

This appendix expands several sections and adds new sections to *SAP Information Lifecycle Management: The Comprehensive Guide* by Iwona Luther, Nicole Fernandes, Frank Buschle, and Carsten Pluder (SAP PRESS, 2020).

2 Chapter 2: Basic Functions in SAP ILM

This section expands coverage of Section 2.3.5: Object Assignment.

If you want to use the new transaction `ILMARA_N`, note that you can use it to create, change, and delete audit areas. To assign ILM objects to an audit area, you must use a separate transaction `ILMARA_OBJ_N` (Assign ILM object to Audit Area). For the sake of simplicity, we only mention transaction `ILMARA` in this book and its appendix in the context of audit areas. You decide whether you want to use transaction `ILMARA_N` and `ILMARA_OBJ_N` as an alternative. Note that as of release SAP S/4HANA 2023 FPS02, the SAP Reference IMG contains only the new transactions.

2.5 Policies

This section expands coverage of Section 2.5: Policies.

As of SAP S/4HANA 2023 FPS02, the new transaction `IRM_POLICY` (ILM Policies) is available as an alternative. (It was introduced because it replaces transaction `IRMPOL` as of SAP S/4HANA Cloud Public Edition 2408 (and only there). In SAP S/4HANA Cloud Public Edition, you can only transport your settings using the new method.)

For the sake of simplicity, we will only mention transaction `IRMPOL` in the context of rule set maintenance in this book and its appendix. However, you can use both transactions. Note that only the new transaction is displayed in the IMG as of SAP S/4HANA 2023 FPS02.

2.5.3 Special Characters in Condition Fields: Transaction `IRM_CUST_CSS`

This section expands coverage of Section 2.5.3 Special Characters in Condition Fields: Transaction `IRM_CUST_CSS`.

As of SAP S/4HANA 2023 FPS02, the following new transactions are available as an alternative to Transaction `IRM_CUST_CSS`:

- `V_LRM_HND_CS_GEN` (Handling Special Characters (Generic)). You can map special characters across ILM objects here.
- `VC_IRM_CSS` (Maintain Customer Specific Settings for an ILM Object). You can map special characters for a selected ILM object.

2.5.4 Creating Rules in a Policy

This section expands coverage of Section 2.5.4: Creating Rules in a Policy.

Let's discuss what to do if the retention period of the data is not yet fully known?

Let's recall, that if you have successfully created an archive file, have assigned a retention period to it and have stored the file in an ILM-enabled storage system, you can extend the retention period retroactively, but you can't shorten it (see also Section 2.6.6, "Conversion of Archive Files").

What to do, if you want to archive data under the control of SAP ILM, although its retention period has not yet been finally clarified. It is advisable not to enter a retention period that is too long (since it cannot be shortened). We recommend that you enter a retention period that results in a date that is later than or the same as 12/31/9999 when calculating the date as of which the archive file can be destroyed. Example: You archive data created in 2024 and have entered in the corresponding policy a retention period of 8,000 years which shall start with the creation year of the data. In this case, ILM enters the value "unknown" as the expiration date for the archive file. You can set this "date" to a correct date using the conversion in Transaction `ILM_CHANGE_RET` (Change Expiration Date, see Section 2.7.5, "Transaction `ILM_CHANGE_RET`") as soon as you know the final retention period of the data and have entered it in the corresponding policy.

We would like to draw your attention to how you can check whether you are using an ILM object correctly. As of the releases specified in SAP Note 3318721 (Enabling the usability enhancements for ILM processes through preconditioned checks), Transaction `ILM_CONSTCY_CHK` is available for this purpose. For example, it displays a red traffic light if an ILM object is assigned to an audit area for which there is no live policy.

2.5.6 Facilitating Rule Maintenance with Rule and Object Groups

This section expands coverage of Section 2.5.6: Facilitating Rule Maintenance with Rule and Object Groups.

As of SAP S/4HANA 2023 FPS02, the new transaction `IRM_CUST_OBJGRP` (ILM: Manage Object Groups) is available as an alternative to transaction `IRM_CUST_CSS`. It was created due to a design error of transaction `IRM_CUST_CSS`. As you may have noticed, when you create object groups or rule groups, you specify the associated policy category, which is then named at the top of the screen, see for example: see in the book **Figure 2.31** Transaction `IRM_CUST_CSS`: Create Object Group") However, this information is missing in the last step (see in the book **Figure 2.32** Transaction `IRM_CUST_CSS`: Assign SAP ILM Objects). This could lead to problems (e.g. to overwriting of your settings) if you want to use the same names for object or rule groups for the same ILM objects both for residence and retention rules.

2.5.7 Overview of Your Policies

This section expands coverage of Section 2.5.7: Overview of Your Policies.

Another option for you to get an overview of your policies is transaction `IRMPOL_OVERVIEW` (Overview of ILM Rules). You can use this transaction to get a comfortable overview of your rules.

2.6 WebDAV, BC-ILM Certification, and SAP ILM Store Browser

This section expands coverage of Section 2.6: WebDAV, BC-ILM Certification, and SAP ILM Store Browser.

Further Information About BC-ILM Certification

For a list of certified vendors, see. www.sap.com/icc. There, choose **Browse the certification directory** and click in the upper part of the window **Search for Solution Providers**. In the free text search, type for example the certification scenario “BC-ILM 3.1”, or “S/4-BC-ILM 1.0”.

Currently, no new certification contracts are issued for the certification scenario BC-ILM 3.1 (non-SAP S/4HANA). Certificates that have already been issued and are valid are still displayed. Today, an SAP partner can (re-)certify (new) storage systems only according to the certification scenario S/4-BC-ILM 1.0.

2.6.6 Conversion of Archive Files

This section expands coverage of Section 2.6.6: Conversion of Archive Files.

In case of an archive run that has already been created under the control of SAP ILM and for which new files were created by the file conversion, note that replaced archive files that are subject to a minimum retention period that has not yet expired cannot be destroyed automatically. Here you have to wait until the minimum retention period has expired. For replaced archive files with an undefined minimum retention period, SAP ILM automatically sets the minimum retention period to the date of the day on which the conversion took place. You can use this to destroy such files on the following day.

2.7 Data Destruction Functions

2.7.1 Methods of Data Destruction

This section expands coverage of Section 2.7.1 Methods of Data Destruction.

SAP ILM can be used to destroy:

- Data from the database
- Archive Files

We can also differentiate between the destruction of leading objects or linked documents, that is, attachments to leading objects.

You can use SAP ILM to destroy the following types of attachments:

- ArchiveLinkk (entries in the **TOAnn** tables).
- Generic Object Services (GOS),
- Business Document Service (BDS)
- Document Management Attachments (DMS/DMS)

You can also use SAP ILM to destroy print lists (entries in table **TOADL**).

2.7.2 Destruction from the Database via an Archiving Object

This section expands coverage of Section 2.7.2: Destruction from the Database via an Archiving Object.

Use transaction ILM_DESTRUCTION or SARA?

If you want to destroy data from the database using an archiving object, both transaction `ILM_DESTRUCTION` and `SARA` lead to the same target. You can therefore call transaction `SARA` and enter the name of the archiving object. However, you can also call transaction `ILM_DESTRUCTION`. In the **Type of Data to Be Destroyed** group box, select **Data from the Database** and enter the name of the ILM object that is assigned to the archiving object. When you then maintain the variant of the archive write program, it only offers the ILM action Data Destruction – the other ILM actions are grayed out. This corresponds to your original command for destroying data from the database.

New Subsection in Chapter 2, Section 2.7: Destruction of User Master Data

This is a new section, not present in the original edition.

In this section, we would like to draw your attention to a fairly new data destruction object with the name `IDENTITY` (Complete Deletion of User Master Data) and the ILM object of the same name. You can use these objects to destroy user master data (user master record) in your system. Their destruction with SAP ILM is only possible if the user master data has previously been blocked. For details, see further down in Section 4.4.8 “Blocking of User Master Data”.

If you have successfully blocked a user master record, you can destroy it. The destruction of blocked user master data is therefore the last stage of the lifecycle of user master data shown in Figure 2.1.

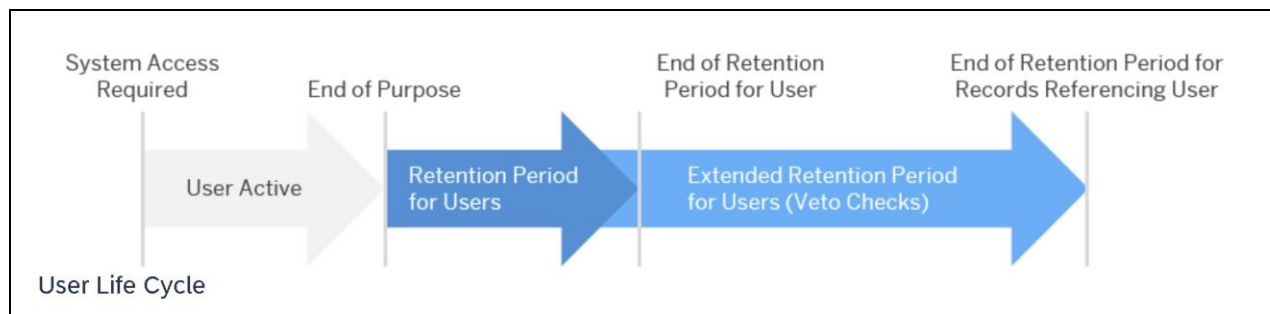


Figure 2.1: User Life Cycle (<https://help.sap.com/>)

You can define the retention periods using the ILM object `IDENTITY`. If required, this can be done for each user group or for the address type of the user. Which of these two fields are offered to you as condition fields when defining a policy in transaction `IRMPOL` depends on whether you are using SAP ERP, SAP S/4HANA, SAP S/4HANA Cloud Private Edition, or SAP S/4HANA Cloud Public Edition.

User master data should not be destroyed as long as an application references this user in its data records. The user can be referenced in an application, for example, as the person who created or last changed application data. All applications that reference user data in their data records must therefore create so-called veto checks for the data destruction object `IDENTITY`. If you execute a destruction run for the data destruction object `IDENTITY`, the following occurs:

1. SAP ILM checks for blocked user master data. (These are held in the blocked area.)

2. If this exists, SAP ILM checks for each user master record whether the retention time you have defined has elapsed.
3. If this is the case, all created veto checks are run. (SAP already delivers some veto checks in the standard system.) This enables each application to check for itself whether it uses data records that reference the user to be destroyed.
4. If this is the case, the application in question triggers a veto.

As a result, the check for the destructibility of the user master data has ended – it cannot yet be destroyed. At this point, you could correctly note that the user master data inherits the retention time from the data records that reference it. This is exactly what the extended retention period for users (veto checks) in Figure 2.1 refers to.

To optimize performance, the system notes which applications have processed the user master data without a veto. The next time you start a destruction run for this user, these applications are no longer asked. It continues with the application that last vetoed. If the user master data can be destroyed this time, the user is deleted. This includes the corresponding entries in the blocked area and in transaction **SUIM** (User Information System). As a result, it is also possible to create a user with this user name again.

Finally, the destruction of the user data is logged in the Security Audit Log (FU8). In addition to the creation of application logs or lists (depending on your decision on the selection screen of the data destruction program).

Note: Categories for Veto Checks

The veto checks just described are assigned to categories. The category determines the sequence in which the checks are processed. For example, the highest category 1 is intended for applications with which users perform development activities in the system. For reasons of system integrity, this category is regarded as the strictest category and is processed at the start.

For more information about the different categories, see the SAP documentation on destroying user master data, or see SAP Notes 2826256 (Deletion and blocking of user master records), 2836524 (Configuration Standard Veto Checks for Blocking and Deleting User Master Records), 550718 (FAQ: Deleting users), and 2826256 (Deletion and blocking of user master records).

The concept described above stipulates that a user master data that is still referenced in an application table is destroyed. It can happen if there is no veto check for this reference. Since SAP delivers the veto checks for the SAP applications, this usually means that such a reference was considered uncritical by SAP.

You can find the list of veto checks delivered by SAP in the SAP documentation. The standard veto checks delivered by SAP include, for example:

- Checks for entries in tables for generic change documents (**CHANGEDOCS**).
- Checks for entries in tables for generic audit trails (**DBTABLOG**).
- Checks for entries in tables for generic desktop office integrations (**SAPOFFICE**).
- Checks for entries in tables for generic user and role administration (**USERADMIN**).

Generic veto checks for SAP developments are delivered in table `USRVETOTABLES` (SAP: Usage of the User in Tables). These checks are intended for applications/tables other than those mentioned above. As a customer, you can also add veto checks by entering the table to be checked and the field to be checked in the registration table `USRVETOTABLES_C` (Customer: Usage of the User in Tables). Use this registration table for your custom code as well as for SAP developments. (The table `USRVETOTABLES` may only be processed by SAP. If you use this table, there is a risk that your entries e.g. may be lost during a release upgrade.)

If the generic veto checks do not meet your requirements, you can program your own checks. To do this, create an implementation for the method `IF_IDENTITY_VETO_DELETE~CHECK_DELETE` and enter the class in which you implement this method together with the appropriate category in the table `USRVETO`.

You are certainly wondering how the system can ensure that archived data is also taken into account during the veto checks. For the new development required for this, see SAP Note 2837367 (Veto checks for references to user master records in archived data). Technically, this new development for the veto checks for references to user master records in archived data is based on an index that is built during the archiving deletion run.

