38 Message about replace of existing items

The strategy for import is "**replace not merge**". This means that update of an existing item in the target environment is achieved by a total replacement of the item by the data in the XML file.

If an import updates (replaces) existing items in the target environment, the import displays a standard message. The messages is similar to:



This is a normal and important message. You must be familiar with all the metadata items listed and the intended updates.

Press **OK** to allow the import to proceed and update the metadata listed in the message. Press **Cancel** to stop the import.

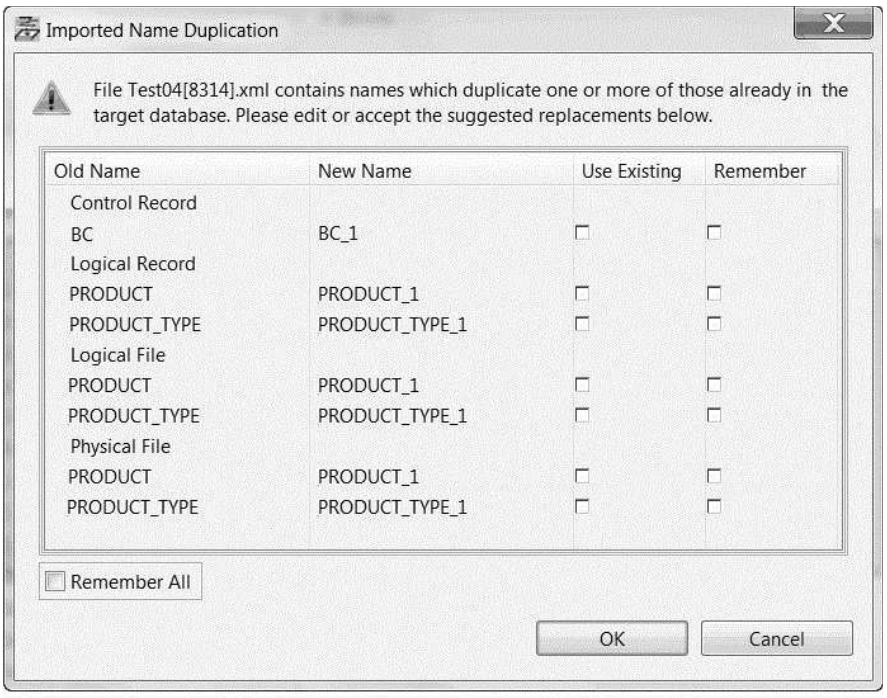
39 Message about name duplications

This message only occurs for an import with a **tick for Replace IDs**.

This message occurs when the XML file has at least one metadata name that already exists in the target environment. The ID numbers of the metadata items with the duplicated names may be different. The message offers a choice of actions for each case where there is a duplicated name.

This message is a normal message and the correct choices depends on deciding the best choices for your data.

An example of the message is below:



The message gives a list of metadata items in the XML file where the name already exists in the target environment. There items are listed in the column "**Old Name**". The item type is also listed as a heading above a group of names.

Each item listed has a **suggested new name and a box "Use Existing"**. The choices for the "Use Existing" box are:

- EITHER put a **tick for "Use Existing"**. This means no import occurs for that metadata item. Instead, any references to that item use the existing data. A tick for "Use Existing" results in a blank value in the "New Name" column because the new name is not used.
- OR ensure a **blank for "Use Existing"**. This means that item in the XML file is given the new name suggested, and is imported with a new ID number. After the import completes, the target environment contains both the existing item (with the old name) and the new item from the XML file (with the new name).

Your choice for each listed metadata item is to tick "Use Existing" or leave this field blank. Clicking the box repeatedly alternates between a tick and a blank value.

Each item listed also has a **box "Remember"**. The choices are:

- EITHER put a **tick** for "Remember" if you prefer to make a decision **once** for "Use Existing" for that metadata item. Your decision applies every time that metadata name appears in the import, which may occur many times.
- OR ensure a **blank** for "Remember" if you prefer to make a decision **every time** for "Use Existing" for that metadata item. With this choice, you may need to make a decision multiple times on "Use Existing" for the same metadata item in one import.

Tick "Remember All" to ensure all items listed have a tick for "Remember". Clicking the box repeatedly alternates between a tick and a blank value. Removing the tick for "Remember All" removes the tick from all "Remember" boxes listed.

Your choice is to tick some or all of the "Remember" boxes and possibly the "Remember All" box.

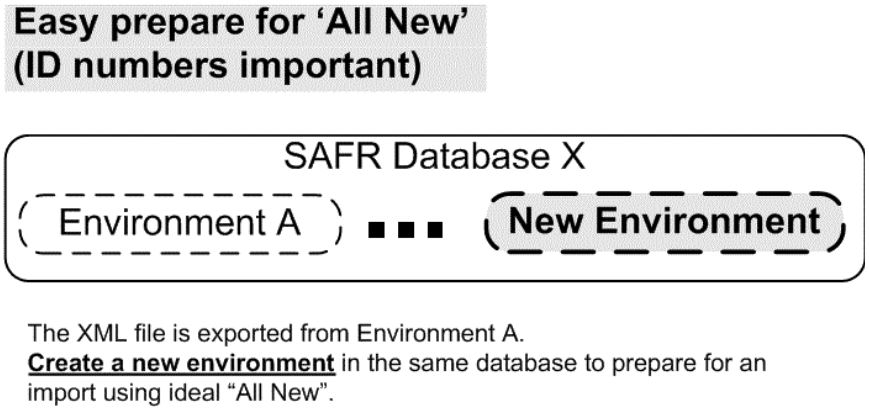
When your choices are complete, press **OK** to allow the import to proceed with your choices. Press **Cancel** to stop the import.

40 Prepare for ideal 'All New' - ID numbers important

This section is relevant only when ID numbers are important to your company, as discussed in section "10 Are ID numbers important to your company?" on page 42.

There is an easy way and a hard way to prepare for this ideal.

The easy way is to create a new environment in the same database as the exported items. This is only possible if the exported items were all from one database.



A totally new environment has no names, so all imported names are automatically new. A new environment in the same database has the same range, so the ID numbers in the XML file are always 'missing' in this new environment.

If you can create a new environment in this way, go to section "50 Perform the import" on page 67.

The hard way to prepare for ideal 'All New' is as follows:

1. Check that all names in the XML file do not exist in the target environment. To determine this, see section "94 Does a name already exist in an environment?" on page 78.

   If some names in the XML file exist in the target environment, either change the name in the XML file or in the target environment. If this cannot be fixed, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation" on page 64.

   When all names pass this check, then proceed to the next step.
2. Check that all ID numbers in the XML file are 'missing' and in range in the target environment. To determine this, see sections:

- "93 How to find a range of ID numbers" on page 77
- "92 Is an ID number 'missing' in an environment?" on page 77

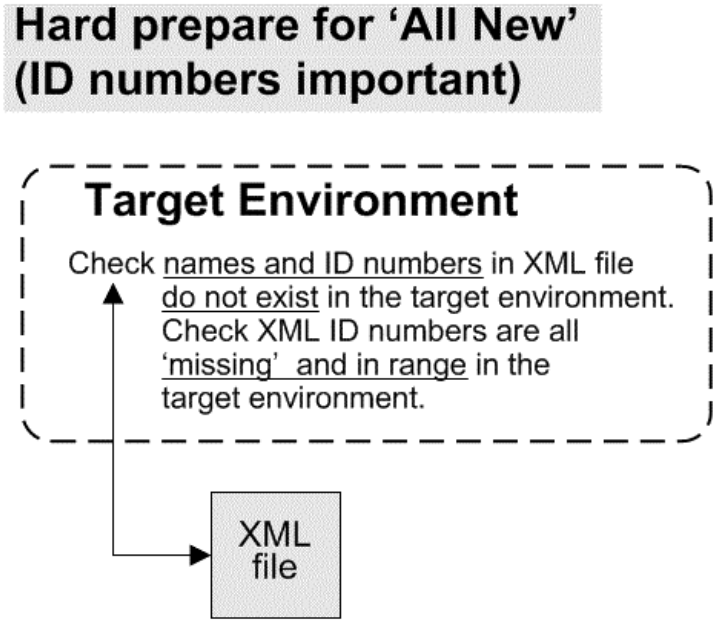
If some ID numbers in the XML file exist in the target environment, then possible fixes are:

- EITHER change the ID number in the XML file,
- OR delete the item containing that ID number in the target environment.

If the fixes above are not appropriate, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation" on page 64.

If all ID numbers pass this check, then proceed to the next step.

3. Preparation is complete - go to section "50 Perform the import" on page 67.



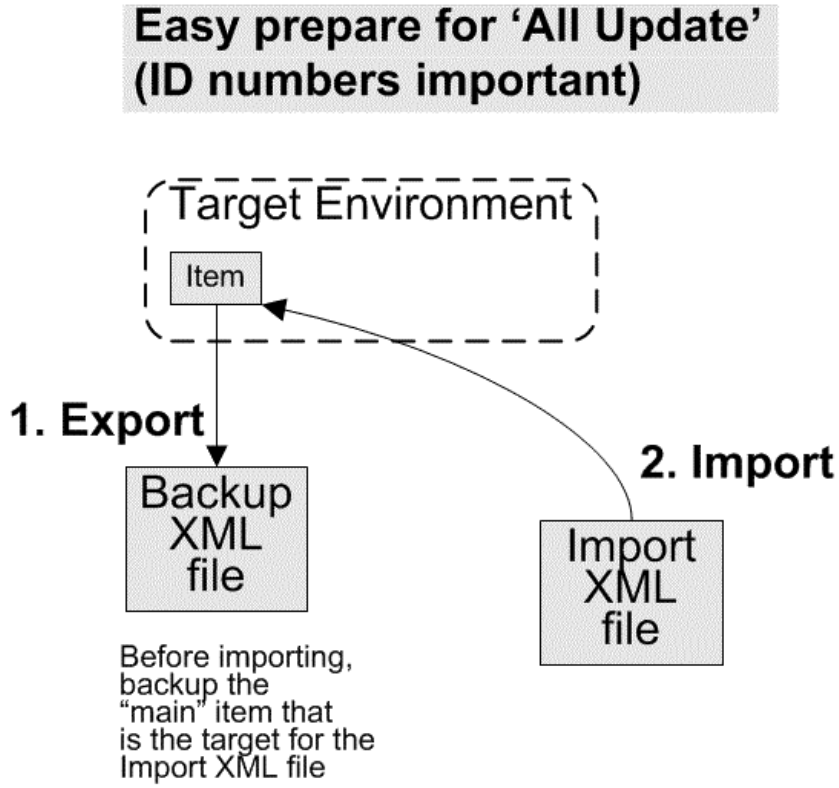
41 Prepare for ideal 'All Update' - ID numbers important

This section is relevant only when ID numbers are important to your company, as discussed in section "10 Are ID numbers important to your company?" on page 42.

There is a risk that an import only appears to conform to be the 'All Update" ideal and that the import may perform unwanted update or delete to existing data in the target environment. Some preparation is necessary to avoid this risk.

There is an easy way and a hard way to prepare for this ideal while eliminating risk.

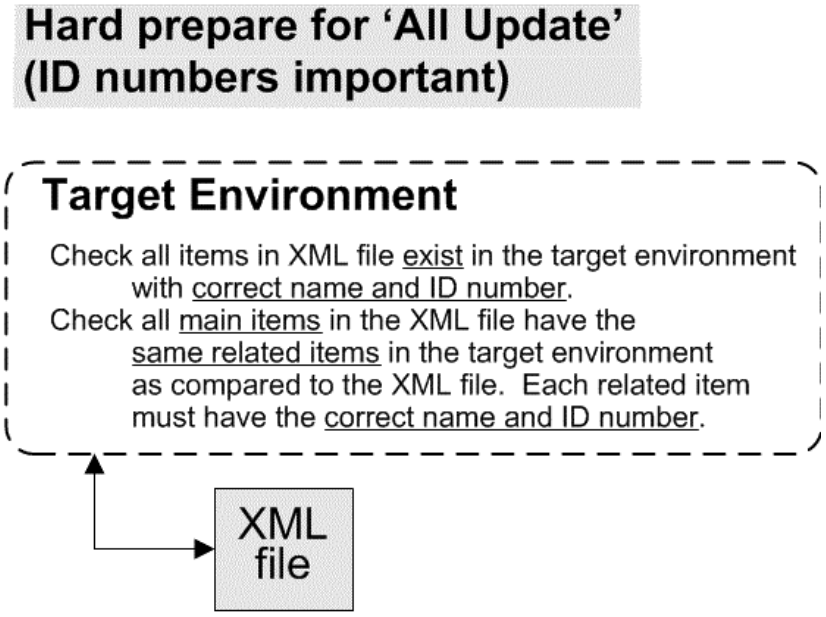
The easy way to prepare is to **take a backup of the "main" item** for the planned import. This means that if the import deletes or modifies metadata that you still require, you can use the backup to restore your data.



Instructions for the easy way are as follows:

1. The XML file you are planning to import can be called Import XML file (see diagram above). The actual name of that file describes the "main" item. Find that "main" item in the target environment. If the "main" item does not exist in the target environment, then the "easy" preparation is complete, and you can now proceed to "50 Perform the import" on page 67.
2. If the "main" item exists in the target environment, backup that item using topic "**Exporting metadata**" in the Administration Guide. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
3. The previous step creates an XML file that we can call the Backup XML file (see diagram above). Once that file exists, the "easy" preparation is complete. Proceed to "50 Perform the import" on page 67.

The "hard" way to prepare for ideal 'All Update' is to carefully check the XML file and the target environment before performing the import.



The "hard" way to prepare is as follows:

1. Be aware that the strategy for import is "**replace not merge**". This means that existing related items in the target environment are replaced by the data in the XML file. Read these sections:
   - "70 Possible problem D - delete of relationships" on page 74
   - "71 Fix problem D" on page 75
   - "72 Prevent problem D" on page 76
2. Check that all combinations of name and ID number in the XML file exist in the target environment. Do this as follows:
   a. Choose the next name for a metadata item in the XML file.
   b. Check that name exists in the target environment, by using section "94 Does a name already exist in an environment?" on page 78.
   c. If the name in the XML file does not exist in the target environment, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation" on page 64.
   d. If the name in the XML file exists in the target environment, check that the ID number in the target environment for this item is the same as given in the XML file.
   e. If the ID number is different to the XML file, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation" on page 64.
   f. If the ID number matches the XML file, then return to step 2a above to check the next name in the XML file. If all combinations of name and ID number in the XML file pass this check, then proceed to step 3 below.
3. Check that the main item in the XML file has the same related items when you compare the XML file to the target environment. Do this as follows:
   a. Find the main metadata item in the XML file.
   b. Find the same item in the target environment using section "94 Does a name already exist in an environment?" on page 78.

c. Make a list of the name and ID number for the related items in the target environment for that main item. The related items depend on the main item type - for a list of possible related item types, see section "03 What is import of metadata?" on page 40. Make a list of the actual related items for each possible type of related item.

d. Check that the list of related items from the target environment are also related items in the XML file. The name and ID number must both match for each related item.

e. If the related items are different, decide if the differences are acceptable or not. If all changes are acceptable, proceed to the next step below. If the differences are not acceptable, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation" on page 64.

4. Preparation is complete - go to section "50 Perform the import" on page 67.

44 Prepare for ideal 'All New' - ID numbers NOT important

This section is relevant only when ID numbers are not important to your company, as discussed in section "10 Are ID numbers important to your company?" on page 42.

There is an easy way and a hard way to prepare for this ideal.

The easy way is to create a new environment in any database.

A totally new environment has no names, so all imported names are automatically new.

If you can create a new environment, go to section "50 Perform the import" on page 67.

The hard way to prepare for ideal 'All New' is as follows:

1. Check that all names in the XML file do not exist in the target environment. To determine this, see section “94 Does a name already exist in an environment?” on page 78.

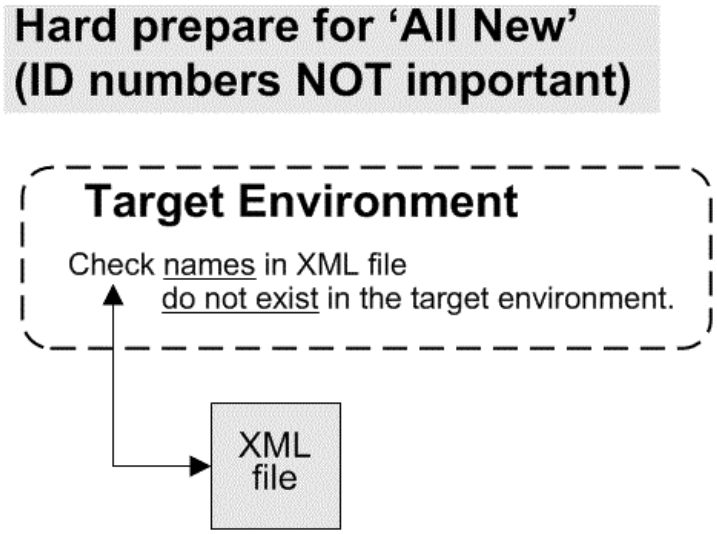
   If some names in the XML file exist in the target environment, either change the name in the XML file or in the target environment. If this cannot be fixed, then this ideal does not apply so proceed to section “49 Prepare for a non-ideal situation” on page 64.

   When all names pass this check, then proceed to the next step.
2. Check that all ID numbers in the XML file are 'missing' and in range in the target environment. To determine this, see sections:
   - “93 How to find a range of ID numbers” on page 77
   - “92 Is an ID number 'missing' in an environment?” on page 77

   If some ID numbers in the XML file exist in the target environment, then possible fixes are:
   - EITHER change the ID number in the XML file,
   - OR delete the item containing that ID number in the target environment.

   If the fixes above are not appropriate, then this ideal does not apply so proceed to section “49 Prepare for a non-ideal situation” on page 64.

   If all ID numbers pass this check, then proceed to the next step.
3. Preparation is complete - go to section “50 Perform the import” on page 67.



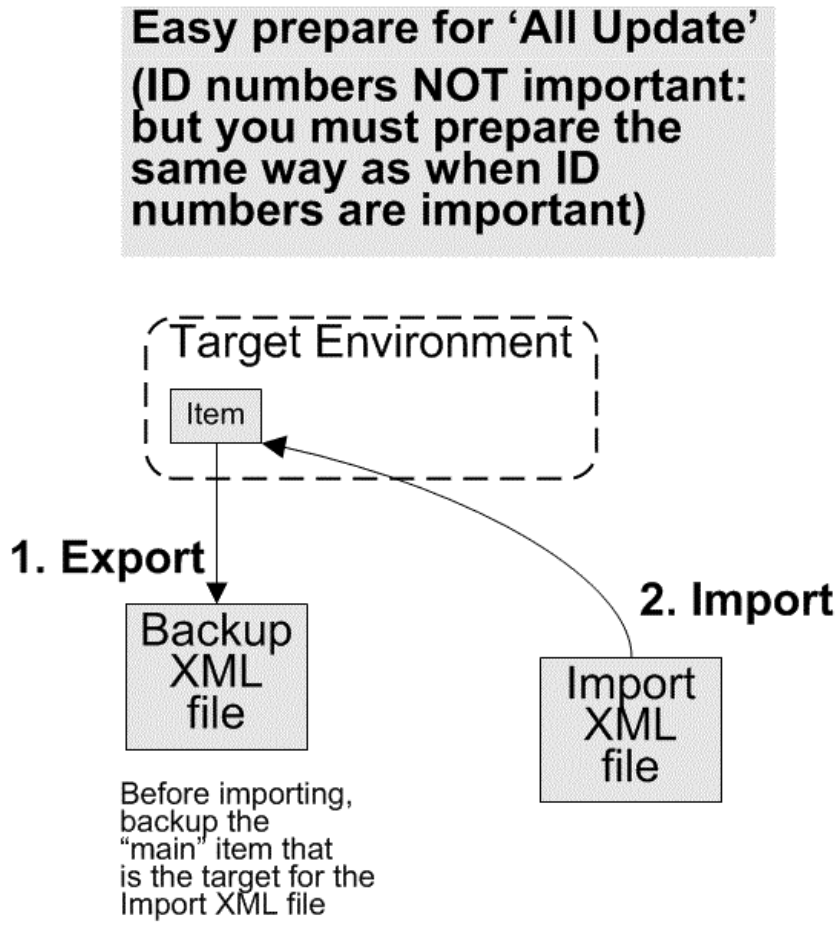
45 Prepare for ideal 'All Update' - ID numbers NOT important

This section is relevant only when ID numbers are not important to your company, as discussed in section “10 Are ID numbers important to your company?” on page 42.

There is a risk that an import only appears to conform to be the 'All Update" ideal and that the import may perform unwanted update or delete to existing data in the target environment. Some preparation is necessary to avoid this risk.

There is an easy way and a hard way to prepare for this ideal while eliminating risk.

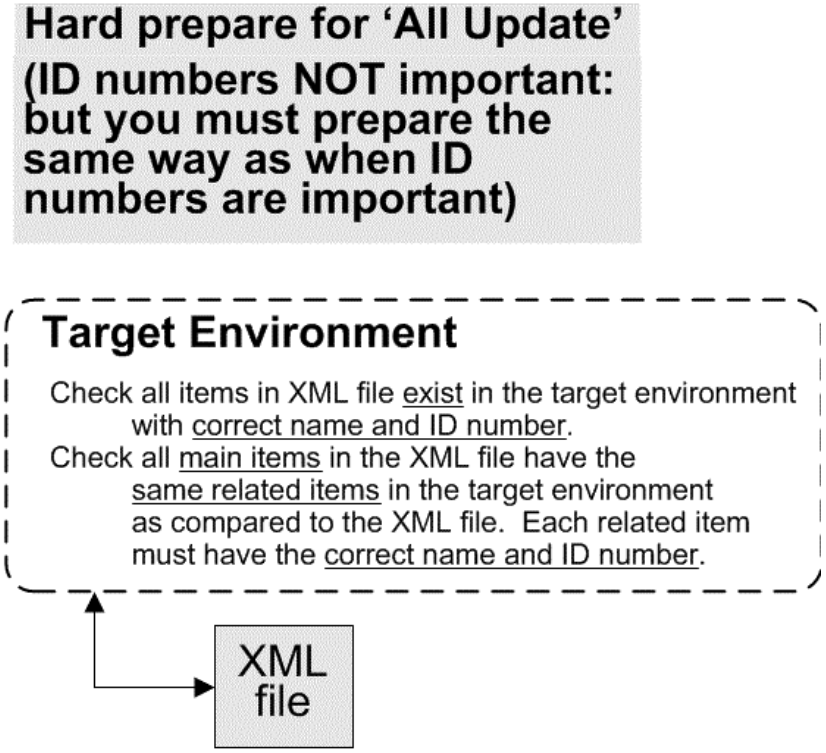
The easy way to prepare is to **take a backup of the "main" item** for the planned import. This means that if the import deletes or modifies metadata that you still require, you can use the backup to restore your data.



Instructions for the easy way are as follows:

1. The XML file you are planning to import can be called Import XML file (see diagram above). The actual name of that file describes the "main" item. Find that "main" item in the target environment. If the "main" item does not exist in the target environment, then the "easy" preparation is complete, and you can now proceed to "50 Perform the import" on page 67.
2. If the "main" item exists in the target environment, backup that item using topic "**Exporting metadata**" in the Administration Guide. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
3. The previous step creates an XML file that we can call the Backup XML file (see diagram above). Once that file exists, the "easy" preparation is complete. Proceed to "50 Perform the import" on page 67.

The "hard" way to prepare for ideal 'All Update' is to carefully check the XML file and the target environment before performing the import.



The "hard" way to prepare is as follows:

1. Be aware that the strategy for import is "**replace not merge**". This means that existing related items in the target environment are replaced by the data in the XML file. Read these sections:
   - "70 Possible problem D - delete of relationships" on page 74
   - "71 Fix problem D" on page 75
   - "72 Prevent problem D" on page 76
2. Check that all combinations of name and ID number in the XML file exist in the target environment. Do this as follows:
   a. Choose the next name for a metadata item in the XML file.
   b. Check that name exists in the target environment, by using section "94 Does a name already exist in an environment?" on page 78.
   c. If the name in the XML file does not exist in the target environment, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation" on page 64.
   d. If the name in the XML file exists in the target environment, check that the ID number in the target environment for this item is the same as given in the XML file.
   e. If the ID number is different to the XML file, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation" on page 64.

f. If the ID number matches the XML file, then return to step 2a above to check the next name in the XML file. If all combinations of name and ID number in the XML file pass this check, then proceed to step 3 below.

3. Check that the main item in the XML file has the same related items when you compare the XML file to the target environment. Do this as follows:
   a. Find the main metadata item in the XML file.
   b. Find the same item in the target environment using section "94 Does a name already exist in an environment?" on page 78.
   c. Make a list of the name and ID number for the related items in the target environment for that main item. The related items depend on the main item type - for a list of possible related item types, see section "03 What is import of metadata?" on page 40. Make a list of the actual related items for each possible type of related item.
   d. Check that the list of related items from the target environment are also related items in the XML file. The name and ID number must both match for each related item.
   e. If the related items are different, decide if the differences are acceptable or not. If all changes are acceptable, proceed to the next step below. If the differences are not acceptable, then this ideal does not apply so proceed to section "49 Prepare for a non-ideal situation."
4. Preparation is complete - go to section "50 Perform the import" on page 67.

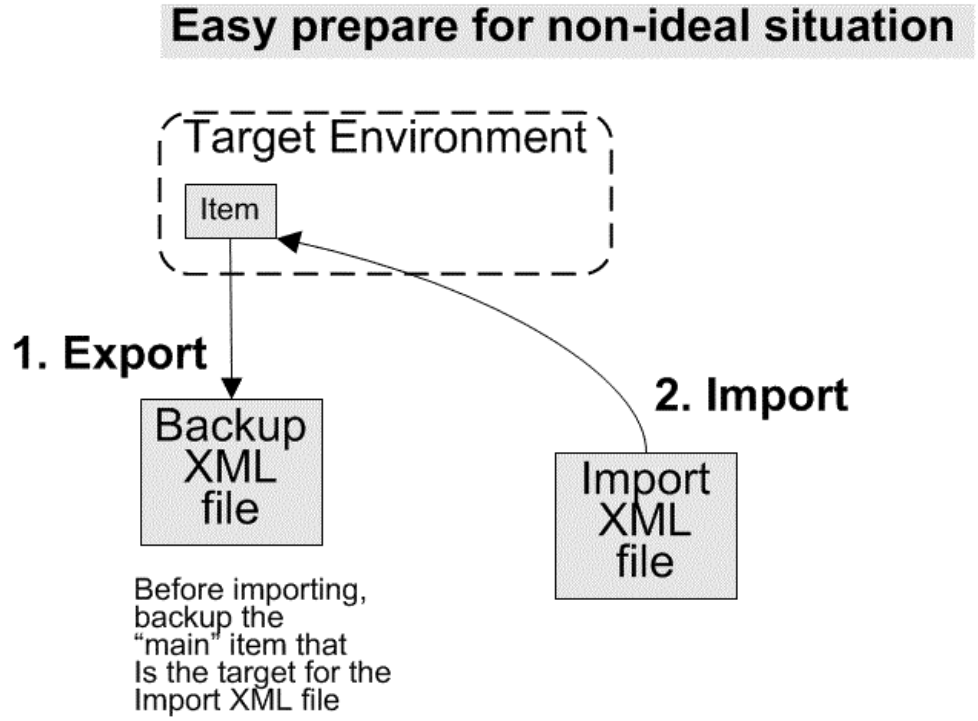
49 Prepare for a non-ideal situation

Non-ideal means your import involves a combination of creating new items and updating existing items. With care, this import is useful.

Preparing for a non-ideal situation is the same for ID numbers important or not important.

There is an easy way and a hard way to prepare for a non-ideal situation.

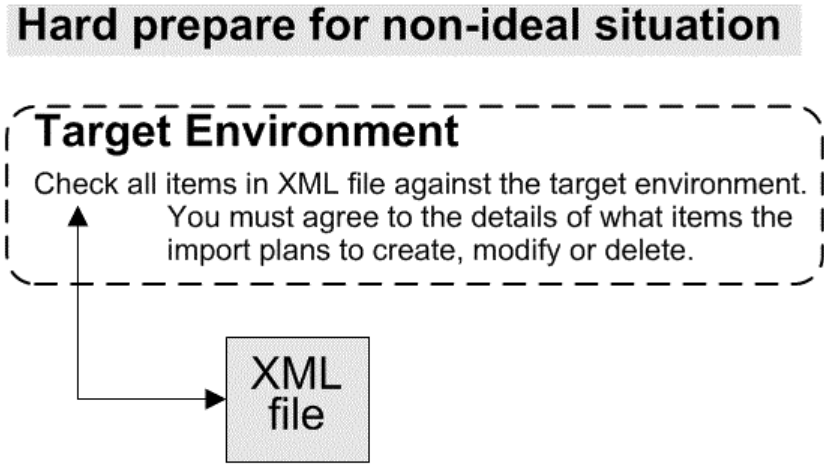
The easy to prepare is to **take a backup of the "main" item** for the planned import. This means that if the import deletes or modifies metadata that you still require, you can use the backup to restore your data.



Instructions for the easy way are as follows:

1. The XML file you are planning to import can be called Import XML file (see diagram above). The actual name of that file describes the "main" item. Find that "main" item in the target environment. If the "main" item does not exist in the target environment, then you must choose the "Hard" way to prepare given below, because there is a risk that the import may damage some related items of metadata that already exist in the target environment.
2. If the "main" item exists in the target environment, backup that item using topic "**Exporting metadata**" in the Administration Guide. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
3. The previous step creates an XML file that we can call the Backup XML file (see diagram above). Once that file exists, the "easy" preparation is complete. Proceed to "50 Perform the import" on page 67.

The hard way to prepare for a non-ideal situation is to carefully check the XML file and the target environment before performing the import.



Do the following:

1. Read these sections:
   - “60 Possible data problems during import” on page 67
   - “61 Possible problem A - import ID out of range” on page 69
   - “62 Fix problem A” on page 69
   - “63 Prevent problem A” on page 70
   - “64 Possible problem B - import ID and name in different items” on page 70
   - “65 Fix problem B” on page 71
   - “66 Prevent problem B” on page 72
   - “67 Possible problem C - import name exists and ID not found” on page 72
   - “68 Fix problem C” on page 73
   - “69 Prevent problem C” on page 74
   - “70 Possible problem D - delete of relationships” on page 74
   - “71 Fix problem D” on page 75
   - “72 Prevent problem D” on page 76
2. Be aware that the strategy for import is "**replace not merge**". This means that existing items in the target environment are replaced by the data in the XML file. This may result in existing data being deleted, as discussed in section “70 Possible problem D - delete of relationships” on page 74.
3. Ensure you know the answers to these questions:
   - What items does import of the XML file **create** in the target environment?
   - What items does import of the XML file **update** in the target environment?

   To answer these questions, use the following sections:
   - “91 Does an ID number exist in an environment?” on page 76
   - “92 Is an ID number 'missing' in an environment?” on page 77
   - “93 How to find a range of ID numbers” on page 77
   - “94 Does a name already exist in an environment?” on page 78

   Adjust your target environment and XML file as required to prevent problems found in Step 1 above.
4. Preparation is complete - go to section “50 Perform the import” on page 67.

50 Perform the import

Do the following:

1. Ensure you have prepared using one of the following sections:
   - “40 Prepare for ideal 'All New' - ID numbers important” on page 56
   - “41 Prepare for ideal 'All Update' - ID numbers important” on page 57
   - “44 Prepare for ideal 'All New' - ID numbers NOT important” on page 60
   - “45 Prepare for ideal 'All Update' - ID numbers NOT important” on page 61
   - “49 Prepare for a non-ideal situation” on page 64
2. Import may result in change to existing metadata in the target environment. It is recommended that you backup all relevant metadata in the target environment before you import, so that you can restore the existing data if required. The relevant metadata is listed in each XML file for import - the "main" item and the "related" items. For complete safety, backup all metadata in the target environment into XML files. You may have already done this as part of Step 1 above.

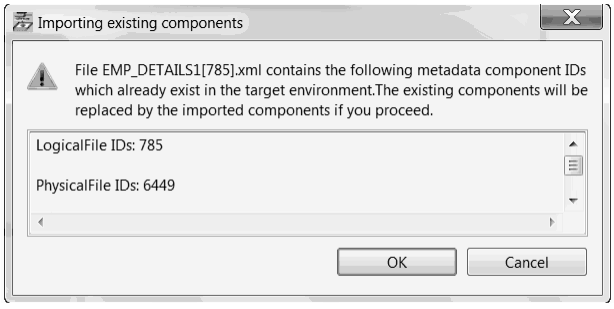
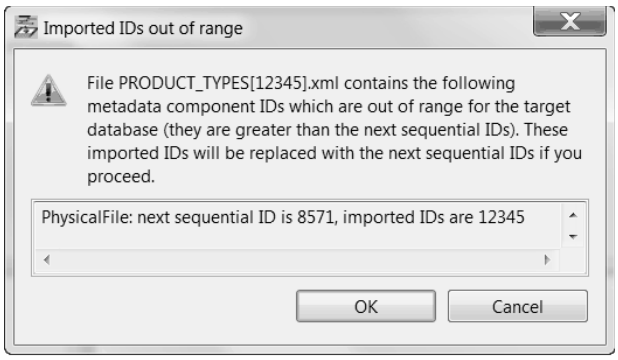
   A backup is done using an export - see help topic "**Export metadata overview**". That topic is elsewhere in this PDF - see the table of contents.
3. Perform the import using help topic "**Importing metadata**" in the Administration Guide. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
4. If the import has problems, see “60 Possible data problems during import.”

   If that section is not sufficient, see help topic "**Import Utility errors**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
5. The WE log file contains data about imports you perform in this session. See section “97 Import history in log file” on page 79.

60 Possible data problems during import

The section below contains some common errors and is useful for planning your import and a learning how to handle problem. For a full list of possible error messages, see help topic "**Import Utility errors**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

The table below describes possible problems and solutions:

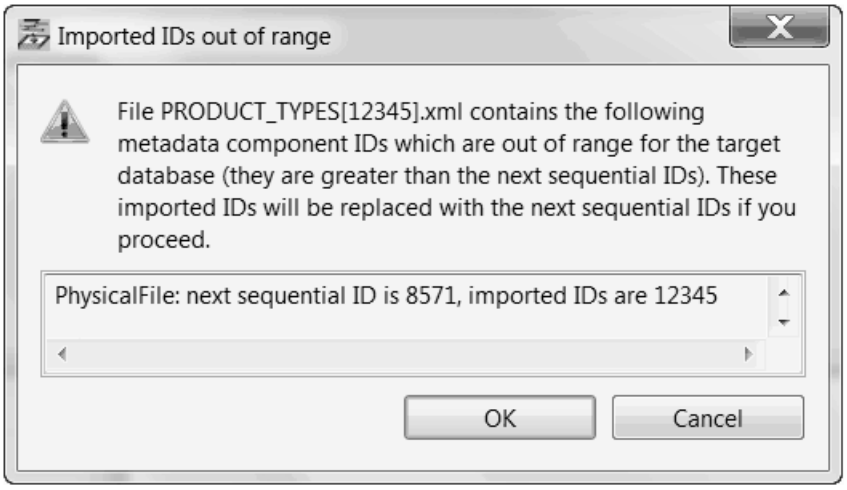
Possible problem	Solution
	This is a normal message that displays before the import updates existing metadata items. **You have a choice to update the metadata items listed.** Be aware that updating the metadata items listed may affect other data in the environment. For example, update of a logical record may affect many views and lookup paths and logical files in the target environment. Press **Cancel** to stop the import. Press **OK** to allow the import to proceed and update the metadata listed in the message.
	This is an example of "problem A". See sections • "61 Possible problem A - import ID out of range" on page 69 • "62 Fix problem A" on page 69
"**Fail**" appears in the "**Import Utility**" screen on the row for this import. When you highlight the row with "Fail" a message appears in the "Errors" section of the screen. An example error message for importing a physical file is: **The Physical File name 'AAAA' already exists. Please specify a different name.**	This is either "problem B" or 'problem C". See sections: • "64 Possible problem B - import ID and name in different items" on page 70 • "65 Fix problem B" on page 71 • "67 Possible problem C - import name exists and ID not found" on page 72 • "68 Fix problem C" on page 73
After an import, a metadata item is missing or has missing related items. For example, a logical file may have less related physical files.	This may be "problem D". See sections • "70 Possible problem D - delete of relationships" on page 74 • "71 Fix problem D" on page 75

61 Possible problem A - import ID out of range

An import can assign an ID number to a metadata item that is different to the ID number in the XML file. This is discussed in section "22 Import can increase a range by 1" on page 46.

This problem occurs when the ID number is the XML is too big for the relevant range in the target environment. The import must assign a lower ID number to fit the range in the target environment, and the range in the target environment is increased by 1.

Possible problem A is indicated by a warning message similar to the following:



Click **OK** to allow the import to assign lower ID numbers than shown in the XML file. Take this action if your company is comfortable with the changed ID numbers.

Click **Cancel** to stop the import.

This problem can never occur in a "restore" type of import. This is because a range only gets larger, so during a "restore" import the range can always include the ID numbers in the XML file.

For the same reason, this problem never occurs in a "copy" type of import where the target environment is in the same SAFR database as the export environment.

This problem may only occur for a "copy" import where the target environment is in a different SAFR database to the export environment. For that situation, the problem occurs when at least one of the ranges in the target environment is smaller then the export environment and cannot fit at least one ID number in the XML file. If that occurs, the import assigns a lower ID number in the target environment than shown in the XML file.

62 Fix problem A

This problem is described in section "61 Possible problem A - import ID out of range".

If you incorrectly click **OK** on the screen "**Imported IDs out of range**" then the import creates metadata items with an incorrect ID number. Fix this by deleting the metadata items created. The affected items were listed in the warning message similar to that shown in “61 Possible problem A - import ID out of range” on page 69. Another way to find the affected items is to check the XML file.

63 Prevent problem A

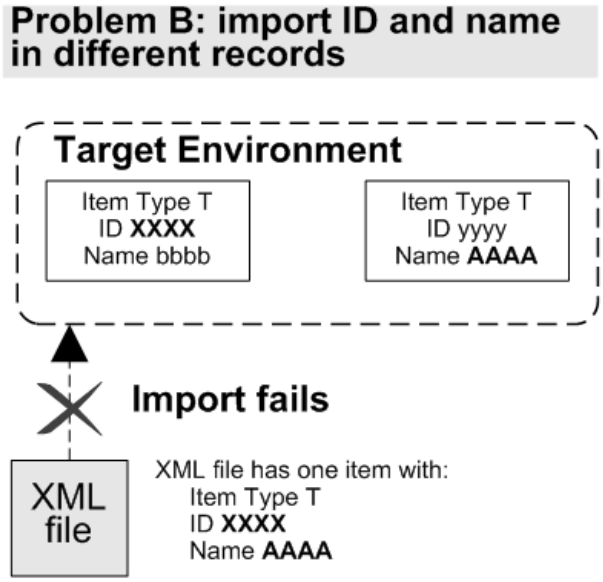
This problem is described in section “61 Possible problem A - import ID out of range” on page 69.

This problem can be prevented as follows:

Options to prevent this problem	Details
Target environment is a new environment in the same SAFR Database where the XML file was exported.	1. Perform the 'easy' way in section “40 Prepare for ideal 'All New' - ID numbers important” on page 56. 2. **Import** to the new environment.
Create dummy items in the target environment to ensure the range accommodates the ID numbers in the XML file.	1. **Analyze the XML file** for the metadata type, ID number and name in each record for import. 2. **Analyze the target environment** to find the metadata types that require more ID numbers to accommodate the XML file data. 3. **Create dummy metadata items** of the correct type to increase the range in the target environment, so that the ID numbers in the XML file records are "in range". Ensure that the ID number for the import has the correct name (not a dummy name). 4. **Check if the names in the XML file exist in the target environment.** It may be acceptable that the name exists with the correct ID number in the target environment so the that the import updates that metadata item. Check that all related items are included - see section “70 Possible problem D - delete of relationships” on page 74. Make any changes to the target environment or the XML file as appropriate. 5. **Import** to the target environment.

64 Possible problem B - import ID and name in different items

In this case, a record in the XML file refers to two items in the target environment, as shown below.



On the "**Import Utility**" screen, this import is marked with result "**Fail**" and when you highlight the line with "Fail" a message appears in the "Errors" section of the screen.

An example error message for importing a physical file is: **The Physical File name 'AAAA' already exists. Please specify a different name.**

In this example, the message is identical to the example in section “67 Possible problem C - import name exists and ID not found” on page 72.

65 Fix problem B

This problem is described in section “64 Possible problem B - import ID and name in different items” on page 70.

Fix as follows:

1. **In the XML file, find the record of the correct type** that has the name "AAAA'.
2. **Check that the name is correct in the XML file.** The correct name is based on your choice of name, and on a review of existing names in the target environment. Check all names of that metadata item type in the target environment to see if your choice of name is appropriate. You may choose to change the name of this record in the XML file.
3. **Check that the ID number is correct in the XML file.** The correct ID number is either a missing ID number in the target environment (for import of a new metadata item), or the ID number of an existing metadata item (for update of that metadata item).
4. **Alternatively, you may choose to change the target environment** to ensure this import can proceed. Options are deleting or renaming one or more existing metadata items.

5. Once the XML or target environment are updated, then **repeat the import**. Note that if you update the XML file, there may be errors in the format of the XML file that stop the import from completing.

66 Prevent problem B

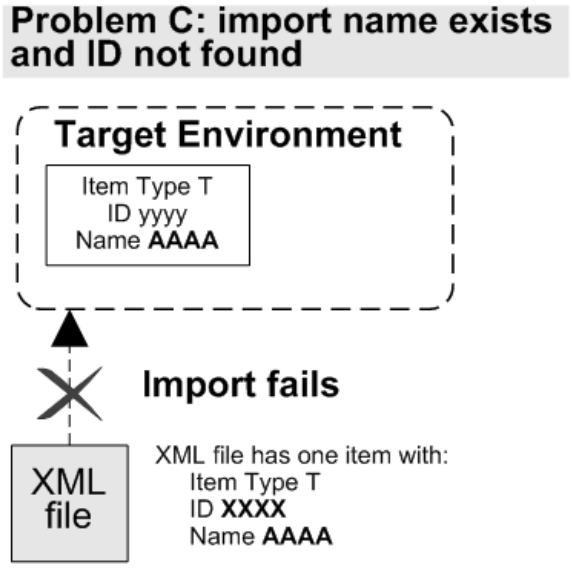
This problem is described in section "64 Possible problem B - import ID and name in different items" on page 70.

Some preparation before the import can prevent this problem occurring.

Options to prevent this problem	Details
Target environment is a new environment	1. Perform the 'easy' way in section "40 Prepare for ideal 'All New' - ID numbers important" on page 56. 2. **Import** to the new environment.
Analyze the XML file and the target environment to detect and fix problems before the import starts.	1. **Analyze the XML file** for the metadata type, ID number and name in each record for import. 2. **Check that each name is correct in the XML file.** The correct name is based on your choice of name, and on a review of existing names in the target environment. Check all names of that metadata item type in the target environment to see if your choice of name is appropriate. You may choose to change the name of this record in the XML file. 3. **Check that the ID number is correct in the XML file.** The correct ID number is either a missing ID number in the target environment (for import of a new metadata item), or the ID number of an existing metadata item (for update of that metadata item). 4. **Alternatively, you may choose to change the target environment** to ensure this import can proceed. Options are deleting or renaming one or more existing metadata items.

67 Possible problem C - import name exists and ID not found

In this case, the record in the XML file is only matched in the target environment with the name of the metadata item. The ID number in the record is not anywhere in the target environment.



On the "**Import Utility**" screen, this import is marked with result "**Fail**" and when you highlight the line with "Fail" a message appears in the "Errors" section of the screen.

An example error message for a physical file is: **The Physical File name 'AAAA' already exists. Please specify a different name.**

In this example, the message is identical to the example in section “64 Possible problem B - import ID and name in different items” on page 70.

68 Fix problem C

This problem is described in section “67 Possible problem C - import name exists and ID not found” on page 72.

Fix as follows:

1. **In the XML file, find the record of the correct type** that has the name "AAAA'.
2. **Check that the name is correct in the XML file.** The correct name is based on your choice of name, and on a review of existing names in the target environment. Check all names for that metadata item type in the target environment to see if your choice of name is appropriate. You may choose to change the name of this record in the XML file.
3. **Check that the ID number is correct in the XML file.** The correct ID number is either a missing ID number in the target environment (for import of a new metadata item), or the ID number of an existing metadata item (for update of that metadata item).
4. **Alternatively, you may choose to change the target environment** to ensure this import can proceed. Options are deleting or renaming one or more existing metadata items.

5. Once the XML or target environment are updated, then **repeat the import**. Note that if you update the XML file, there may be errors in the format of the XML file that stop the import from completing.

69 Prevent problem C

This problem is described in section “67 Possible problem C - import name exists and ID not found” on page 72.

Some preparation before the import can prevent this problem occurring.

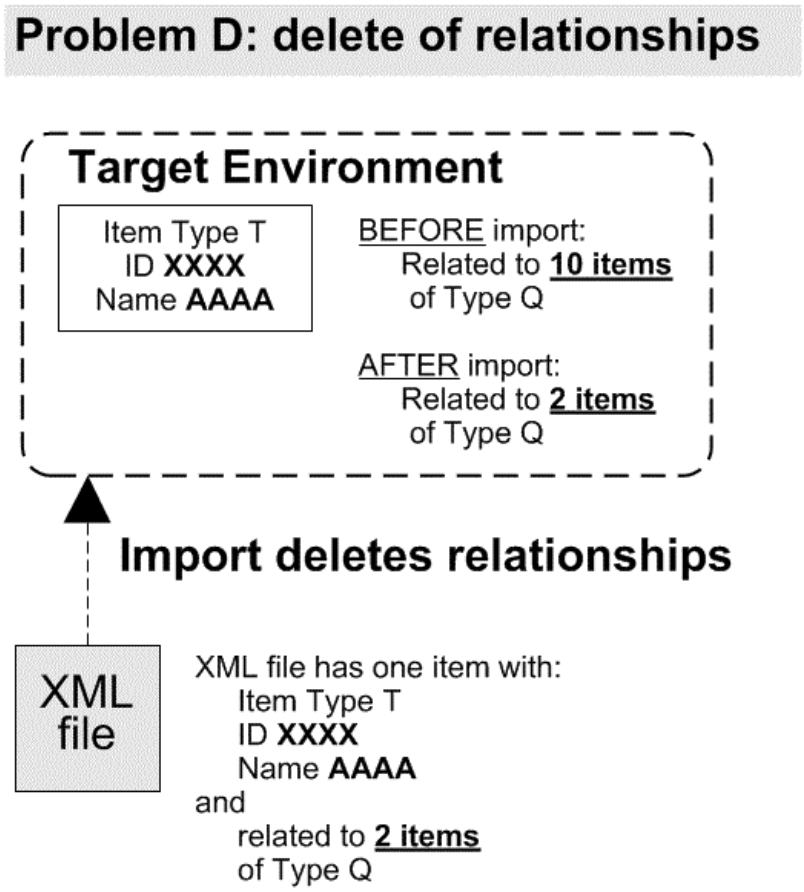
Options to prevent this problem	Details
Target environment is a new environment	1. Perform the 'easy' way in section “40 Prepare for ideal 'All New' - ID numbers important” on page 56. 2. **Import** to the new environment.
Analyze the XML file and the target environment to detect and fix problems before the import starts.	1. **Analyze the XML file** for the metadata type, ID number and name in each record for import. 2. **Check that each name is correct in the XML file.** The correct name is based on your choice of name, and on a review of any existing names in the target environment. Check all names of that metadata item type in the target environment to see if your choice of name is appropriate. You may choose to change the name of this record in the XML file. 3. **Check that the ID number is correct in the XML file.** The correct ID number is either a missing ID number in the target environment (for import of a new metadata item), or the ID number of an existing metadata item (for update of that metadata item). 4. **Alternatively, you may choose to change the target environment** to ensure this import can proceed. Options are deleting or renaming one or more existing metadata items.

70 Possible problem D - delete of relationships

In this case, the main record in the XML file has a matching main metadata item in the target environment. The problem is the XML file has different "related" items compared to the target environment. In this case, an import deletes existing relationships and replaces with the relationships in the XML file. This may be a problem for your company.

Be aware that the strategy for import is "**replace not merge**". This means that existing relationships in the target environment are deleted and replaced by the relationships in the XML file.

For example, a metadata item of type "T" has related items of type "Q". The XML file only has two related items of type Q, whereas the target environment as 10 related items of type Q. An import replaces 10 related items with two related items. The import has resulted in delete of at least eight existing relationships.



There is **no error message** warning of the delete of relationships.

Delete of existing relationships may be OK for your company for some data. No action is required if this is true.

71 Fix problem D

This problem is described in section "70 Possible problem D - delete of relationships" on page 74.

The only fix is to restore from a backup of the metadata that was deleted. The backup may be in XML files, or in a "whole" database backup.

Restore from backup as follows:

Type of backup available	How to restore
"Whole" database backup using the given utilities for that database tool. The tool may be, for example, DB2.	Use the given utilities for that database tool.

Type of backup available	How to restore
XML file(s) created by help topic "**Export metadata overview**".	For each backup XML file, go to section “50 Perform the import” on page 67

72 Prevent problem D

This problem is described in section “70 Possible problem D - delete of relationships” on page 74.

If the delete of existing relationships is a possible problem, consider the following actions:

Options to prevent this problem	Details
Backup target environment before import.	1. Before the import, **export** all metadata items that may be overwritten by an import. The exported XML files are the backups. 2. If an import deletes relationships that are required, restore the relationships using an **import from a backup**. The import recreates the deleted relattionships.
Analyze the XML file and the target environment to detect and fix problems before the import starts.	1. **Analyze the XML file** for the metadata type, ID number and name in each record for import. 2. **Check all related metadata in the XML file.** Make a list of the related items, the metadata types, ID numbers and names. 3. **Check the target environment.** List the equivalent metadata items in the target environment and check the related items. If there are differences, decide if the differences are OK or a problem. 4. **If the differences are a problem, modify the XML as appropriate.** This may require a new export to create a new XML file which you can then modify if required.

90 Structure of XML files

See help topic "**XML structure for metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

91 Does an ID number exist in an environment?

An XML file specifies an ID number for a certain metadata item. To determine if that ID number exists in a particular environment, do the following:

1. Log into the Workbench using the relevant environment.
2. In the Navigator, **click on the relevant metadata type**.
3. The Metadata List displays a list of existing metadata items of that type. **Click on the heading "Id"** to sort the list into descending order for ID number. If you click this heading multiple times, the list varies between descending and ascending order for ID number.

4. Look for the particular ID number in the Metadata List. If the ID number is not listed, then the ID number does not exist in that environment.

92 Is an ID number 'missing' in an environment?

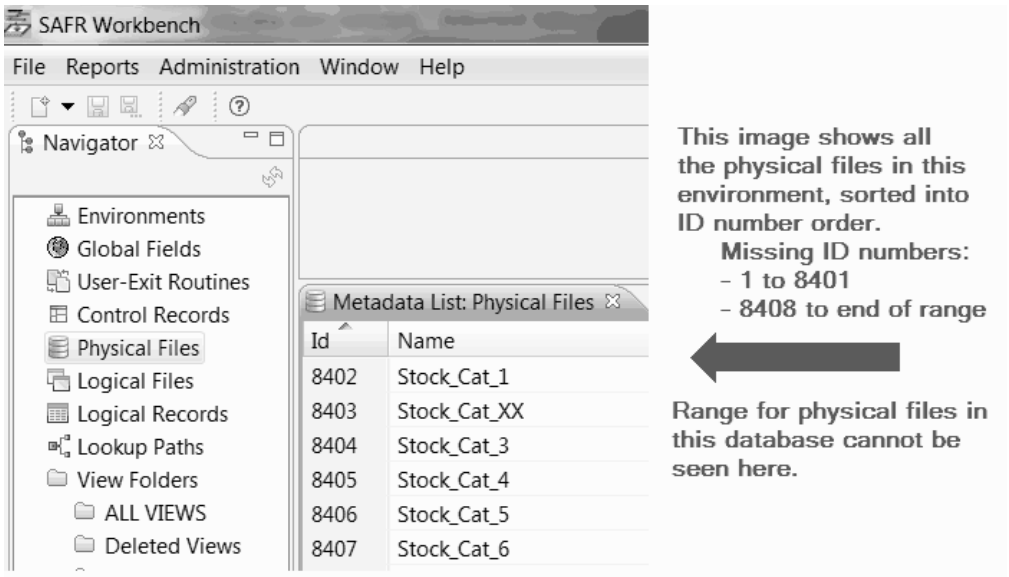
For the purposes of this help topic, a 'missing' ID number must meet two requirements:

- The ID number does not exist for that metadata type in that environment.
- The ID number is "in range" for that metadata type in that environment.

An XML file specifies an ID number for a certain metadata item. To determine if that ID number is 'missing' from a particular environment, do the following:

1. You must first determine the range of ID numbers for the relevant metadata type in that environment using section “93 How to find a range of ID numbers.” The ID number you are checking must be equal to or less than the range for the metadata type.
2. Log into the Workbench using the relevant environment.
3. In the Navigator, **click on the relevant metadata type**.
4. The Metadata List displays a list of existing metadata items of that type. **Click on the heading "Id"** to sort the list into descending order for ID number. If you click this heading multiple times, the list varies between descending and ascending order for ID number.
5. Look for the particular ID number in the Metadata List. If the ID number is not listed, then the ID number is 'missing' from that environment.

When the Metadata List is sorted into ID number order, most 'missing' ID numbers are clearly visible, as shown below.



93 How to find a range of ID numbers

A range of ID numbers applies to each metadata type in a database. The database may contain multiple environments. To import to a target environment, it is important to know the range of ID numbers for the relevant metadata type.

A range always starts at 1. The upper limit is the information you need.

There is an easy way and a hard way to find the upper limit to the range.

The easy way is to create a new "dummy" item of that metadata type in the target environment. The ID number of that "dummy" item is (at that moment) the upper limit for the range for that metadata item in that database. Delete that "dummy" item.

The hard way to find the upper limit of the range is as follows:

1. Log into the Workbench using the an environment in the target database for the intended import.
2. In the Navigator, **click on the relevant metadata type**.
3. The Metadata List displays a list of existing metadata items of that type. **Click on the heading "Id"** to sort the list into descending order for ID number. If you click this heading multiple times, the list varies between descending and ascending order for ID number. Make a note of the highest ID number in that Metadata List.
4. Follow the above steps for every environment in the database. The upper limit of the range is the largest ID number for that metadata type across all environments in that database.

94 Does a name already exist in an environment?

An XML specifies a name for a certain metadata item. To determine if that name already exists in that particular environment, do the following:

1. Log into the Workbench using the relevant environment.
2. In the Navigator, **click on the relevant metadata type**.
3. The Metadata List displays a list of existing metadata items of that type. **Click on the heading "Name"** to sort the list into descending order for name. If you click this heading multiple times, the list varies between descending and ascending order for name.
4. Look for the particular name in the Metadata List for the appropriate metadata type. If the name is listed, then the name exists in that environment.

95 Comparison of export and import functions in WW and WE

This online help is provided for the SAFR Workbench called **WE (Workbench Eclipse)**.

The export and import functions were provided in an older version of the SAFR Workbench called **WW (Windows Workbench).**

At a high level, the export and import in WW and WE are similar but not compatible. For example, an XML file created by an export in WW cannot be imported into WE.

A **comparison of the export functions of WW and WE** is as follows:

- WE provides export for only these metadata types:
  - Physical file,
  - Logical file,
  - Logical record,
  - Lookup path,

  - View.
- XML files exported from WE have a different format for date/time stamp compared to XML files exported from WW. The formats are as follows:
  - WW format in the USA is: **MM/DD/YYYY HH:MM:SS AMPM** (at midnight the time part is omitted)
  - WW format in some other countries is: **DD/MM/YYYY HH:MM:SS AMPM** (at midnight the time part is omitted)
  - WE format is: **YYYY-MM-DD HH:MM:SS** (24 hour clock)
- WE does not request selecting a SAFR database at the start of the export function. WE export uses the database for your current session.
- WE provides a choice of location for the XML file that is generated by the export.

A **comparison of the import functions of WW and WE** is as follows:

- WE import performs more checks on the XML files and their relationship with existing metadata in the target environment.
- WE import is more consistent and reliable.
- WE import ensures that serious potential problems are blocked and the import cannot proceed.
- WE import provides a warning for minor potential problems, and gives a choice to proceed or cancel the import.
- Overall, the WE import function preserves the integrity of the metadata in the target environment after import is complete.

97 Import history in log file

For an introduction to WE log files, see topic "**WE log file overview**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

The WE log file records data about any imports you perform in this session. An example extract from a WE log file is below:

```
20/09/2013 3:41:30 PM
com.ibm.safr.we.model.utilities.importer.ImportUtility showParams
INFO: >>>IMPORT PARAMETERS...
Target Env Id   = 110
Component Type  = View
View Folder Id  = 869
Replace Ids     = yes
XML files...
PRODUCT_REPORT_ONLY_01[8472].xml
20/09/2013 3:41:51 PM com.ibm.safr.we.TimingMap report
INFO: DAO Method timings
20/09/2013 3:41:51 PM com.ibm.safr.we.model.SAFRModelCount report
INFO: Model counts
20/09/2013 3:41:51 PM
com.ibm.safr.we.model.utilities.importer.ImportUtility logErrors
INFO: >>>IMPORT COMPLETED without errors.
```

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Logical files overview

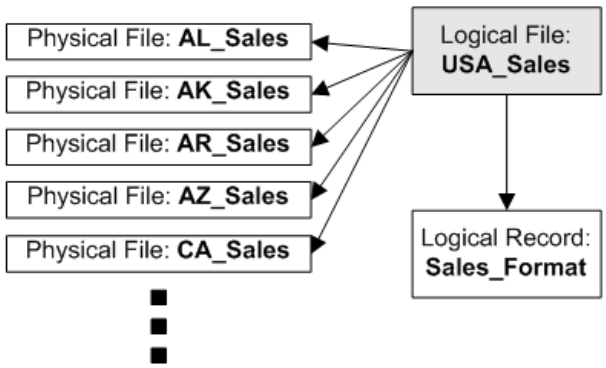
01 Summary of this topic

The sections in this topic are as follows:

10 What is a logical file?

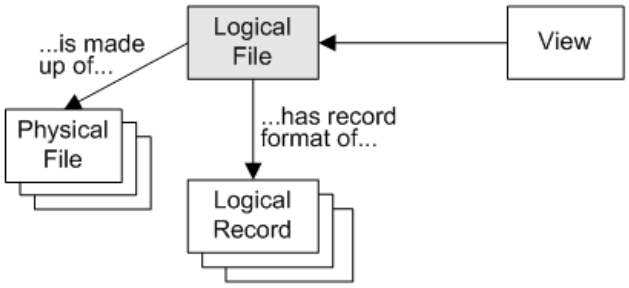
A logical file is a collection of **physical files with the same record layout**.

For example, your company may have **sales files for all 50 states of the USA**. Each sales file is a physical file, and all share the **same record layout in a logical record**. Together all these sales files form one logical file which contains the entire USA sales data for your company.



Logical files process much faster in SAFR because all the physical files can process in parallel. This is one of the speed advantages of SAFR.

The mandatory requirement is that all physical files in the logical file share the same record layout. The record layout is called a logical record. **Every logical file has at least one logical record.**



For an introduction to physical files, see topic "**Physical Files overview**". For an introduction to logical records, see topic "**Logical Records overview**". These topics are elsewhere in this PDF - see the table of contents.

20 How do I use a logical file?

All users in that environment can **use logical files in a view**. For more see these topics:

- "**Views overview**",
- "**Views - advanced overview**",
- "**Creating views**",
- "**Modifying views**".

The overview topics are elsewhere in this PDF - see the table of contents. To find the other topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

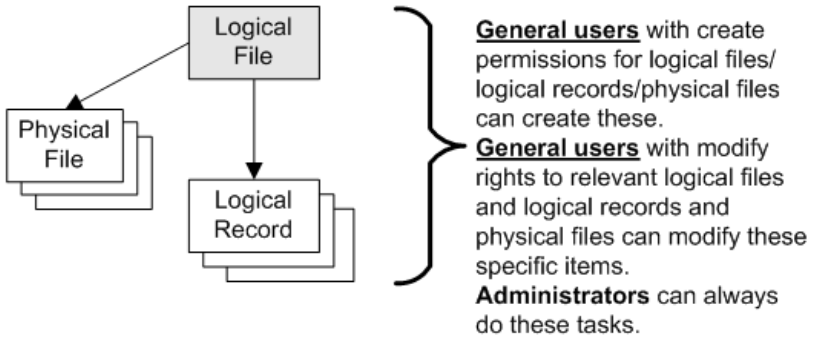
30 How do I know which Logical Files to use?

Check if you can read logical file records as follows: in the **Navigator**, click on "**Logical Files**", and in the **Metadata List** double click on any logical file. If the name is grey then you do not have read access to that logical file.

General users need to be aware of help topic "**What metadata do I want to see?**" To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

See your system or environment administrator if you need more access to logical files in your environment, or if you need more logical files created and you do not have the authority to do this yourself.

50 How do I create or modify a Logical File?



System administrators and environment administrators can always **create or modify logical files**.

General users can **create or modify logical files** if the group for login has the following authorities:

- **Create Logical File** permission in the relevant environment,
- **Create Logical Record** permission in the relevant environment (if required),
- **Create Physical File** permission in the relevant environment (if required),
- **Modify or Delete rights to a particular logical file(s)** in that environment. (This allows modify of the logical file itself and the links to physical files),
- **Modify or Delete rights to a relevant logical record(s)** in that environment (if required),
- **Modify or Delete rights to a relevant physical file(s)** in that environment (if required).

For more on these authorities, see topics "**Groups overview**" , "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

Once the group has above authorities, users in that group can **create or modify logical files** by using the tasks below, which are **administrator tasks:**

- "**Creating logical files**",
- "**Creating logical records**",
- "**Creating physical files**",
- "**Modifying logical files**" (to modify the logical file itself and the links to physical files),
- "**Modifying logical records**" (to modify the links between logical files and logical records),
- "**Modifying physical files**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

90 How do I delete a logical file?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Logical records overview

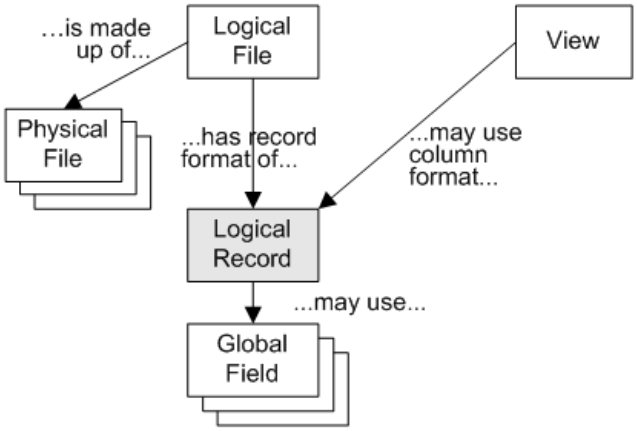
01 Summary of this topic

The sections in this topic are as follows:

10 What is a logical record?

A Logical Record can be one of two things:

- A **record format** for one or more **logical files**
- The **format of columns** in a report in one or more **views**



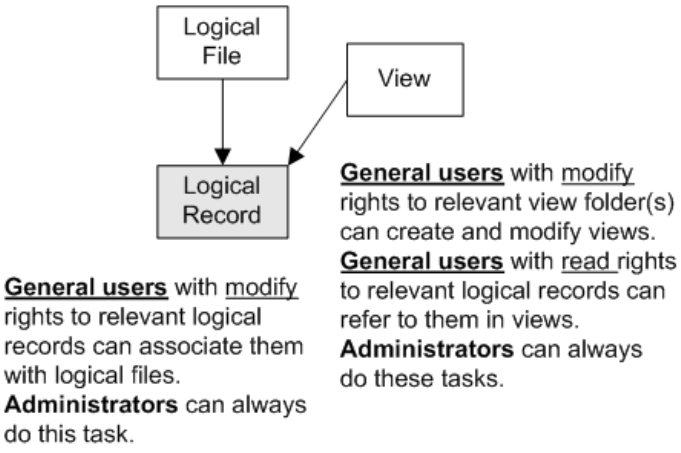
A logical record consists of fields that **may be global fields**.