If you have carried out archiving before you have activated the "Lock and Delete User Master Records" function described in this section, you must subsequently set up this index for old, relevant archive files. The program `RSARCH_USR_VETO_INDEX` is available for this purpose. Here you can set up the required index data for a specific archiving object or for a specific archiving session. (From a technical point of view, index entries are created in the `ADMI_USR_VETO*` tables.)

Note that for each archive file, the system remembers when it last built the index. Archive files that have already been processed are not taken into account by the program during later index build runs. If required, you can override this behavior by entering a desired date in the **Build To** field.

Data destruction object IDENTITY and legal holds

It is also worth mentioning that the destruction of user data is not possible if the user is referenced in instances of BOR object types (e.g. in documents) that have been included in a legal hold (see section 3.6, "Legal Case Management"). It does not matter whether the BOR object type was included manually or using the E-Discovery function. This also applies if these instances have been archived. For more information about this, see SAP Note 2910459 (Veto checks for references to user master records in archived data associated with the legal case). Destruction is not possible as long as the legal case does not yet have the status 50 (Case Closed).

2.7.6 Enhancement Spot ES_ILM_DESTRUCTION

This section expands coverage of Section 2.7.6: Enhancement Spot ES_ILM_DESTRUCTION.

Note that the BAdI `BADI_ILM_DESTR_FILE_VETO` is called when destroying archive files in transaction `ILM_DESTRUCTION` (Data Destruction). It is not called if you start an archiving session with the ILM action Data Destruction. In this case, you should do the following:

- If you are responsible for the archiving object yourself, you should carry out the additional checks in the coding of the archive write program.

- If you are not responsible for the archiving object, you should implement method `OPERATE_ON_DESTRUCTION` of BAdI `BADI_ILM_PRE_DESTR_OBJ_ACTION` (Clean-Up Activities Prior to Destroying a Data Object with Data Archiving). We describe it in more detail in this section.

Make sure that your implementation has a filter so that it is not called for all archive files of all available archiving objects, but only for those of your archiving object.

From the information about the BAdIs of the enhancement spot `ES_ILM_DESTRUCTION`, the following procedure results when destroying archive files from the certified storage:

1. Call of the BAdI method `BADI_ILM_DESTR_FILE_VETO-CHECK_BEFORE_DESTRUCTION`. If there is a veto, further processing is terminated.
2. Call of the BAdI method. `BADI_ILM_PRE_DESTR_FILE_ACTION-OPERATE_ON_DESTRUCTION`
3. Destruction of archive file in production mode.
4. Call of BAdI method `BADI_ILM_DESTR_WITH_ARKEY-DESTROY_ADD_INFO` during processing in production mode.
5. Call of BAdI method `BADI_ILM_PRE_DESTR_FILE_ACTION-OPERATE_AFTER_FILE_DESTRUCTION`

As mentioned above, we want to describe in this section the BAdI `BADI_ILM_PRE_DESTR_OBJ_ACTION` (Clean-Up Activities Prior to Destroying a Data Object with Data Archiving) in more details. It has the following methods:

- `OPERATE_ON_DESTRUCTION` (Preparation for Clean Up Activity in a Remote System)
- `OPERATE_ON_FILE_DESTRUCTION` (Follow-Up of Cleanup Activity in Remote System)

In contrast to the BAdIs mentioned above, this one is called when you start an archiving run with the ILM action Data Destruction. This enables you to delete corresponding data (e.g. from the remote system). The method `OPERATE_ON_DESTRUCTION` is called in the delete program by the function module `ARCHIVE_GET_NEXT_OBJECT` (Read Data Object from Archive File) for each data object to be destroyed immediately before the destruction. The method `OPERATE_ON_FILE_DESTRUCTION` (Follow-Up of Cleanup Activity in Remote System) is called in the delete program for each data object to be destroyed immediately after the destruction of the archive file.

As mentioned in the context of vetos during the destruction of archive files, the method `OPERATE_ON_DESTRUCTION` is also available if you want to prevent the destruction of data in the case of the ILM action Data Destruction. A use case for this would be that the application (business process) A needs the data from an application B even though the retention time has expired. The corresponding check cannot already be performed in the archive write program (because application B is not responsible for archiving object A, or because A does not offer any check exits for its archive write program) and must therefore take place in the delete program of archiving object A. As described in the SAP documentation, application B must trigger the exception `CX_ILM_BADI` with a suitable message in the method `OPERATE_ON_DESTRUCTION`. This stops the processing of the delete program. The archive file remains in the status Write Completed and does not switch to Deletion Completed. (At this point, you can also see the advantage of performing such additional checks in the archive write program, because then individual object instances and not the entire archive file can be excluded from processing.)

Let's complete the information about the enhancement spot `ES_ILM_DESTRUCTION` using the BAdI `BADI_ILM_DESTRUCTION_OBJECT` (Veto Check and Destruction of Additional Information for ILM Data Destruction with Data Destruction Objects). As you can see from its description, this BAdI is used for data destruction objects (and not for archiving objects such as the BAdIs discussed above). It provides two methods:

- `CHECK_VETO_BEFORE_DESTRUCTION`
- `DESTROY_ADDITIONAL_INFO`

You can use the first method to veto the destruction of data. You can use the second method to destroy additional information (e.g. entries from related tables in the customer namespace). SAP itself uses the second method in data destruction programs of SAP data destruction objects to destroy related attachments.

New Subsection in Chapter 2, Section 2.7: Examples of Destroying Master Data

This is a new section, not present in the original edition.

We want to round off the knowledge about the data destruction functions of SAP ILM with some examples. In the following sections, we select the following four master data as examples: the customer, the supplier, the contact person, and the central business partner. Figure 2.2 shows you the four related ILM objects in transaction `ILMARA` (Audit Area Processing) for which you need to define suitable, productive retention periods.

Policy Category: Retention Rules					
Assignment of Objects to Audit Area					
Select Tables and Fields		Show Checksums			
O...	Object Category	ILM Object		Description	Object Assignment
O...	SAP Business Suite	AL_DOCUMENTS		Documents Archived Using ArchiveLink	<input checked="" type="checkbox"/>
O...	SAP Business Suite	BC_SFLIGHT		Sample ILM Object (Flight Model)	<input checked="" type="checkbox"/>
O...	SAP Business Suite	BC_SFLIGHT_DESTRUCTION		Sample Object for Data Destruction	<input checked="" type="checkbox"/>
O...	SAP Business Suite	CA_BUPA	←	SAP Business Partners	<input checked="" type="checkbox"/>
O...	SAP Business Suite	CHANGEDOCU		Change Documents	<input checked="" type="checkbox"/>
O...	SAP Business Suite	FI_ACCKNVK	←	Contact (Data Destruction)	<input checked="" type="checkbox"/>
O...	SAP Business Suite	FI_ACCPAYB	←	Vendor Master Data	<input checked="" type="checkbox"/>
O...	SAP Business Suite	FI_ACCRECV	←	Customer Master Data	<input checked="" type="checkbox"/>
O...	SAP Business Suite	FI_DOCUMNT		Financial Accounting Documents	<input checked="" type="checkbox"/>

Figure 2.2: Transaction `ILMARA` – ILM-Objects for customer, supplier, contact person, and the central business partner

You must enter the value **Start of Retention Time** as the time reference. The start of the retention period is determined during the end of purpose check, that is, when the master data is blocked.

Destroying Customer Master Data

To destroy the customer data, use the archiving object `FI_ACCRECV` (customer master data).

In detail this means that you have to perform three separate runs in the following sequence: you must first destroy sales and distribution data (SD), then financial data (FI), and finally the general data. You select this in the **Data to Be Archived** selection field, as shown in Figure 2.3.

Variant Attributes	
Data To Be Archived	SD SD Data
Customer Master Data	
Customer	BIT665_00B
Company Code	
Sales Organization	
Selections	
Min.No.of Days in the System	

Figure 2.3: Selection screen of the archiving object `FI_ACCRECV` (customer master data)

Let's take a look at what options you have to enhance the archiving object `FI_ACCRECV` (customer master data). In the enhancement spot `ARC_FI_ACCRECV` (archiving object `FI_ACCRECV`), you can use the following BAdIs to make additional checks in the archiving of customer master data:

- `FI_ACCRECV_CHECK` (Archiving Customer Master Data (General Part): Additional Archivability Checks)
- `FI_ACCRECV_CHECK_FI` (Archiving Customer Master Data (FI Part): Additional Archivability Checks)
- `FI_ACCRECV_CHECK_SD` (Archiving Customer Master Data (SD Part): Additional Archivability Checks)

The enhancement spot `ARC_FI_ACCRECV` also offers you the following BAdIs to archive or delete entries of additional tables (e.g. tables in the customer namespace) with the archiving object `FI_ACCRECV` (customer master data):

- `FI_ACCRECV_WRITE` (Archiving of Customer Master Data (General Part): Archive Additional Tables)
- `FI_ACCRECV_WRITE_FI` (Archiving of Customer Master Data (FI Part): Archive Additional Tables)
- `FI_ACCRECV_WRITE_SD` (Archiving of Customer Master Data (SD Part): Archive Additional Tables)

If you implement the method `DELETE` (Delete Additional Data from Database) of the last 3 mentioned BAdIs, you can delete entries in additional tables. This method is called in the delete program of the archiving object `FI_ACCRECV` (customer master data) if you have started the archiving run with the ILM action Archiving or Data Destruction.

We do not recommend that you archive blocked master data, this means that you use the ILM action Archiving. (We recommend that you destroy blocked master data (this means that you use the ILM action Data Destruction). However, if you want to do this, you can write entries from additional tables to the archive if you implement the method WRITE (Write Additional Data to Archive) of the three mentioned BAdIs (`FI_ACCRECV_WRITE*`).

As you can see from the name of all six BAdIs of the enhancement spot `ARC_FI_ACCRECV`, you can consider additional data from the following three areas of customer master data: general part, FI part, and SD part. This is reflected in the parameters of the BAdI methods. For example the methods `WRITE` and `DELETE` from the BAdIs `FI_ACCRECV_WRITE*` allow you to select additional data for the general part depending on the customer number (for the FI part depending on the customer number and the company code, and for the SD part depending on the customer number and the sales organization). For more information, see the BAdI documentation.

Note: BAdI `ARC_OBJECT_ADD_TABLE`

If you use the BAdIs to archive entries from additional tables (e.g. tables in the customer namespace), you must add the names of the relevant tables to the structure definition of the archiving object. The BAdI `ARC_OBJECT_ADD_TABLE` (Adding structures to the structure definition of an archiving object) and its method `ADD_STRUCTURES` (Extend the Structure Definition of an Archiving Object) are used for this purpose.

Destroying Vendor Master Data

Next, let's look at the vendor master data. To destroy these, use archiving object `FI_ACCPAYB` (vendor master data).

Also here, you need to execute three separate destruction runs, first the materials management (MM) data, then the financial data (FI), and finally the general data (see Figure 2.4).

The screenshot displays the selection screen for the archiving object `FI_ACCPAYB`. The 'Data To Be Archived' dropdown menu is open, showing four options: 'MD General Data', 'FI FI Data', 'MD General Data', and 'MM MM Data'. The 'Vendor Master Data' section contains three input fields: 'Vendor', 'Company Code', and 'Purch. Organization', each followed by a 'to' field. The 'Selections' section at the bottom has a field for 'Min.No.of Days in the System'.

Figure 2.4: Selection screen of the archiving object `FI_ACCPAYB` (vendor master data)

You also have options to enhance the archiving object `FI_ACCPAYB` (vendor master data). In the enhancement spot `ARC_FI_ACCPAYB` (Archiving object `FI_ACCPAYB`), you can use the following BAdIs to make additional checks in the archiving of vendor master data:

- `FI_ACCPAYB_CHECK` (Supplier Master Data Archiving: Additional Check of General Part)
- `FI_ACCPAYB_CHECK_FI` (Vendor Master Data Archiving: Additional Check of Finance)
- `FI_ACCPAYB_CHECK_MM` (Vendor Master Data Archiving: Additional Check of Materials Management)

The enhancement spot `ARC_FI_ACCPAYB` also provides you with three BAdIs for archiving entries from additional tables (for example tables in the customer namespace) with the archiving object `FI_ACCPAYB`:

- `FI_ACCPAYB_WRITE` (Vendor Master Data Archiving: Archive Additional Tables of General Part)
- `FI_ACCPAYB_WRITE_FI` (Vendor Master Data Archiving: Archive Additional Tables of Finance)
- `FI_ACCPAYB_WRITE_MM` (Vendor Master Data Archiving: Archive Additional Tables of Materials Management)

If you implement the method `DELETE` (Delete additional data from database) for these BAdIs, you can delete entries from additional tables. This method is called in the delete program of the archiving object `FI_ACCPAYB` if you have started the archiving run with the ILM action Archiving or Data Destruction.

We do not recommend that you archive blocked master data. However, if you want to you can write entries from additional tables to the archive, if you use the `WRITE` method (Write additional data to the archive) of the three named BAdIs (`FI_ACCRECV_WRITE*`).

Using the `WRITE` and `DELETE` methods from the BAdIs `FI_ACCPAYB_WRITE*` as an example, you can select additional data for the general part of the vendor master data depending on the vendor number (for the FI part depending on the vendor number and the company code, and for the MM part depending on the vendor number and the purchasing organization). For more information, see the BAdI documentation.

Note: BAdI ARC_OBJECT_ADD_TABLE

If you use the BAdIs to archive entries from additional tables (e.g. tables in the customer namespace), you must add the names of the relevant tables to the structure definition of the archiving object. The BAdI `ARC_OBJECT_ADD_TABLE` (Adding structures to the structure definition of an archiving object) and its method `ADD_STRUCTURES` (Extend the Structure Definition of an Archiving Object) are used for this purpose.

Destroying Business Partner Master Data

You destroy the central business partner using the archiving object `CA_BUPA` (SAP Business Partner).

To display the selection screen for this archiving object correctly, you must first start the following programs:

- `BUPSELG0` (Generation Program for SELECT OPTIONS for Differentiation Types) and
- `BUPSELG5` (Generation Program for SELECT OPTIONS for Change Document Object)

This is because the corresponding archive write program `BUSOBARCH` (BDT: Archiving Program) uses the *Business Data Toolset* (BDT) and is used by several archiving objects.

You can also only start the archive write program from transaction **SARA** (Archive Administration) and not, for example, using transaction **SE38** (ABAP Editor) because transaction **SARA** tells the program which archiving object it is. The program then only displays the selection parameters that are intended for this archiving object.

If a business partner is a customer, you must first destroy the related data using archiving object **FI_ACCRECV** (Customer master data). This principle also applies to the following categories of business partners:

- **Suppliers**

Here, you use the archiving object **FI_ACCPAYB** (Vendor master data).

- **Contract Accounts**

Here, you use the archiving object **FI_FICA** (FI-CA: Contract Accounts).

This sequence applies to SAP S/4HANA, SAP S/4HANA Cloud Private Edition, and SAP S/4HANA Cloud Public Edition. In SAP ERP, it's exactly the other way around. A prerequisite here is that the business partner is linked to a customer/vendor using customer/vendor integration (CVI).

We recommend that you destroy the master data of customers, suppliers, and central business partners after the retention period has expired using the ILM action Data Destruction. Therefore, we do not recommend that you archive the data first because you cannot unblock archived master data if required. Therefore, using the ILM action Archiving is not a prerequisite for destroying blocked master data. Note this if you plan to archive master data.

Hinweis: Preprocessing Program Versus Blocking

If you have activated the business functions required for the EoP check and for simplified blocking and deletion with SAP ILM (see Section 4.3.1, "For Blocking of Master Data in Transaction **SPRO**"), blocking the master data is the prerequisite for its archiving or destruction. You then no longer need to schedule a preprocessing program.

Destroying Contact Person Master Data

Master data for the contact person, that is, the contact person for a customer or vendor, is destroyed together with the corresponding master data of the customer or vendor. Therefore, if you have blocked the customer or vendor completely – that is, including all contact persons – the data for the contact person will be destroyed together with the data for the customer or vendor.

You can also block and then destroy the contact person independently of the leading object (customer or vendor). The data destruction object **FI_ACCKNVK** (Contact (Data Destruction)) and the ILM object of the same name are available for this purpose.

The contact persons for a business partner are themselves created as business partners of the category Person. This means that you link several business partners with each other and do not use the object **FI_ACCKNVK** (Contact (Data Destruction)).

3 Chapter 3: Additional Functions in Retention Management

3.1 SAP ILM Notifications

This section expands coverage of Section: 3.1: SAP ILM Notifications.

Note: Possible Recipients

The list of recipients is no longer limited since SAP Note 2959645 (ILM Notifications Framework - Multiple Recipients DDIC objects). Possible recipients are documented in the SAP documentation for the ILM notifications.

The recipients are responsible for replicating the data to other systems. This means that a corresponding implementation in the SAP Business Suite or SAP S/4HANA system is a prerequisite for reading the ILM notification data and transferring it using the required communication channel.

3.6 Legal Case Management

3.6.4 Set Legal Case

This section expands coverage of Section 3.6.4: Set Legal Case.

Note: When to use button Add Objects and when E-Discovery

If you want to add more than one instance (e.g. sales order) in a legal case, or if you want to add one or more instances in a legal case together with all dependent documents (e.g. the material document, billing document, the corresponding FI and CO documents, and so on), use the *E-Discovery function*, which you call using the **E-Discovery** button. Here, the data (instances) relevant for the legal case is searched for and added to the legal case using search criteria that you specify. You specify the search criteria using a variant for a so called *E-Discovery program*.

If you want to add only one instance (e.g. sales order) in a legal case without all dependent documents, you can use the **Add Objects** button.

In previous releases, you had to enter the logical system from which the instance (the document) originated, when you clicked the **Add Objects** button to add one entity of a BOR object type in a legal case. As of the releases specified in SAP Note 2901307 (Enabling multiple origin in ILM legal hold management), this is no longer necessary and the column with the same name no longer appears in this and similar places.

Once the system has added in the legal case the entity you specified, among other things, the following information is displayed for each BOR object type entity:

- BOR object type and its short description.
- Key of the entity and the logical system to which it belongs.
- The origin. The system can determine whether the instance exists in the database or in the archive. If it was found in the archive, the **Archive Files** column shows how many archive files it was found in (usually in one). You can click on the number specified there and a dialog box appears with the names of the archive files. Note that the number of archive files found may not be displayed correctly until you display the legal case again.

- The **Attachments** column shows whether ArchiveLink attachments exist for the instance (the document). The system searches for these attachments automatically and does not require any further information from you because the key of the ArchiveLink tables consists of the BOR object type and key that have already been displayed. As a result, the legal hold automatically extends to the corresponding attachments.
- Which user and when has added the instance (the document).
- Whether the instance was added manually or using an **E-Discovery** button.

Note: Class CL_EDISC_REF_COLLECTOR

If the alternatives displayed in the input help (F4) for the **Program** field when starting e-Discovery are not sufficient, contact SAP. Alternatively, you can use the class **CL_EDISC_REF_COLLECTOR** (Collect References to Business Objects in Global Memory) to implement customer-specific E-Discovery reports. This class transfers the BOR object type instances found using the E-Discovery report to the corresponding legal case. For more information about the correct use of this class, see the reports that the input help (F4) display for the **Program** field.

If you have implemented customer-specific E-Discovery reports and want to see them listed in the input help, you can do this using the IMG activity **Register E-Discovery Selection Programs** (in the SAP Reference IMG under **Information Lifecycle Management • Retention Management**)

In the **Variant** field, you enter a variant for the selected program. You can use the **Edit** button to the right of the field to create or change a variant. You can specify exactly which documents for which organizational units, and so on that you want to add in the legal case.

You select the **Background Processing** checkbox if you want the system to perform all work as a background job after you confirm the popup window.

If you select this checkbox, the **Server Group** and **Max Processes** fields are ready for input. In the **Server Group** field, you can select an application server on which to run the E-Discovery report. In the **Max Processes** field, you can enter the maximum number of work processes to be created on this application server.

Let's look at another special function: You may want to add a print list or an ArchiveLink attachment in the legal case. Let us also assume that this attachment does not have a leading object or that you do not want to add this object in the legal case. In this case, click the **Data Category** button marked in Figure 3.1 and choose **ArchiveLink Documents**. The heading of the list of objects below now changes to **ArchiveLink Documents**.

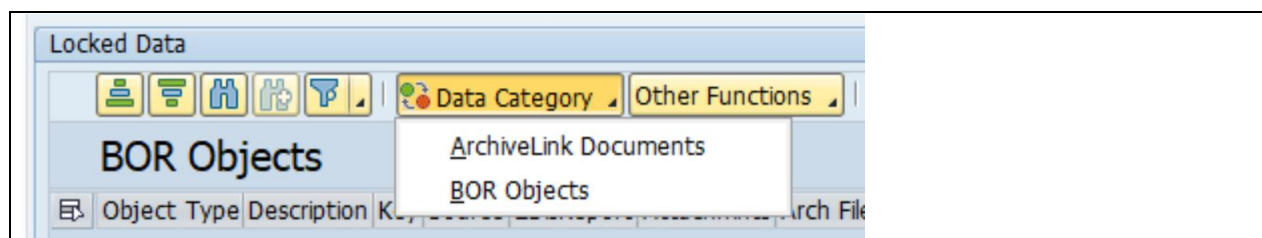


Figure 3.1: Transaction ILM_LHM – Add Print List or ArchiveLink Attachment to a Legal Case 1/2


Now choose the **Add Objects** button. (The **E-Discovery** button is not available in this case.) In the dialog box that appears, you can select the required documents. In our example, this is a print list that we want to add in the legal case. If this is also your target, select the entry **AL_DL** (ArchiveLink: Print Lists) in the **Document Area** field and the ID of the print list in the **Document ID** input field (see Figure 3.2). Confirm the dialog box by clicking the green checkmark or press **ENTER**.

Document Area:

Info.	<input type="text"/>	to	<input type="text"/>	
Storage Date	<input type="text"/>	to	<input type="text"/>	
Business Object	<input type="text"/>	to	<input type="text"/>	
Document Type	<input type="text"/>	to	<input type="text"/>	
User	<input type="text"/>	to	<input type="text"/>	
Short Text	<input type="text"/>	to	<input type="text"/>	
Storage Sys.	<input type="text"/>	to	<input type="text"/>	
Document ID	<input type="text" value="42010AEF3E3F1EDDA8D..."/>			

Hit List	Storage Date
AL_DL (1 Hits)	
Print Lists	
TEST SAP 1	02.02.2023

Figure 3.2: Transaction ILM_LHM – Add Print List or ArchiveLink Attachment to a Legal Case 2/2

The required document is then displayed in the lower part of the dialog box. Select it. (It is important that you select the entire row. To do this, you can, for example, click on the small icon on the left in the corresponding line.) Now confirm that the selected document is to be added in the legal case by clicking the export icon () at the bottom right in Figure 3.2.

You now know how to add structured objects (e.g. a sales document) and unstructured documents (e.g. an ArchiveLink print list) in a legal case. Next, we will describe the functions that are available to you after you have successfully added the required objects/documents in the legal case.

If required, you can edit the list of objects with Legal Hold. For example, you can delete entries from the list. To do this, you can use the **Delete Selected Rows** () icon marked in Figure 3.3. Note that objects are of course not physically deleted.

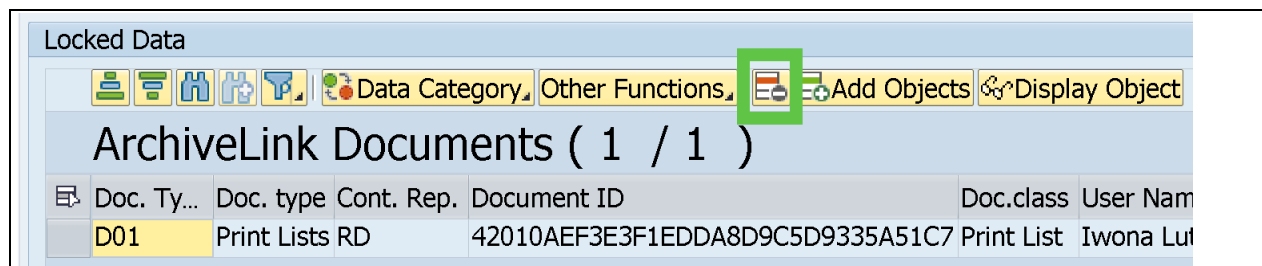


Figure 3.3: Transaction ILM_LHM – Delete Documents from a Legal Case

New Section in Chapter 3: Information Retrieval Framework

This is a new section, not present in the original edition.

Transparency requirements are a fundamental pillar of data protection law. In this chapter, we explain the Information Retrieval Framework (IRF), which supports to fulfil information obligations to a data subject and the right to data portability.

Transparency requirements are among the data subject rights in the General Data Protection Regulation (GDPR) and are distinguished between the right of access to data in accordance with Art. 15 GDPR and the right to advance information in accordance with Arts. 13 and 14 GDPR.

A very significant but logically justified difference between the advance information and the information pursuant to Art. 15 GDPR is that in the case of the information, the data itself must be disclosed. IRF helps you find and extract the data to be disclosed to a data subject. As report that can provide suitable information (obviously excluding purely organizational information) it is also able to support you in creating advance information and the record of processing activities

Furthermore, as a report that can output all personal data is IRF also sufficient to formally fulfil the right to data portability under Art. 20 GDPR, if it converts the data into a machine-readable format.

At this point, you will be wondering why the authors confront you with data protection law issues in a book about SAP ILM. Our motivation for this is that the personal data to be disclosed usually concerns exactly the data that may only be processed for the respective purpose and must be blocked and/or deleted after the end of the purpose.

By using SAP ILM for blocking and deletion of personal data (see Chapter 4 "Simplified Blocking with SAP ILM"), you have already identified the data and the associated ILM objects relevant for information retrieval. You may also have considered and adjusted your data model concerning the intended purposes

(see section "Data Controller Rule Framework (DCRF)"). There is therefore no further reason why you shouldn't be able to provide information for a specific purpose.

However, the ILM objects identified in this way only form the basis for setting up and using IRF. Because the IRF-based search and the output of data mostly use functionalities outside of SAP ILM.

Further information on data protection in general and IRF in particular

We invite you to read further information on the purpose-related collection, processing, and storage limitation of personal data in our book "GDPR and SAP: Data Privacy with SAP Business Suite and SAP S/4HANA " (Lehnert, Luther, Christoph, Pluder, SAP PRESS 2018).

You will there find a more comprehensive presentation of the IRF in Chapter 8 "Information Retrieval Framework". In this book, we limit ourselves to an overview of the most important IRF functionalities

Concept of the Information Retrieval Frameworks

Based on the idea that information must be provided about the personal data that is deleted via SAP ILM, the IRF accesses these ILM objects, thereby achieving congruence between deletion and information.