For further introductions, see:

- **Logical Files overview**,
- **Physical Files overview**,
- **Views overview**,

- **Global Fields overview**.

20 How do I use a logical record?



All users in that environment can use **logical records in a view**, to describe column headings in a formatted report, as shown in these topics:

- **Creating views**,
- **Modifying views**.

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

Logical records are used in logical files as the record format. For this usage, see heading **How do I create or modify a logical record?** below.

30 How do I know which logical record to use?

Check if you can read logical records as follows: in the **Navigator**, click on "**Logical Records**", and in the **Metadata List** double click on any logical record. If the name is grey then you do not have read access to that logical record.

General users need to be aware of help topic "**What metadata do I want to see?**" To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

See your system or environment administrator if you need more access to logical records in your environment, or if you need more logical records created and you do not have the authority to do this yourself.

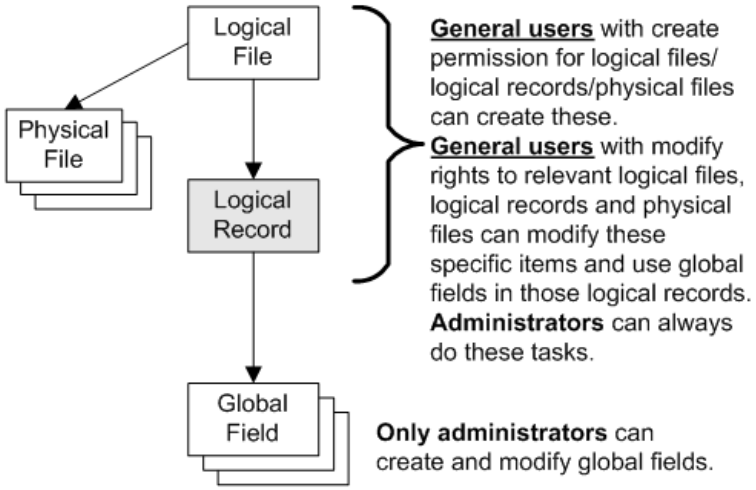
40 Logical Record Report

To see a **report on selected logical records**, use one of these topics:

- FAQ topic "**How do I generate a Logical Record Report?**"
- Task "**Generating reports on metadata**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

50 How do I create or modify a logical record?



System administrators and environment administrators can always **create or modify logical records**.

General users can **create or modify logical records** if the group for login has the following authorities:

- **Create Logical File** permission in the relevant environment (if required).
- **Create Logical Record** permission in the relevant environment.
- **Create Physical File** permission in the relevant environment (if required).
- **Modify or Delete rights to the relevant logical files** in that environment. (This allows modify of the logical file itself and the links to physical files.)
- **Modify or Delete rights to the logical records** in that environment.
- **Modify or Delete rights to the relevant physical files** in that environment (if required).

For more on these authorities, see topics "**Groups overview**" , "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

Once the group has the above authorities, users in that group can **create or modify logical records** by using the tasks below, which are **administrator tasks:**

- **Creating logical files**,
- **Creating logical records**,
- **Creating physical files**,
- **Modifying logical files** (to modify the logical file itself and the links to physical files),
- **Modifying logical records** (to modify the links between logical files and logical records),
- **Modifying physical files**.

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

90 How do I delete a logical record?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Logic text overview

What is logic text?

Logic text is optional text added to a view that changes how the results of the view. There are four types of logic text:

- Logic text 1: Extract Record Filter
- Logic text 2: Extract Column Assignment
- Logic text 3: Format Column Calculations
- Logic text 4: Format Record Filter

Under "**Related Concepts**" below, click on links to overviews for these.

Logic text 1: Extract Record Filter overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is logic text for Extract Record Filter?

All logic text is optional. The main objective of this logic text is to select or skip input records for processing a view.

Specifically, this logic text changes the **records selected for processing** in the extract phase of the SAFR Performance Engine. The records selected during this phase are the **input records** for processing the view(s) in that run of the Performance Engine.

This logic text can also write selected input record to extract files.

All Extract Record Filter logic text is performed at the start of the extract phase in SAFR processing.

For an introduction to SAFR phases, see topic "**SAFR phases overview**". That topic is elsewhere in this PDF - see the table of contents.

20 Where do I find syntax and examples of this logic text?

The syntax of this logic text and examples are described in topic "**Logic text 1: Extract Record Filter**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I create logic text for Extract Record Filter?

This logic text is part of a view and is associated with a view source file. To create this logic text in an existing view, do the following:

1. Ensure you are on the "**Edit View**" screen for the relevant view. For help with displaying this screen, see topic "**Modifying Views**" in the General User Guide.
2. The view must already have at least one view source defined. If there is no view source defined, see topic "**Edit View (View Editor tab) screen help**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
3. Ensure you are on the "**View Editor**" tab. If the "**View Properties**" tab is displayed, click the **Show Grid / Properties** button. or press F9 or select **Edit -> Show Grid/Properties**.
4. In the column immediately below "View Editor", if there is a plus + sign to the left of '**View Source**", double click the plus sign to expand the list of view sources.
5. Click in the cell with a relevant view source name under "**View Sources**".
6. Double click on the cell to the right of "**Record Filter**" and click  .
7. Type your logic text in the window "**Create New Extract Record Filter**" or "**Edit Extract Record Filter**".
8. Click ✓ to check if the logic text is valid.
9. Fix any errors shown in the "**Logic Text Validation Errors**" window.
10. When the new logic text is complete and valid,
    - **EITHER** click  (the save icon),
    - **OR** select **File, Save**,
    - **OR** press **Ctrl+S**.
11. You may wish to close some of the open windows, such as "**Logic Text Helper**" and "**Create New Extract Record Filter**" or "**Edit Extract Record Filter**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Logic text 2: Extract Column Assignment overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is logic text for Extract Column Assignment?

All logic text is optional. The objective of this logic text is to create columns for processing in a view, and possibly to write certain records to files.

Specifically, this logic text **defines column values** in files processed in the extract phase of the SAFR Performance Engine. Writing of input records to files is also controlled in this logic text.

For an introduction to SAFR phases, see topic "**SAFR phases overview**". That topic is elsewhere in this PDF - see the table of contents.

20 Where do I find syntax and examples of this logic text?

The syntax of this logic text and examples are described in topic "**Logic text 2: Extract Column Assignment**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I create logic text for Extract Column Assignment?

This logic text is part of a view and is associated with a column and a view source file. To create this logic text in an existing view, do the following:

1. Ensure you are on the "**Edit View**" screen for the relevant view. For help with displaying this screen, see topic "**Modifying Views**" in the General User Guide.
2. The view must already have at least one view source defined. If there is no view source defined, see topic "**Edit View (View Editor tab) screen help**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
3. Ensure you are on the "**View Editor**" tab. If the "**View Properties**" tab is displayed, click the **Show Grid / Properties** button.  or press F9 or select **Edit -> Show Grid/Properties**.
4. In the column immediately below "View Editor", if there is a plus + sign to the left of '**View Source**", double click the plus sign to expand the list of view sources.
5. Click on a cell in the relevant column that is also in the row for the relevant view source file.
6. The "**Column Source Properties**" window appears. Double click "**Column Source Type**" and select "**Formula**" from the drop down box.
7. Double click "**Column Source Value**" and click  .
8. Type your logic text in the window "**Create New Extract Column Assignment**" or "**Edit Extract Column Assignment**".
9. Click ✓ to check if the logic text is valid.
10. Fix any errors shown in the "**Logic Text Validation Errors**" window.
11. When the new logic text is complete and valid,
    - **EITHER** click  (the save icon),
    - **OR** select **File, Save**,

- **OR** press **Ctrl+S**.

12. You may wish to close some of the open windows, such as "**Logic Text Helper**" and "**Create New Extract Column Assignment**" or "**Edit Extract Column Assignment**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Logic text 3: Format Column Calculations overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is logic text for Format Column Calculations?

The logic text "Format Column Calculations" can also be called "**Format-Phase Calculations**".

All logic text is optional. The objective of this logic text is to create columns based on numeric calculations for processing in a view. This logic text is limited to **numeric columns** in a view.

Specifically, this logic text **defines calculations** for columns in the format phase of the SAFR Performance Engine. This phase is optional, because the extract phase may be sufficient to produce the results of that view.

For an introduction to SAFR phases, see topic "**SAFR phases overview**". That topic is elsewhere in this PDF - see the table of contents.

20 Where do I find syntax and examples of this logic text?

The syntax of this logic text and examples are described in topic "**Logic text 3: Format Column Calculations**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I create logic text for Format Column Calculations?

This logic text is part of a view and is associated with a column during the format phase. To create this logic text in an existing view, do the following:

1. Ensure you are on the "**Edit View**" screen for the relevant view. For help with displaying this screen, see topic "**Modifying Views**" in the General User Guide.
2. The view must have a format phase. If there is no format phase or you are not sure, see topic "**Edit View (View Properties, General tab) screen help**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

3. Ensure you are on the "**View Editor**" tab. If the "**View Properties**" tab is displayed, click the **Show Grid / Properties** button.  or press F9 or select **Edit -> Show Grid/Properties**.
4. Select a **numeric column** that is not part of the sort key. (A numeric column is any Data Type that is not Alphanumeric.) In that column **double click** on the cell for row "**Format-Phase Calculation**". That cell may be grey, which means this logic text cannot be applied. This may be because the column is part of the sort key, or is not numeric. This may also be because there is no format phase.
5. Click  .
6. Type your logic text in the window "**Create New Format-Phase Calculation**" or "**Edit Format-Phase Calculation**".
7. Click ✓ to check if the logic text is valid.
8. Fix any errors shown in the "**Logic Text Validation Errors**" window.
9. When the new logic text is complete and valid,
   - **EITHER** click  (the save icon),
   - **OR** select **File, Save**,
   - **OR** press **Ctrl+S**.
10. You may wish to close some of the open windows, such as "**Logic Text Helper**" and "**Create New Format-Phase Calculation**" or "**Edit Format-Phase Calculation**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Logic text 4: Format Record Filter overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is logic text for Format Record Filter?

Format Record Filter logic text can also be called **Format-Phase Record Filter** logic text.

All logic text is optional. The objective of this logic text is to select or skip output records for a view.

Specifically, this logic text changes the **records selected for output** in the format phase of the SAFR Performance Engine. The format phase is optional, because the extract phase may be sufficient to produce the results of that view.

For an introduction to SAFR phases, see topic "**SAFR phases overview**". That topic is elsewhere in this PDF - see the table of contents.

20 Where do I find syntax and examples of this logic text?

The syntax of this logic text and examples are described in topic "**Logic text 4: Format Record Filter**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I create logic text for Format Record Filter?

This logic text is part of a view and is associated with the format phase. To create this logic text in an existing view, do the following:

1. Ensure you are on the "**Edit View**" screen for the relevant view. For help with displaying this screen, see topic "**Modifying Views**" in the General User Guide.
2. The view must have a format phase. If there is no format phase or you are not sure, see topic "**Edit View (View Properties, General tab) screen help**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
3. Ensure you are on the "**View Properties**" tab. If the "**View Editor**" tab is displayed, click the **Show Grid / Properties** button.  or press F9 or select **Edit -> Show Grid/Properties**.
4. Go to the "**View Properties, Format Phase**" tab, and click on the "**Create**" or "**Edit**" button under heading" **Format-Phase Record Filter**".
5. Type your logic text in the window "**Create New Format-Phase Record Filter**" or "**Edit Format-Phase Record Filter**".
6. Click ✓ to check if the logic text is valid.
7. Fix any errors shown in the "**Logic Text Validation Errors**" window.
8. When the new logic text is complete and valid,
   - **EITHER** click  (the save icon),
   - **OR** select **File, Save**,
   - **OR** press **Ctrl+S**.
9. You may wish to close some of the open windows, such as "**Logic Text Helper**" and "**Create New Format-Phase Record Filter**" or "**Edit Format-Phase Record Filter**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

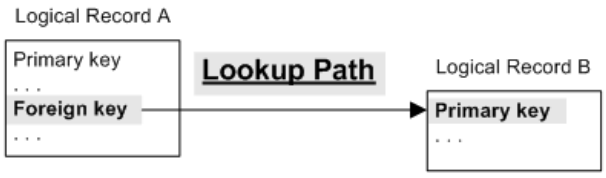
Lookup paths overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is a lookup path?

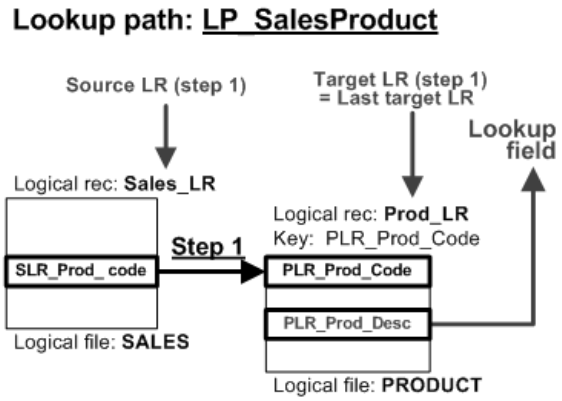
A lookup path is a method where a field in record A allows access to a completely different record B. A field in record A is a "foreign key" that can access record B where that field is a "primary key". This is similar to the DB2 situation called a "left outer join".



The purpose is that a field in record A now gives access to all the information in record B.

Consider an example: for each sales transaction we need to know the product description .

In this example, call the lookup path **LP_SalesProduct**.



Note the following:

- Each step has a **source** logical record and a **target** logical record.
- Each step must **create a primary key** for the target logical record.
  That primary key can be made up of:
  - A field in the source logical record (for example the "foreign key" might be all you need).
  - Extra fields in the source logical record, when a primary key may be a combination of other fields.
  - A constant - see section “34 Constant in a lookup path step” on page 96
  - A symbol - see section “36 Symbol in a lookup path step” on page 97
- In the example above the lookup path has only one step. For multiple steps, see section “30 Multi-step lookup paths” on page 95.
- The purpose of the example above is to access the field "**PLR_Prod_Desc**". This field is the **lookup field** for the example above. Any field in any of the target logical records can be a lookup field.

20 Where to call lookup paths in a view

You can call a lookup path from four locations, as given in the sections below:

1. “22 Call a lookup from a Column Lookup Field”
2. “24 Call a lookup from a Sort Key Title” on page 94
3. “26 Call a lookup from Extract Column Assignment logic text” on page 94
4. “28 Call a lookup from Extract Record Filter logic text” on page 94

22 Call a lookup from a Column Lookup Field

Aspect	Details
Where to call lookup path	A column in a view that has **Column Source Type** set to Lookup Field.
Phase for lookup path	Extract phase.
How to call the lookup path	See topic "**Edit View (View Editor tab) screen help**" and look in the table of actions for the following rows: • "**Define a column source type**" • "**Define a lookup field**"

Aspect	Details
Reasons to use this	This is useful when a column in a view is simply the result of a lookup path. For example, use a column is the product name which is the result of using the product id in a lookup path. The lookup path provides the value and no further processing is necessary.

24 Call a lookup from a Sort Key Title

Aspect	Details
Where to call lookup path	A sort key column that has a **Sort Key Title.**
Phase for lookup path	Format phase.
How to call the lookup path	See topic "**Edit View (View Editor tab) screen help**" and look in the table of actions for the following rows: • "**Define a column source type**" • "**Make a sort key**" • "**Modify a sort key**"
Reasons to use this	Purely for a sort key title that needs a value supplied by a lookup path. For example, the sort key may be a product id and the sort key title is the product name.

26 Call a lookup from Extract Column Assignment logic text

Aspect	Details
Where to call lookup path	A column in a view that has **Column Source Type** set to Formula. The logic text needs to be created and modified for this column.
Phase for lookup path	Extract phase.
How to call the lookup path	See topic "**Edit View (View Editor tab) screen help**" and look in the table of actions for the following rows: • "**Define a column source type**" • "**Define a formula**"
Reasons to use this	This is necessary when the lookup is part of a more complex calculation. For example, the product id can be input to a lookup that finds a discount percentage for that product. Once found the discount percentage is part of a new calculation of the product price, so performing the lookup is only part of the processing required to calculate the price. This situation needs logic text.

Note that when a lookup path is typed in logic text, you can optionally include the lookup field name in the logic text. For full details, see topic "**Syntax: lookup paths**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

28 Call a lookup from Extract Record Filter logic text

Aspect	Details
Where to call lookup path	Click on a view source file so that the "**View Source Properties**" screen appears. Create or modify logic text in the field "**Record Filter**".
Phase for lookup path	Extract phase.

Aspect	Details
How to call the lookup path	See topic "**Edit View (View Editor tab) screen help**" and look in the table of actions for the following rows: • "**Create Extract Record Filter logic text**" • "**Modify or delete Extract Record Filter logic text**"
Reasons to use this	This is when a lookup is required to select or deselect input file records for processing. For example, the product id can be input to a lookup path that provides the product category, and the view can select records from only certain product categories to process.

Note that when a lookup path is typed in logic text, you can optionally include the lookup field name in the logic text. For full details, see topic "**Syntax: lookup paths**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 Multi-step lookup paths

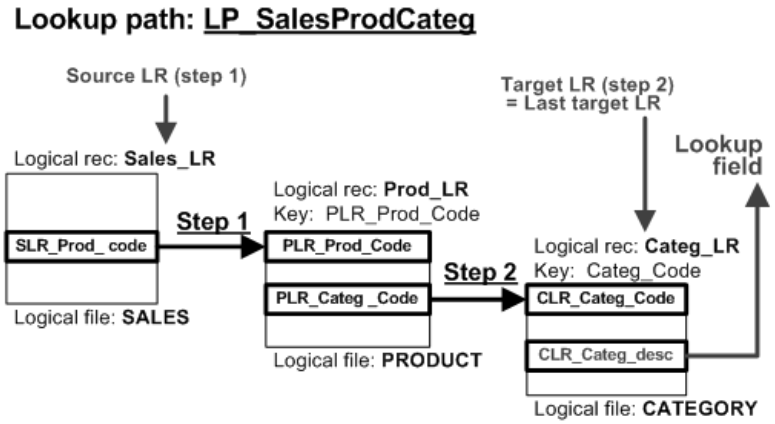
A lookup can have two ore more steps that allows access to more logical records in a single lookup path.

In the example below, for each sales transaction we need to know the product category description.

In this example, call the lookup path **LP_SalesProdCateg**.

The **steps** are as follows:

1. Use **SLR_Prod_code** to access the **Prod_LR** logical record.
2. Use **PLR_Categ_code** to access the **Categ_LR** logical record.

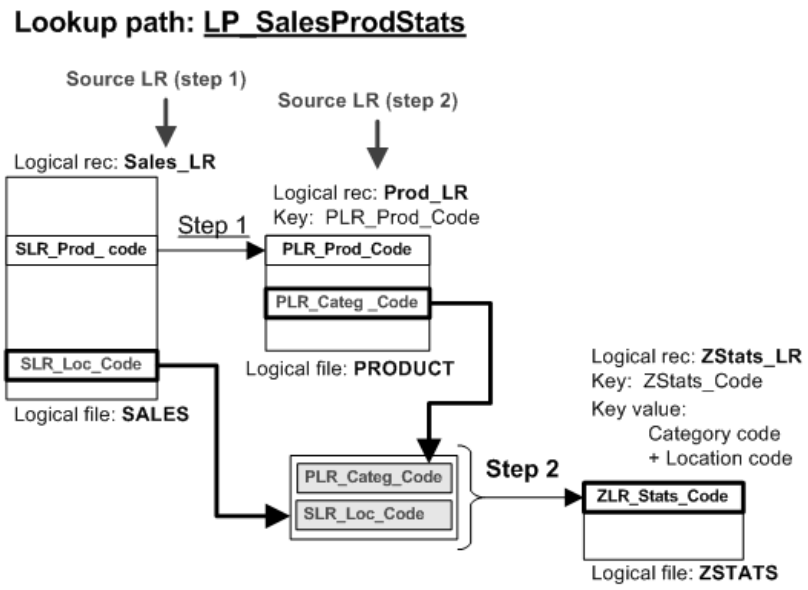


32 Use fields from the current and previous steps

Each step in a lookup path creates a primary key. That process can use the following fields:

- Fields in the source logical record for the current step, and
- Fields in the source logical records for previous steps.

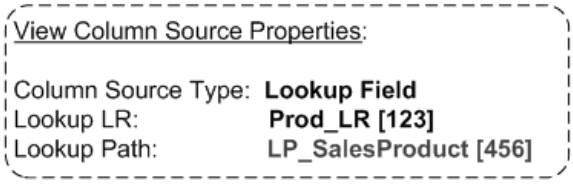
In the example below, step 2 uses fields from Step 1 and 2.:



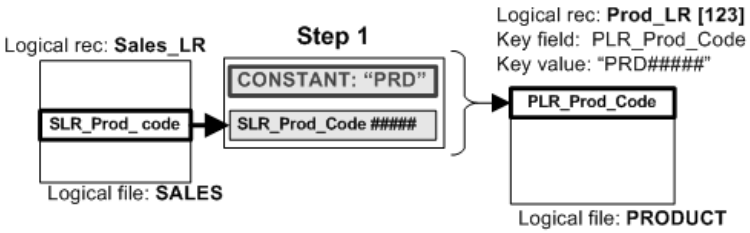
34 Constant in a lookup path step

Each step of a lookup path creates the primary key to a target logical record. That primary key may contain a constant value.

In the example below, lookup path **LP_SalesProduct** uses a constant. The primary key to Prod_LR consists of "PRD" followed by 5 digits. The lookup path use a constant for the "PRD" and the 5 digits come from field **SLR_Prod_code**:



Lookup path: LP_SalesProduct [456]



A constant works for all calls to lookup paths listed in section “20 Where to call lookup paths in a view” on page 93.

36 Symbol in a lookup path step

A symbol is a compromise between a data field and a constant. The symbol value changes (unlike a constant) and is not a data field in the source logical record.

A symbol value has a default value in the lookup path, so you only need to change the symbol for exceptions. If symbol always uses the default value, then the symbol is effectively a constant.

A symbol is necessary because the primary key changes structure as given in this example below:

- For products from 00001 to 49999, the primary key is "PRD#####".
- For products from 50000-59999, the primary key is "WHL#####" (where WHL is for wholesale).
- For products from 60000 onwards, the primary key is "RET#####" (where RET is for retail).

In the above example, "PRD" is default value, and this is changed to "WHL" or "RET" if the product is in the appropriate range of numbers. How is the symbol changed from the default value?

A symbol can have a non-default value set only in logic text. This means **this is only possible in:**

- Section “26 Call a lookup from Extract Column Assignment logic text” on page 94
- Section “28 Call a lookup from Extract Record Filter logic text” on page 94

A symbol **always uses the default value** in:

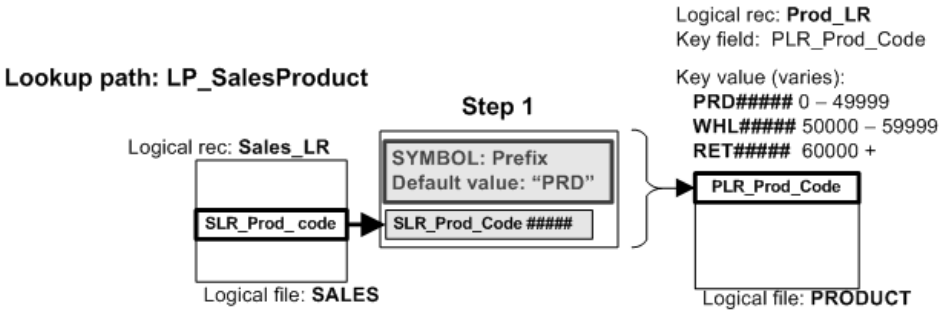
- Section “22 Call a lookup from a Column Lookup Field” on page 93
- Section “24 Call a lookup from a Sort Key Title” on page 94

To set a symbol to a non-default value in logic text, the syntax requires "**$**" (a dollar sign) before the symbol name. The actual symbol name always starts with a letter.

Lookup path LP_SalesProduct uses a **symbol called Prefix** which has a default value of "PRD". In Extract Column Assignment logic text, the call to LKProduct is different depending on the value of the 5 digits. This is shown below:

```
Extract Column Assignment logic text for a column in a view:

    IF {SLR_Product_Code} < 50000
       THEN Column = {LP_SalesProduct}
       ELSE IF {SLR_Product_Code} < 60000
                THEN Column = {LP_SalesProduct ; $Prefix="WHL"}
                ELSE Column = {LP_SalesProduct ; $Prefix="RET"}
            ENDIF
    ENDIF
```



Notice how the first call to LP_SalesProduct does not give a value for symbol Prefix so the default value of "PRD" applies.

Note that a lookup path can only be called from logic text in the extract phase, which means either **Extract Record Filter** or **Extract Column Assignment.** For an introduction to logic text, see topic "**Logic Text overview**". That topic is elsewhere in this PDF - see the table of contents.

40 Effective date lookup paths

An effective date lookup means the lookup finds a target logical record with the correct key and the correct date.

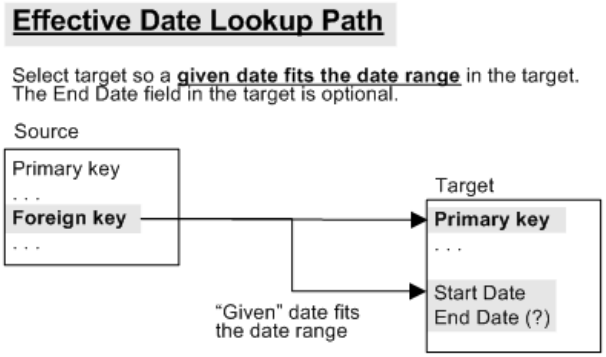
This feature can be used for any call to a lookup, as shown in section "20 Where to call lookup paths in a view" on page 93.

An effective date lookup needs the following:

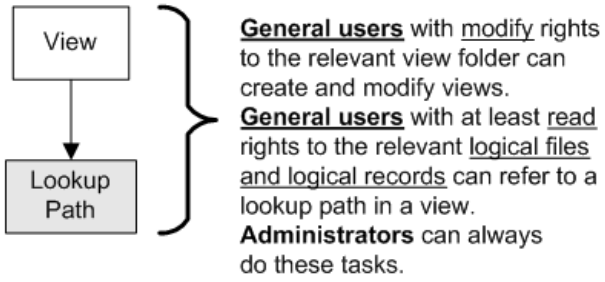
- The target logical record to have a **date range**.

  This date range exists if at least one date field in the target logical record is marked "**Start date**". Optionally, the record can also have a date field marked "**End date**". These two dates **cannot** be part of the primary key of that logical record.
- The view to supply a "**given date**". The given date selects a target logical record by date.

  The "**given date**" can be one of:
  - A **field** in the **source** logical record,
  - The **run date** for that run of the Performance Engine,
  - A **constant** value given in the view or logic text calling the lookup path.



50 How do I use a lookup path?



Lookup paths can be used in the **extract phase** and **format phase** processing for a view.

Choose a location to call a lookup path by using section "20 Where to call lookup paths in a view" on page 93.

You need to know these things about a lookup to use it:

- The **lookup path name**.
- The **last target logical record**.
- The **last target logical file**.
- The **lookup field**. This could be any field in the last target logical record.
- Any **effective date** requirements for the lookup path.
- The **default and required values** for any **symbols** in that lookup path.

  For example, in "36 Symbol in a lookup path step" on page 97 if the product number it 50001 then the default value is "PRD" and the required value is 'WHL".

  If the required value of the symbol is different from the default value then this is only changed in logic text. The two types of logic text available for this are **Extract Record Filter** and **Extract Column Assignment**

A lookup path can only be used if the **status** of the lookup path is **active**. If the status if inactive, fix all problems with the lookup path and save.

See the following topics:

- "**Creating lookup paths**",
- "**Creating views**",
- "**Modifying lookup paths**",
- "**Modify views**",
- "**Batch activate lookup paths**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

Any user with access to an environment can create or modify lookup paths in that environment.

60 How do I know which lookup path to use?

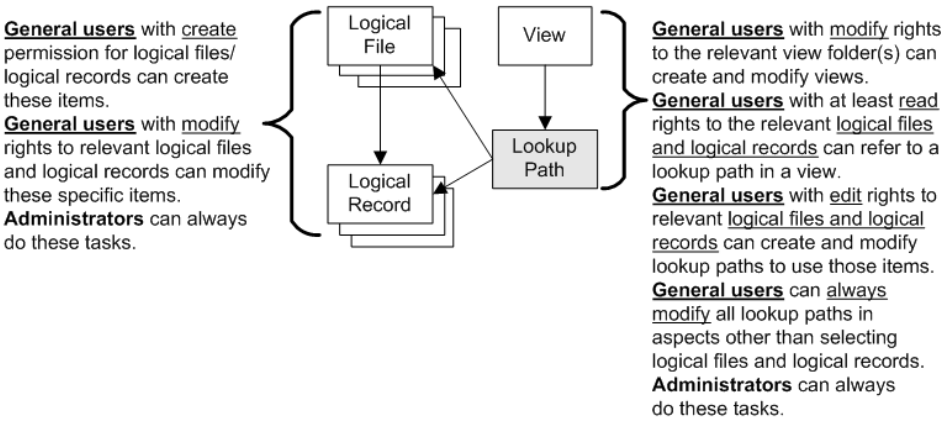
Read lookup paths as follows: in the **Navigator**, click on "**Lookup Paths**", and in the **Metadata List** double click on any lookup path.

General users need to be aware of help topic "**What metadata do I want to see?**" To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

See your system or environment administrator if you need more access to logical files and logical records in your environment, or if you need more logical files and logical records created and you do not have the authority to do this yourself.

Another way is to review lookup paths is to use the report described in the next section.

70 How do I create or modify a lookup path?



General users can **create or modify logical files and logical records** if the group for login has the following authorities:

- **Create Logical File** permission in the relevant environment.
- **Create Logical Record** permission in the relevant environment.
- **Create Physical File** permission in the relevant environment (if required).
- **Modify or Delete rights to the specific logical files** in that environment.

- **Modify or Delete rights to the specific logical records** in that environment.
- **Modify or Delete rights to the specific physical files** in that environment (if required).

General users can **create and modify lookup paths**, in general, with some restrictions. To create or update a lookup path to use **new logical files and logical records**, the group selected during login must have at least **edit** rights to those relevant logical files and logical records. All other aspects of a lookup path can be created or modified by all general users at all times.

General users can **modify a view to use a lookup path** if the group selected during login has at least **modify** rights to the relevant view folder that contains the view and at least **read** rights to the relevant logical files and logical records.

System administrators and environment administrators can always do the above tasks.

For more on these authorities, see topics "**Groups overview**" , "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

If you have enough authority, **create or modify lookup paths and refer to them in views** by using the tasks below, which are **general user tasks**:

- "**Creating lookup paths**",
- "**Creating views**",
- "**Modifying lookup paths**",
- "**Modify views**",

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

80 Lookup Path Report

All users in an environment can run this report on any lookup path, using these topics:

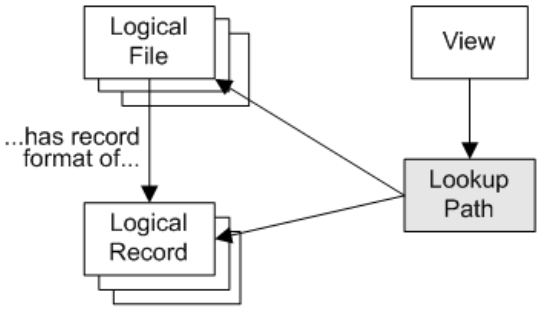
- FAQ topic "**How do I generate a Lookup Path Report?**"
- Task "**Generating reports on metadata**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

85 Metadata relevant to lookup paths

A lookup path is referred to in a **view**. The reference could be in a column definition or in logic text.

A lookup path refers to **logical files** and the fields in the associated **logical records**.



This topic is connected to the following topics:

- "**Logical files overview**",
- "**Logical records overview**",
- "**Views overview**",

These topics are elsewhere in this PDF - see the table of contents.

90 How do I delete a lookup path?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Metadata overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is metadata?

Metadata is the group of items defined inside the SAFR Workbench that enable work to get done.

To use the SAFR Workbench, you need these types of items:

- **General users** who login and configure SAFR to produce results for your company.
- **Groups** with members that are general users and environment administrators.
- **Environment administrators** who prepare environments for general users.
- **System administrators** who create environments and nominate environment administrators and also prepare environments for general users.

To configure SAFR to produce results for your company, users create and modify these types of items:

- **Views** to describe the results that SAFR must produce.
- **Logical files** to describe collections of input and output data.
- **Logical records** to describe record layouts for logical files.
- **Physical files** to describe actual datasets on the mainframe systems in your company.
- **Global fields** to describe common fields of data in the mainframe systems in your company. For example, a customer identification number might be a global field.
- **Lookup paths** to describe how one or more source files can lookup a target file by using fields of data.
- **Control records** to control how a view processes data.
- **View folders** to contain collections of views.
- **User-exit routines** to performs some unique and necessary computing task specifically for your company.
- **Environments** to contain all the items above in logical collections, such as **Sales** or **Accounts_Receivable**.

For example, your SAFR Workbench needs general users, groups and environments. System and environment administrators setup the environments with and other items so that general users can do work. General users create views and lookup paths that you need for SAFR to produce results.

20 Items of metadata

An item of metadata is a particular example, such as a particular view.

Every metadata item has the following:

- A **name**.

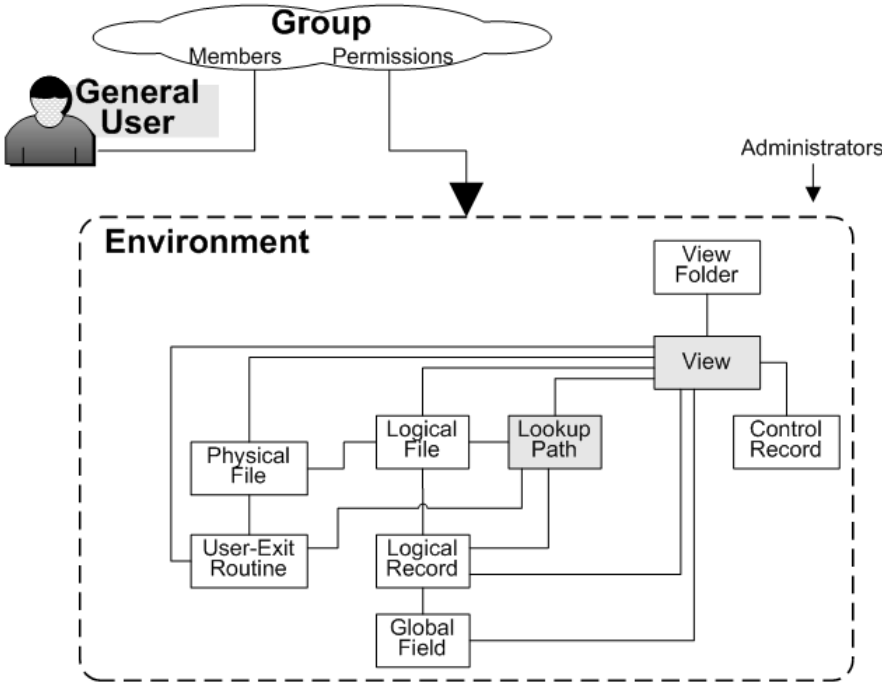
  For example, a logical file might be named "**Sales_USA**".
- An **ID** number.

  For example, logical file "Sales_USA" might have an ID number of **123**. ID numbers in SAFR are often written inside square brackets: **[123]**.

  Often, both the name and ID are given in the format "name[ID]", for example "**Sales_USA[123]**".
- A **type**.

  For the example of "**Sales_USA[123]**" the type is **logical file**.

30 How metadata item types are related



The diagram above shows the metadata types in SAFR. The lines show where metadata types are related. The relationship may be one to many or many to many, and may be only in one direction. If there is no line then there is no direct relationship.

System and environment administrators can create and modify all the items types shown above.

General users can create and modify the shaded items, which produce the most obvious results for SAFR. General users can have extra authorities granted by membership of a group. The extra authorities can allow a general user to create and modify other items. Granting of extra authorities is a way of implementing specialized job roles for general users in your company.

All the items in the diagram above have overview topics. Authorities are discussed in topic "**Groups overview**".

Overviews topics are elsewhere in this PDF - see the table of contents.

50 General users can choose the metadata to see

General users get access to environments and metadata by the group selected during login, and can potentially become an environment administrator. This contrasts with system administrators who have unlimited access to every environment and every item of metadata.

This means that general users can choose the metadata to see on screens in the workbench:

- EITHER see ALL metadata items, and items are grey if the user has no rights to that item
- OR see ONLY the metadata items where the user has rights.

For more about this, see topic "**What metadata do I want to see?**" To find this topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

60 Permissions to access metadata

Different users of the SAFR Workbench have different permissions to login to environments and access metadata.

For details of how these permissions work, see topics "**Groups overview**" and "**Groups - advanced overview**". Those topics are elsewhere in this PDF - see the table of contents.

70 Reports on metadata

All users can access these reports. Use one of these topics:

- FAQ topic "**How do I generate an Environment Security Report?**
- FAQ topic "**How do I generate a Logical Record Report?**
- FAQ topic "**How do I generate a Lookup Path Report?**
- FAQ topic "**How do I generate a View Properties Report?**
- FAQ topic "**How do I generate a View Column Report?**
- FAQ topic "**How do I generate a View Column PIC Report?** (this gives the COBOL PIC clause appropriate for each column in the view)
- Task "**Generating reports on metadata**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

80 Searching lists of metadata

In some places in the SAFR Workbench, there are lists of metadata. In some cases, these lists can be searched to assist working with a long list of items.

For instructions on how to search these lists, see topic "**Searching lists of metadata**".

To find this topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

90 How do I delete metadata?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

95 More information on metadata

See topic "**Metadata - advanced overview**". This topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Metadata - advanced overview

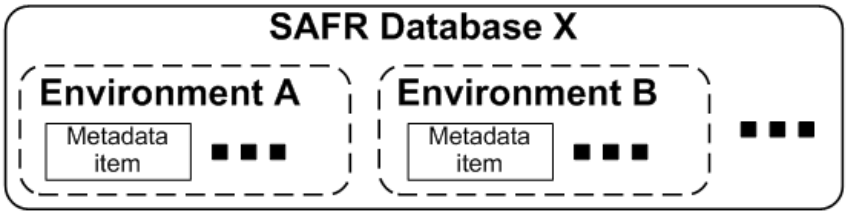
01 Summary of this topic

The sections in this topic are as follows:

10 Knowledge you need first

This topic assumes you are familiar with the topic "**Metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

20 Metadata, environments and databases



All work in the SAFR Workbench is saved in a SAFR **database**.

A SAFR **database** can contain many **environments**. The current database and environment are always shown at the bottom of the screen in the SAFR Workbench. Each individual metadata item is stored in a particular environment in a particular database.

30 Copy metadata to another environment or database

The following are the available methods to copy metadata to another environment or database:

- **Copy using a combination of export and import**.
  This copy uses the two steps: (1) create the export XML file(s) from the source environment, and (2) import the XML file(s) into the target environment. The XML files are a "backup" of what was copied and can be reused

ADVANTAGES: The source environment and target environment can be in different SAFR databases. Export and import steps are separate and can be done by different people at different times and at different locations. A general user can perform the export(s). Exported data can be imported to multiple target environments.

DISADVANTAGES: the same - this method takes longer because export and import are separate steps. An administrator is required for the import(s).

HOW TO PERFORM: see section “40 Copy using export and import.”

- **Copy using Migration**.

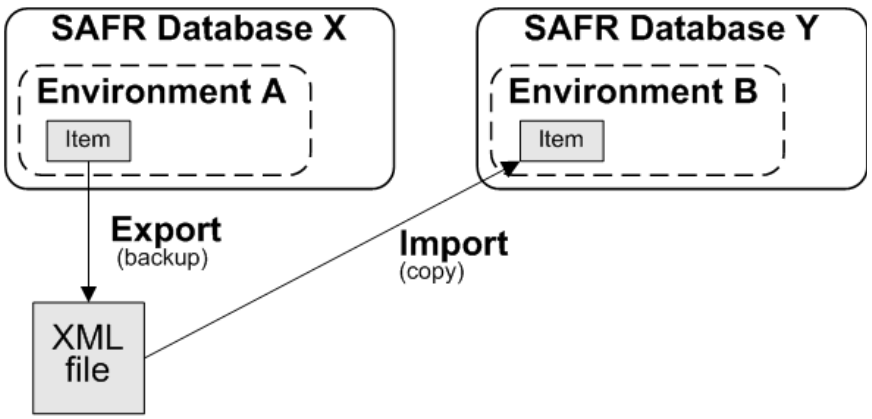
  This copy happens in one utility - the Migration Utility.

  ADVANTAGES: faster because the copy happens on one screen. A general user can perform this entire task. Another advantage is that control records and user-exit routines can be the "main" item for migration, whereas these cannot be the "main" item for a combination of export and import. Note that export and import can still copy a control record or user-exit routine, but these are copied as part of a main item that is a different type.

  DISADVANTAGES: The target environment must be in the same database as the source environment. This method requires special security authority for any general user. There is no "backup" of the data that was copied afterwards.

  HOW TO PERFORM: see section “50 Copy using Migration” on page 108.

40 Copy using export and import



The copy is performed in two steps:

1. Create backups using **Export** of metadata components from the source environment (see Environment A above).

   See "**Export metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

2. Copy metadata using **Import** into the target environment (see Environment B above). Environment B may be in a different database, or the same database.

   See "**Import metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

   IMPORTANT: You need to read help topic "**Import metadata overview**" because:

   - An import can be blocked by data problems in certain situations.

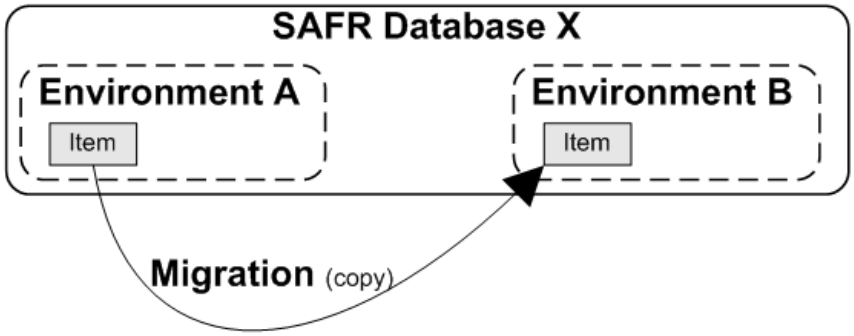
- An import can generate error messages that require a choice of action.
- An import can delete or modify data that you still require. Sometimes this occurs without a warning message.

Help topic "**Import metadata overview**" explains how to fix import problems and also to prevent import problems. It is recommended you do not attempt an import before you read that topic.

50 Copy using Migration

This is only possible when the source and target environments are in the same SAFR Database.

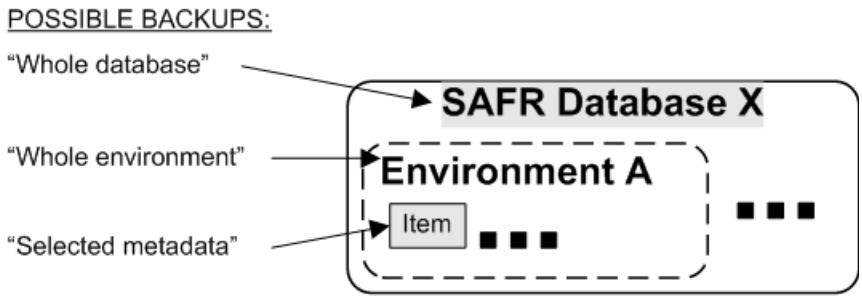
An administrator in both environments can always run this utility. A general user needs the "Migrate In" run permission in the target environment, and must have read rights to the metadata to be copied in the source environment.



For details, see "**Migrate metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

60 Methods to backup and restore metadata

The following are the available methods to backup and restore metadata:



- "**Whole database**" backup.
  This backup uses the given utilities for that database tool. That tool may be, for example, DB2. Such a backup contains **all metadata in all environments** in that Workbench database.

ADVANTAGES: all data in the database can be restored to a certain point in time. Every type of metadata is included, unlike the "whole environment" and "selected metadata" backups below which cover only five metadata types.

DISADVANTAGES: the same - a restore affects all data in the database. You may require to restore only part of the database and retain all other data.

HOW TO PERFORM: use the given utilities for that database tool (see the database support team in your company). If you require to use a "whole database" backup to restore only part of the data in the database, see section “90 Partial restore from a "whole database" backup” on page 111.

- Backup a **whole environment** of metadata into XML files.

  This backup reads a **particular environment** and creates backups in XML files of five types of metadata.

  ADVANTAGES: all metadata of five types in an environment can be restored to a certain point in time without affecting any other environments in that database.

  DISADVANTAGES: restore requires import of many XML files. This is a lengthy process.

  HOW TO PERFORM: see section “80 Backup and restore: "whole environment"” on page 110.

- Backup **selected items** of metadata in XML files.

  This backup creates backups in XML files of selected metadata items only of five types.

  ADVANTAGES: selected metadata can be restored to a certain point in time without affecting the other metadata in your database.

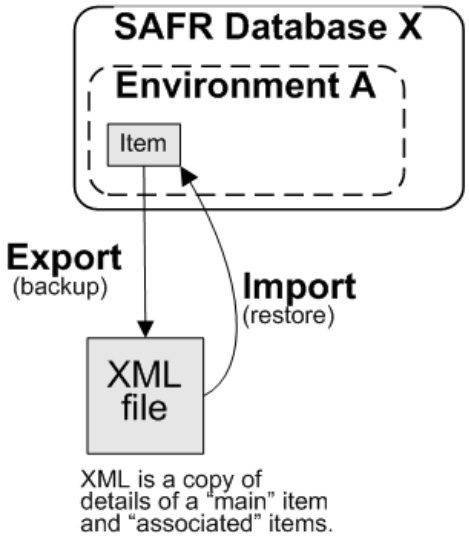
  DISADVANTAGES: the same - a restore affects only selected data in the environment. You may require to restore a whole environment or a whole database.

  HOW TO PERFORM: see the next section below.

70 Backup and restore: "selected metadata"

This applies to only these metadata types:

- Logical file,
- Logical record,
- Lookup path,
- Physical file,
- View.



Do the following:

1. Create backups using **Export**.

   See "**Export metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

2. Restore from backups using **Import**.

   See "**Import metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

   IMPORTANT: You need to read help topic "**Import metadata overview**" because:

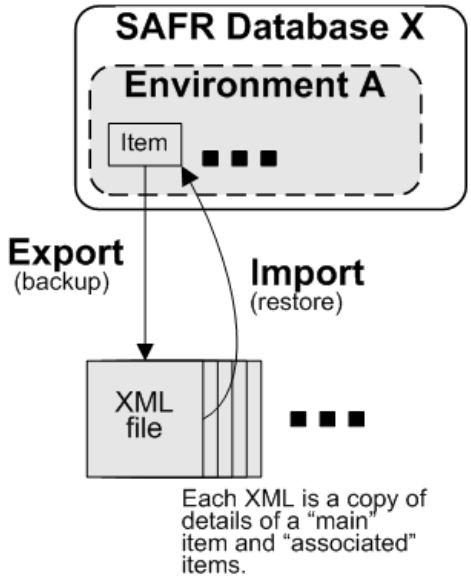
   - An import can be blocked by data problems in certain situations.
   - An import can generate error messages that require a choice of action.
   - An import can delete or modify data that you still require. Sometimes this occurs without a warning message.

   Help topic "**Import metadata overview**" explains how to fix import problems and also to prevent import problems. It is recommended you do not attempt an import before you read that topic.

80 Backup and restore: "whole environment"

This includes only these metadata types in an environment:

- Logical file,
- Logical record,
- Lookup path,
- Physical file,
- View.



Do the following:

1. Create backups using **Export**.

   See help topic "**Export metadata overview**" for a section "**Backup an entire environment**". That help topic is elsewhere in this PDF - see the table of contents.

2. Restore from backups using **Import**.

   For more details, see "**Import metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

   IMPORTANT: You need to read help topic "**Import metadata overview**" because:

   - An import can be blocked by data problems in certain situations.
   - An import can generate error messages that require a choice of action.
   - An import can delete or modify data that you still require. Sometimes this occurs without a warning message.

   Help topic "**Import metadata overview**" explains how to fix import problems and also to prevent import problems. It is recommended you do not attempt an import before you read that topic.

90 Partial restore from a "whole database" backup

If you require a restore of only part of the metadata in a "whole database" backup, do the following:

1. Recover the "whole database" backup to an empty second database such as a new test database.
2. Copy only the required metadata from this second database to the target database using section “30 Copy metadata to another environment or database” on page 106.

95 Environment Checker Report

This report shows all environments for a particular a metadata item name. This is useful when planning to update or delete a metadata item, because the same action may be necessary for the same item name in other environments.

All users can generate this report, using one of these topics:

- FAQ topic "**How do I generate an Environment Checker Report?**"
- Task "**Finding a metadata item name in all environments**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Migrate metadata overview

01 Summary of this topic

This topic describes how to use the Migration Utility to copy some metadata from a source environment to a target environment in the same SAFR Database.

Read this topic before running the Migration Utility.

Below is a summary of this topic.

1. For an introduction see:
   - "02 Knowledge you need first"
   - "10 Introduction to migration" on page 113
2. For the basics see:
   - "20 Main and Related items" on page 113
   - "30 What data is migrated?" on page 114
   - "40 Minimum migration" on page 114
   - "45 Maximum migration" on page 116
   - "50 Critical issues for Migration" on page 119
   - "60 Items to backup before a migration" on page 119
3. To perform see:
   - "70 Prepare for migration" on page 120
   - "75 Run the migration" on page 122
4. Some useful extra sections are:
   - "80 Find a metadata name in an environment" on page 122
   - "85 Find an ID number in an environment" on page 123
   - "90 Migration history in log file" on page 123
   - "100 Need more on this page?" on page 124

02 Knowledge you need first

This topic assumes you are familiar with these topics:

- "**Metadata overview**",
- "**Metadata - advanced overview**".

- "**WE Security overview**".

These topics are elsewhere in this PDF - see the table of contents.

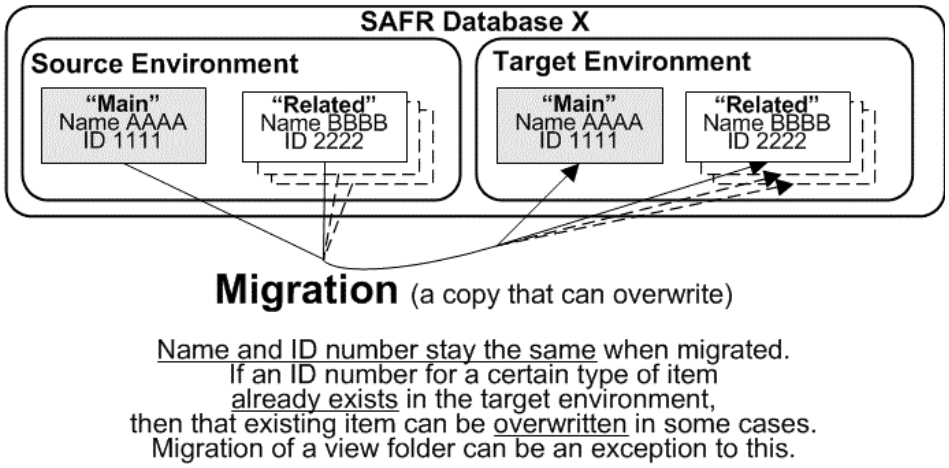
10 Introduction to migration

The Migration Utility provides a method to copy some metadata from a source environment to a target environment. Both environments must be in the same SAFR Database.

You select a "main" item to be migrated. There are some "related items" for that main item. Details of what data is migrated are shown in sections below.

What is important in this introduction is that the name and ID number stay the same for the main and related items migrated to the target environment. There can be an exception when the main item is a view folder.

Note that if an ID number for a certain type of item already exists in the target environment, then that existing item can be overwritten in some cases.



20 Main and Related items

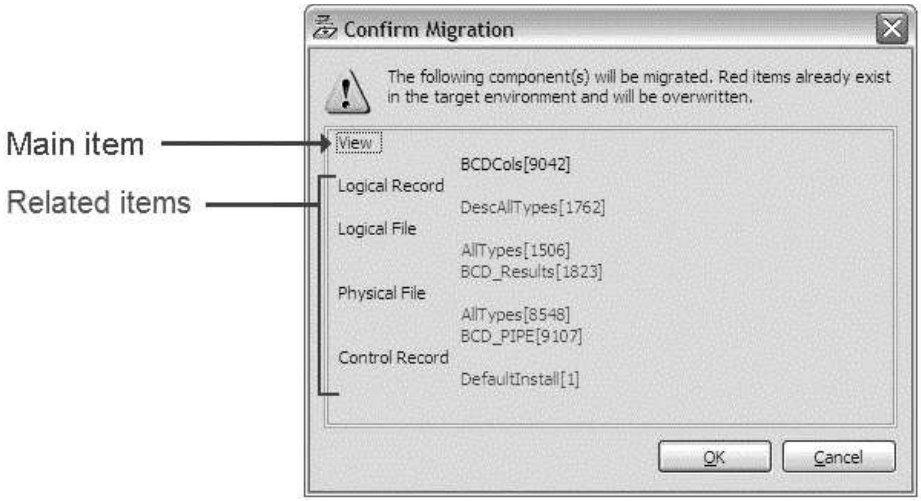
The table below shows the **possible** main and related items:

"Main" item	"Related" items
Control record	(none)
Global field	(none)
Logical file	Physical file, user-exit routine
Logical record	Logical file, physical file, user-exit routine
Lookup path	Logical record, logical file, physical file, user-exit routine
Physical file	User-exit routine
User-exit routine	(none)

"Main" item	"Related" items
View	Control record, lookup path, logical file, logical record, physical file, user-exit routine. Note that some of these items (such as lookup path and user-exit routine) may be referenced only in logic text. For an active view, related items that are referenced in logic text are also migrated.
View folder	View and all the related items of a view.

These possible related items are based on database relationships. For a particular main item, there may be none or one or multiple related items of the types given above.

The main and related items are displayed on the "Confirm Migration" screen during a migration. An example is below:



30 What data is migrated?

Data	Migrated?
Main item	Always.
Links to related items	Always.
Related items	Only for a maximum migration

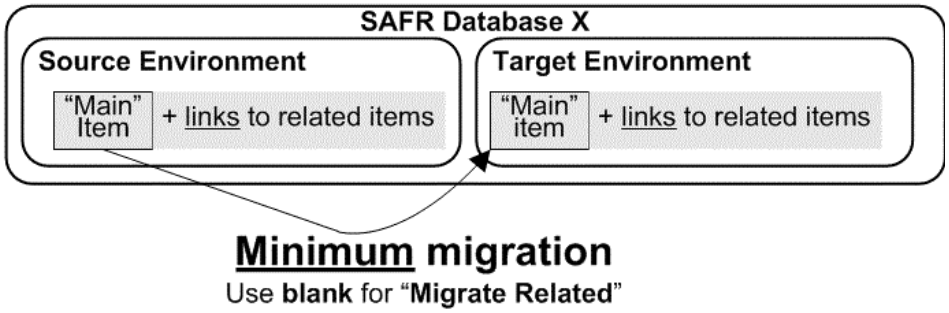
For more on this, see sections

- "40 Minimum migration"
- "45 Maximum migration" on page 116

40 Minimum migration

To perform a minimum migration, **remove the tick** in the box "**Migrate Related**" on the Migration Utility screen.

A minimum migration is shown in the diagram below:



A minimum migration performs the following:

- Copies the main item. This item is either created or overwritten in the target environment using the sane ID number.

  This copy does not occur when the main item is a view folder and you select a value for "Target View Folder".
- Copies the links from the main item to the related items, which replace any existing links in the target environment.

  No links are copied when the main item is a view folder, because the links are a field in the view itself. In this case, the links are only copied when the views and related items are migrated, which happens only for a maximum migration of that view folder. For a view folder, always use a maximum migration.

A minimum migration has strict conditions, as follows:

- **If the main item name already exists** in the target environment with a different ID number then the **migration fails**.

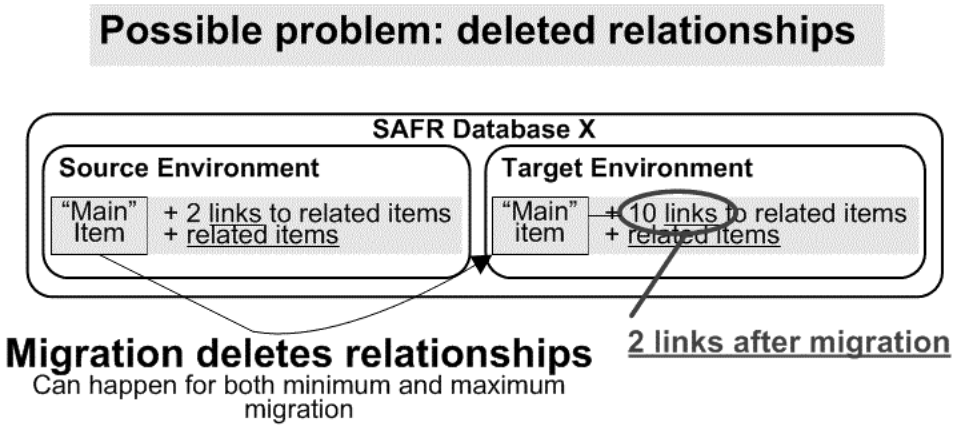
  This condition is omitted when the main item is a view folder and you select a value for "Target View Folder".
- **If any related item ID number does NOT exist** in the target environment then the **migration fails**.

  It is OK for a related item in the target environment to have the correct ID number and a different name.

  This condition is omitted when the main item is a control record, global field or user-exit routine, because there are no related items.

Either condition can block the migration and result in an error message. All error messages and solutions are covered in topic "**Migration Utility messages**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

Be aware that a migration may delete relationships in the target environment. The relationships are the links between a main item and the related items.



This occurs when the number of links in the source environment is less than in the target environment. There may be no warning message for this. This may be appropriate for your data or this may be a problem.

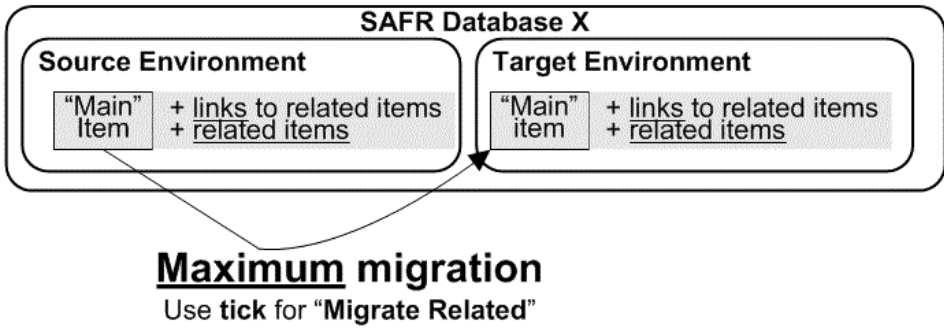
To prevent this possible problem, you must be aware of the data in your migration. A backup is recommended before a migration to ensure it is possible to recover data after a migration. See section "60 Items to backup before a migration" on page 119.

To show the progress of the migration, text displays below the Navigator:

45 Maximum migration

To perform a maximum migration, **ensure a tick** in the box "**Migrate Related**" on the Migration Utility screen.

A maximum migration is shown in the diagram below:



A maximum migration performs the following:

- Copies the main item. This is either created or overwritten in the target environment using the same ID number.

  No copy of the main item occurs when the main item is a view folder and you select a value for "Target View Folder".
- Copies the related items themselves. Each item is either created or overwritten in the target environment using the existing ID number for that item.

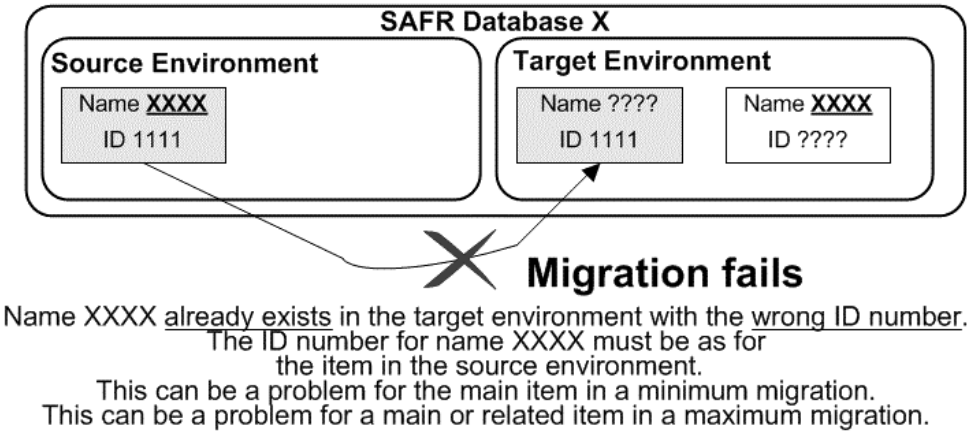
  If the main item is a view then some related items (such as lookup path and user-exit routine) may be referenced only in logic text. For an active view, related items that are referenced in logic text are also migrated.

  This means that if the main item is a view, then there are potentially many related items migrated. If the main item is a view folder, then a maximum migration means all views inside the view folder are migrated.
- Copies links from the main item to the related items, replacing all existing links in the target environment.

A maximum migration has one strict condition: **if the name of a main or related item already exists** in the target environment with a different ID number then the **migration fails**.

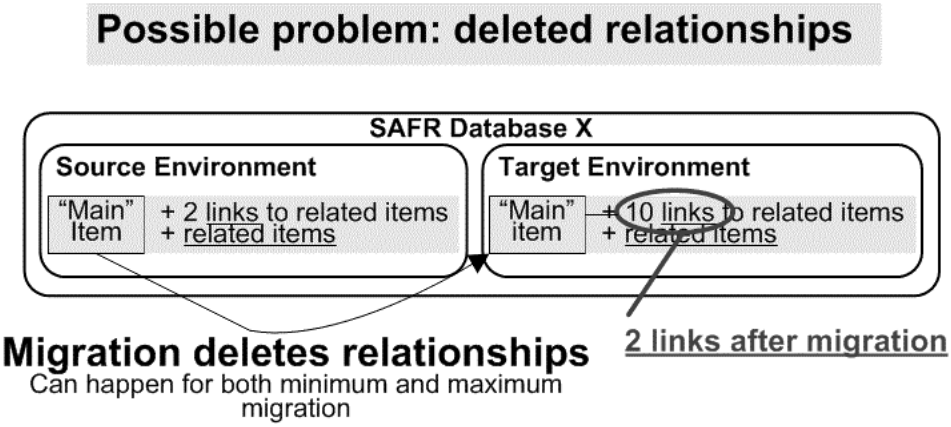
This condition does not apply to the main item when the main item is a view folder and you select a value for "Target View Folder".

When this condition applies the diagram below shows the situation:



If the target environment is empty, then a maximum migration always works, because the target environment never meets the condition above.

Be aware that a migration may delete relationships in the target environment. The relationships are the links between a main item and the related items.



This occurs when the number of links in the source environment is less than in the target environment. There may be no warning message for this. This may be appropriate for your data or this may be a problem.

To prevent this possible problem, you must be aware of the data in your migration. A backup is recommended before a migration to ensure it is possible to recover data after a migration. See section "60 Items to backup before a migration" on page 119.

To show the progress of the migration, text displays below the Navigator:

50 Critical issues for Migration

The critical issues are:

- The choice of minimum or a maximum migration.
- The name and ID numbers of the main and related items in the source environment.
- Whether the target environment has items with the same name and/or ID number as those in the previous point. This affects the following conditions:
  - For a minimum migration, **if the main item name already exists** in the target environment with a different ID number then the **migration fails.**

    This condition is omitted when the main item is a view folder and you select a value for "Target View Folder".
  - For a minimum migration, **if any related item ID number does NOT exist** in the target environment then the **migration fails.**

    It is OK for a related item in the target environment to have the correct ID number and a different name.

    This condition is omitted when the main item is a control record, global field or user-exit routine, because there are no related items.
  - For a maximum migration, **if the name of a main or related item already exists** in the target environment with a different ID number then the **migration fails.**

    This condition does not apply to the main item when the main item is a view folder and you select a value for "Target View Folder".
- The number of links between the main item and the related items, both in the source and target environments.