The following basic ideas are behind the preceding explanations:

- The relevant ILM objects containing potentially person-related data are known.
- An IRF data model (hereinafter also referred to as a data model) can be automatically generated from these ILM objects, whereby the IRF uses the functionality of the Generic Smart Search (GSS).
- The generated IRF data model is manually adjusted (see Chapter 1.1.3 "Creating an IRF Data Model")
- The GSS performs an automated extraction of all associated data for each purpose based on the IRF data model.
- The result can be checked by the data protection officer and, if necessary, transmitted to the data subject after a revision (see Chapter 1.1.5, "Carrying out an Information Request").

Note: Use of IRF in SAP S/4HANA Cloud Public Edition

IRF is available in SAP S/4HANA Cloud Public Edition. The underlying IRF data model is specified by SAP and cannot be changed. Your use of IRF therefore mainly focuses on carrying out the information request for the ILM objects you have identified and the corresponding purpose setup.

Setup and Personalization of IRF

To use the IRF, it is necessary to perform several steps for activation and setup. We refrain from explaining these steps here, as they are described in detail in the SAP application help. You can find this information in the SAP Help Portal by searching for the product **ABAP_PLATFORM_NEW**, selecting your used release followed by searching for the term **IRF** in the document. Select the correct search result (usually found first) and follow the instructions described under **Setting Up the System for the IRF**.

Similar and further information about the functionalities described below can be found in the IRF Guide, which is available as an attachment in SAP Note 2646204 ("Consolidated Note for Information Retrieval Framework").

As already mentioned, enables IRF a purpose-based data search. You define the purposes required for this and assign the associated ILM objects. The configuration as shown in Figure 3.4 also allows the definition of filter criteria (related to the respective IRF data model) to search only certain data of an ILM object per purpose. This can be used, for example, to select only data from a specific organizational unit (such as a company code). Additionally, you have extension options through the definition and assignment of data collectors (see SAP Note 2897023), which can be implemented independently of a generated data model for data collection from other data sources.

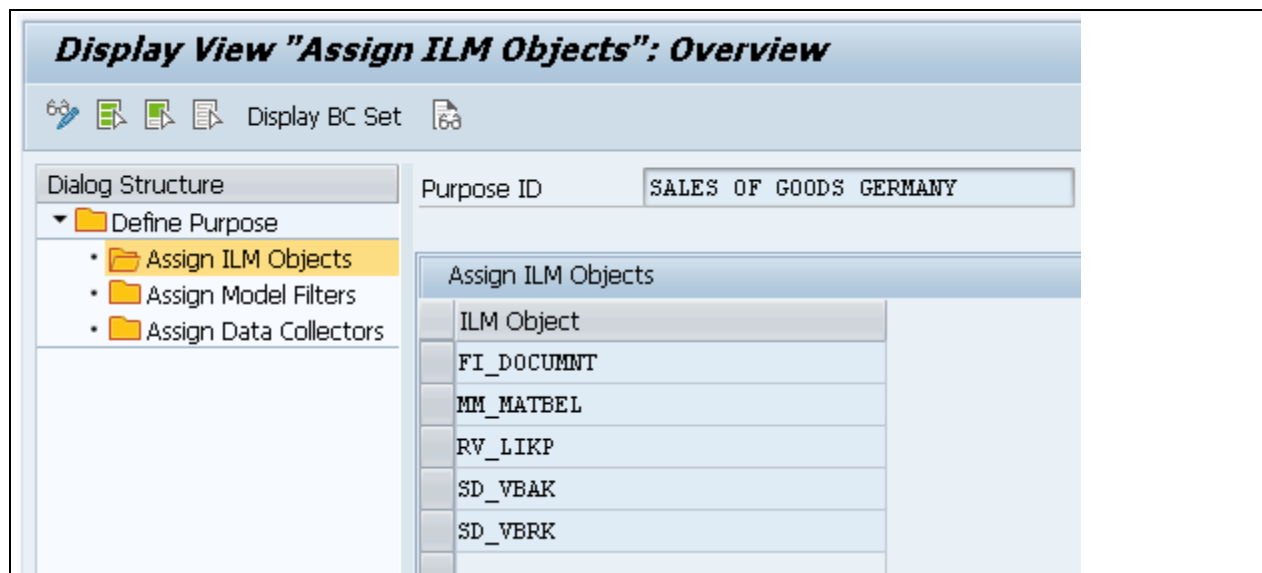


Figure 3.4: Definition of purposes for IRF

If you use the **Data Controller Rule Framework (DCRF)**, you only need to complete the purposes in IRF. The IRF interacts with the DCRF and automatically copied the purposes maintained in DCRF as well as the associated ILM objects.

Note: SAP S/4HANA Cloud Public Edition

Purposes are centrally maintained including the related data controller and data categories about the related personal data. Purposes in status “Active” can be used as IRF purposes to collect personal data. If required, additional ILM objects can be added to IRF purposes.

With the definition and use of profiles in the data search as shown in Figure 3.5, you can implement various personalization options to influence the collected data:

- Limitation of the scope of the data to be searched via the data models, including the exclusion of certain tables
- Options for hiding or masking data
- Change of values for tables and fields in the data model
- The option to consider attachments in data collection. Further information can be found, for example, in SAP Note 2964095 ("Enablement of attachments retrieval from IRF").

- The option to consider archived data in data collection. Further information can be found in SAP Notes 3220336 ("Retrieval of data from Archives") and 3306887 ("IRF Data Retrieval from Archive - Summary and FAQ").
- Settings for downloading the data, including the desired download format

The screenshot shows the 'Display Profile: PROFILE SALES GERMANY' interface. At the top, there are icons for help, search, and information. Below this, the profile name 'PROFILE SALES GERMANY' is displayed next to its status 'Active'. A description field is present but empty. Two tabs, 'General Settings' and 'Table Settings', are visible, with 'General Settings' being the active one. The main content area is divided into four sections: 'Data Collection', 'Download', 'Data Masking', and 'Data Hiding'. In the 'Data Collection' section, 'Maximum Hierarchy Depth (Model)' is set to 2, 'Maintain Exclusion List' is unchecked, 'Archived Data' and 'Attachments' are checked, 'Tagging Characters' is set to '*', and 'Email Notification' is unchecked. The 'Download' section shows 'Format' set to 'TEXT', with 'Attachments', 'In Background', 'Hidden Tables and Fields', and 'Masked Fields' all unchecked. The 'Data Masking' section has 'Masking Character' set to '*', 'Masking Length' set to 10, and 'Maintain Masking List' unchecked. The 'Data Hiding' section shows 'Hide All Fields' and 'Maintain Hide List' unchecked, while 'Hide Empty Fields' is checked. At the bottom of the 'Data Hiding' section, there are three buttons: 'Tables', 'Fields', and 'Empty Fields'.

Figure 3.5: Maintenance of profiles in IRF

Creation of an IRF Data Model

The work of creating an IRF data model is first carried out in the development system. Before the finished data model is transported to the production system, a final test is carried out in the test system.

Generate Initial IRF Data Model

As a first step, we get an overview of the database tables identified as relevant for an ILM object and generate an IRF data model. It is also necessary to determine what the so-called **start table** and **start field** (called **start points** as a combination) are to be able to perform a search using this information.

We start the transaction `DTINF_ADJUST_MODEL` (Adjust IRF data model) and enter the ILM object `MM_MATBEL` ("Material Management: Material Documents") as our example, activate the checkbox "Include SAP Modeling BAdI" and press "Simulate". The result is shown in Figure 3.6.

The screenshot shows the 'IRF Modeling' SAP interface. At the top, there's a header 'IRF Modeling' and a sub-header 'Show Additional Modeling Options'. Below this, a 'Generate Model' tab is active. The 'ILM Object' field contains 'MM_MATBEL'. There are 'Simulate' and 'Generate' buttons, and a 'Generation Log' dropdown set to 'None'. Two checkboxes are present: 'Allow Deletions in Existing Model' (unchecked) and 'Include SAP Modeling BAdI' (checked). Below these are two tables: 'Table Links' and 'Field Links', each with a toolbar above it.

ILM Object	From Table	To Table	Action
MM_MATBEL	BUT000	MSEG	Add
MM_MATBEL	KNA1	MMINKON_UP	Add
MM_MATBEL	KNA1	MSEG	Add
MM_MATBEL	LFA1	MMINKON_UP	Add
MM_MATBEL	LFA1	MSEG	Add
MM_MATBEL	MKPF	DTINF_ATTCH_LINK	Add
MM_MATBEL	MSEG	MARI	Add
MM_MATBEL	MSEG	MKPF	Add

From Table	From Field	To Table	To Field	Action	Link Type
BUT000	PARTNER	MSEG	SERVICEPERFORMER	Add	Reference
KNA1	KUNNR	MMINKON_UP	KUNNR	Add	Reference
KNA1	KUNNR	MSEG	KUNNR	Add	Reference
LFA1	LIFNR	MMINKON_UP	LIFNR	Add	Function Module
LFA1	LIFNR	MSEG	LIFNR	Add	Function Module
MKPF	MBLNR	DTINF_ATTCH_LINK	FIELD_1	Add	Function Module
MKPF	MJAHR	DTINF_ATTCH_LINK	FIELD_2	Add	Function Module
MSEG	LIFNR	MARI	LIFNR	Add	Reference
MSEG	MBLNR	MARI	MBLNR	Add	Reference
MSEG	MBLNR	MKPF	MBLNR	Add	Reference
MSEG	MJAHR	MARI	MJAHR	Add	Reference
MSEG	MJAHR	MKPF	MJAHR	Add	Reference
MSEG	ZEILE	MARI	ZEILE	Add	Reference

Figure 3.6: Simulated IRF data model

On the left side, the **table links** are shown, and on the right side, the corresponding **field links**. In this case, we benefit from the SAP-delivered implementation of the BAdI `BADI_DTINF_ILM_OBJ_TABLES` for the ILM object. You can see the difference in the simulation result if you do not set the checkbox "Include SAP Modeling BAdI" and repeat the action. Since no data model exists yet for the ILM object, all table relationships and field links, as seen in the "Action" columns, will be "Add" to the data model.

When we look at the listed tables, we see that material documents can be found starting from the business partner. Specifically, these are the tables BUT000 (field PARTNER) for the business partner and KNA1 (field KUNNR) and LFA1 (field LIFNR) for the customers and suppliers belonging to the business partner in SAP S/4HANA.

In the transaction `DTINF_ADJUST_MODEL`, we now press "**Generate**" to create an initial IRF data model. You will be asked for a transport request in which the generated data model should be saved. Select the previously created transport request.

If an IRF data model already exists for the ILM object and hinders the generation, you can delete the data model of an ILM object as follows: Click on "**Show Additional Modeling Options**", select the "**Delete Model**" tab, enter MM_MATBEL as our ILM object, and press "**Delete**" (Figure 3.7).

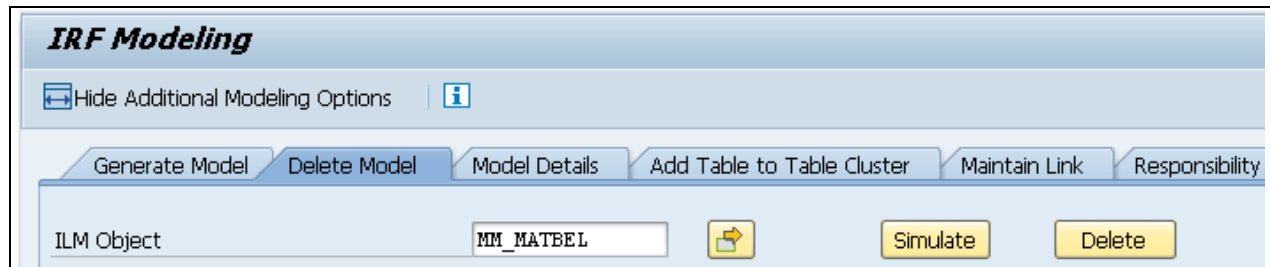


Figure 3.7: Deletion of an IRF data model

Create BAdI Implementations

We must always change the IRF data models so that there is a valid start point, which in most cases is likely to be the business partner. This start table must appear in the "**From Table**" column. For simplicity, we focus on the KNA1 table as an example. The corresponding start field should point to a field of the table to be searched, which can be used for data search. In our example, this is the field link from the KNA1 table and the KUNNR field to the KUNNR field of the MSEG table.

We also need appropriate table relationships and field links for all other data to be found. If this is not the case for one of your required IRF data models, you can develop your own implementation of the BAdI `BADI_DTINF_ILM_OBJ_TABLES_C` (and the corresponding methods) to change or correct the IRF data model. The procedure for developing BAdI implementations is not covered here and is referred to the corresponding ABAP documentation. As a template for your implementation, we recommend looking at SAP-delivered implementations in the BAdI `BADI_DTINF_ILM_OBJ_TABLES`.

In our example, the data model seems simple - for other ILM objects, this will certainly be different, and creating your own BAdI implementation will be useful. This also applies if you have extended application objects with customer-specific tables that should be supplemented accordingly via a BAdI implementation. You can also find implementation help in the appendix of the IRF guide, as previously mentioned, as an attachment in SAP Note 2646204 ("Consolidated Note for Information Retrieval Framework").