  If there are less links in the source environment, then a migration deletes some links in the target environment. There may be no warning message about this. You must be aware of the data changes. A backup is recommended before a migration to ensure it is possible to recover data after a migration. See section “60 Items to backup before a migration.”
- If a migration may create an unacceptable problem in the target environment, then the migration does not proceed and an error message appears. You must fix the problem in the error message in order to retry the migration.

  All error messages and solutions are covered in topic "**Migration Utility messages**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

60 Items to backup before a migration

When you perform a migration, the main and related items are listed on a screen message called "**Confirm Migration**". An example is below:

On the Confirm Migration screen, look for red items and note the item type, name and ID number. These are the ID numbers (for that type of item) that a migrate overwrites in the target environment.

It is recommended to backup in the target environment the **red** item(s) on this screen before proceeding with the migration.

This can be done as follows:

1. When you see the Confirm Migration screen, **note the red item types, names and ID numbers, then click Cancel.**
2. Backup these items in the target environment. The backup allows restore of any data that is overwritten by the migration.

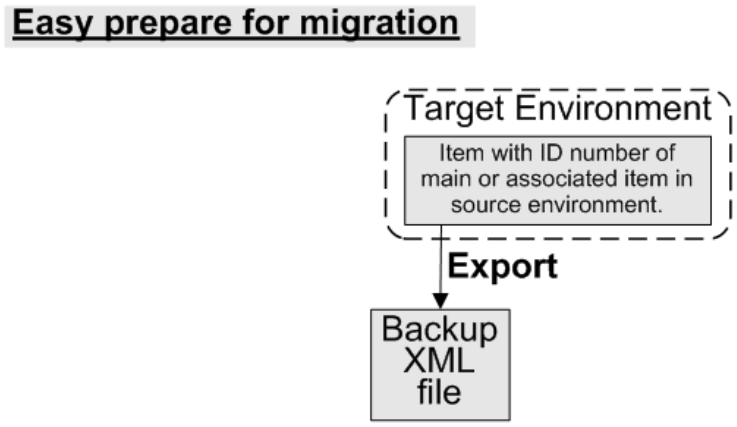
   A backup can be done using an export - see help topic "**Export metadata overview**". That topic is elsewhere in this PDF - see the table of contents.
3. Repeat the migration, and click OK on the Confirm Migration screen.

70 Prepare for migration

There is an easy way and a hard way to prepare for a migration.

The **easy** way is as follows:

1. Backup in the target environment the red items shown on the Confirm Migration screen. For details, see section “60 Items to backup before a migration” on page 119.



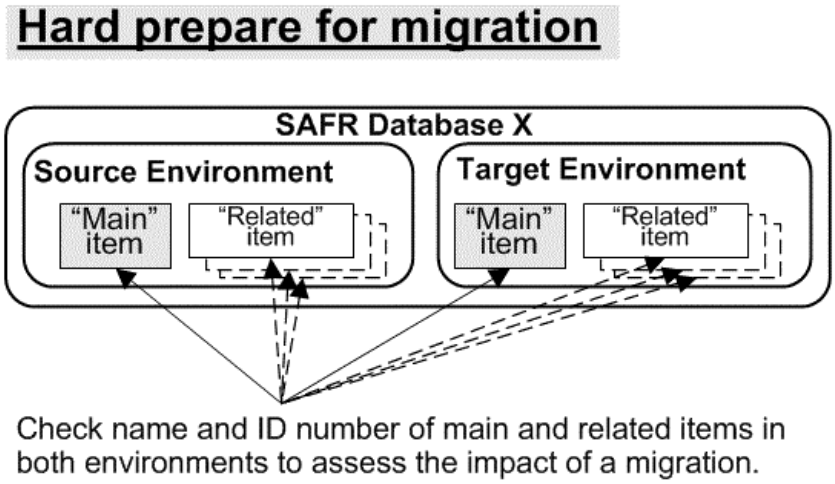
This easy preparation gets even easier in the following situation:

- Make the target environment a new (empty) environment,
- Use a maximum migration.

In the above situation, there is no backup required because the target environment is empty.

2. Proceed to "75 Run the migration" on page 122.

The **hard** way to prepare for a migration is shown below:



1. Ensure you have read all previous sections in this topic.
2. List the names and ID numbers of the main item and related items in the source environment. This is done by finding the main item. The details of the related items are usually listed on one of the screens for the main item. For example, the screen for a logical file lists the physical files related to that logical file. To find the main item, use one of these sections:
   - "80 Find a metadata name in an environment" on page 122
   - "85 Find an ID number in an environment" on page 123

3. Check if the main item name already exists in the target environment, using the section “80 Find a metadata name in an environment.”

   If the ID number of the main item already exists in the target environment, this item may be overwritten. If the name of the main item exists in an item with a different ID number then the migration will fail. If the main item ID number exists but the name does not exist in the target environment, decide if you want the record with the correct ID number overwritten. If so, continue with the next step below. If not, then do not migrate this main item.
4. Check if all the related items already exist in the target environment. The name and ID number must be correct in each case. This is mandatory for a minimum migration. To find each related item, use one of these sections:
   - “80 Find a metadata name in an environment”
   - “85 Find an ID number in an environment” on page 123

   If a related item does not exist in the target environment then do the following:
   - EITHER consider using a maximum migration,
   - OR consider performing a migration of a missing related item from the source environment to the target environment first. When this is complete for all related items, the original migration can proceed.
5. If “main” item exists in the target environment, check for any links to related items in the target environment that are not present in the source environment. If so, decide if you are comfortable losing those links after the migration. If you are comfortable, continue to the next step below. If you are not comfortable, do not migrate this main item.
6. Preparation is complete - go to section “75 Run the migration.”

75 Run the migration

1. Ensure you have performed section “70 Prepare for migration” on page 120.
2. It is recommended to backup in the target environment the red items shown on the Confirm Migration screen. For details, see section “60 Items to backup before a migration” on page 119.
3. Perform "**Migrating metadata**" in the Administration Guide. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

   Error messages and solutions are covered in topic "**Migration Utility messages**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".
4. The WE log file contains data about migrations you perform in this session. See section “90 Migration history in log file” on page 123.

80 Find a metadata name in an environment

1. Log into the Workbench using the relevant environment.
2. In the Navigator, **click on the relevant metadata type**.
3. The Metadata List displays a list of existing metadata items of that type. **Click on the heading "Name"** to sort the list into descending order for name. If you click this heading multiple times, the list varies between descending and ascending order for name.
4. Look for the particular name in the Metadata List. If the name is listed, then the name exists in that environment. Double click that name in the list to view the details of that metadata item.

   If the name is grey, then you do not have enough security authority to view that metadata item.

General users should be aware of the topic "**What metadata do I want to see?**" To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

For more details of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

85 Find an ID number in an environment

1. Log into the Workbench using the relevant environment.
2. In the Navigator, **click on the relevant metadata type**.
3. The Metadata List displays a list of existing metadata items of that type. **Click on the heading "Id"** to sort the list into descending order for ID number. If you click this heading multiple times, the list varies between ascending and descending order for ID number.
4. Look for the particular ID number in the Metadata List. If the ID number is listed, then the ID number exists in that environment. Double click that ID number in the list to view the details of that metadata item.

   If the name is grey, then you do not have enough security authority to view that metadata item.

   General users should be aware of the topic "**What metadata do I want to see?**" To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

   For more details of security, see topic "**WE Security overview**". That topic is elsewhere in this PDF - see the table of contents.

90 Migration history in log file

For an introduction to WE log files, see topic "**WE log file overview**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

The WE log file records data about any migrations you perform in this session. An example extract from a WE log file is below:

```
20/09/2013 11:42:05 AM com.ibm.safr.we.model.utilities.Migration showParams
INFO:
>>>>> MIGRATION PARAMETERS  .............

Source Environment = Red[110]
Target Environment = Green[190]
Component Type     = Logical File
Component          = PRODUCT[957]
Target View Folder = [not specified]
Migrate Related    = yes
=======================================
20/09/2013 11:42:11 AM com.ibm.safr.we.model.utilities.Migration doValidate
INFO: Validate PRODUCT[6382]
20/09/2013 11:42:12 AM com.ibm.safr.we.internal.data.dao.DB2PhysicalFileDAO
getDuplicatePhysicalFile
INFO: No duplicate Physical File in Env 190 with name : PRODUCT
20/09/2013 11:42:12 AM com.ibm.safr.we.model.utilities.Migration doValidate
INFO: Validate PRODUCT[957]
20/09/2013 11:42:12 AM com.ibm.safr.we.internal.data.dao.DB2LogicalFileDAO
getDuplicateLogicalFile
INFO: No duplicate Logical File in Env 190 with name : PRODUCT
20/09/2013 11:42:13 AM com.ibm.safr.we.model.utilities.Migration doStore
INFO: Component Store TimingStore: com.ibm.safr.we.model.LogicalFile 0 secondsStore:
com.ibm.safr.we.model.PhysicalFile 0 seconds
20/09/2013 11:42:14 AM com.ibm.safr.we.TimingMap report
INFO: DAO Method timingsDB2PhysicalFileDAO.updatePhysicalFile 0 seconds
20/09/2013 11:42:14 AM com.ibm.safr.we.model.SAFRModelCount report
INFO: Model countscom.ibm.safr.we.model.associations.FileAssociation: 1 instances
com.ibm.safr.we.model.LogicalFile: 1 instancescom.ibm.safr.we.model.PhysicalFile: 1
instances
20/09/2013 11:42:14 AM com.ibm.safr.we.model.utilities.Migration$MessageRecorder
writeMessages
WARNING:
>>>>> MIGRATION WARNINGS ..............

Confirm Migration
The following component(s) will be migrated. Red items marked with '*' exist in the
target environment and will be overwritten.
Logical File    PRODUCT[957]*Physical File      PRODUCT[6382]*
=========================================
20/09/2013 11:42:14 AM com.ibm.safr.we.model.utilities.Migration$MessageRecorder
writeMessages
INFO:
>>>>> MIGRATION COMPLETED.
```

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Parallelism overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is parallelism?

One of the biggest advantages of SAFR is the speed of producing results. Parallelism means allowing SAFR to run multiple processes in parallel, so that the overall work is done faster.

There are four methods to create parallelism in SAFR:

1. **Physical File Partitions.** Partition a logical file into many physical files. With careful implementation, each physical file can be processed in parallel with the others.
2. **DB2 Key Range Partitions.** This method accesses DB2 Tables which have been partitioned by Key Ranges using SQL. With careful implementation, this speeds access to DB2.
3. **DB2 via VSAM Access.** This method accesses the VSAM partitions that underly DB2. This allows the fastest sequential access to data in DB2.
4. **Pipes** allow multiple views to run in parallel.

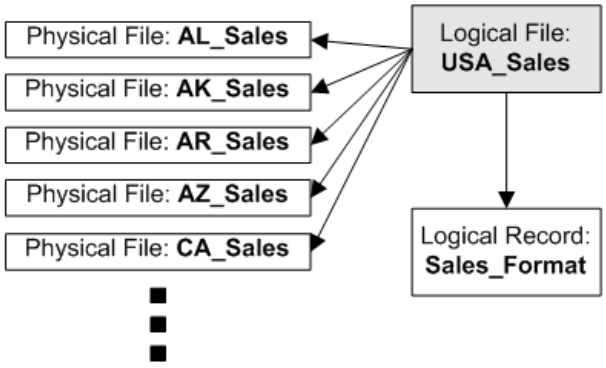
Each of these methods is discussed below.

20 Physical File Partitions (1): choose a field

A logical file may consist of a single physical file. To make SAFR run faster, partition the physical file into many physical files, based on a field that you choose.

A good example is to split the **USA_Sales** physical file into a physical sales file for each state in the USA. Here the field for the partitions is the state of each sales transaction. There is still one logical USA_Sales file and this now consists of 50 physical files. SAFR can run a separate process for each physical file, so the USA_Sales logical file can be processed by 50 processes in parallel, instead of by one process working on the entire logical file.

Notice that all the partitions are still needed to process all the sales. For field partitions, normally all partitions are included in processing.

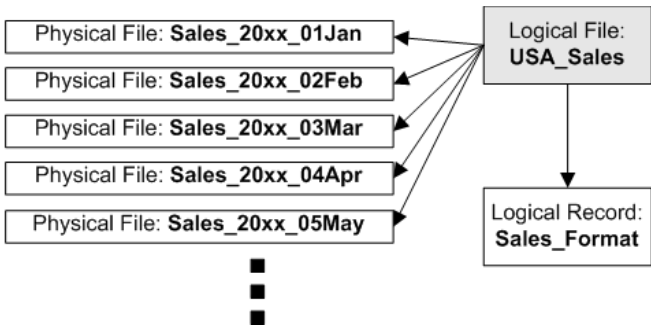


Partitions needs to be planned early in the design for implementing SAFR.

30 Physical File Partitions (2): time periods

Another way to partition a logical file is to create physical files for time periods.

An example is to split the USA_Sales physical file into three physical files, as shown below:

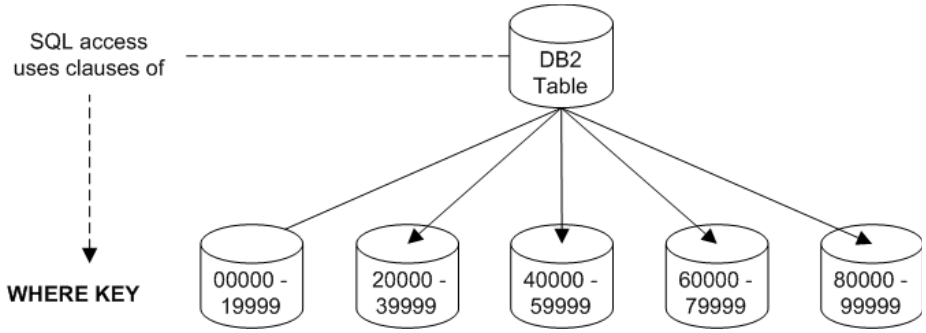


The advantages of this partition are that SAFR only needs to read the relevant time partitions for a report, and can ignore the rest.

Partitions needs to be planned early in the design for implementing SAFR. The time partitions can be combined with fields partitions - for example, each state can have separate time partitions.

40 DB2 Key Range Partitions

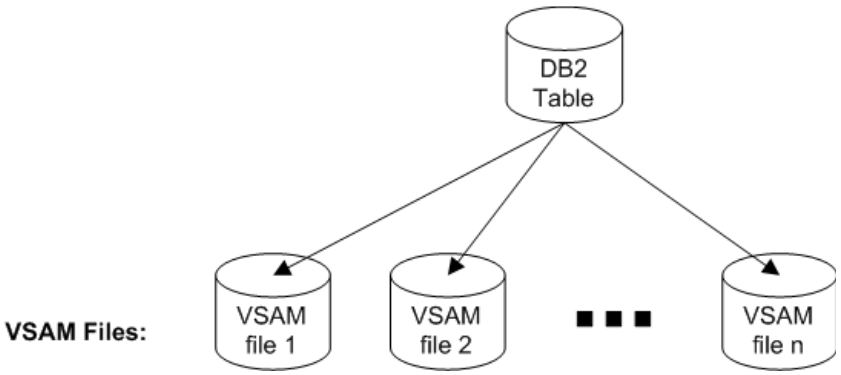
A DB2 table can have different key ranges stored in different partitions. The Physical File (in the SAFR Workbench) for the DB2 table must have **DB2 via SQL** and the SQL text must have **WHERE** clauses for the different key ranges. Each of the different key ranges can be processed in parallel by SAFR, which is faster than processing the entire table as a single file.



This method requires careful work to ensure that the file contain strictly only the key ranges specified, and all the possible keys are covered by the ranges.

50 DB2 via VSAM access

Any DB2 table has underlying VSAM partitions. SAFR can process the underlying VSAM partitions directly in parallel. The DB2 Physical File (in the SAFR Workbench) must be marked **DB2 via VSAM** and some changes made to the JCL that runs the Performance Engine to identify all the VSAM partitions.



This is the fastest method for SAFR to access DB2 in a sequential way.

60 Pipes

The pipe itself is a virtual file that stays in memory and is passed between views without Input/Output. Pipes are defined as a Physical File of type **Pipe** in the SAFR Workbench.

See the topic "**Pipes overview**". That topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

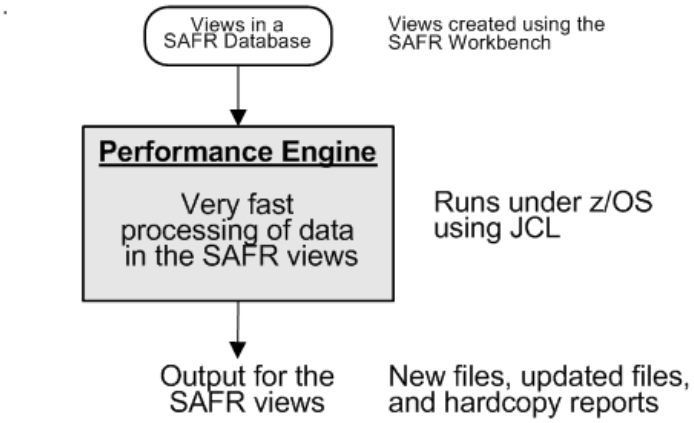
Performance Engine (PM) overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is the SAFR Performance Engine?

The Performance Engine is the part of SAFR that runs on your company mainframe systems and produces the actual results of SAFR. The Performance Engine runs under the operating system **z/OS** as a batch job using JCL.



For more on SAFR overall and the parts of SAFR, see topic "**SAFR overview - START HERE**". That topic is elsewhere in this PDF - see the table of contents.

The Performance Engine is a batch mainframe stream that produces results for the views you select.

20 Phases in the Performance Engine

The processing of the Performance Engine can be divided into phases. These can be summarized as follows:

1. **Select Phase** - where the Performance Engine reads the views you have selected.
2. **Logic Phase** - where views are translated into Logic Tables for processing.
3. **Reference Phase** - where the Logic Tables are consolidated.
4. **Extract Phase** - where input logical files are read and all columns defined (some columns may be new). Optionally, this phase can select or skip input records. Optionally, records can be written to special output files.
5. **Format Phase** - an optional phase. Optionally, this phase performs some arithmetic operations such as totals. Optionally, this phase can also select or skip records for output.

For more, see topic "**SAFR phases overview**". That topic is elsewhere in this PDF - see the table of contents.

30 Views and logic text

Control the processing of the Performance Engine by using views and logic text in the SAFR Workbench.

Views define the output of SAFR - the logical files and the columns of data and the form of the output (such as disk files or reports).

Logic text allows control over the extract and format phases of the Performance Engine. Logic text controls processing such as select or skip of records and defining column values.

For more, see topics "**Workbench (WE) overview**", "**Views overview**" and "**Logic text overview**". These topics are elsewhere in this PDF - see the table of contents.

40 Components of the Performance Engine

The Performance Engine consists of the following components:

- **Main Programs**. These are the main steps in the JCL for the Performance Engine.

  For more, see topic "**PM programs overview**". That topic is elsewhere in this PDF - see the table of contents.
- **Subprograms**. These are shared program modules installed in the Performance Engine that are normally hidden behind the other components.

  For more, see topic "**PM subprograms overview**". That topic is elsewhere in this PDF - see the table of contents.
- **User-exit routines**. Some user-exit routines are provided by IBM in the Performance Engine (in addition to any user-exit routines written by your company).

  For more, see topic "**PM user-exit routines overview**". That topic is elsewhere in this PDF - see the table of contents.
- **Utilities**. Some useful utility programs are provided by IBM in the Performance Engine.

  For more, see topic "**PM utilities overview**". That topic is elsewhere in this PDF - see the table of contents.

80 How do I optimize SAFR?

See topic "**SAFR optimization overview**" for details. This assumes you have read all other overviews and are familiar with SAFR overall.

That topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Physical files overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is a physical file?

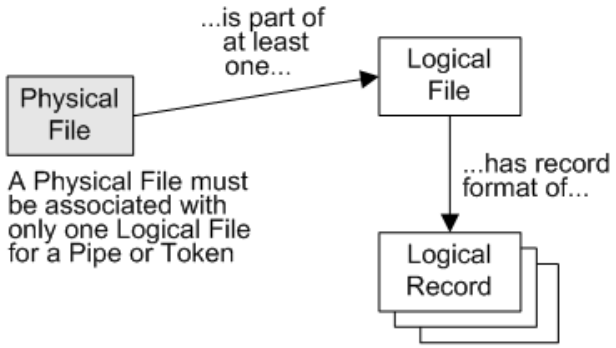
A physical file is something on your mainframe system. A physical file can be any of the following:

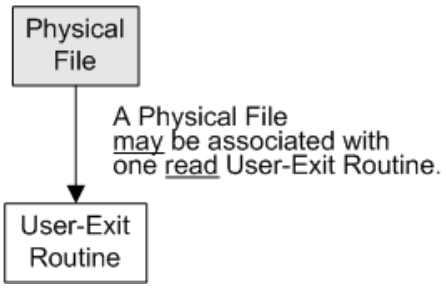
- A **disk file** (this includes DB2 tables),

- A **pipe** - see topic "**Pipes overview**",
- A **token** - see topic "**Tokens overview**",
- A **tape file**.

The two overview topics are elsewhere in this PDF - see the table of contents.

A physical file is not used directly in a view. Instead, a physical file is associated with at least one logical file. One physical file can be associated with many logical files, and one logical file can be associated with many physical files. The record format of the logical file (and any associated physical files) is given by one or more logical records. A view can use a logical file directly, but not a physical file.





If the physical file is a pipe or token, then the physical file can only be associated with one logical file, as indicated above.

Also shown above, a physical file can be associated with one read user-exit routine.

For more on all of the above, see these topics:

- "**Logical files overview**",
- "**Logical records overview**",
- "**Pipes overview**"
- "**Tokens overview**"
- "**User-Exit Routines overview**"
- "**Views overview**",
- "**Views - advanced overview**".

These topics are elsewhere in this PDF - see the table of contents.

20 How do I use a physical file?

A physical file is used when the associated logical file is used in a view or a lookup path. To ensure a physical file is used, ensure the physical file is associated with at least one logical file.

The association between the physical file and one or more logical files is controlled by the logical file. See these topics:

- "**Creating logical files**",
- "**Modifying logical files**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

For access rights to logical files, see the heading "**How do I create or modify a physical file?**" below.

A physical file is indirectly used in a view, when the view includes the logical file associated with the physical file. All users in that environment can use **logical files in a view**. For more see these topics:

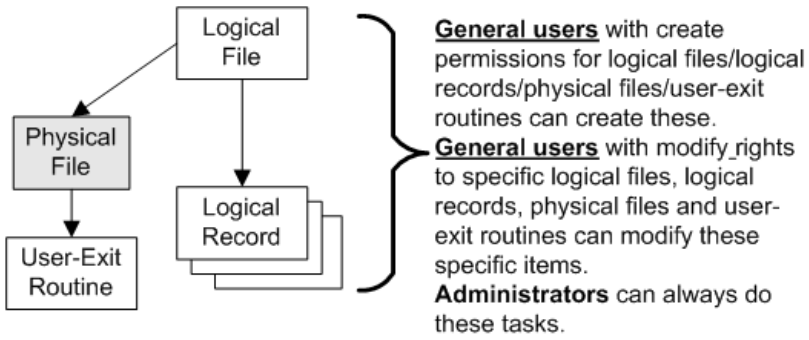
- "**Creating views**",
- "**Modifying views**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I know which physical file to use?

See your system or environment administrator for advice on physical files in your environment.

50 How do I create or modify a physical file?



System administrators and environment administrators can always **create or modify physical files** and the other items above.

General users can **create or modify physical files** if the group for login has the following authorities:

- **Create Logical File** permission in the relevant environment (if required).
- **Create Logical Record** permission in the relevant environment (if required).

- **Create Physical File** permission in the relevant environment.
- **Modify or Delete rights to this particular logical file** in that environment (if required - this allows modify of the logical file itself and links to physical files.)
- **Modify or Delete rights to the relevant logical records** in that environment (if required).
- **Modify or Delete rights to the relevant physical files** in that environment.

For more on these authorities, see topics "**Groups overview**" , "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

Once the group has the above authorities, general users in that group can **create or modify physical files** by using the tasks below, which are **administrator tasks:**

- "**Creating logical files**",
- "**Creating logical records**",
- "**Creating physical files**",
- "**Modifying logical files**" (if required to modify the logical file itself and the links to physical files),
- "**Modifying logical records**" (if required to modify the links between logical files and logical records),
- "**Modifying physical files**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

90 How do I delete a physical file?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Pipes overview

01 Summary of this topic

The sections in this topic are as follows:

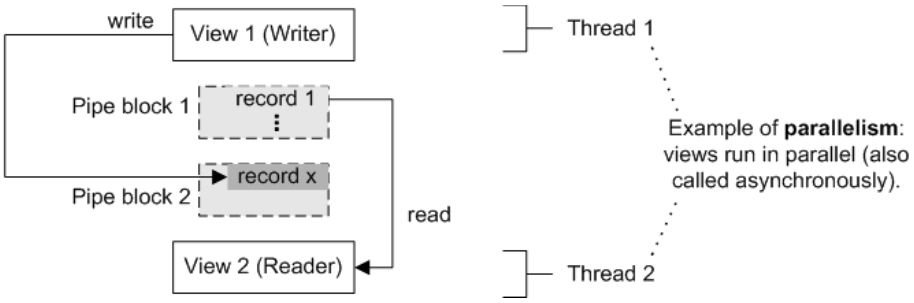
10 What is a pipe?

A pipe is a virtual file passed in memory between two or more views. A pipe is defined as a Physical File of type **Pipe** in the SAFR Workbench.

Pipes save unnecessary input and output. If View 1 outputs a disk file and View 2 reads that file, then time is wasted for output from View 1 and input to View 2. If there is a pipe placed between View 1 and View 2, then the records stay in memory, and no time is wasted. The pipe consists of blocks, which group records together.

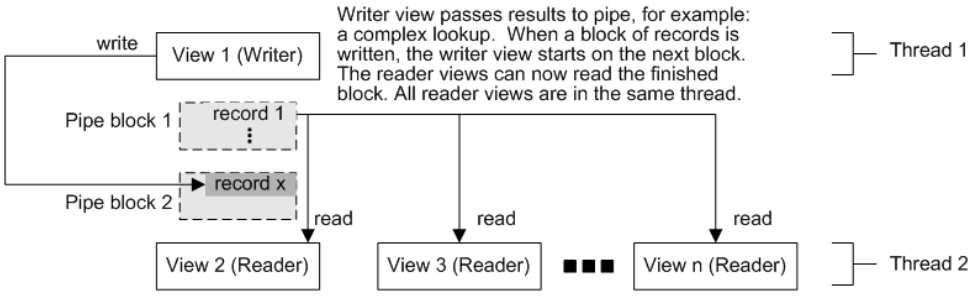
Another advantage of pipes is that View 2 runs in parallel to View 1. After View 1 has finished writing block 1 of records to the pipe, View 2 can process block 1 at the same time that View 1 is preparing block 2. This advantage is part of parallelism, which improves SAFR performance. Without this, View 2 would have to wait until all of View 1 is complete.

Another way of describing this parallelism is that View 1 is in a different thread to View 2.



Another advantage is that multiple views can read from the same pipe at the same time. All the reader views can read block 1 once the writer view has finished writing all records to block 1. Hence, if View 1 does a complex lookup and writes the results to the pipe, then all the reader views save the time to perform that lookup. All the reader views are in the same thread (shown as Thread 2 below), which is different form the thread of the writer view (shown as Thread 1 below). This is the parallelism advantage discussed above.

The parallelism advantage is the reason pipes are encouraged to improve SAFR performance.

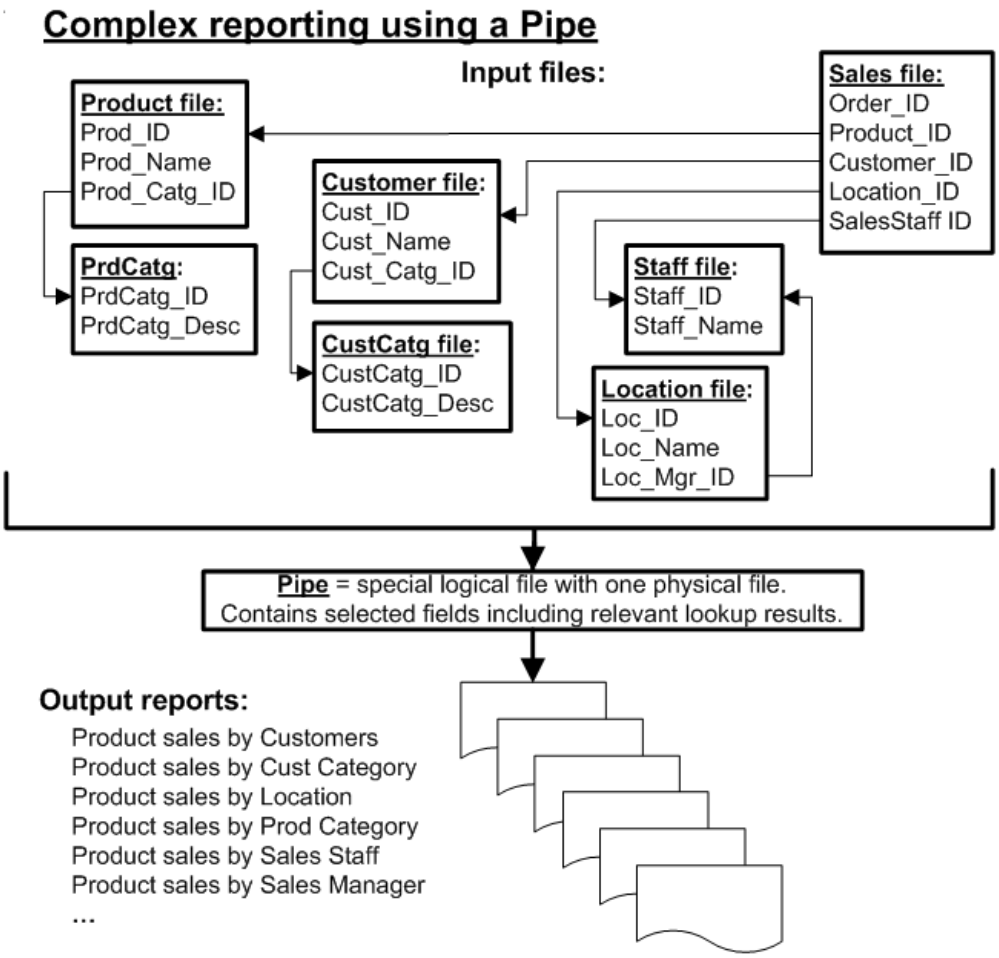


A limitation of pipes is that the only data the reader views can see is in the pipe. The reader views do not (automatically) have access to the source files of View 1. If the reader views require this access, then consider using a token - see topic "**Tokens overview**".

That topic is elsewhere in this PDF - see the table of contents.

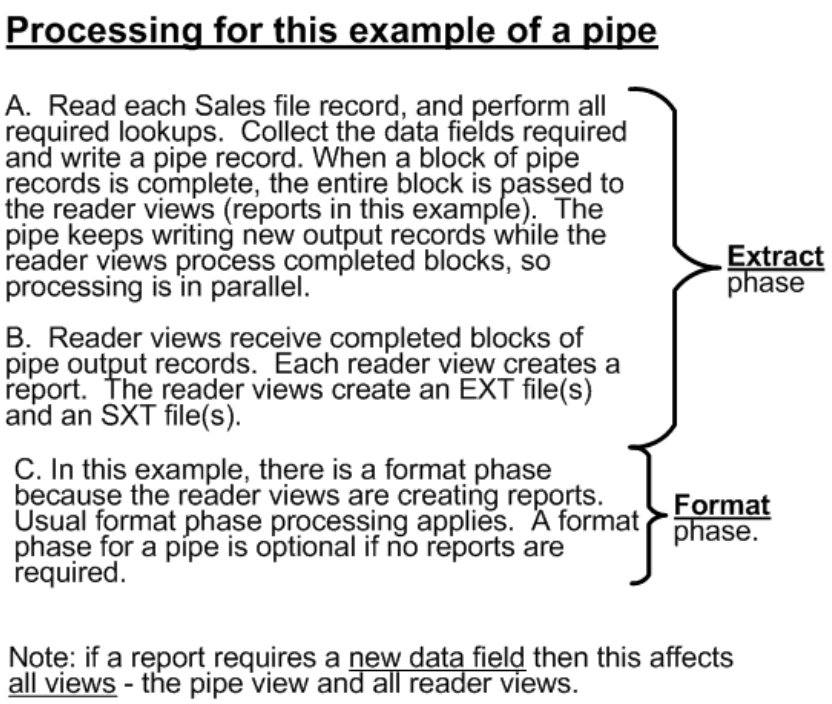
20 Example of a Pipe

SCENARIO: one sales file produces at least six reports, and the reports require data from other files. The Sales file has only ID numbers for products and customers for example, and the actual product names, product category names, customer names and customer category names are required.