For situations where it is not possible to establish an adequate link between two tables via a field-to-field link by reference, the BAdI method `ADD_FIELD_LINKS` provides the option to create the field link with the link type "**Function Module**" with a correspondingly implemented function module. This function module is executed at runtime of the search and can return a modified or more complex selection condition for the search or can provide already the desired data as the result of the search. As we see in Figure 3.8, SAP already delivers various such function modules for the ILM object MM_MATBEL. One reason here is, for example, the search for material documents in the MSEG table for supplier numbers. Because correctly, the search in the LIFNR field must also search for supplier numbers in the LLIEF, EMLIF, and DISUB_OWNER fields using an OR condition. The link type "**Reference**" always uses an AND condition for multiple field links between 2 tables. This does not work in this case, as usually not all 4 fields of the MSEG table will contain the same supplier number.

Field Links					
From Table	From Field	To Table	To Field	Link Type	Link Value
BUT000	PARTNER	MSEG	SERVIC...	Reference	
KNA1	KUNNR	MMINKON_UP	KUNNR	Reference	
	KUNNR	MSEG	KUNNR	Reference	
LFA1	LIFNR	MMINKON_UP	LIFNR	Function ...	MMIM_DTINF_LFA1_MMINKON_UP
	LIFNR	MSEG	LIFNR	Function ...	MMIM_DTINF_LFA1_MSEG
MKPF	MBLNR	DTINF_ATT...	FIELD_1	Function ...	DTINF_GET_ATTACHMENTS
	MJAHR		FIELD_2	Function ...	DTINF_GET_ATTACHMENTS
MSEG	LIFNR	MARI	LIFNR	Reference	
	MBLNR		MBLNR	Reference	
	MJAHR		MJAHR	Reference	
	ZEILE		ZEILE	Reference	
	MBLNR	MKPF	MBLNR	Reference	
	MJAHR		MJAHR	Reference	

Figure 3.8: Function Modules for Complex Field Links

IRF Test Data Model

After an initial revised data model is available, it must be validated and possibly revised in an iterative process.

To test, we need data, i.e., we need to select a business partner for whom material documents exist. Use this to start the transaction **DTINF_TEST_MODEL** (IRF Test Data Model). As a result, you will see a table hierarchy with the tables according to the data model confirming the correctness of the data model.

It is recommended to view all selection conditions based on our defined field links between the tables. This allows us to check if the link or data selection is correct.

It is also recommended to check the results of the individual links for errors. If IRF has not found data from a table, the implementation of the BAdI method GET_TABLES or one of the other methods must be corrected accordingly, the IRF data model recreated, checked, and the search executed again.

If the result is satisfactory, the transport requests (BAdI implementation, IRF data model, and purpose) can be transported to the test system and tested there again.

If the result in the test system is also satisfactory, the requests can be transported to the production system, otherwise, further corrections are carried out in the development system.

Carrying Out an Information Request

After the IRF data model meets the requirements, we can simulate which data would be determined for an affected person requesting information about their personal data.

Start Data Collection



To perform the disclosure, start the transaction `DTINF_START_COLL` (Start Data Collection). Here, the search criteria are predefined, unlike in the `DTINF_TEST_MODEL` transaction. For example, only business partner, customer, or supplier numbers can be searched. The corresponding start points are already stored in the SAP system.







Enter your test data and press "**Execute**". You will receive confirmation that the data collection has been successfully initiated. In the background, a regularly running job is executed, which is responsible for reading the data.

Manage Data Collection

Start the transaction `DTINF_PROC_COLL` (Process Data Collection Results), and you will see a list as shown in Figure 3.9. The search for your request first appears in the status colour yellow (being executed), other possible status colours are red (error) and green (successfully completed).

View Data Collection

Request List


Status	Data Subject ID Type	Data Subject ID	Language	Last Updated By	Started On	Last Updated On
	Customer Number	SES	EN	PLANNING	04/04/2025 04:22:02	04/04/2025 04:29:04

Figure 3.9: Request List of Data Collections

Note: Execution of Data Collections

The individual searches are not started in parallel to keep the system load within limits. This cannot be influenced by the user. Once an individual search is completed (or aborted), another search is initiated.

By double-clicking on the search request, a new page opens (see Figure 3.10).

Profiles

Customer Number SES

PurposeSales of Goods GermanyGOProfileNONE

Purpose: Sales of Goods Germany

General Data in Customer Master

Document Segment: Material

Header: Material Document

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material

Document Segment: Material


Figure 3.10: Result of a Data Collection

Unlike before when testing the data model, you now no longer see the details, but a hierarchical result of the data collection.

At the top, a dropdown list shows all purposes for which a search has been conducted. The result corresponds to what was previously determined using DTINF_TEST_MODEL, but with a different representation: There is a hierarchical table tree that can be individually expanded. And after double-clicking on a row in the tree, all details of the selected entry appear on the right side.

The data is already prepared in a more readable format. You can directly change the data here or influence it with a corresponding profile selection to enable easier understanding of the transmitted data for the affected person.

If the user has the appropriate authorization, the search result can be downloaded for example as a text file

by clicking on  (**Download**). How these downloaded data are further supplemented or edited and how they reach the customer is at the discretion of the responsible company.

Summary

In the past (without IRF), it was necessary to know a multitude of various reports, transactions, or apps to implement a disclosure request, associated with a lot of manual work and possibly without direct support from business processes. You have now seen that this can be carried out almost fully automated by using ILM objects – and the entire procedure is possibly already automatically prepared in a form understandable to the affected person, as required by the GDPR.

However, this does not work without your own contribution: Only through the iterative process of creating an IRF data model and analyzing the search result can correct IRF data models be created that deliver the desired result.

Where the limit of the data scope lies, which is made available to the affected person, somewhere between the minimum (a single page of paper, which is probably not sufficient) and the possible maximum (hundreds or thousands of pages of paper), must be decided individually.

4 Chapter 4: Simplified Blocking with SAP ILM

4.2 Overview of Solutions

This section expands coverage of Section 4.2: Overview of Solutions.

Note: Meaning of the residence time for the audit areas ARCHIVING and BUPA_DP

As you have read in section 2.3, "Audit Area: A Reason to Define Retention Rules", you must create residence rules for blocking master data (customer, vendor, central business partner, contact person) in audit area **BUPA_DP** (Business Partner EoP). You can only use this audit area for this (no other audit area and no copy of it). Note that the **BUPA_DP** audit area has no special meaning in connection with the retention rules. It is an audit area like any other for the retention periods.

In section 2.4, "Policy Categories", you have also learned that the residence period defines the period over which the data is to remain in the database before it is archived. With the concept of simplified blocking and deletion, the term residence period has a second meaning: Residence times are the times in which data can be processed as part of the purpose. But how can you distinguish between the two areas of application? This is very simple: In the first case, the audit area **ARCHIVING** (data archiving) is used. In the second case, the audit area **BUPA_DP** (Business Partner EoP) is used.

Note: New audit area BLOCKING

SAP Note 3374438 (ILM: New audit area **BLOCKING** for residence rules (application-specific blocking of data)), you can use the new audit area **BLOCKING**. It is used to block other master data or transaction data than the four types of master data we discussed so far (customer, vendor, central business partner and the contact person). We mention it at this point so that you can interpret it correctly in transaction **ILMARA**, even if currently there are only a few applications that use this new audit area (so that you would have to create residence time rules for it). The audit area is intended for applications that meet the following criteria:

- applications that deliver a data destruction object (less often, but also possible: an archiving object)
- applications that control the blocking of data using a special indicator (attribute) that provides the information "is blocked"
- Applications that offer a preprocessing program for the data destruction object (or archiving object) that you can use to set the indicator, that is, to block. (The preprocessing program should be written in

such a way that it only takes residence times from this audit area into account.) This allows you to define when the data can be blocked. (The preprocessing program would, of course, perform further application-specific checks to ensure that the data can be blocked.) After the retention period has expired, you would execute the data destruction program of the data destruction object as usual for final data deletion. (In the case of an archiving object, you could use the archive write program as usual for data archiving.)

Note: More complex system landscapes with multiple leading and dependent systems

The standard customer/supplier/business partner EoP check was initially designed to work only in a system landscape with one leading and multiple dependent master data systems. A more complex system landscape with multiple leading and dependent systems was initially not supported. This has changed in the meantime. The following SAP Notes describe how you can register the leading systems for more complex system landscapes:

- SAP Note 2883431 (Customer/Vendor master data destruction: End of Purpose Check scenario enhancements - Part 1 as well as the other SAP Notes mentioned there)
- SAP Note 2883350 (Customer/Vendor master data destruction: End of Purpose Check scenario enhancements - Introduction)
- SAP Note 3276021 (Business Partner Master Data: End of Purpose scenario Enhancements - Multi Lead System Group Concept FAQ Details)

The mentioned SAP Notes explain how to create the *system landscape groups* introduced for this purpose (*system groups* for short) to define your leading and dependent systems. In a second step you can use the *Business Rule Framework plus* (BRFplus) to assign these system groups to the customer, supplier, or business partner master data.

Some of the mentioned SAP Notes contain a PDF document that describes the constellations supported for complex system landscapes both graphically and textually.

4.4 Blocking in the Business-Related View

4.4.1 Blocking Transactional Data in a Business Process

This section expands coverage of Section 4.4.1: Blocking Transactional Data in a Business Process

Note: Two purposes that the entry of an authorization group fulfills

Entering the authorization group in a policy fulfills two purposes:

- On the one hand, it ensures that the personal data in the corresponding archive files is blocked.
- On the other hand, it defines who can access the data blocked in this way

The master data behaves differently. The authorization group entered here only fulfills the second purpose. You block master data as described in section 4.4.3 “Blocking Master Data in a Business Process”.

Note: Improvement of message long texts and detailed information in application log

The log also contains a **Ltx** column that contains the long text with further details about the message. As of the corrections from SAP Note 2908849 (EoP check - Improvements of messages in the application log), the long texts have been improved. They contain for customers and suppliers or contact persons that you want to block, information such as the next check date or the start of the retention period. If there is a long text, you see a question mark symbol in the **Ltx** column, which you can click to call up the long text. The log displayed also contains a **Detail** column. As of SAP Note 3199558 (End of Purpose Check Log Extension), clicking the magnifying glass icon in this column provides additional information about open documents if a customer/vendor could not be blocked during the EoP check.

You can use the following transactions to view or delete data in the **SoRT** tables:

- **CVP_SORT_MONITOR** (Monitor Start of Retention Time)
You use this transaction to display the data records for the start of the retention period for customers, vendors, and contact persons (entries in the **SoRT** tables).
- **CVP_SORT_MONITOR_DEL** (Monitor and Delete SoRT records)
You use this transaction to delete the data records for the start of the retention period for customers, vendors, and contact persons. (entries in the **SoRT** tables).
- **BUPA_SORT_MONITOR** (Display SoRT records)
You use this transaction to display the data records for the start of the retention period for the central business partner. (entries in the **SoRT** tables).
- **BUP_SORT_MONITOR_DEL** (Monitor and Delete SoRT records)
This transaction is used to delete the data records for the start of the retention period for the central business partner. (entries in the **SoRT** tables).

4.4.8 Blocking of User Master Data

This is a new section, not present in the original edition. It follows Section 4.4.7: Possible Combinations and Sequences.

User master data (*user master record*) is also personal data. In this section, we show you another technical procedure for blocking personal master data to deepen knowledge about blocking master data. In the case of user master data, there is a slight modification to the procedure discussed in this book so far.

If a user master record is created, you can identify this system user and give it the required authorizations. Applications can also store references to this user, such as “created by” or “last changed by”.

A consequence of the deletion of a user master record that is still referenced by applications would be, if you don't use SAP ILM, that a system administrator could create a new user master record that describes a completely different person, but (accidentally) has the same user name (system field **SY-UNAME**). As a result, SAP ILM does not allow the deletion of user master data until no more applications reference this user and the retention periods for user master data that you have defined have elapsed.

This new functionality of blocking and destroying user master records under control of SAP ILM is delivered with SAP Notes 2826256 (Deletion and blocking of user master records) and 2836524 (Configuration Standard Veto Checks for Blocking and Deleting User Master Records). You activate it by

creating a table entry with the ID **USER_DEL_LOCK_ENTRY** (if it does not yet exist) in the table **PRGN_CUST** (Customizing settings for authorization process) and entering the value **YES** in the **PATH** field.

In the following sections, we will introduce this new feature.

Blocking of User Master Data

Figure 2.1 illustrates the lifecycle of the user master data again. You are already familiar with this figure from the chapter 2.7.7 “Destruction of User Master” further above in this appendix.

As usual, you would delete the user in transaction **SU01** (User Maintenance). However, since you have activated the blocking and deletion of the user master data under ILM control, it does not disappear completely. Instead, the user ID along with key information, such as the address data, is moved to its own, newly created blocking area (table). As of this point in time, the user master record is regarded as blocked. This has the following effects:

- You will no longer be able to find the user in the value help.
- You will also no longer be able to access the user in an existing application record based on the user ID.
- You will also not be able to recreate a user with the same user ID.

Note: Access to Blocked User Master Data

Access to the blocking area just mentioned, in which the blocked user master data is retained, is particularly protected. Only users who have the authorization object **S_USER_BLK** can access it. The authorization object is not part of the administration authorization of standard users. It is intended for persons with special requirements for which transaction **SU06** (Deleted Users – Lock Entries) is available. You can view the blocked user master data here.

It is important for you to know that the time of blocking the user is considered as the start of retention time of the user. Exactly this point in time refers to the retention time that you must define in SAP ILM. ILM object **IDENTITY** is available for this purpose.