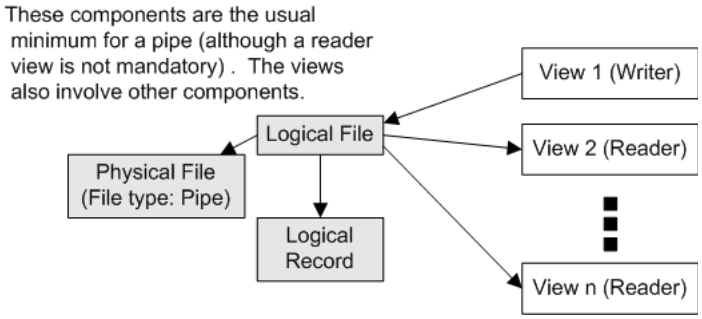
The solution is a **pipe**, which is a special type of logical file that has a single physical file. The complex lookups required are performed, and relevant data is written in records to the pipe. The reports read the pipe records, and generate the reports while new pipe records are still being written.

Below are the processing steps for this complex reporting.



30 What are the components of a pipe?

A pipe is configured by a physical file of file type **Pipe**. There is one logical file and at least one logical record associated with that. There must be one writer view. There may be zero or one or more reader view(s) for that logical file (pipe).



Pipes should be tested initially by using a disk file. Check the disk file after the views have run to verify the writer has produced the correct data. Only after the writer and reader views are tested thoroughly using a disk file should a pipe be used.

40 Strengths and weaknesses

The strengths of pipes are:

- Avoid unnecessary output and input of pipe data.
- Allows multiple reader views to run in parallel to writer view - which is an example of parallelism.

- Avoids the overhead of subroutine calls (which are necessary for common key buffers).
- Suits requirement for different logical records to be written to a common format for later processing.
- Creates an environment similar to many software applications: the writer view is a "driver" and the reader views respond to data received.
- Maintenance is easier since writer view changes are only made once.

The weakness of pipes are:

- Reader views can only see the pipe data, not any original source files used by the writer view.
- Requires that reader views are independent (asynchronous) compared to the writer view.
- Is less practical when there is a complex relationship between a group of different logical records.

Overall, pipes are encouraged when performance is critical and it is possible to use pipes. Pipes may be used in combination with Common Key Buffers.

See also topics "**Common Key Buffer overview**" and "**Tokens overview**" and "**Parallelism overview**".

These topics are elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

SAFR optimization overview

The methods to achieve optimization of SAFR are to optimize the Performance Engine (PE). These methods are

- Common Key Buffers
- Parallelism
- Pipes
- Tokens

Under "**Related Concepts**" below, click on links to overviews for these.

SAFR phases overview

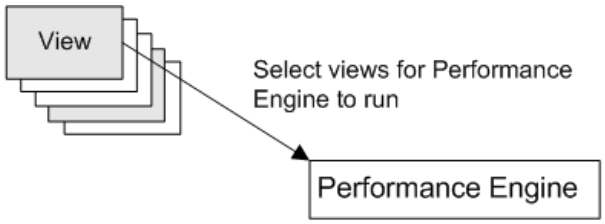
01 Summary of this topic

The sections in this topic are as follows:

02 What are SAFR phases?

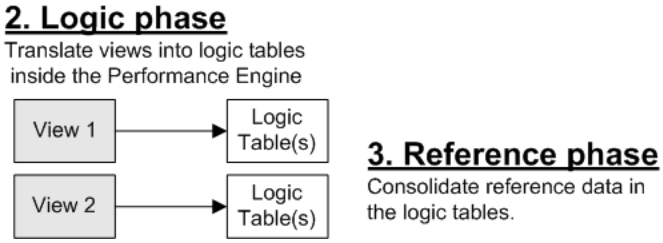
A SAFR phase is a part of the processing done in the SAFR Performance Engine. For an introduction to the SAFR Performance Engine, see topics **SAFR overview - START HERE** and **Performance Engine (PE) overview**. These topics are elsewhere in this PDF - see the table of contents.

10 Select phase



This phase selects views for this run of the Performance Engine. You can influence this phase by the views selected in the JCL to run the Performance Engine.

20 Logic phase and Reference phase

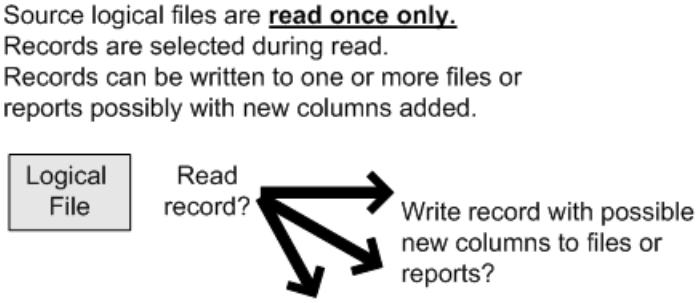


The **logic phase** translates the selected views into one or more logic tables each.

The **reference phase** consolidates the logic tables so that references to objects resolve successfully.

These two phases are internal to SAFR and users have no direct influence. Users selected the views, so this is how users indirectly affect these phases.

30 Extract phase

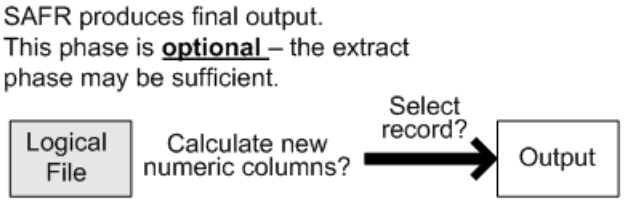


This phase involves an important feature of SAFR: source logical files are read once only, which avoids wasting time and resources on multiple reads of the same data. Users can directly influence this phase by using logic text of types:

- **Logic text 1: Extract Record Filter** – this controls which records are selected from the source logical files.
- **Logic text 2: Extract Column Assignment**- this controls new columns and which records are written to which files.

For more, see overview topics for these types of logic text. These topics are elsewhere in this PDF - see the table of contents.

40 Format phase



This phase produces the output from the SAFR views. This phase is optional, because the output from the extract phase above may be sufficient. The format phase provides more choices for output than the extract phase. Users can directly influence this phase by using logic text of types:

- **Logic text 3: Format Column Calculation** logic text - this controls numeric calculations for new columns.
- **Logic text 4: Format Record Filter** logic text – this controls which records are selected for output.

For more, see overview topics for these types of logic text. These topics are elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Tokens overview

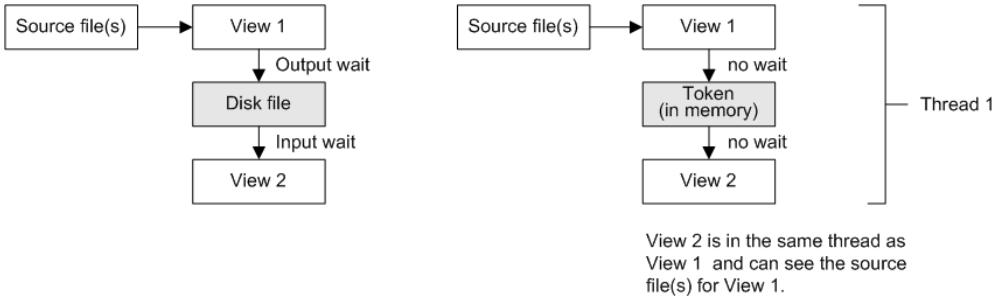
01 Summary of this topic

The sections in this topic are as follows:

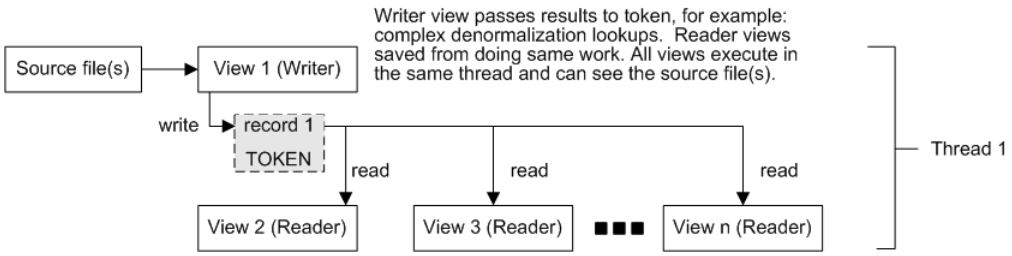
10 What is a token?

A token is a named memory area. The name is the logical file id number and the logical record id number. The name is used for communication between two or more views.

Tokens save unnecessary input and output. Token reading views are called immediately after the token data is written. If View 1 outputs a disk file and View 2 reads that file, then time is wasted for output from View 1 and input to View 2. If there is token placed between View 1 and View 2, then the records stay in memory, and so there is no time wasted.



An advantage is that multiple views can read from the same token. Token reader views are called one at a time in the sequence number of the view id number. For example, if View 1 does complex denormalization lookups and writes the results to the token, then all token reader views save the time to perform those lookups. Another advantage is that once the token data is written, the reader views get access immediately, without waiting for a block to be finished.



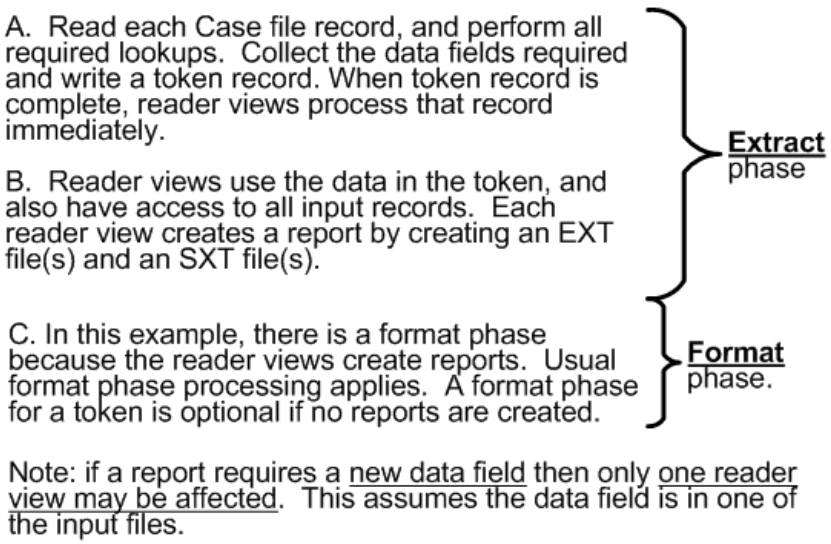
A limitation of tokens is that there is no parallelism - all views run in the same thread. If the reader views need only the data in the token logical record, then consider using a pipe - see topic "**Pipes overview**".

That topic is elsewhere in this PDF - see the table of contents.

20 Example: Token

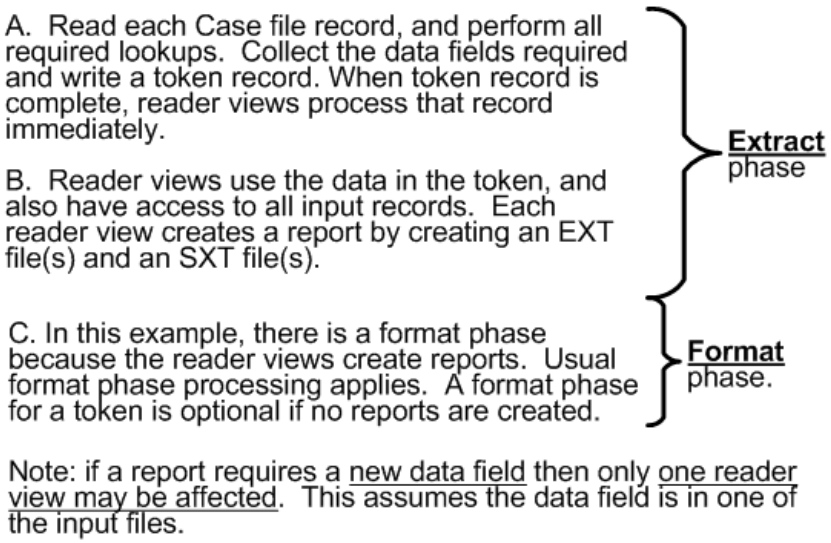
This section describes a complex reporting problem, and how a technique called a "token" can provide a solution.

SCENARIO: a hospital want many reports involving patient, diagnosis and treatment records. The data fields in the reports may change at any time in an unpredictable way, depending on the medical issues at the time. The input files and the minimum lookups are shown below.



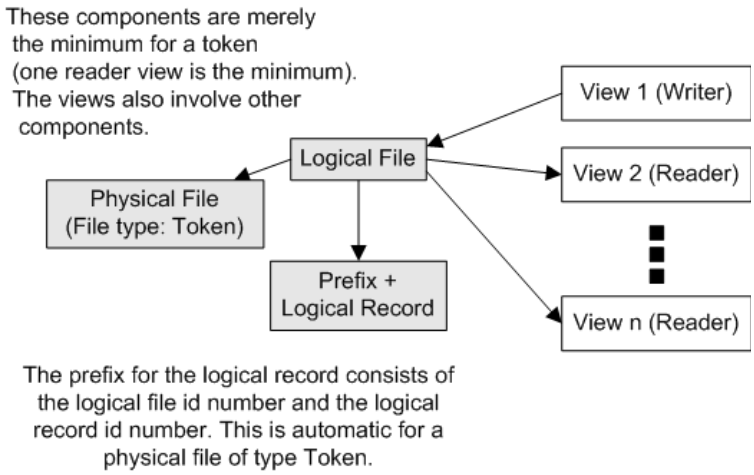
The solution is a **token**, which is a special type of logical file. The complex lookups required are performed, and relevant data is written in records to the token. Once the token record is written, the reports immediately process that record.

Below are the processing steps for this complex reporting.



30 What are the components of a token?

A token is configured by a physical file of file type **Token**. There is one logical file and at least one logical record associated with that. There must be at least two views: one as a writer view and the other(s) as reader view(s) for that logical file (token).



There can be more than one token in a thread, because the writer view may write more than one type of record. The situation of the writer view creating only one type of token record is called a **Mutually Exclusive Token**. If the writer view can write more than one type of token, this is called a **Non-mutually exclusive token**.

40 Strengths and weaknesses

The strengths of tokens are:

- Avoid unnecessary output and input of token data.
- Allows multiple token reader views to benefit from work done by the token writer view.
- Avoids the overhead of subroutine calls (which are necessary for common key buffers).
- Once token data is written, the reader views gain access immediately (one reader view at a time). There is no waiting for a block to be finished.
- Suits requirement for different logical records to be written to a common format for later processing.
- Allows all token reader views to access the same set of source file(s) as the writer view, which may need be necessary in unusual cases.
- Maintenance is easier since writer view changes are only made once.

The weakness of tokens are:

- There is no parallelism - all token reader views run in the same thread.
- Is less practical when there is a complex relationship between a group of different logical records.

Overall, tokens are best used when the token reader views need access to the same set of source files that the writer view has access to. Tokens may be used in combination with Common Key Buffers.

See also topics "**Common Key Buffer overview**", "**Pipes overview**" and "**Parallelism overview**".

These topics are elsewhere in this PDF - see the table of contents.

100 Need this page completed?

The task of completing this page has not yet been scheduled. If you wish to raise its priority, please email **AskSAFR@us.ibm.com** .

Users overview

01 Summary of this topic

The sections in this topic are as follows:

10 Three types of user for the SAFR Workbench

Produces results using SAFR:

- Receives information from the administrators on SAFR login, the environments and the components available.
- Creates and activates lookup paths and views. This produces the reports and updated data that are the goals of using SAFR in the company.

Environment Administrator

Prepares SAFR for work by general users:

- Creates and maintains all components required in a specific environment except for users and groups.
- Can modify group permissions for that environment.
- (If required) performs general user work.

System Administrator

Prepares SAFR for work by general users:

- Creates and maintains all users, groups, environments and components required by all general users.
- Can modify all group permissions for all environments. Decides environment administrator access.
- (If required) performs general user work.

The SAFR Workbench has these users:

- **General users** who focus on producing results for your company using SAFR.

  A general user specifies an environment and a group at login time. The group provides the access authorities.
- **Environment administrators** who focus on preparing an environment for work by general users.

An environment administrators specifies an environment and a group at login time. The group provides the access authorities.

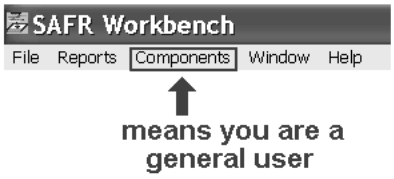
- **System administrators** who focus on preparing all environments for work by environment administrators and general users.

  A system administrator specifies an environment only at login time (and does not specify a group). A system administrator is not affected by group membership themselves. A system administrator decides group membership for all other users, which effectively decides access authorities for all general users and environment administrators.

For more on the access authorities mentioned above, see topics "**Groups overview**", "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

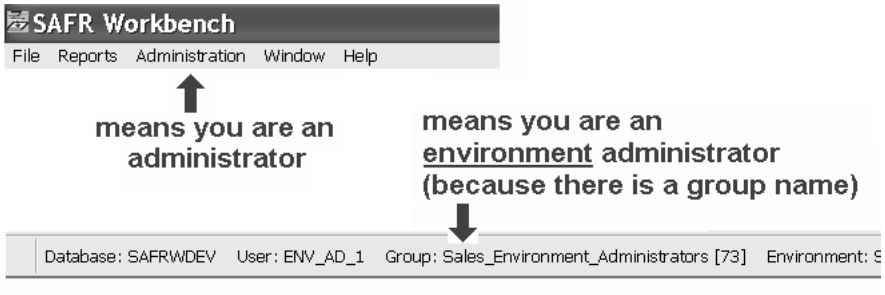
20 Am I a general user?

Only **general users** have a menu "**Components**":



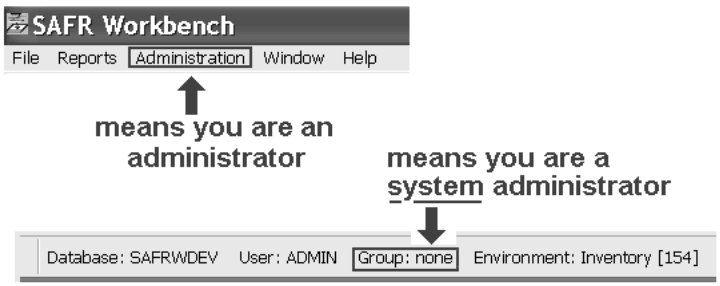
30 Am I an environment administrator?

Environment administrators always have a menu "**Administration**" and a **group name** at the bottom:



40 Am I a system administrator?

System administrators always have a menu "**Administration**" and "**Group: none**" at the bottom:



50 General user topics to read

Recommended topics to read are as follows:

- **Overviews** (all topics),
- **General users guide** (all topics).

The overview topics are elsewhere in this PDF - see the table of contents. To find the other topics in a PDF, see chapter "Cross reference of topics and PDF files".

55 How do I delete a user?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

60 Administrator topics to read

Recommended topics to read are as follows:

- **Overviews** (all topics),
- **Administrators guide** (all topics).

The overview topics are elsewhere in this PDF - see the table of contents. To find the other topics in a PDF, see chapter "Cross reference of topics and PDF files".

90 The SAFR User ID and your mainframe systems

A SAFR User ID gives you access to the SAFR Workbench. You login to the SAFR Workbench using this User ID.

For the SAFR User ID to work you must **already have a mainframe User ID** to access the mainframe computer systems in your company.

The SAFR User ID does not have to be the same as your mainframe User ID. What is important is that you have both.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

User-exit routines overview

01 Summary of this topic

The sections in this topic are as follows:

10 Short name is "exit"

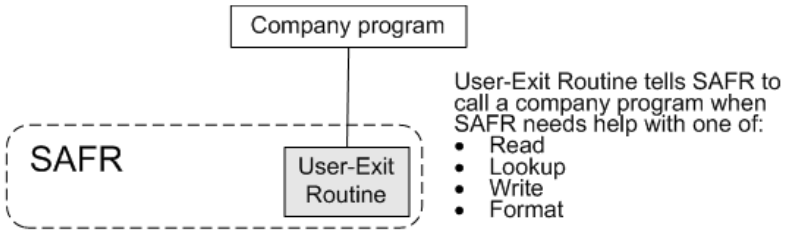
The term "**exit**" is simply the short name for "**user exit routine**". Both terms are used interchangeably in this documentation.

The term "**user-exit routine**" is commonly used in WE. The term "**exit**" is more commonly used in PE.

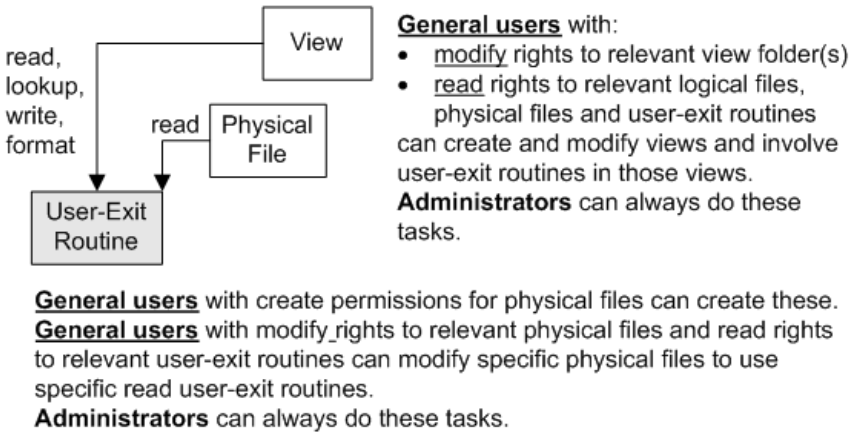
20 What is a user-exit routine?

A user-exit routine is a program specifically written for your company, which performs some unique and necessary computing task. There are four types of user-exit routine:

- **Read** - use for special processing during a read of your mainframe data. For example, SAFR can only **read encrypted data** with the help of a read user-exit routine.
- **Lookup** - use for special processing in a step for a Lookup Path. For example, SAFR can only handle **exceptions for some ranges of key values** with the help of a lookup user-exit routine.
- **Write** - use for special processing during a write of your mainframe data. For example, SAFR can only **write encrypted data** with the help of a write user-exit routine. Also called "**output**" user-exit routines.
- **Format** - use for special processing during output of reports. For example, a format user-exit routine might customize jumping to new pages in certain places in a report.



30 How do I use a user-exit routine?



User-Exit Routines are used in views and physical files- see these topics:

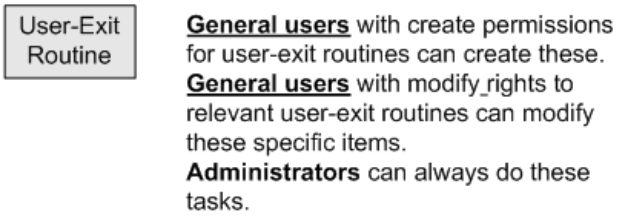
- "**Physical Files overview**",
- "**Views overview**",
- "**Views - advanced overview**",
- "**Creating physical files**",
- "**Creating views**",
- "**Modifying physical files**",
- "**Modifying views**".

The overview topics are elsewhere in this PDF - see the table of contents. To find the other topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

40 How do I know which user-exit routine to use?

See your system or environment administrator for advice on user-exit routines in your environment.

50 How do I create or modify a user-exit routine?



System administrators and environment administrators can always **create or modify user-exit routines**.

General users can **create or modify user-exit routines** if the group for login has the following authorities:

- **Create User-Exit Routine** permission in the relevant environment.
- **Modify or Delete rights to the relevant user-exit routines** in that environment.

For more on these authorities, see topics "**Groups overview**" , "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

Once the group has the above authorities, users in that group can **create or modify user-exit routines** by using the tasks below, which are **administrator tasks:**

- **Creating user-exist routines**,
- **Modifying user-exit routines**.

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

90 How do I delete a user-exit routine?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Views overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is a view?

A view configures the SAFR Performance Engine to produce results for your company. A view is a description of an input file(s), processing specifications and an output file. Views are prepared in the SAFR Workbench.

At a simplified level, the main components of a view are:

- The **input file(s)** called a view source file(s).

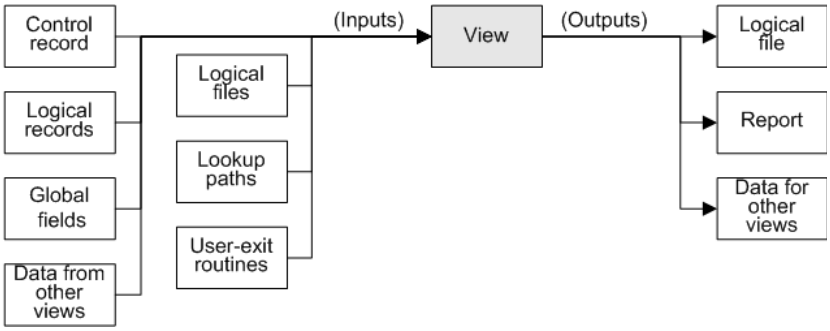
- The **output file** which can be a flat file or hardcopy report. A flat file can be a delimited file (for example, a CSV file).
- The **columns** which are the fields in the output file. Each column is one of the following:
  - **EITHER** a source file field,
  - **OR** a constant,
  - **OR** a formula which is implemented in **logic text**,
  - **OR** a lookup field which uses a **lookup path**.

  If the view has a format phase, then at least one column must be part of the **sort key.**

For an introduction to logic text and lookup paths, see topics "**Logic text overview**" and "**Lookup path overview**". For an introduction to phases, see topic "**SAFR phases overview**". Those topics are elsewhere in this PDF - see the table of contents.

A view is the most complex component in the SAFR metadata. This overview (and the advanced overview) provide an introduction only. You must create or modify a view to become familiar with all the data involved in a view.

20 What are the possible inputs and outputs of a view?



All of the **inputs** shown above are **metadata** components in SAFR - see "**Metadata Overview**". That topic is elsewhere in this PDF - see the table of contents.

30 How is a view used?

Views are run in the SAFR Performance Engine.

A view must be "**Active**" before the view can be run in the Performance Engine. A view is created, modified and activated by work in the SAFR Workbench.

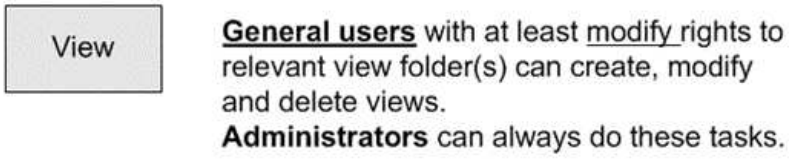
40 How do I know which view to use?

In the SAFR Workbench, look at views for the environments you have access to. If the appropriate view does not exist, either copy and/or modify an existing view, or create a new view.

45 How do I copy a view?

See topic "**Copying metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

50 How do I create or modify a view?



General users with at least modify rights to the relevant view folder(s) can create, modify and delete views in the view folder(s). Administrators can always do these tasks in any view folder in the environment. For instructions on creating or modifying views, see these topics:

- "**Creating views**",
- "**Modifying views**".

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

60 Metadata reports on views

To see a **report on selected views**, use one of these FAQ topics:

- "**How do I generate a View Properties Report?**"
- "**How do I generate a View Column Report?**"
- "**How do I generate a View Column PIC Report?**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

90 How do I delete a view?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

95 More information

See topic "**Views - advanced overview**". That topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Views - advanced overview

This topic assumes you are familiar with the topic "**Views overview**". That topic is elsewhere in this PDF - see the table of contents.

01 Summary of this topic

This topic covers the following:

- “10 Logic Text and formulas”
- “20 Activating a view”
- “30 What metadata components does a view access?”
- “50 Optimize view processing” on page 152
- “90 How to delete a view” on page 152
- “100 Need more on this page?” on page 153

10 Logic Text and formulas

Logic text is a scripting language that can be added to a view to change the behaviour. Logic text can **select records in input or output files**. Logic text can implement **formulas** for columns in a view. See topic "**Logic text overview**". That topic is elsewhere in this PDF - see the table of contents.

20 Activating a view

During creation of a view, or immediately after modification of any part of a view, the view becomes "**Inactive**".

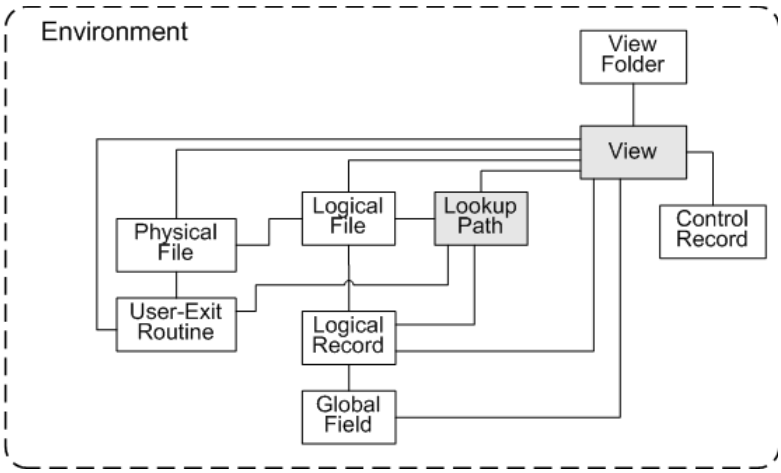
The view must be activated before the view can be run in the SAFR Performance Engine. Activation means all parts of the view are validated. The validation displays any error messages that prevent the view becoming '**Active**".

For details of how to activate a view, see these topics:

- "**Creating views**"
- "**Modifying views**"
- "**Batch activating views**"

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 What metadata components does a view access?



The diagram above shows the metadata types that have a direct relationship to a view. The lines show where metadata types are related. The relationship may be one to many or many to many, and may be only in one direction. If there is no line then there is no direct relationship.

System and environment administrators can create and modify all the components shown above.

General users can create and modify the shaded components, which produce the most obvious results for SAFR. General users can have extra authorities granted by membership of a group. The extra authorities can allow a general user to create and modify other components. Granting of extra authorities is a way of implementing specialized job roles for general users in your company.

All the components in the diagram above have overview topics. Authorities are discussed in topic "**Groups overview**".

Overviews topics are elsewhere in this PDF - see the table of contents.

50 Optimize view processing

The methods are:

- **Parallelism** - allowing (where possible) to do processing in parallel.
- **Common Key Buffers** - to load into memory at the same time all relevant records with a shared key.
- **Pipes** - to allow one view to pass data in memory to another view(s). This technique allows parallel processing.
- **Tokens** - to pass data between views in a manner similar to a pipe. The token means that the receiving view(s) must wait until the sending view has finished processing an entire record. This method has advantages and disadvantages compared to pipes.

For more details, see these topics:

- "**Parallelism overview**",
- "**Common Key Buffers overview**",
- "**Pipes overview**",
- "**Tokens overview**".

These topics are elsewhere in this PDF - see the table of contents.

90 How to delete a view

You must have the modify right to the view folder containing the view. To delete a view do the following:

1. In the **Navigator**, select the relevant view folder.
2. In the **Metadata List**, click on the relevant view to highlight it.
3. Click ✖ or press the **Delete** key.

Views are not deleted immediately.

When you delete a view, then the view is placed in a folder "**Deleted Views**" and the status is made inactive.

Later, the task "**Empty "Deleted Views" folder**" permanently deletes views in that folder. That topic is elsewhere in this PDF - see the table of contents.

An alternative method to achieve the permanent delete is to delete views from the "**Deleted Views**" folder. When you confirm a delete from "**Deleted Views**", the view is deleted permanently.