Blocking A Business Partner That Is Linked to a User Master Record

A business partner can only be blocked if no active user is assigned to it. In other words, a business partner cannot be blocked as long as it is linked to a user master record that is displayed in transaction **SU01**.

For this reason, the user management is involved in the end of purpose (EoP) check. In this way, it can determine whether a user master record (that is linked to a business partner) is still actively used in. To do this, user management delivers the application name **USER** (User Management). You can assign specific residence times for this application name, if required.

If you have blocked user master data, you can destroy it.

7 Chapter 7: Implementing an SAP ILM GDPR Project

7.2 Stages of an SAP ILM GDPR Project

7.2.6 Implementation and Testing

This section is an addition to the end of Section 7.2.6: Implementation and Testing.

Unblock or Not?

It is possible to unblock blocked personal master data. Various settings are available for how this process can be designed, e.g. the enforcement of a dual control principle when unblocking or various extension options via BAdIs. In practice, the question arises as to the extent to which an unblocking process is used productively at all. The hurdles for an actual block are generally quite high; for example, any transaction must be completed in all applications and an application-specific retention period must have expired. After blocking, the personal master data and all transaction data used disappear from the visibility of the normal user.

In the context of unblocking, the question then arises as to how high the probability is that, for example, a blocked customer will return, i.e. must be unblocked in the system. It must also be clarified how a possibly existing but blocked master data instance can be found in the system and by whom. In general, blocking is orchestrated centrally. Destruction, on the other hand, is decentralized in the individual systems and not coordinated. In a complex landscape, can lead to a customer existing in some systems while he has already been destroyed in other systems. This means that dependent systems can destroy their blocked personal master data depending on the retention rules and the data before the leading system. When unblocking, it must be ensured that the destroyed master data is recreated or replicated.

In practice, this means that in many projects, unblocking as a process is not set up at all or is set up very simply. Instead of dealing with the unblocking process in detail, in such cases new instances of customers, suppliers, contacts or business partners are created directly.

Testing in the Quality Assurance Environment

As already emphasized, there should be a balance between effort and benefit when testing. In this section, we explain which factors should be considered when testing the destruction of transaction data and the blocking and destruction of personal master data in the quality assurance environment. The starting point for our explanations in this section is a completed SAP ILM implementation in the sandbox systems. All ILM objects were technically tested there and then considered for each application and within the overall process from a business perspective. As a rule, companies maintain a quality assurance landscape for various quality assurance tests, which mirrors the productive landscape and has a relatively up-to-date data status. This landscape is also used for the final acceptance tests of the ILM implementation project.

Setting Up the Quality Assurance Environment

The business functions and services for SAP ILM are first activated in the quality assurance landscape. The ILM settings are transferred to the quality assurance systems via transport requests. The completeness of these settings must be checked there to ensure that all necessary transports are complete for the subsequent go-live. The final retention periods must also be determined and set in the system by now at the latest. It is

often a challenge to obtain the retention periods for other countries. As a workaround, you can work with assumptions up to this point.

Simulation Runs

In order to successfully complete the testing of transaction data destruction, not only a technical check is required, but also confirmation of the successful tests by the applications responsible.

From a technical point of view, as already explained, it must be checked whether the ILM retention rules have been implemented and are being applied correctly by the ILM rule engine. It must also be ensured that the SAP standard archiving checks can correctly evaluate the destruction criteria for the transaction data.

From an application perspective, application users should be involved to confirm that data destruction can be performed according to business requirements. It is difficult to establish measures and criteria to assess whether data destruction is performed properly. It is therefore strongly recommended to simulate data destruction in the quality assurance system (or a system with production data). The data destruction should be carried out for the oldest financial year or years, either for all or for selected organizational units and with the final retention rules.

The results and statistics of the data destruction show, among other things, how many data units cannot be destroyed. If the oldest data units have already expired or are close to the end of their retention period, a high volume of data that cannot be destroyed usually means incomplete business processes or problems with data quality. In both cases, the application team should be involved to investigate the cause. After destruction, application users should run key transactions and reports in the quality assurance system to check whether unexpected results (e.g. incorrectly accumulated figures) can be observed.

Simulating the blocking of personal master data in the quality assurance system is also crucial for a smooth go-live. The simulation run should focus on both the technical and the application perspective. The technical focus is on determining the optimal configuration for executing the blocking report, i.e. defining the scope of the individual variants, the work packages and the number of parallel processes.

As a result, the runtime of the initial execution of the block run in the production system can be estimated. The result of the simulation illustrates how many person master data records are blocked in accordance with the defined residence rules. There is no golden rule or guideline for application experts and data protection officers to define the residence time of individual applications. Therefore, application stakeholders should be involved to review the list of blocked personal master data and confirm whether the defined dwell time policy best meets the business needs and benefits.

Evaluation of the Application Logs

In order to carry out a simulation run, it is important to understand how the application logs (of transactions `BUPA_PRE_EOP` (Block Business Partners) and `CVP_PRE_EOP` (Block Customer&Supplier Master Data)) can be used.

Transaction `BUPA_PRE_EOP` displays the application log in three levels of detail, namely Summary only, Status per business partner and Complete:

- The summary only setting can be used to generate statistics that show how many business partners can be blocked (test mode) or are blocked (productive mode). As a rule, this option is suitable for creating block reports with a large number of business partners.

- The setting status per business partner provides the next level in the application log by listing the information for each processed business partner as to whether it can be blocked or not. This level of the application log is suitable for executing a transaction with a list of business partners.
- Finally, the complete information setting shows information at application level. It is normally only used for unit tests (see Figure 7.1).

Ty..	Message Text	Long	Details	ID Ty..	Bus. Partner	CoCd	Application Name	Logical system	Status
■	Bus. activities for customer 0001001231 not found in application EHFND_CCM		🔍	1	1001231		EHFND_CCM	HE4CLNT400	🔗
■	Bus. activities for customer 0001001231 not found in application EHSDS_OR		🔍	1	1001231		EHSDS_OR	HE4CLNT400	🔗
■	Bus. activities for customer 0001001231 not found in application EHSM_PRC		🔍	1	1001231		EHSM_PRC	HE4CLNT400	🔗
■	Bus. activities for customer 0001001231 not found in application EHS_PS_REP		🔍	1	1001231		EHS_PS_REP	HE4CLNT400	🔗
■	Bus. activities for customer 0001001231 not found in application ERP_ALLOC		🔍	1	1001231		ERP_ALLOC	HE4CLNT400	🔗
■	Bus. activities for customer 0001001231 not found in application ERP_CON		🔍	1	1001231		ERP_CON	HE4CLNT400	🔗
■	Bus. activities completed for customer 0001001231 in application ERP_CUST	?	🔍	1	1001231		ERP_CUST	HE4CLNT400	🔗
■	Cannot block customer 0001001231 with company code 1010			1	1001231	1010		HE4CLNT400	
■	Bus. activities for cust. 0001001231 with CoCd 1010 not found in application CMM_MTM_ACC		🔍	1	1001231	1010	CMM_MTM_ACC	HE4CLNT400	🔗
■	Bus. activities for cust. 0001001231 with CoCd 1010 not found in application CMM_PEV		🔍	1	1001231	1010	CMM_PEV	HE4CLNT400	🔗
■	Bus. activities for cust. 0001001231 with CoCd 1010 not found in application CRM_IF		🔍	1	1001231	1010	CRM_IF	HE4CLNT400	🔗
■	Bus. activities for cust. 0001001231 with CoCd 1010 not found in application ERP_ALLOC		🔍	1	1001231	1010	ERP_ALLOC	HE4CLNT400	🔗
■	Bus. activities completed for customer 0001001231 with CoCd 1010 in application ERP_CUST	?	🔍	1	1001231	1010	ERP_CUST	HE4CLNT400	🔗
■	Bus. activities for cust. 0001001231 with CoCd 1010 not found in application ERP_EF		🔍	1	1001231	1010	ERP_EF	HE4CLNT400	🔗
■	Bus. activities for cust. 0001001231 with CoCd 1010 not complete in appl. ERP_FI	?	🔍	1	1001231	1010	ERP_FI	HE4CLNT400	🔗
■	2 partner records not blocked							HE4CLNT400	

Figure 7.1: Simulation run for a block run in the application log (transaction BUPA_PRE_EOP) - Complete display

At first glance, transaction CVP_PRE_EOP does not offer comparable flexibility in controlling the scope of the application protocol. The application log is controlled here via process switches. You can use the process switches APPL_LOG_FILTER_MSG_1 (see Figure 7.2), APPL_LOG_FILTER_MSG_2 and APPL_LOG_FILTER_MSG_3 to define a list of messages that are displayed in the application log. For details, please refer to SAP Note 2779658.

Process Switch	Process Switch Description	Process Switch Value
<input type="checkbox"/> APPL_LOG_FILTER_MSG_1	Filter selected messages in the Application Log-1	CVP_DP_ILM(085,159,162,168,171)]

Figure 7.2: Customizing for process switch APPL_LOG_FILTER_MSG_1

Figure 7.3 shows an example of an application log in transaction CVP_PRE_EOP that has been filtered accordingly using the process switch.

Ty...	Message Text	Long	Det.	I...	Bus. Partner	CoCd	Application Name	Logical system	Status
■	2 partner records are selected for processing; see long text	?						HE4CLNT400	
■	Execution in 'Test mode'	?						HE4CLNT400	
■	Execution in 'Overall Check' mode	?						HE4CLNT400	
■	Cannot block customer 0001001231			1	1001231			HE4CLNT400	
■	Cannot block customer 0001001231 with company code 1010			1	1001231	1010		HE4CLNT400	
■	Bus. activities for cust. 0001001231 with CoCd 1010 not complete in appl. ERP_FI	?	?	1	1001231	1010	ERP_FI	HE4CLNT400	
■	2 partner records not blocked							HE4CLNT400	

Figure 7.3: Application log in transaction CVP_PRE_EOP with filter set

When analyzing the application protocols, always bear in mind that blocking and destroying master data may require follow-up actions in connected systems. It must therefore be examined whether the data is also handled properly there and whether a new (re)replication can be ruled out.

Example: Connected BW system

If a master data is destroyed in an SAP system, the connected SAP Business Warehouse (SAP BW) must be informed of this via a notification. As part of the tests, some master data should therefore be blocked or destroyed in order to subsequently check whether such a notification was generated. In the BW system itself, it should be checked whether the notification was received and the necessary actions were triggered.

Understanding the Ongoing Business Purpose

During the test phase, the most frequently asked question will be: “Why can't this business partner (customer/supplier/contact person) be blocked?” Although the application log can illustrate which applications have ongoing business purposes that have not yet been completed, it does not show the details. For example, you cannot see from the log which documents have not been fully processed. Therefore, a good understanding of the logic behind the EoP check is essential for carrying out the test. The application users or testers may lack such an understanding.

In order to understand how the current purpose was recognized, additional detailed logs are provided in transaction CVP_TEST_ONE_EOP_APP (see Figure 7.4). SAP Note 3199558 summarizes the most important information about this special function. In addition, the SAP note lists the EoP applications that support the provision of detailed logs.

Date/Time/User	Number	External ID	Object text	Subobject Text	Transaction C...	Program	Mode	Log Number
19.04.2024 11:02:54 D066396	8		Customer/Vendor Loc...	Lock Customers	CVP_TEST_O...	CVP_CL_APPL...	Dialog processing	\$000002
▲ Problem Class Medium	1							
■ Problem Class Additional Information	7							

Ty...	Message Text	Long	L	Bus. Partner	CoCd	Application Name	Logical system	Status
■	Application ERP_MM belongs to application component MM-PUR-GF-DPP	1	R302	R300	ERP_MM	HE4CLNT400		
▲	The information shown below is computed on demand	1	R302	R300	ERP_MM	HE4CLNT400		
■	***	1	R302	R300	ERP_MM	HE4CLNT400		
■	Plants with Customer were detected.	1	R302	R300	ERP_MM	HE4CLNT400		
■	Therefore remove the Customer assignment from the Plant.	1	R302	R300	ERP_MM	HE4CLNT400		
■	Plant R302	1	R302	R300	ERP_MM	HE4CLNT400		
■	To find further open documents check the following table(s)/field(s):	1	R302	R300	ERP_MM	HE4CLNT400		
■	Check table T001W with customer field KUNNR.	1	R302	R300	ERP_MM	HE4CLNT400		

Figure 7.4: Additional information in the application log

A key result of the activities in the quality assurance environment described in this section is the finalization of the operating concept.

7.3 Complex System Landscapes in an SAP ILM GDPR Project

7.3.2 Managing Different Groups of Personal Master Data

This section is an addition to the end of Section 7.3.2: Managing Different Groups of Personal Master Data.

Several Leading Systems with Customers and Suppliers

For some years now, SAP Business Suite has offered the option of configuring the EoP check of customers or vendors so that it is only executed via dedicated sub-areas of the customer or vendor master record on different, dependent systems. In this section, we look at some examples of system landscapes in which there are customers or suppliers that only exist on one part of the system landscape. This means that a system may be the leader for one subset of the data, but takes on the role of the dependent system for the other subset. It is important that the different subsets are disjoint. Together they form the total quantity of all customer or supplier master data.

Figure 7.5 shows two system landscapes in which such a scenario has been supported to date.