This pattern is not affected by the fact that if a view is stored in the "**Deleted Views**" folder that view is also listed in the "**All Views**" folder. Deleting a view from the "**Deleted Views**" folder also deletes the view from the "**All Views**" Folder.

100 Need more on this page?

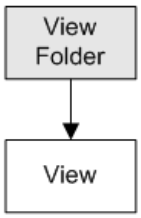
If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

View folders overview

01 Summary of this topic

The sections in this topic are as follows:

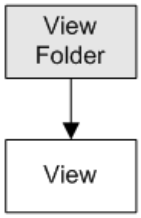
10 What is a view folder?



A view folder is a folder (directory) that **stores views in the SAFR Workbench**. A view can only be stored in one view folder.

Each user has a default view folder, and a user can override this default when they store a view.

20 How do I use a view folder?



General users with modify rights to specific view folders can create views inside them.
Administrators can always do this task.

Use a view folder to store views - see these topics:

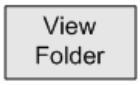
- **Views overview**,
- **Views - advanced overview**,
- **Creating views**,
- **Modifying views**.

The overview topics are elsewhere in this PDF - see the table of contents. To find the other topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I know which view folder to use?

When you save a view, choose from the drop down list of view folders available in that environment. Check with other general users on use of view folders.

50 How do I create or modify a view folder?



General users with create permissions for view folders can create these.
General users with modify rights to specific view folders can modify these.
Administrators can always do these tasks.

System administrators and environment administrators can always **create or modify view folders**.

General users can **create or modify view folders** if the group for login has the following authorities:

- **Create View Folders** permission in the relevant environment.
- **Modify or Delete rights to the relevant view folders** in that environment.

For more on these authorities, see topics "**Groups overview**" , "**Groups - advanced overview**" and "**WE Security overview**". These topics are elsewhere in this PDF - see the table of contents.

Once the group has the above authorities, users in that group can **create or modify view folders** by using the tasks below, which are **administrator tasks:**

- **Creating view folders**,

- **Modifying view folders**.

To find these topics in a PDF, see chapter "**Cross reference of topics and PDF files**".

90 How do I delete a view folder?

See topic "**Deleting metadata**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

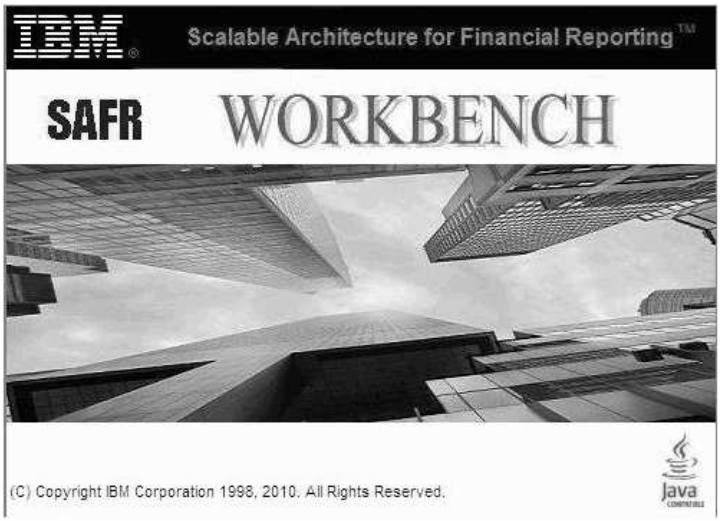
If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Workbench overview

01 Summary of this topic

The sections in this topic are as follows:

02 Introduction



The **SAFR Workbench** is software that you use to configure SAFR in order to do work.

The SAFR Workbench software runs on a PC and communicates to the mainframe computers in your company.

The SAFR Workbench can also be called **WE** which stands for **Workbench Eclipse.**

Eclipse refers to a programming platform that was invented by IBM and is now free and open source software.

In 2010, IBM upgraded the original SAFR **Windows Workbench (WW)** software to use Eclipse, and added greater functionality.

10 What do I use the workbench for?

Use the workbench to configure SAFR by configuring metadata. Metadata covers the following items:

- Users (meaning general users, environment administrators and system administrators),
- Groups,
- Environments,
- Physical Files,
- Logical Files,
- Logical Records,
- Global Fields,
- Control Records,
- User-Exit Routines,
- View Folders,
- Views,
- Lookup Paths.

For more, see topic "**Metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

20 How do I login to the workbench?

For instructions on how to start the workbench software, see your system administrator.

To login, see topic "**Logging into the SAFR Workbench**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 How do I use the screens in the workbench?

See topic "**Basics of using the SAFR Workbench**" for details. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

40 Keyboard shortcuts in the workbench

See topic "**What are the keyboard shortcuts?**" for details.

To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

50 Virtualization for the workbench

Virtualization means the workbench can be installed on a server so that multiple workstations can run using the server copy of the software, for example using Citrix.

Virtualization uses three Windows user profiles:

- **Local Profile area**

  The path to the Local Profile area for the workbench depends on the version of Windows:

  – In **Windows XP**, the path is "**C:\Documents and Settings\<user>\Local Settings\Application Data\SAFR\Workbench Eclipse**".

  – In **Windows 7**, the path is "**C:\Users\<user>\AppData\Local\SAFR\ Workbench Eclipse**".

  The Local Profile area stores potentially large files of data. If required, the workbench automatically creates these sub-folders:

  – **csv folder** containing CSV files of exported dependency checker reports

  – **log folder** containing log files about workbench sessions

  – **pdf folder** containing PDF reports

  – **workspace folder** containing Eclipse RCP work files for running the workbench

  – **xml folder** containing XML files of exported metadata.

- **Roaming Profile area**

  The path to the Roaming Profile area for the workbench depends on the version of Windows:

  – In **Windows XP**, the path is "**C:\Documents and Settings\<user>\ Application Data\SAFR\Workbench Eclipse**".

  – In **Windows 7**, the path is "**C:\Users\<user>\AppData\Roaming\SAFR\ Workbench Eclipse**".

  The Roaming Profile area stores data that is mirrored on the Windows Domain server.

  The Roaming Profile area stores the following unique data for that user:

  – Values entered by the user on the SAFR Connection Manager screen.

  – Any user updates to the WE log file path.

    For more on WE log files, see topic "**WE log file overview**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

  – The user choice for Old Compiler during login.

  – The User ID for the last login to the workbench.

  – The views recently accessed by the user.

  Any user can create/delete/modify a local database connection name, or can modify/default a global database connection name.

  For more on database connection names, see topic "**SAFR Connection Manager screen help**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

- **All Users Profile area**

  This is a profile area on the server used for virtualization. The path to the All Users Profile area for the workbench depends on the version of Windows:

  – In **Windows XP**, the path is "**C:\Documents and Settings\All Users\Application Data\SAFR\Workbench Eclipse**".

– In **Windows 7**, the path is "**C:\ProgramData\SAFR\Workbench Eclipse**".

The All Users Profile area optionally stores default data for all users of the workbench.

80 How do I optimize SAFR?

See topic "**SAFR optimization overview**" for details. This assumes you have read all other overviews and are familiar with SAFR overall.

That overview topic is elsewhere in this PDF - see the table of contents.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

WE log file overview

01 Summary of this topic

The sections in this topic are as follows:

10 What is the log file?

A log file is a trace of database activity in WE. This file is useful when debugging problems and when contacting IBM for help with a problem.

A simple example is below:

```
29/08/2014 3:54:15 PM com.ibm.safr.we.ui.dialogs.SAFRLogin <init>
INFO: ====== START SAFR WORKBENCH SESSION ======
29/08/2014 3:54:15 PM com.ibm.safr.we.ui.dialogs.SAFRLogin <init>
INFO: Workbench Eclipse (WE) Version 4.14
29/08/2014 3:54:23 PM com.ibm.safr.we.internal.data.ConnectionFactory getConnection
INFO: Connecting to database:
URL    jdbc:db2://SERVER01.YOURCO:1234/ABCDEF
Schema SAFRDB01
Userid johndoe
29/08/2014 3:54:30 PM com.ibm.safr.we.internal.data.ConnectionFactory getConnection
INFO: Got connection  com.ibm.db2.jcc.t4.b@79127912
29/08/2014 3:54:31 PM com.ibm.safr.we.ui.dialogs.SAFRLogin checkUser
INFO: Stored Procedure Version is SD4.14.002.116
29/08/2014 3:54:40 PM com.ibm.safr.we.ui.dialogs.SAFRLogin saveAndClose
INFO: SAFR Login Details:
SAFR Userid  JOHNDOE
Environment  Accounts[107]
Group        ACCOUNTS[35]
Authority    Normal user
29/08/2014 3:54:40 PM com.ibm.safr.we.ui.dialogs.SAFRLogin saveAndClose
INFO: Compiler version is PM 4.14.008 (new compiler)
29/08/2014 3:55:07 PM com.ibm.safr.we.internal.data.dao.DB2ViewDAO getDuplicateView
INFO: No duplicate View in Env 107 with name : General_Report
```

Note the following:

- The first 21 lines down to "**INFO: Compiler version**" indicate a successful login to WE.

- The last two lines are normal when creating a new view (in this case called "General_Report"). The messages indicate that the name "General_Report" is not used for any existing view, so this name is available for a new view.

20 Opening a log file inside WE

See topic "**Opening a log file**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

30 Where is the WE log file?

Use topic "**Changing Log Path**" to display the current log file path. To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

The first time the workbench executes, a default log file path is provided. Users can change to a different path by using the topic "**Changing Log Path**".

In a virtualization environment such as Citrix, a default log file path can be provided by the administrators.

For more on virtualization, see topic "**Workbench overview**". To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

40 Naming system for log files

WE log files are named:

WE.cycle.log.instance	Example: **WE.1.log** Note the "instance" part of the name is blank in this example.

The parts of the naming system are explained below:

Naming component	Notes
WE	Always these letters
cycle	**Cycle** indicates which session of the workbench, as follows: • Cycle starts at 0 and then counts 1, 2 or 3 and so on. • **Cycle 0** means the **current** instance of a session in the workbench. • **Cycle 1** means the **previous** session in the workbench. Cycle 2 means the session before, and so on. • When your session ends, and you start a new session later, the **names of existing log files change**. For the current session, the log file is **WE.0.log** by WE. After you finish this session and start a new session later, the log file is renamed to **WE.1.log** by WE. As new sessions finish and start, the log files continue to change name.
log	Always these letters

Naming component	Notes
instance	**Instance** indicates multiple sessions running on your workstation at the same time, as follows: • Instance starts at **blank**, and then is a number: 1, 2 or 3 and so on. • **Instance blank** means the **first** of any multiple instances in the workbench. If you never run multiple sessions at the same time, then the instance is always blank. • **Instance 1** means the **second multiple** session running at the same time. Instance 2 means the third of multiple session, and so on. • For example, for the first current session, the log file is **WE.0.log** and the second instance running at the same time is named **WE.0.log.1** by WE. A third multiple instance is named **WE.0.log.2** by WE. As explained for cycle, after all these sessions end and a new session starts later, those log files are renamed to **WE.1.log** and **WE.1.log.1** and **WE.1.log.2** by WE. Notice that the cycle number changes but the instance numbers do not change.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

WE Security overview

This topic helps all users (a general user, environment administrator and system administrator) to learn how security works in the workbench. Note that general users can perform extra tasks if the login group has appropriate security. Note that an environment administrator requires the login group to have administrator access.

This topic assumes you are familiar with all the metadata in the SAFR Workbench - see topic "**Metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

01 Summary of this topic

This topic describes security from various points of view:

  - “52 Security for environments” on page 167
  - “54 Security for global fields” on page 167
  - “56 Security for logical files” on page 167
  - “58 Security for logical records” on page 168
  - “60 Security for lookup paths” on page 168
  - “62 Security for physical files” on page 169
  - “64 Security for user-exit routines” on page 169
  - “66 Security for views” on page 170
  - “68 Security for view folders” on page 170
- **Reports** - see “90 Security for reports” on page 170
- **Utilities** - see “91 Security for utilities” on page 171
- **Copy data between environments**
  - See “92 Security for export and import” on page 172
  - See the **Migration Utility** in “91 Security for utilities” on page 171
  - See “93 Security for XML files used in export and import” on page 172
- The last section is: “100 Need more on this page?” on page 172

10 Security phase "edit rights"

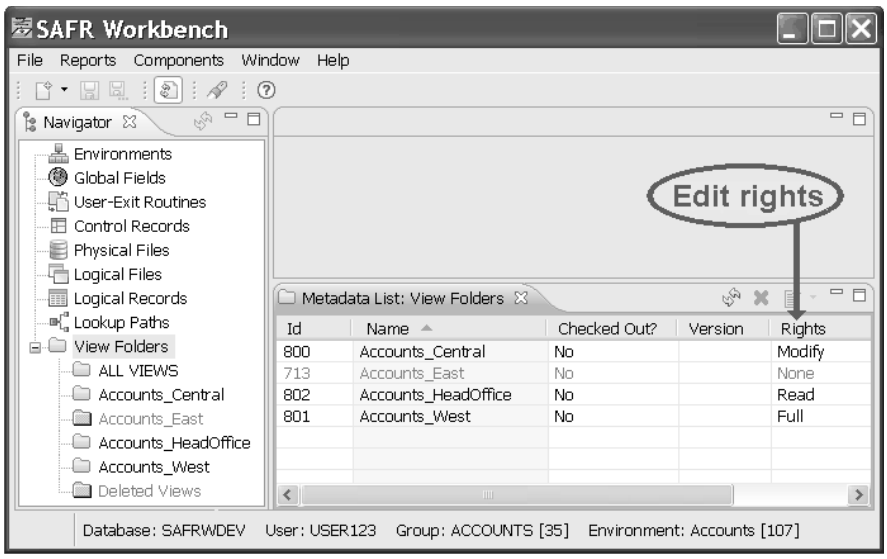
In a particular environment, the phrase "**edit rights**" means the rights to an item of the following types:

- Logical file,
- Logical record,
- Physical file,
- User-exit routine,
- View folder.

The edit rights possible are:

- **No rights at all**.
- **Read** right which allows both display and usage of an item. For example, a user needs the read right to a logical record in order to refer to that logical record in a view.
- **Modify** right which implies read as well.
- **Delete** right which implies the modify and read rights as well. This right is also called "**full**" rights.

Edit rights can be seen in column "**Rights**" in the **Metadata List**. For example, if you click on "**View Folders**" in the **Navigator**, the Metadata List may appear as follows:



Edit rights are setup by administrators - see these tasks:

- "**Modifying group permissions by environment**"
- "**Modifying group permissions by group**"

To find these tasks in a PDF, see chapter "**Cross reference of topics and PDF files**".

12 Security phrase "create permission"

In a particular environment, the phrase "**create permission**" means the permission to create a new item of the following types:

- Logical file,
- Logical record,
- Physical file,
- User-exit routine,
- View folder.

For example, the create permission for physical files means a general user can create a new physical file at any time.

Create permissions are setup by administrators - see these tasks:

- "**Modifying group permissions by environment**"
- "**Modifying group permissions by group**"

To find these tasks in a PDF, see chapter "**Cross reference of topics and PDF files**".

14 Security phrase "run permission for utilities"

In a particular environment, the above phrase means permission to run the following utilities in WE:

- **Batch Activate Lookup Paths.** This utility checks lookup paths are ready to use in an environment, and if possible sets the status to "active". For more, see topic "**Lookup Paths overview**". That topic is elsewhere in this PDF - see the table of contents.

- **Batch Activate Views.** This utility checks views are ready to run in the SAFR Performance Engine, and if possible sets the status to "active". For more, see topic "**Views overview**". That topic is elsewhere in this PDF - see the table of contents.
- **Migration Utility.** This utility copies selected metadata from a source environment to target environment in the same SAFR Database. For more, see topic "**Migrate metadata overview**". That topic is elsewhere in this PDF - see the table of contents.

There is one group run permission:

- **Migrate In**. This provides access to all three utilities:
  - **Migration Utility** where the target environment is this environment ,
  - **Batch Activate Lookup Paths** and **Batch Activate Views** in this environment.

The above run permission can be applied to a general user in one more environments. Administrators in an environment always have this run permission.

The run permission for utilities is setup by administrators - see these tasks:

- "**Modifying group permissions by environment**"
- "**Modifying group permissions by group**"

To find these tasks in a PDF, see chapter "**Cross reference of topics and PDF files**".

20 General users and all users - what you can always do

For the login environment, every user **can always:**

- Modify their own user record.
- Read any control record, global field and lookup path.
- Delete any lookup path.
- Read the environment record.
- Generate an Environment Security Report. This report lists all the groups in that environment.
- Generate an Environment Checker Report. This report shows all environments for a particular metadata item name.

All the above only applies to those environments where the login group has access, except for the Environment Checker Report which can be run by any user and always covers all environments.

General users should also be aware of the topic "**What metadata do I want to see?**" To find that topic in a PDF, see chapter "**Cross reference of topics and PDF files**".

22 General users - what you can possibly do

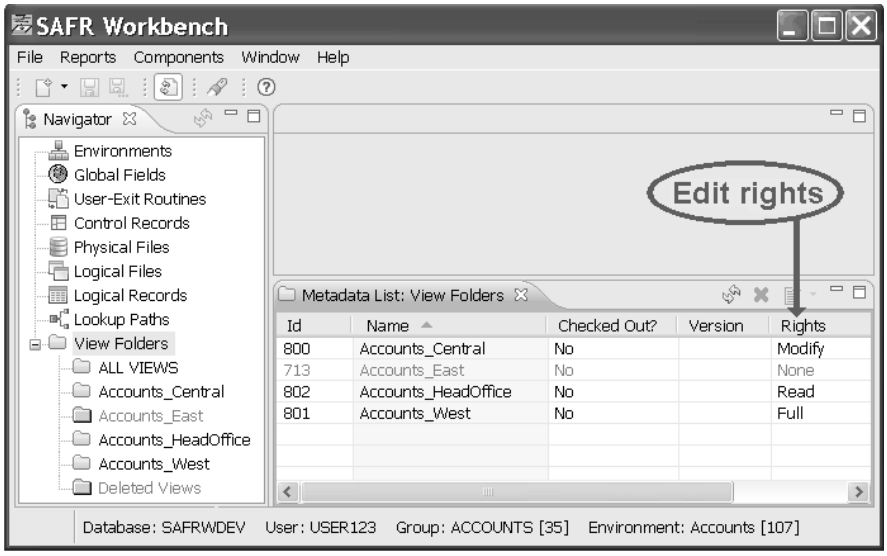
General users get most access from the **group** selected during login.

Normally, the login group provides general user access which is indicated when the "**Components**" menu displays in the SAFR Workbench.

The login group provides environment administrator user access when the "**Administration**" menu displays in the SAFR Workbench. If this applies, see “30 Security for an environment administrator” on page 165.

When the login group provides general user access, the most important part of that access is edit rights. Edit rights are important in deciding what a general user can possibly do. For a definition of edit rights, see section “10 Security phase "edit rights"” on page 161.

Edit rights can be seen in column "**Rights**" in the **Metadata List**. For example, if you click on "**View Folders**" in the **Navigator**, the Metadata List may appear as follows:



Let's take a detailed look at how edit rights (and some other rights) affect a general user.

For the login environment, and for any environment where the login group has access, a general user **can possibly**:

- Read a view, only if the login group has at least the read right to the view folder that contains the view.

  If the user has only the read right to the view folder, this allows the user to click the activate button in a view without allowing the user to save the results. This allows a user in this situation to check if a view can be activated.
- Copy a view only if the login group has:
  - The read right to the view folder that contains the view to be copied,
  - The modify right to the view folder to store the copied view.
- Create, modify or delete a view only if the login group has at least the modify right to the view folder that contains the view.
- Create a logical file, logical record, physical file, user-exit routine or view folder only if the login group has the create permission for these item types.
- Read, modify, or delete a particular logical file, logical record, physical file, user-exit routine, view folder only if the login group has the read. modify. or delete edit right for those particular items.
- Create and modify any lookup path, but a general user may only select logical records and logical files on which they have edit rights. If a general user does not have edit rights on the logical records or logical files which are already selected in an existing lookup path, they can still modify the lookup path to use different logical records or logical files where that general user does have the

edit rights. In this situation, once the lookup path is modified, the general users cannot change the lookup path back to the original selections for logical records or logical files, because that general user does not have edit rights for this.

- Refer to a lookup path in a view only if the login group has:
  - At least the modify right to the view folder that contains the view.
  - At least the read right to the relevant logical records and logical files used in that lookup path.
- Run the utilities called Batch Activate Lookup Paths and Batch Activate Views only if the login group has the Migrate In run permission for the relevant environment.
- Migrate (copy) metadata from another environment only if the login group has:
  - At least the read right to the relevant metadata in the source environment.
  - At least the Migrate In run permission in the target environment.

  Once a successful copy of metadata using the Migration Utility is complete, the general user automatically has at least the read right in the target environment to the metadata that was copied in. If the general user had more rights than the read right before the migration, then the extra rights are retained.

If you require more security authority in your login group, see your administrator.

24 General users - what you can never do

It is **not possible** for a general user to:

- Display a list of all user or group names in the Navigator shown on the left of the WE screen. This is only possible for a system administrator.
- Create, read, modify or delete any user, group, or environment. with one exception: a general user can read and modify their own user record. All other create, read, modify, and delete actions for users, groups and environments are performed only by administrators.
- Create, modify or delete any control record or global field. This is only performed by administrators

30 Security for an environment administrator

An environment administrator gets most access in WE from the **group** selected during login.

When the login group has administrator access for an environment, then users who login with that combination of group and environment are automatically an environment administrator. If those users login with a different combination of group and environment, then those users be general users in that session. Each login session depends on the credentials provided during login.

In the login environment and any environment where the login group provides administrator access, **an environment administrator can:**

- Create, read, modify or delete any logical file, logical record, lookup path, physical file, user-exit routine, view folder, view, control record, or global field.
- Run any utility.
- Modify their own user record.
- Modify a group in that environment. An environment administrator can only change edit rights, create permissions and the run permission in that group.

Effectively, the environment administrator can control most of the general user access to that environment by modifying groups.
- Read and modify the relevant environment records.

It is **not possible** for an environment administrator to perform these system administrator tasks:
- Display a list of all user or group names in the Navigator shown on the left of the WE screen.
- Create, read, modify or delete any user with one exception: an environment administrator can read and modify their own user record.
- Create or delete any group.
- Create or delete an environment.

35 Security for groups

In any environment, a **system administrator** can create, read, modify or delete groups. For any group, a system administrator controls:
- Which users are members of the group.
- Which environments the group can access.
- Edit rights for particular metadata items in an environment.
- Create permissions in each environment.
- Run permission for utilities in each environment.
- Administrator access which makes each member of the group an environment administrator.

In summary, a system administrator can do all actions for any group in any environment.

In the login environment or in an environment where the login group provides administrator access, an **environment administrator** can modify an existing group, as follows:
- Change create permissions for types of metadata,
- Change run permission for utilities,
- Change edit rights to particular items of metadata.

An environment administrator can control some of the general user access to that environment.

40 Security for a system administrator

It is only possible for one system administrator to create another system administrator.

A system administrator has complete access to all parts of WE.

In any environment, a system administrator can:
- Create, read, modify or delete any metadata item, including all users, groups and environments.
- Run any utility.

In particular, only a system administrator controls which users are an environment administrator.

50 Security for control records

In any environment, a **system administrator** can always create, read, modify and delete control records and refer to control records in views.

In the login environment, an **environment administrator** can always create, read, modify and delete control records and refer to control records in views.

In the login environment, **general users** can:

- Always read control records.
- Modify a view to refer to a control record only if the login group has at least the modify right to the view folder that contains the view.

52 Security for environments

In any environment, a **system administrator** can always create, read, modify and delete environment records.

In the login environment or in an environment where the login group provides administrator access, an **environment administrator** can always read and modify the environment record.

In the login environment or in an environment where the login group has access, a **general user** can always read the environment record.

54 Security for global fields

In any environment, a **system administrator** can always create, read, modify and delete global fields and refer to global fields in views.

In the login environment, an **environment administrator** can always create, read, modify and delete global fields and refer to global fields in views.

In the login environment, **general users** can:

- Always read global fields.
- Modify a view to refer to a global field only if the login group has at least the modify right to the view folder that contains the view.
- Modify a logical record to refer to a global field only if the login group has at least the modify right to the relevant logical record.

56 Security for logical files

In any environment, a **system administrator** can always create, read, modify and delete logical files and refer to logical files in views.

In the login environment, an **environment administrator** can always create, read, modify and delete logical files and refer to logical files in views.

In the login environment, **general users** can:

- Create a particular logical file only if the login group has at least the create permission for logical files.
- Read, modify or delete a particular logical file only if the login group has at least the read, modify or delete right to that particular logical file.
- Refer to a logical file in a view only if the login group has:

- At least the modify right to the view folder that contains the view.
- At least the read right to the logical file.

58 Security for logical records

In any environment, a **system administrator** can always create, read, modify and delete logical records and refer to logical records in views.

In the login environment, an **environment administrator** can always create, read, modify and delete logical records and refer to logical records in views.

In the login environment, **general users** can:

- Create a particular logical record only if the login group has at least the create permission for logical records.
- Read, modify or delete a particular logical record only if the login group has at least the read, modify or delete right to that particular logical record.
- Refer to a logical record in a view only if the login group has:
  - At least the modify right to the view folder that contains the view.
  - At least the read right to the logical record.

60 Security for lookup paths

In any environment, a **system administrator** can always create, read, modify and delete lookup paths and refer to lookup paths in views.

In the login environment, an **environment administrator** can always create, read, modify and delete lookup paths and refer to lookup paths in views.

In the login environment, **general users** can:

- Always read and delete lookup paths.
- Create and modify any lookup path, but a general user may only select logical records and logical files on which they have edit rights. If a general user does not have edit rights on the logical records or logical files which are already selected in an existing lookup path, they can still modify the lookup path to use different logical records or logical files where that general user does have the edit rights. In this situation, once the lookup path is modified, the general users cannot change the lookup path back to the original selections for logical records or logical files, because that general user does not have edit rights for this.
- Refer to a lookup path in a view only if the login group has:
  - At least the modify right to the view folder that contains the view.
  - At least the read right to the logical records and logical files used in that lookup path.
- Run Batch Activate Lookup Paths only if the login group has:
  - **EITHER** system or environment administrator access to the environment for the lookup path(s).
  - **OR** the "Migrate In" run permission to the environment for the lookup path(s).

Notice that security for lookup paths and views can sometimes create unusual situations. For example, consider this example:

1. User_1 has edit access to these logical files and logical records (all ending in "1"):

- LogFile_Source1,
- LogRec_Source1,
- LogiFile_Target1,
- LogiRec_Target1.

User_1 has NO read access to other logical files and logical records.

2. User_2 has edit access to these logical files and logical records (all ending in "2"):
   - LogFile_Source2,
   - LogRec_Source2,
   - LogiFile_Target2,
   - LogiRec_Target2.

   User_2 has NO read access to other logical files and logical records.
3. User_1 **creates Lookup_Special** and selects the logical file and logical record names ending in "1".
4. User_2 **modifies Lookup_Special** to use the logical file and logical record names ending in "2".
5. Problem: User_1 can no longer refer to Lookup_Special in a view. User_1 can modify Lookup_Special back to the names ending in "1", but cannot modify Lookup-Special retaining the names ending in "2".

The problem for User_1 can be solved in one of the following ways:

- **EITHER** ask User_1 and User_2 to create separate lookup paths with different names. The two users can now work independently.
- **OR** ask an administrator to give User_1 and User_2 edit access to **all** the logical file and logical records mentioned above. When complete, the two users have equal rights to update Lookup_Special.

62 Security for physical files

In any environment, a **system administrator** can always create, read, modify and delete physical files and refer to physical files in views.

In the login environment, an **environment administrator** can always create, read, modify and delete physical files and refer to physical files in views.

In the login environment, **general users** can:

- Create a particular logical file only if the login group has at least the create permission for logical files.
- Read, modify or delete a particular physical file only if the login group has at least the read, modify or delete right to that particular physical file.
- Refer to a physical file in a view only if the login group has:
  - At least the modify right to the view folder that contains the view.
  - At least the read right to the physical file.

64 Security for user-exit routines

In any environment, a **system administrator** can always create, read, modify and delete user-exit routines and refer to user-exit routines in views.

In the login environment, an **environment administrator** can always create, read, modify and delete user-exit routines and refer to user-exit routines in views.

In the login environment, **general users** can:

- Create a particular user-exit routine only if the login group has at least the create permission for user-exit routines.
- Read, modify or delete a particular user-exit routine only if the login group has at least the read, modify or delete right to that particular user-exit routine.
- Refer to a user-exit routine in a view only if the login group has:
  - At least the modify right to the view folder that contains the view.
  - At least the read right to the user-exit routine.

66 Security for views

In any environment, a **system administrator** can always create, read, modify and delete views.

In the login environment, an **environment administrator** can always create, read, modify and delete views.

In the login environment, **general users** can:

- Read a particular view only if the login group has at least the read right to that view folder that stores the view.
- Create, modify or delete a particular view only if the login group has at least the modify right to that particular view folder that stores the view.
- Run Batch Activate Views only if the login group has:
  - **EITHER** at least the modify right to the view folder that contains the views to be activated,
  - **OR** the "Migrate In" run permission to the environment for the view(s).

68 Security for view folders

In any environment, a **system administrator** can always create, read, modify and delete view folders and create, read, modify and delete views in those view folders.

In the login environment, an **environment administrator** can always create, read, modify and delete view folders and create, read, modify and delete views in those view folders.

In the login environment, a **general user** can:

- Create a particular view folder record only if the login group has at least the create permission for view folders.
- Read, modify or delete a particular view folder record only if the login group has at least the read, modify or delete right to that particular view folder.
- Read a view in a view folder only if the login group has at least the read right to the view folder that contains the view.
- Create, modify, delete a view in a view folder only if the login group has at least the modify right to the view folder that contains the view.

90 Security for reports

In any environment, a **system administrator** can always run any report.

In the login environment or in an environment where the login group provides administrator access, an **environment administrator** can always run any report.

In the login environment, **general users** can:

- Run a Logical Record Report only if the login group has at least the read right to the logical records selected.
- Run a Logical File Report only if the login group has at least the read right to the logical files selected.
- Run a View Properties Report only if the login group has at least read right to the view folder(s) for the views selected.
- Run a View Columns Report only if the login group has tat least read right to the view folder(s) for the views selected.
- Run a View Column PIC Report only if the login group has at least read right to the view folder(s) for the views selected.
- Run an Environment Security Report for any environment that the login group has access to. This report shows the groups in that environment.

In any environment, **all users** can:

- Generate an Environment Checker Report. This report shows all environments for a particular a metadata item name.

91 Security for utilities

As background, see the above section “14 Security phrase "run permission for utilities"” on page 162.

In any environment, a **system administrator** can always run a utility.

An **environment administrator**is an administrator only in the login environment or in an environment where the login group provides administrator access. An **environment administrator** can always run:

- Batch Activate Lookup Paths,
- Batch Activate Views.

An **environment administrator** can run the Migration Utility only if the login group has at least both of these requirements:

- Read rights to the metadata to be copied for the source environment. This is provided if the login group provides environment administrator access to the source environment. Alternatively, this is provided if the login group provides enough read rights to a general user in the source environment.
- Migrate In run permission for a general user in the target environment. Alternatively, this is provided if the login group provides environment administrator access to the target environment.

Notice how an environment administrator does not need administrator access in both the source and target environments.

In the login environment, or in an environment where the login group has access, a **general user** can:

- Run Batch Activate Views only if the login group has:
  - **EITHER** at least the modify right to the view folder that contains the views,
  - **OR** the user has the "Migrate In" run permission in that environment.

- Run Batch Activate Lookup Paths only if the login group has the "Migrate In" run permission in that environment.
- Run the Migration Utility only if the login group has:
  - Read rights to the metadata to be copied for the source environment,
  - **AND** the Migrate In run permission for the target environment.

Once a successful copy of metadata using the Migration Utility is complete, the general user automatically has at least the read right in the target environment to the metadata that was copied in. If the general user had more rights than the read right before the migration, then the extra rights are retained.

92 Security for export and import

A **system administrator** can always export and import metadata into and out of any environment. See next section below for security for XML files.

An **environment administrator** can always export and import metadata into and out of the login environment. This also applies to any environment where the login group provides administrator access. See next section below for security for XML files.

A **general user** can export metadata into an XML file only if the login group has at least the read rights to the items selected for export. See next section below for security for XML files.