Both cases could always be configured as part of a GDPR project:

- In the “Standard 1” case, there was only one leading system and one dependent system.
- In the “Standard 2” case, a CRM system is the leading system for the business partner. The dependent systems are the ERP 3 and ERP 4 systems. There is no data replication between the ERP systems. Therefore, each system is configured exclusively as the leading system for customers and suppliers.

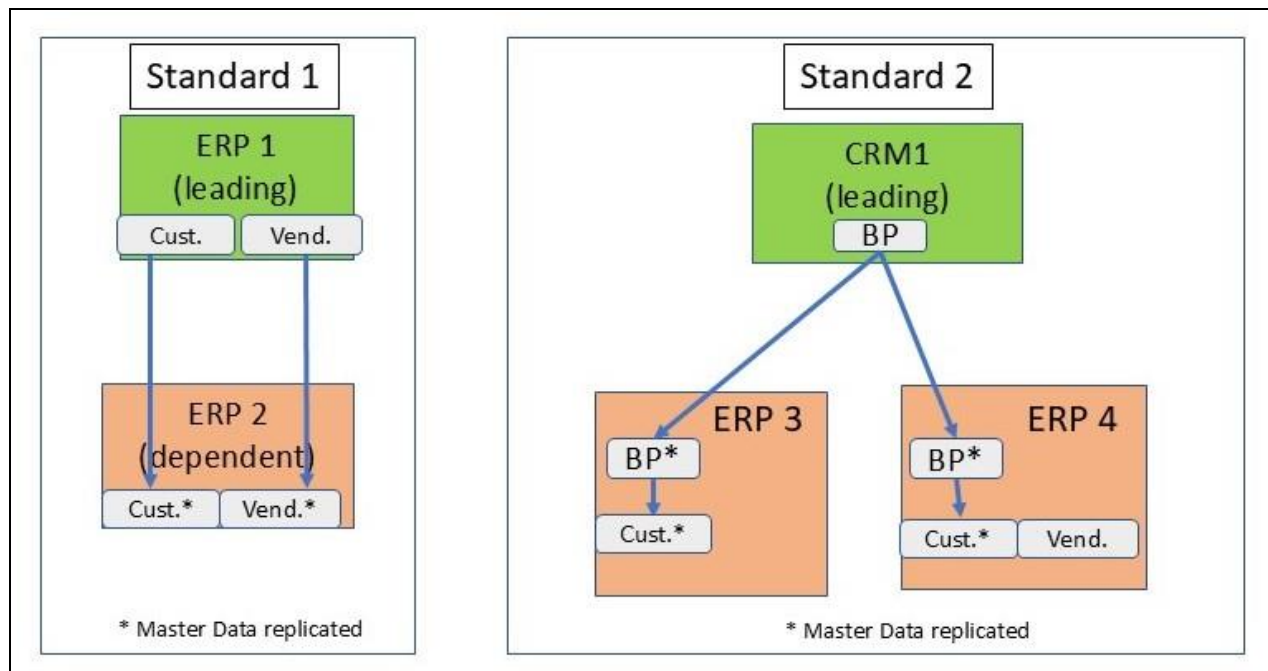


Figure 7.5: Examples of standard landscapes with leading system

An EoP check for customers and suppliers with the replication of master data shown in Figure 7.6 - admittedly somewhat unusual - would not have been possible until now. In order not to make the example too complicated, we will limit ourselves here exclusively to customer data. In this system network, there are four disjoint sets A, B, C, D of customers. Some of these data sets are replicated, but set C, for example, only exists on the ERP 4 system.

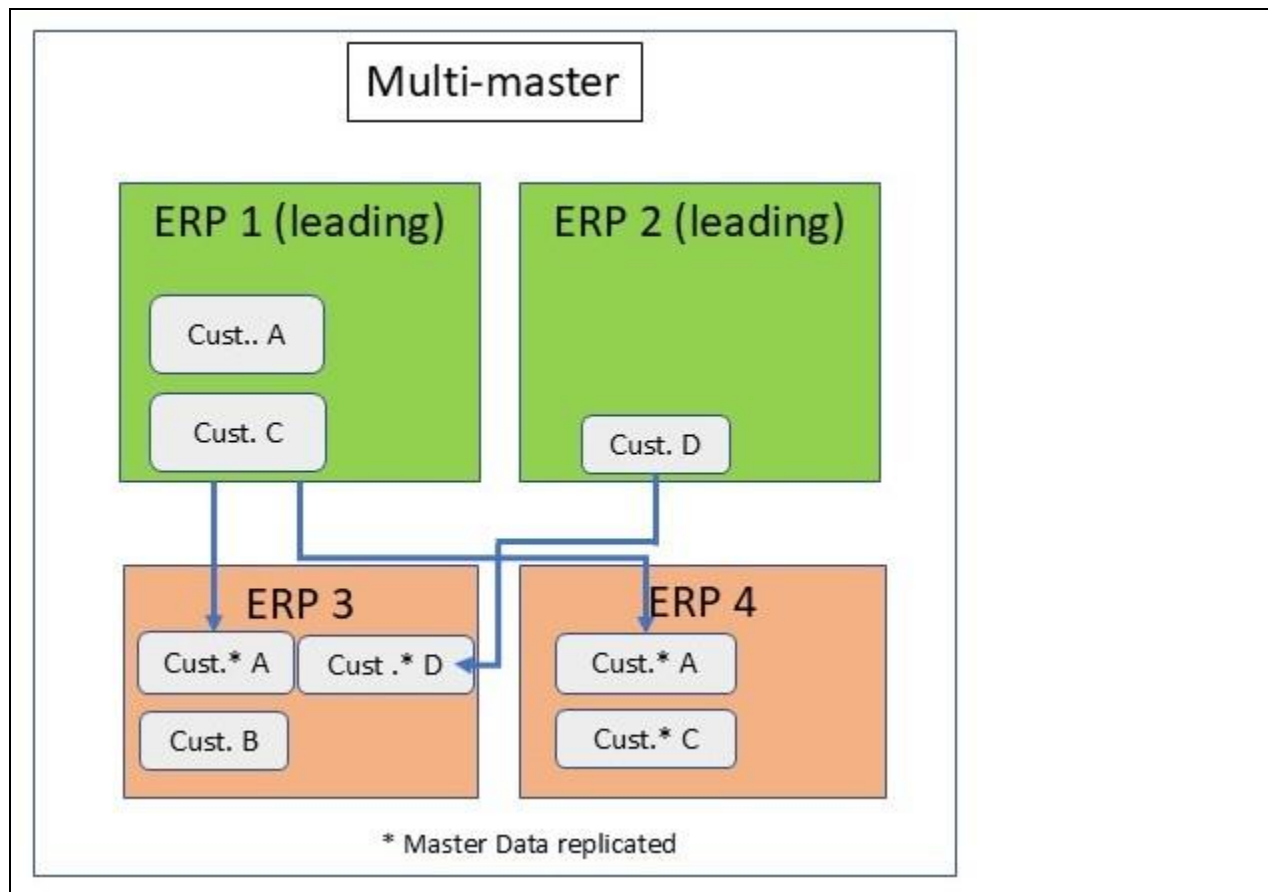


Figure 7.6: Example of a system landscape with several leading systems

For historical reasons, customers and suppliers are replicated in SAP ERP systems via Application Link Enabling (ALE). If a customer or vendor is blocked, this block is first carried out in the leading system. The information “Master date blocked” is then replicated asynchronously via ALE in the dependent systems. In contrast, when a business partner is blocked at the end of the EoP Check, each connected system is informed that this business partner is blocked. Each system then carries out the necessary actions in this system.

To configure the multi-master system functionality, proceed as follows:

- you must first evaluate which disjoint sets of customer master data you have in your system group. In our example, these are the customer groups labeled A, B, C and D. In the same step, assign a logical name for each system, in our case ERP1, ERP2, ERP3 and ERP4.
- then assign the process switch LANDSCAPE_V2_CONFIG_MASTER_SYS with the corresponding system name as the value in each of these systems.
- you must now configure the corresponding systems, i.e. ERP1, ERP2, ERP3 and ERP4. To do this, call up transaction [SPRO](#) (Customizing - Edit project) and use the following IMG path (see Figure 7.7):

Cross-application components - Data protection - blocking and unblocking data - Customer/supplier master data - Registering the system landscape (Advanced) - Basic settings - Define systems.

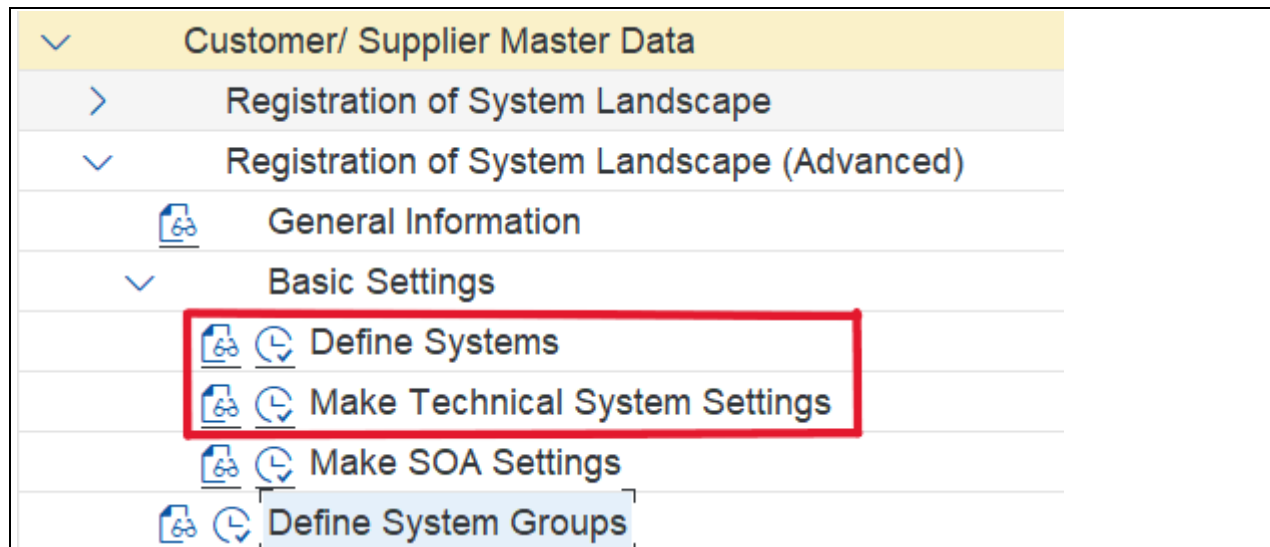


Figure 7.7: IMG activities “Define systems” and “Make technical system settings” in transaction SPRO

Enter the SAP system name for each system ERP1, ERP2, ERP3 and ERP4 and define how the system can be reached via RFC. This configuration takes place in the IMG, under the entry “Technical System Settings”. Figure 7.8 shows an example of a corresponding configuration.

Customer/Vendor EoP check: Technical System Settings			
System	System Description	Logical System	RFC Destination
<input type="checkbox"/> ERP1	QI3CLNT075	QI3CLNT075	QI3CLNT075
<input type="checkbox"/> ERP2	QI3CLNT076	QI3CLNT076	QI3CLNT076
<input type="checkbox"/> ERP3	QI3CLNT077	QI3CLNT077	QI3CLNT077
<input type="checkbox"/> ERP4	QI3CLNT078	QI3CLNT078	QI3CLNT078

Figure 7.8: Example of a system configuration with several leading systems

You can check whether the corresponding systems can be reached via RFC at the same location using a connection test. To do this, click on the Check customizing settings () button in the toolbar or press the key combination (Ctrl) + (F11).

If systems are to be integrated via web services, you can configure this via the SOA settings (Service-oriented Architecture). Please also read section 7.3.4, “Integration of non-ABAP systems”.

This completes the basic configuration. The next step is to define the different system groups. To do this, call up transaction **SPRO** (Customizing - Edit project) and navigate to the following customizing activity: Cross-application Components • Data Protection • Blocking and Unblocking data • Customer/Supplier

Master Data • Registration of System Landscape (Advanced) • Basic Settings • Define System Groups (see Figure 7.9).

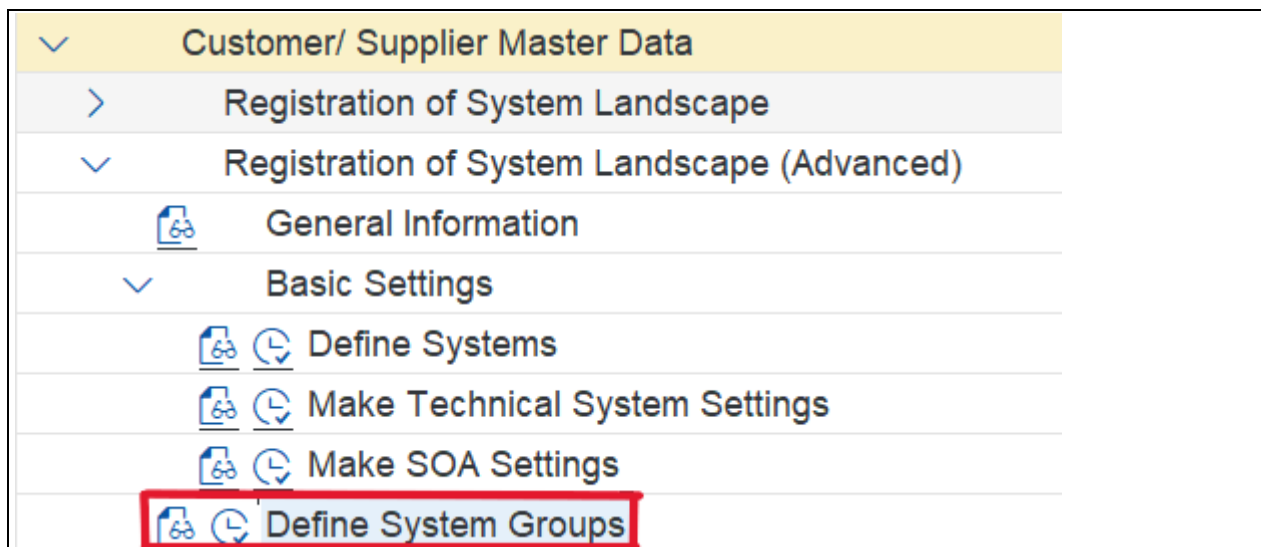


Figure 7.9: IMG activity "Define system groups"

Each system here has between 0 and n dependent systems. The leading system is defined first for each system group (see Figure 7.10). The leading system is always the system in which the data is generated. In the system groups, a group ID is assigned in the first column and supplemented with an explanatory name in column 2. The next columns contain the leading system as well as information on the connection and the leading system. The context menu offers selection options.