General users cannot import metadata - only an administrator can. A general user can possibly run the Migration Utility - see previous section above.

93 Security for XML files used in export and import

The tasks "export metadata" and "import metadata" use XML files. XML files themselves only have the security provided by the operating system that is running the workbench. Normally, all users of the workbench can create, read, modify and delete XML files. Any restrictions on access to XML files provides a further layer of security.

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

XML structure for metadata overview

01 Summary of this topic

The sections in this topic are as follows:

02 Knowledge you need first

This topic assumes you are familiar with these topics:

- "**Export metadata overview**",
- "**Import metadata overview**".

These topics are elsewhere in this PDF - see the table of contents.

10 Relationships between metadata

This topic shows the records inside an XML file containing exported metadata.

The "main" metadata types in an XML file must be one of:

- Physical file,
- Logical file,
- Logical record,
- Lookup path,
- View.

The table below shows the possible "associated" items:

"Main" item	Possible "associated" items
Physical file	User-exit routine
Logical file	Physical file, user-exit routine
Logical record	Logical file, physical file, user-exit routine
Lookup path	Logical record, logical file, physical file, user-exit routine
View	Control record, lookup path, logical record, logical file, physical file, user-exit routine

20 Records in XML files

The possible record names in an XML file are shown in the table below.

Main item	Possible record names in XML file
Physical file	PhysicalFile, Procedure, Server
Logical file	LogicalFile, File-Partition, PhysicalFile, Procedure, Server
Logical record	LogicalRecord, LRField, LR-Field-Attributes, LR-Index, LR-IndexFields, LR-File, LogicalFile, File-Partition, PhysicalFile, Procedure, Server
Lookup path	Join, Join-Source, Join-Target, LogicalRecord, LRField, LR-Field-Attributes, LR-Index, LR-IndexFields, LR-File, LogicalFile, File-Partition, PhysicalFile, Procedure, Server
View	View, View-Columns, View-Sources, View-SourceFields, View-SortKeys, View-LogicText, Join, Join-Source, Join-Target, LogicalRecord, LRField, LR-Field-Attributes, LR-Index, LR-IndexFields, LR-File, LogicalFile, File-Partition, PhysicalFile, Procedure, Server, ControlRecord

Notice that a **control record** metadata item cannot be a "main" item for export, but a control record can be an "associated" item for export of a view. To copy a control

record for one environment to another, you must copy a view that contains that control record.

30 Server record in XML file

The **Server** record is **never** imported. The Server record is created during the export process that creates the XML file.

40 XML records for an exported physical file

The table below shows the record names possible in an XML file for an exported physical file.

Record name	Metadata items involved	Notes
PhysicalFile	Physical file	The physical file metadata item.
Procedure	User-exit routine	A read user-exit routine associated with the physical file.
Server	Environment	See section "30 Server record in XML file"

The XML file contains one **PhysicalFile** record.

The XML file contains zero or one **Procedure** record. A **Procedure** record is present when a read user-exit routine is associated with the physical file.

The XML file contains one **Server** record.

45 XML records for an exported logical file

The table below shows the record names possible in an XML file for an exported logical file.

Record name	Metadata items involved	Notes
LogicalFile	Logical file	The logical file metadata item.
File-Partition	Logical file, physical file	A record associating a logical file and a physical file.
PhysicalFile	Physical file	A physical file associated with the logical file.
Procedure	User-exit routine	A read user-exit routine associated with the physical file.
Server	Environment	See section "30 Server record in XML file"

The XML file contains one **LogicalFile** record.

The XML file contains one or many **File-Partition** records, and the same number of **PhysicalFile** records.

The XML file contains zero or one or many **Procedure** records. A **Procedure** record is present when a read user-exit routine is associated with a physical file.

The XML file contains one **Server** record.

50 XML records for an exported logical record

The table below shows the record names possible in an XML file for an exported logical record.

Record name	Metadata items involved	Notes
LogicalRecord	Logical record	The logical record metadata item.
LRField	Logical record	A field in a logical record (some data).
LR-Field-Attributes	Logical record	A field in a logical record (more data).
LR-Index	Logical record	A field in the logical record sort key (some data).
LR-IndexFields	Logical record	A field in the logical record sort key (more data).
LR-File	Logical record, logical file	A record associating a logical record and a logical file.
LogicalFile	Logical file	A logical file associated with the logical record.
File-Partition	Logical file, physical file	A record associating a logical file and a physical file.
PhysicalFile	Physical file	A physical file associated with the logical file.
Procedure	User-exit routine	A read user-exit routine associated with the physical file.
Server	Environment	See section "30 Server record in XML file" on page 174

The XML file contains one **LogicalRecord** record.

The XML file contains one or many **LRField** records, and the same number of **LR-Field-Attributes** records.

The XML file contains zero or one or many **LR-Index** records, and the same number of **LR-IndexFields** records.

The XML file contains one or many **LR-File** records.

The XML file contains one or many **LogicalFile** records.

The XML file contains one or many **File-Partition** records, and the same number of **PhysicalFile** records.

The XML file contains zero or one or many **Procedure** records. A **Procedure** record is present when a read user-exit routine is associated with a physical file.

The XML file contains one **Server** record.

55 XML records for an exported lookup path

The table below shows the record names possible in an XML file for an exported lookup path.

Record name	Metadata items involved	Notes
Join	Lookup path	The lookup path metadata item.
Join-Source	Lookup path, logical record	A source logical record in the lookup path.
Join-Target	Lookup path, logical record	A target logical record in the lookup path.
LogicalRecord	Logical record	A logical record used in the lookup path.
LRField	Logical record	A field in a logical record (some data).
LR-Field-Attributes	Logical record	A field in a logical record (more data).
LR-Index	Logical record	A field in the logical record sort key (some data).
LR-IndexFields	Logical record	A field in the logical record sort key (more data).
LR-File	Logical record, logical file	A record associating a logical record and a logical file.
LogicalFile	Logical file	A logical file associated with the logical record.
File-Partition	Logical file, physical file	A record associating a logical file and a physical file.
PhysicalFile	Physical file	A physical file associated with the logical file.
Procedure	User-exit routine	A read user-exit routine associated with the physical file.
Server	Environment	See section “30 Server record in XML file” on page 174

The XML file contains one **Join** record.

The XML file contains at least one **Join-Source** record.

The XML file contains at least one **Join-Target** record.

The XML file contains at least two **LogicalRecord** records.

The XML file contains one or many **LRField** records, and the same number of **LR-Field-Attributes** records.

The XML file contains zero or one or many **LR-Index** records, and the same number of **LR-IndexFields** records.

The XML file contains one or many **LR-File** records.

The XML file contains one or many **LogicalFile** records.

The XML file contains one or many **File-Partition** records, and the same number of **PhysicalFile** records.

The XML file contains zero or one or many **Procedure** records. A **Procedure** record is present when a read user-exit routine is associated with a physical file.

The XML file contains one **Server** record.

60 XML records for an exported view

The table below shows the record names possible in an XML file for an exported view.

Record name	Metadata items involved	Notes
View	View	The view metadata item.
View-Columns	View	A column defined in the view editor.
View-Sources	View, logical file	A source logical file used in the view.
View-SourceFields	View	A field in the logical record of a source logical file used in the view.
View-SortKeys	View	A sort key field in the view editor.
View-LogicText	View	Logic text defined in the view.
Join	Lookup path	A lookup path used in the view.
Join-Source	Lookup path, logical record	A source logical record in the lookup path.
Join-Target	Lookup path, logical record	A target logical record in the lookup path.
LogicalRecord	Logical record	A logical record used in the lookup path.
LRField	Logical record	A field in a logical record (some data).
LR-Field-Attributes	Logical record	A field in a logical record (more data).

Record name	Metadata items involved	Notes
LR-Index	Logical record	A field in the logical record sort key (some data).
LR-IndexFields	Logical record	A field in the logical record sort key (more data).
LR-File	Logical record, logical file	A record associating a logical record and a logical file.
LogicalFile	Logical file	A logical file associated with the logical record.
File-Partition	Logical file, physical file	A record associating a logical file and a physical file.
PhysicalFile	Physical file	A physical file associated with the logical file.
Procedure	User-exit routine	A read user-exit routine associated with a physical file, or associated with the view.
Server	Environment	See section "30 Server record in XML file" on page 174
ControlRecord	Control record	The control record associated with the view.

The XML file contains one **View** record.

The XML file contains zero or one or many **View-Columns** records.

The XML file contains zero or one or many **View-Sources** records.

The XML file contains zero or one or many **View-SourceFields** records.

The XML file contains zero or one or many **View-SortKeys** records.

The XML file contains zero or one or many **View-LogicText** records.

The XML file contains zero or one or many **Join** records.

The XML file contains zero or one or many **Join-Source** records.

The XML file contains zero or one or many **Join-Target** records.

The XML file contains zero or one or many **LogicalRecord** record.

The XML file contains zero or one or many **LRField** records, and the same number of **LR-Field-Attributes** records.

The XML file contains zero or one or many **LR-Index** records, and the same number of **LR-IndexFields** records.

The XML file contains zero or one or many **LR-File** records.

The XML file contains zero or one or many **LogicalFile** records.

The XML file contains zero or one or many **File-Partition** records, and the same number of **PhysicalFile** records.

The XML file contains zero or one or many **Procedure** records. A **Procedure** record is present when a read user-exit routine is associated with a physical file. There can also be user-exit routines for other parts of a view.

The XML file contains zero or one **Server** record.

The XML file contains one **ControlRecord** record.

80 How to create and use an XML file for a metadata item

For more, see these topics:

- "**Exporting metadata**",
- "**Importing metadata**",

To find these tasks in a PDF, see chapter "**Cross reference of topics and PDF files**".

100 Need more on this page?

If you need more details to be added to this page, please email **AskSAFR@us.ibm.com** .

Chapter 2. Cross reference of topics and PDF files

How to download a PDF

Go to SAFR Information Center, select **About this Information Center** and select **PDF**. Follow the instructions on that page.

Alphabetical list of topics

Note the following:

- "**InfoCtr4150**" means the PDF called "**SAFR Information Center 4.15.00**" which contains all help topics.
- "**Top 3**" means the PDF called "**Top 3 - Admin Guide, General Users Guide and Overviews**".

Find the required topic in the first column below. The columns to the right show the PDFs that contain that topic.

Topic	PDF	PDF	PDF	PDF
Add View Source errors	Troubleshooting			InfoCtr4150
Add View Source screen help	Screens			InfoCtr4150
Admin Guide	Admin Guide		Top 3	InfoCtr4150
Administrators START HERE	Admin Guide		Top 3	InfoCtr4150
Basics of using the SAFR Workbench	Admin Guide	General Users Guide	Top 3	InfoCtr4150
Batch activate lookup paths	Admin Guide		Top 3	InfoCtr4150
Batch Activate Lookup Paths errors	Troubleshooting			InfoCtr4150
Batch Activate Lookup Paths screen help	Screens			InfoCtr4150
Batch activate views	Admin Guide		Top 3	InfoCtr4150
Batch Activate Views errors	Troubleshooting			InfoCtr4150
Batch Activate Views screen help	Screens			InfoCtr4150
Change Log Path screen help	Screens			InfoCtr4150
Changing Log Path	Admin Guide	General Users Guide	Top 3	InfoCtr4150
Checking metadata dependencies	Admin Guide		Top 3	InfoCtr4150
Clear environment	Admin Guide		Top 3	InfoCtr4150
Clear environment messages	Screens			InfoCtr4150
Column Source Properties errors	Troubleshooting			InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Column Source Properties screen help	Screens			InfoCtr4150
Common Key Buffers overview	Overviews		Top 3	InfoCtr4150
Control records overview	Overviews		Top 3	InfoCtr4150
Copying metadata	Admin Guide		Top 3	InfoCtr4150
Create New Extract Column Assignment errors	Troubleshooting			InfoCtr4150
Create New Extract Column Assignment screen help	Screens			InfoCtr4150
Create New Extract Record Filter errors	Troubleshooting			InfoCtr4150
Create New Extract Record Filter screen help	Screens			InfoCtr4150
Create New Format-Phase Calculation errors	Troubleshooting			InfoCtr4150
Create New Format-Phase Calculation screen help	Screens			InfoCtr4150
Create New Format-Phase Record Filter errors	Troubleshooting			InfoCtr4150
Create New Format-Phase Record Filter screen help	Screens			InfoCtr4150
Creating control records	Admin Guide		Top 3	InfoCtr4150
Creating environments	Admin Guide		Top 3	InfoCtr4150
Creating global fields	Admin Guide		Top 3	InfoCtr4150
Creating groups	Admin Guide		Top 3	InfoCtr4150
Creating logical files	Admin Guide		Top 3	InfoCtr4150
Creating logical records	Admin Guide		Top 3	InfoCtr4150
Creating lookup paths	General Users Guide		Top 3	InfoCtr4150
Creating physical files	Admin Guide		Top 3	InfoCtr4150
Creating user-exit routines	Admin Guide		Top 3	InfoCtr4150
Creating users	Admin Guide		Top 3	InfoCtr4150
Creating view folders	Admin Guide		Top 3	InfoCtr4150
Creating views	General Users Guide		Top 3	InfoCtr4150
Deleting metadata	Admin Guide		Top 3	InfoCtr4150
Deleting metadata messages	Troubleshooting			InfoCtr4150
Dependency Checker errors	Troubleshooting			InfoCtr4150
Dependency Checker screen help	Screens			InfoCtr4150
Do SAFR batch processes run sequentially or in parallel?				InfoCtr4150
Edit Control Record errors	Troubleshooting			InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Edit Control Record screen help	Screens			InfoCtr4150
Edit Environment errors	Troubleshooting			InfoCtr4150
Edit Environment screen help	Screens			InfoCtr4150
Edit Extract Column Assignment errors	Troubleshooting			InfoCtr4150
Edit Extract Column Assignment screen help	Screens			InfoCtr4150
Edit Extract Record Filter errors	Troubleshooting			InfoCtr4150
Edit Extract Record Filter screen help	Screens			InfoCtr4150
Edit Format-Phase Calculation errors	Troubleshooting			InfoCtr4150
Edit Format-Phase Calculation screen help	Screens			InfoCtr4150
Edit Format-Phase Record Filter errors	Troubleshooting			InfoCtr4150
Edit Format-Phase Record Filter screen help	Screens			InfoCtr4150
Edit Global Field errors	Troubleshooting			InfoCtr4150
Edit Global Field screen help	Screens			InfoCtr4150
Edit Group errors	Troubleshooting			InfoCtr4150
Edit Group screen help	Screens			InfoCtr4150
Edit Logical File errors	Troubleshooting			InfoCtr4150
Edit Logical File screen help	Screens			InfoCtr4150
Edit Logical Record errors	Troubleshooting			InfoCtr4150
Edit Logical Record (Assoc. Log. Files tab) screen help	Screens			InfoCtr4150
Edit Logical Record (LR Fields tab) screen help	Screens			InfoCtr4150
Edit Logical Record (LR Properties tab) screen help	Screens			InfoCtr4150
Edit Lookup Path errors	Troubleshooting			InfoCtr4150
Edit Lookup Path (General tab) screen help	Screens			InfoCtr4150
Edit Lookup Path (Lookup Path Definition tab) screen help	Screens			InfoCtr4150
Edit Physical File errors	Troubleshooting			InfoCtr4150
Edit Physical File screen help	Screens			InfoCtr4150
Edit User errors	Troubleshooting			InfoCtr4150
Edit User screen help	Screens			InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Edit User-Exit Routine errors	Troubleshooting			InfoCtr4150
Edit User-Exit Routine screen help	Screens			InfoCtr4150
Edit View (View Editor tab) screen help	Screens			InfoCtr4150
Edit View (View Properties, Extract Phase tab) screen help	Screens			InfoCtr4150
Edit View (View Properties, Format Phase tab) screen help	Screens			InfoCtr4150
Edit View (View Properties, General tab) screen help	Screens			InfoCtr4150
Edit View (View Properties, Header/ Footer tab) screen help	Screens			InfoCtr4150
Edit View errors	Troubleshooting			InfoCtr4150
Edit View Folder errors	Troubleshooting			InfoCtr4150
Edit View Folder screen help	Screens			InfoCtr4150
Empty "Deleted Views" folder	Admin Guide		Top 3	InfoCtr4150
Environment Checker errors	Troubleshooting			InfoCtr4150
Environment Checker screen help	Screens			InfoCtr4150
Environments - advanced overview	Overviews		Top 3	InfoCtr4150
Environments overview	Overviews		Top 3	InfoCtr4150
Examples: COLUMN and COL.nnn statements	Logic text types			InfoCtr4150
Examples: COLUMN statements	Logic text types			InfoCtr4150
Examples: IF with COLUMN and COL.nnn statements	Logic text types			InfoCtr4150
Examples: IF with COLUMN statements	Logic text types			InfoCtr4150
Examples: IF with SELECT	Logic text types			InfoCtr4150
Examples: IF with SKIP	Logic text types			InfoCtr4150
Examples: SELECTIF statements	Logic text types			InfoCtr4150
Examples: SKIPIF statements	Logic text types			InfoCtr4150
Examples: WRITE statements	Logic text types			InfoCtr4150
Export metadata overview	Overviews		Top 3	InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Export Utility errors	Troubleshooting			InfoCtr4150
Export Utility screen help	Screens			InfoCtr4150
Exporting metadata	Admin Guide		Top 3	InfoCtr4150
FAQ				InfoCtr4150
Finding and replacing logic text	Admin Guide		Top 3	InfoCtr4150
Find/Replace Logic Text errors	Troubleshooting			InfoCtr4150
Find/Replace Logic Text screen help	Screens			InfoCtr4150
Finding all environments for a particular a metadata item name	Admin Guide	General Users Guide	Top 3	InfoCtr4150
Finding screen help	Admin Guide	General Users Guide	Top 3	InfoCtr4150
General users guide	General Users Guide		Top 3	InfoCtr4150
General users START HERE	General Users Guide		Top 3	InfoCtr4150
Generating reports on metadata	General Users Guide		Top 3	InfoCtr4150
Global fields overview	Overviews		Top 3	InfoCtr4150
Glossary				InfoCtr4150
Group Membership errors	Troubleshooting			InfoCtr4150
Group Membership screen help	Screens			InfoCtr4150
Group Permissions By Environment (Component Security section) screen help	Screens			InfoCtr4150
Group Permissions By Environment (Environment, Assoc. Groups sections) screen help	Screens			InfoCtr4150
Group Permission By Environment messages	Troubleshooting			InfoCtr4150
Group Permissions (Component Security section) screen help	Screens			InfoCtr4150
Group Permissions (Groups, Assoc. Environs sections) screen help	Screens			InfoCtr4150
Group Permission messages	Troubleshooting			InfoCtr4150
Groups - advanced overview	Overviews		Top 3	InfoCtr4150
Groups overview	Overviews		Top 3	InfoCtr4150
How do I generate a Dependency Checker Report?				InfoCtr4150

Topic	PDF	PDF	PDF	PDF
How do I generate an Environment Checker Report?				InfoCtr4150
How do I generate an Environment Security Report?				InfoCtr4150
How do I generate a Logical Record Report?				InfoCtr4150
How do I generate a Lookup Path Report?				InfoCtr4150
How do I generate a View Column PIC Report?				InfoCtr4150
How do I generate a View Column Report?				InfoCtr4150
How do I generate a View Properties Report?				InfoCtr4150
How many address spaces does SAFR use?				InfoCtr4150
How many processes per address space?				InfoCtr4150
How many SAFR passes will there be?				InfoCtr4150
How scalable is SAFR in z/OS?				InfoCtr4150
I want to see more logic text for all rows in the Find/Replace Logic Text screen				InfoCtr4150
Import metadata overview	Overviews		Top 3	InfoCtr4150
Import Utility screen help	Screens			InfoCtr4150
Import Utility errors	Troubleshooting			InfoCtr4150
Importing metadata	Admin Guide		Top 3	InfoCtr4150
Logging into the SAFR Workbench	General Users Guide		Top 3	InfoCtr4150
Logic text 1: Extract Record Filter	Logic text types			InfoCtr4150
Logic text 1: Extract Record Filter overview	Overviews		Top 3	InfoCtr4150
Logic text 2: Extract Column Assignment	Logic text types			InfoCtr4150
Logic text 2: Extract Column Assignment overview	Overviews		Top 3	InfoCtr4150
Logic text 3: Format Column Calculations	Logic text types			InfoCtr4150
Logic text 3: Format Column Calculations overview	Overviews		Top 3	InfoCtr4150
Logic text 4: Format Record Filter	Logic text types			InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Logic text 4: Format Record Filter overview	Overviews		Top 3	InfoCtr4150
Logic text diagrams 1: Extract Record Filter	Logic text diagrams			InfoCtr4150
Logic text diagrams 2: Extract Column Assignment	Logic text diagrams			InfoCtr4150
Logic text diagrams 3: Format Column Calculation	Logic text diagrams			InfoCtr4150
Logic text diagrams 4: Format Record Filter	Logic text diagrams			InfoCtr4150
Logic Text Helper errors	Troubleshooting			InfoCtr4150
Logic Text Helper screen help	Screens			InfoCtr4150
Logic text overview	Overviews		Top 3	InfoCtr4150
Logic Text Validation Errors message help	Troubleshooting			InfoCtr4150
Logic Text Validation Errors screen help	Screens			InfoCtr4150
Logic text: summary diagrams	Logic text diagrams			InfoCtr4150
Logic text: syntax	Logic text syntax			InfoCtr4150
Logical files overview	Overviews		Top 3	InfoCtr4150
Logical records overview	Overviews		Top 3	InfoCtr4150
Lookup paths overview	Overviews		Top 3	InfoCtr4150
Metadata overview	Overviews		Top 3	InfoCtr4150
Metadata - advanced overview	Overviews		Top 3	InfoCtr4150
Migrate metadata overview	Overviews		Top 3	InfoCtr4150
Migrating metadata	Admin Guide		Top 3	InfoCtr4150
Migration Utility errors	Troubleshooting			InfoCtr4150
Migration Utility screen help	Screens			InfoCtr4150
Modifying control records	Admin Guide		Top 3	InfoCtr4150
Modifying Environments	Admin Guide		Top 3	InfoCtr4150
Modifying global fields	Admin Guide		Top 3	InfoCtr4150
Modifying group membership	Admin Guide		Top 3	InfoCtr4150
Modifying group permissions by environment	Admin Guide		Top 3	InfoCtr4150
Modifying group permissions by group	Admin Guide		Top 3	InfoCtr4150
Modifying groups	Admin Guide		Top 3	InfoCtr4150
Modifying logical files	Admin Guide		Top 3	InfoCtr4150
Modifying logical records	Admin Guide		Top 3	InfoCtr4150
Modifying lookup paths	General Users Guide		Top 3	InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Modifying own user account	General Users Guide		Top 3	InfoCtr4150
Modifying physical files	Admin Guide		Top 3	InfoCtr4150
Modifying user-exit routines	Admin Guide		Top 3	InfoCtr4150
Modifying users	Admin Guide		Top 3	InfoCtr4150
Modifying view folders	Admin Guide		Top 3	InfoCtr4150
Modifying views	General Users Guide		Top 3	InfoCtr4150
New Control Record errors	Troubleshooting			InfoCtr4150
New Control Record screen help	Screens			InfoCtr4150
New Environment errors	Troubleshooting			InfoCtr4150
New Environment screen help	Screens			InfoCtr4150
New Global Field errors	Troubleshooting			InfoCtr4150
New Global Field screen help	Screens			InfoCtr4150
New Group errors	Troubleshooting			InfoCtr4150
New Group screen help	Screens			InfoCtr4150
New Logical File errors	Troubleshooting			InfoCtr4150
New Logical File screen help	Screens			InfoCtr4150
New Logical Record errors	Troubleshooting			InfoCtr4150
New Logical Record (Assoc. Log. Files tab) screen help	Screens			InfoCtr4150
New Logical Record (LR Fields tab) screen help	Screens			InfoCtr4150
New Logical Record (LR Properties tab) screen help	Screens			InfoCtr4150
New Lookup Path errors	Troubleshooting			InfoCtr4150
New Lookup Path (General tab) screen help	Screens			InfoCtr4150
New Lookup Path (Lookup Path Definition tab) screen help	Screens			InfoCtr4150
New Physical File errors	Troubleshooting			InfoCtr4150
New Physical File screen help	Screens			InfoCtr4150
New User errors	Troubleshooting			InfoCtr4150
New User screen help	Screens			InfoCtr4150
New User-Exit Routine errors	Troubleshooting			InfoCtr4150
New User-Exit Routine screen help	Screens			InfoCtr4150
New View (View Editor tab) screen help	Screens			InfoCtr4150

Topic	PDF	PDF	PDF	PDF
New View (View Properties, Extract Phase tab) screen help	Screens			InfoCtr4150
New View (View Properties, Format Phase tab) screen help	Screens			InfoCtr4150
New View (View Properties, General tab) screen help	Screens			InfoCtr4150
New View (View Properties, Header/ Footer tab) screen help	Screens			InfoCtr4150
New View errors	Troubleshooting			InfoCtr4150
New View Folder errors	Troubleshooting			InfoCtr4150
New View Folder screen help	Screens			InfoCtr4150
Opening a log file	General Users Guide		Top 3	InfoCtr4150
Overviews	Overviews		Top 3	InfoCtr4150
Parallelism overview	Overviews		Top 3	InfoCtr4150
Performance Engine (PE) overview	Overviews		Top 3	InfoCtr4150
Physical files overview	Overviews		Top 3	InfoCtr4150
Pipes overview	Overviews		Top 3	InfoCtr4150
Return to login	General Users Guide		Top 3	InfoCtr4150
Rules for all logic text	Logic text diagrams	Logic text syntax	Logic text types	InfoCtr4150
SAFR Connection Manager errors	Troubleshooting			InfoCtr4150
SAFR Connection Manager screen help	Screens			InfoCtr4150
SAFR Login errors	Troubleshooting			InfoCtr4150
SAFR Login screen help	Screens			InfoCtr4150
SAFR optimization overview	Overviews		Top 3	InfoCtr4150
SAFR overview - START HERE	Overviews		Top 3	InfoCtr4150
SAFR phases overview	Overviews		Top 3	InfoCtr4150
Searching lists of metadata	Admin Guide	General Users Guide	Top 3	InfoCtr4150
Sort Key Properties errors	Troubleshooting			InfoCtr4150
Sort Key Properties screen help	Screens			InfoCtr4150
Sort Key Titles errors	Troubleshooting			InfoCtr4150
Sort Key Titles screen help	Screens			InfoCtr4150
Syntax: BEGINS_WITH	Logic text syntax			InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Syntax: COL.nnn in Format Record Filter	Logic text syntax			InfoCtr4150
Syntax: COLUMN and COL.nnn in Extract Column Assignment	Logic text syntax			InfoCtr4150
Syntax: COLUMN and COL.nnn in Format Column Calculations	Logic text syntax			InfoCtr4150
Syntax: COLUMN and COL.nnn statements	Logic text syntax			InfoCtr4150
Syntax: CONTAINS	Logic text syntax			InfoCtr4150
Syntax: ENDS_WITH	Logic text syntax			InfoCtr4150
Syntax: function ALL	Logic text syntax			InfoCtr4150
Syntax: function BATCHDATE	Logic text syntax			InfoCtr4150
Syntax: function CURRENT	Logic text syntax			InfoCtr4150
Syntax: function DATE	Logic text syntax			InfoCtr4150
Syntax: function DAYSBETWEEN	Logic text syntax			InfoCtr4150
Syntax: function FISCALDAY	Logic text syntax			InfoCtr4150
Syntax: function FISCALMONTH	Logic text syntax			InfoCtr4150
Syntax: function FISCALPERIOD	Logic text syntax			InfoCtr4150
Syntax: function FISCALQUARTER	Logic text syntax			InfoCtr4150
Syntax: function FISCALYEAR	Logic text syntax			InfoCtr4150
Syntax: function ISFOUND	Logic text syntax			InfoCtr4150
Syntax: function ISNOTFOUND	Logic text syntax			InfoCtr4150
Syntax: function ISNOTNULL	Logic text syntax			InfoCtr4150
Syntax: function ISNOTNUMERIC	Logic text syntax			InfoCtr4150
Syntax: function ISNOTSPACES	Logic text syntax			InfoCtr4150
Syntax: function ISNULL	Logic text syntax			InfoCtr4150
Syntax: function ISNUMERIC	Logic text syntax			InfoCtr4150
Syntax: function ISSPACES	Logic text syntax			InfoCtr4150
Syntax: function MONTHSBETWEEN	Logic text syntax			InfoCtr4150
Syntax: function PRIOR	Logic text syntax			InfoCtr4150
Syntax: function REPEAT	Logic text syntax			InfoCtr4150
Syntax: function RUNDAY	Logic text syntax			InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Syntax: function RUNMONTH	Logic text syntax			InfoCtr4150
Syntax: function RUNPERIOD	Logic text syntax			InfoCtr4150
Syntax: function RUNQUARTER	Logic text syntax			InfoCtr4150
Syntax: function RUNYEAR	Logic text syntax			InfoCtr4150
Syntax: function YEARSBETWEEN	Logic text syntax			InfoCtr4150
Syntax: functions	Logic text syntax			InfoCtr4150
Syntax: functions Q1, Q2, Q3 and Q4	Logic text syntax			InfoCtr4150
Syntax: IF statements	Logic text syntax			InfoCtr4150
Syntax: IF statements in Extract Column Assignment	Logic text syntax			InfoCtr4150
Syntax: IF statements in Extract Record Filter	Logic text syntax			InfoCtr4150
Syntax: IF statements in Format Column Calculations	Logic text syntax			InfoCtr4150
Syntax: IF statements in Format Record Filter	Logic text syntax			InfoCtr4150
Syntax: LIKE	Logic text syntax			InfoCtr4150
Syntax: lookup paths	Logic text syntax			InfoCtr4150
Syntax: MATCHES	Logic text syntax			InfoCtr4150
Syntax: SELECT and SELECTIF statements in Extract Record Filter	Logic text syntax			InfoCtr4150
Syntax: SELECT and SELECTIF statements in Format Record Filter	Logic text syntax			InfoCtr4150
Syntax: SELECT, SELECTIF, SKIP and SKIPIF statements	Logic text syntax			InfoCtr4150
Syntax: SKIP and SKIPIF statements in Extract Record Filter	Logic text syntax			InfoCtr4150
Syntax: SKIP and SKIPIF statements in Format Record Filter	Logic text syntax			InfoCtr4150
Syntax: string comparison	Logic text syntax			InfoCtr4150
Syntax: WRITE statements	Logic text syntax			InfoCtr4150
Syntax: WRITE statements in Extract Column Assignment	Logic text syntax			InfoCtr4150
Syntax: WRITE statements in Extract Record Filter	Logic text syntax			InfoCtr4150
Tokens overview	Overviews		Top 3	InfoCtr4150
User-exit routines overview	Overviews		Top 3	InfoCtr4150

Topic	PDF	PDF	PDF	PDF
Users overview	Overviews		Top 3	InfoCtr4150
View Activation Errors message help	Troubleshooting			InfoCtr4150
View Activation Errors screen help	Screens			InfoCtr4150
View folders overview	Overviews		Top 3	InfoCtr4150
View Source Properties errors	Troubleshooting			InfoCtr4150
View Source Properties screen help	Screens			InfoCtr4150
Views - advanced overview	Overviews		Top 3	InfoCtr4150
Views overview	Overviews		Top 3	InfoCtr4150
WE log file overview	Overviews		Top 3	InfoCtr4150
WE Security overview	Overviews		Top 3	InfoCtr4150
What are the keyboard shortcuts?	Admin Guide	General Users Guide	Top 3	InfoCtr4150
What does SAFR stand for?				InfoCtr4150
What does WE stand for?				InfoCtr4150
What metadata do I want to see?	General Users Guide		Top 3	InfoCtr4150
What's new in this Information Center				InfoCtr4150
Where is the WE log file?	Admin Guide	General Users Guide	Top 3	InfoCtr4150
Why use SAFR?				InfoCtr4150
Workbench overview	Overviews		Top 3	InfoCtr4150
XML structure for metadata overview	Overviews		Top 3	InfoCtr4150

Appendix: Notices

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

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

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

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

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 Corporation
Software Interoperability Coordinator, Department 49XA

3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

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 Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

Trademarks

IBM, the IBM logo, and 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.

Index

W

X

Z

Printed in USA