Dialog Structure		Define System Groups				
Define System Groups		GrpID	Group Description	Leading System	System Description	Connection Type to Leading System
Assign Dependent Systems to System Group		<input type="checkbox"/> GROUPA	Customer / Supplier A (general)	ERP1	QI3CLNT075	Service Connection
		<input type="checkbox"/> GROUPB	Customer / Supplier B (sales)	ERP2	QI3CLNT076	RFC Connection
		<input type="checkbox"/> GROUPC	Customer / Supplier C (goods supply)	ERP1	QI3CLNT075	RFC Connection
		<input type="checkbox"/> GROUPD	Customer / Supplier D (Competitors)	ERP4	QI3CLNT078	RFC Connection
		<input type="checkbox"/> GROUPE	Customer / Supplier E (Payer / Invoicing)	ERP1	QI3CLNT075	RFC Connection
		<input type="checkbox"/> GROUPF	Customer / Supplier F (Customer from BP)	ERP1	QI3CLNT075	RFC Connection

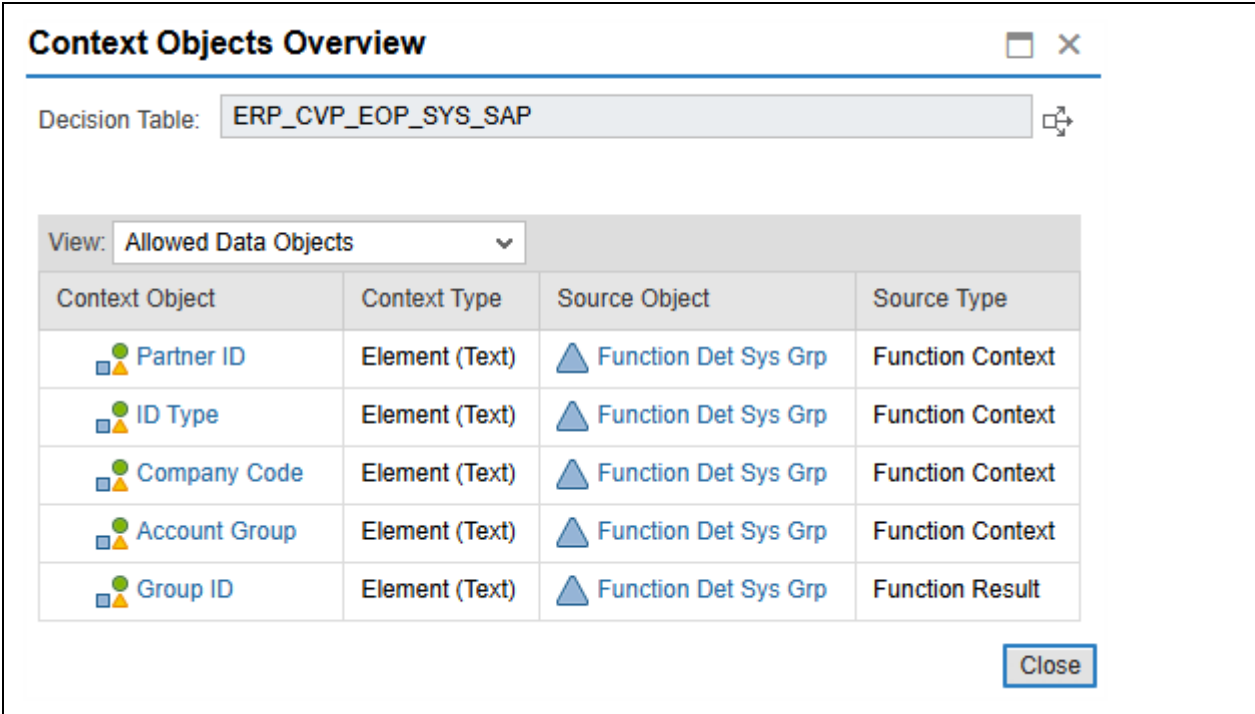
Figure 7.10: Defining leading systems for system groups

Then enter the dependent systems with the corresponding connection types for each group (see Figure 7.11).

Dialog Structure		Define System Groups				
Define System Groups		Group ID: GROUPA				
Assign Dependent Systems to System Group		Assign Dependent Systems to System Group				
		Dependent System	System Description	Connection Type to Dependent System	SeqNo	Connection Type to Leading System
		<input type="checkbox"/> ERP2	QI3CLNT076	Service Connection	1	Service Connection
		<input type="checkbox"/> ERP3	QI3CLNT077	Service Connection	2	Service Connection

Figure 7.11: Assigning dependent systems to the system group

The last step is to define which customer or supplier data belongs to which system group. This is usually very customer-specific and must therefore also be configured on a customer-specific basis. SAP also offers the option of programming this assignment via the Business Rule Framework (BRFplus, transaction [BRFPLUS](#)) if the predefined operating fields are not sufficient. This requires knowledge of the framework, but the most obvious condition fields are already available in the standard system (see Figure 7.12).



Context Object	Context Type	Source Object	Source Type
Partner ID	Element (Text)	Function Det Sys Grp	Function Context
ID Type	Element (Text)	Function Det Sys Grp	Function Context
Company Code	Element (Text)	Function Det Sys Grp	Function Context
Account Group	Element (Text)	Function Det Sys Grp	Function Context
Group ID	Element (Text)	Function Det Sys Grp	Function Result

Figure 7.12: Configuring permitted data objects for a system in BRFplus

The standard condition fields (see illustration) can be extended by programming. The values of the system groups defined in this way should also normally be disjoint. If this is not the case, an evaluation priority can be defined for each rule.

Note: Multi-master system approach for country systems

In a system landscape with a large number of systems, it is common for there to be a leading system for all master data (i.e. for all company codes, customer and supplier master data) and other ERP systems for the transactional data of the various countries. Only master data belonging to these countries should be replicated in the systems of the individual countries. Without the multi-master system concept, the EoP check would query all systems into which this master data is replicated for each master data. This can make an EoP check very error-prone and slow, as all systems must be online and accessible at the time of the EoP check. With the multi-master system concept, the EoP check can be optimized so that only the systems to which the master data in question has been replicated are checked.

Note: System approach with several leading systems at the business

At the time of writing, the first SAP notes on the multi-master system solution based on the business partner had just been published. See Master Note 2883350 The concept is based on the concept of the multi-master system for customers and suppliers.

7.4 Business Processes and Data Controller

7.4.5 How Do I Get the Data Out of the System Again?

This is a new section, not present in the original edition. It follows Section 7.4.4: Employee Authorizations.

Master data, i.e. the data of the customer, supplier, central business partner or contact person, can only be destroyed once the transactional data (all other data) has left the system. The destructions are therefore subject to dependencies and destruction sequences. Perhaps the following mnemonic will help you understand the basic principle: “The data must leave the system in the order in which it entered it.”

Figure 7.13 shows an example of a destruction sequence for some ILM objects in the SAP Business Suite.

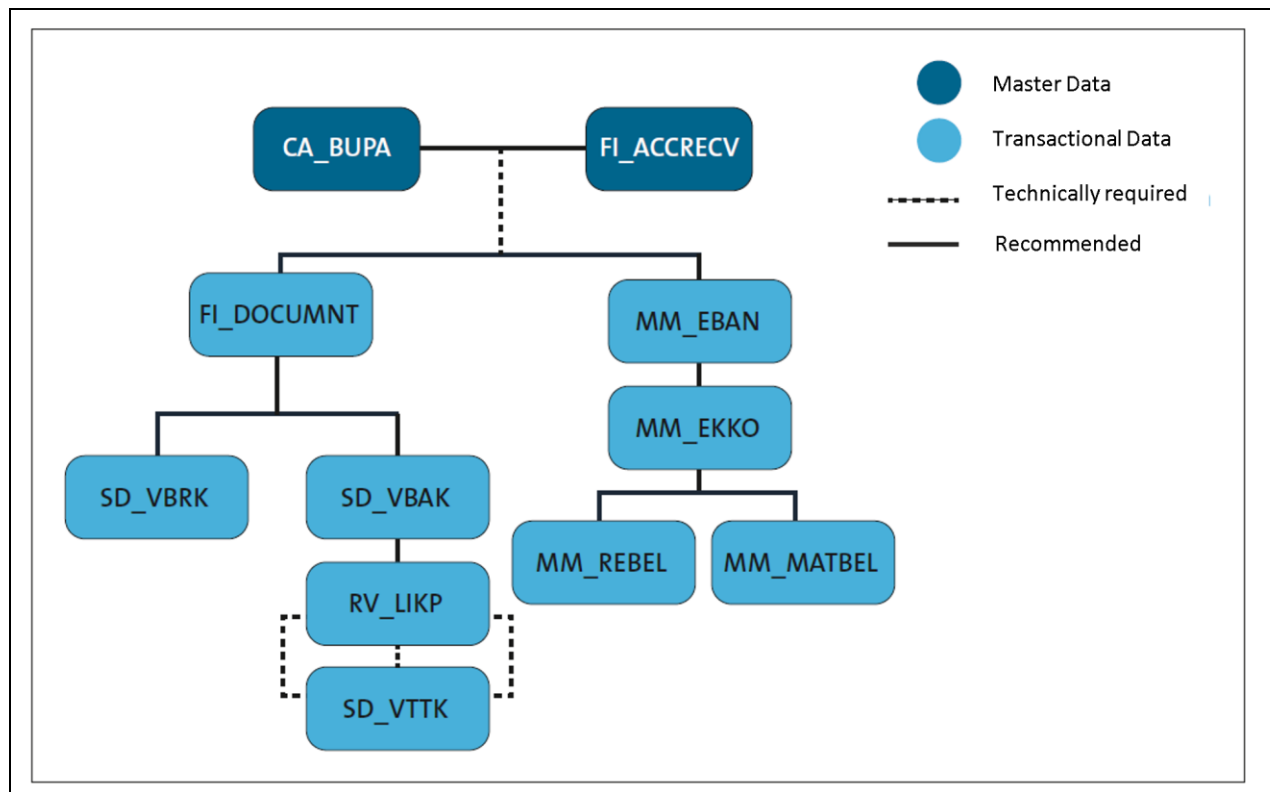


Figure 7.13: Example destruction sequence for some ILM objects

The following sequence should therefore generally be followed. The master data is blocked first, followed by the transaction data and then the master data is destroyed. These individual steps of the destruction process are described in more detail in the following sections.

Step 1: Blocking master data

Destroying master data requires this master data to be blocked first. The time between the block and the actual destruction must be at least one day, so the master data may not be destroyed until the day after the block at the earliest.

In transaction `CVP_PRE_EOP` (Block customer and supplier master data), please note the different check levels in the “Data to be processed” selection.

These are:

- CP: Check contact person only
- FI: Check at company code level only
- AL: Check at all data levels

If you only want to block customers or suppliers at company code level, select the “FI” option as the data to be processed. In conjunction with a sophisticated ILM set of rules, it is possible to destroy the company code of the customer or vendor the next day if no data exists for the customer or vendor in this company code. However, if you want to block the customer or supplier altogether, you need the “AL” option. If a customer or supplier is then blocked, it can only be destroyed once all company codes have been destroyed and the retention period for the data of the individual company codes has expired. The same applies to contact persons.

Step 2: Destroying transaction data

As you have already learned, structured and unstructured data is grouped into ILM objects according to business processes and their technical logic. The life cycle of data in an SAP system is usually managed using a status concept. Possible statuses can be, for example, in process, open or completed. In a destruction concept, the rules according to which destruction is to take place are defined on a process and country-specific basis. These rules are then stored in Retention Management. Once the retention requirements have been met, the data must be destroyed.

As described in section 2.3.6, “ILM actions”, it is possible either to archive the data first and destroy it later or to destroy it without first archiving it. Using the archiving object `FI_DOCUMENT` as an example, we will now look at how the data in an ILM project is destroyed and how the destruction can be checked.

The archiving object `FI_DOCUMENT` is used for financial accounting documents. You can determine the tables from which such documents are destroyed using transaction `DB15` (Data Archiving: DB Tables). You can find out which header table belongs to the archiving object in transaction `AOBJ` (Definition of Archiving Objects or DOBJ - Definition of Data Destruction Objects). Navigate here to the **Structure definition** folder. In Figure 7.14, for example, you can see that the table `BKPF` is the header table; it is at the top of the list.

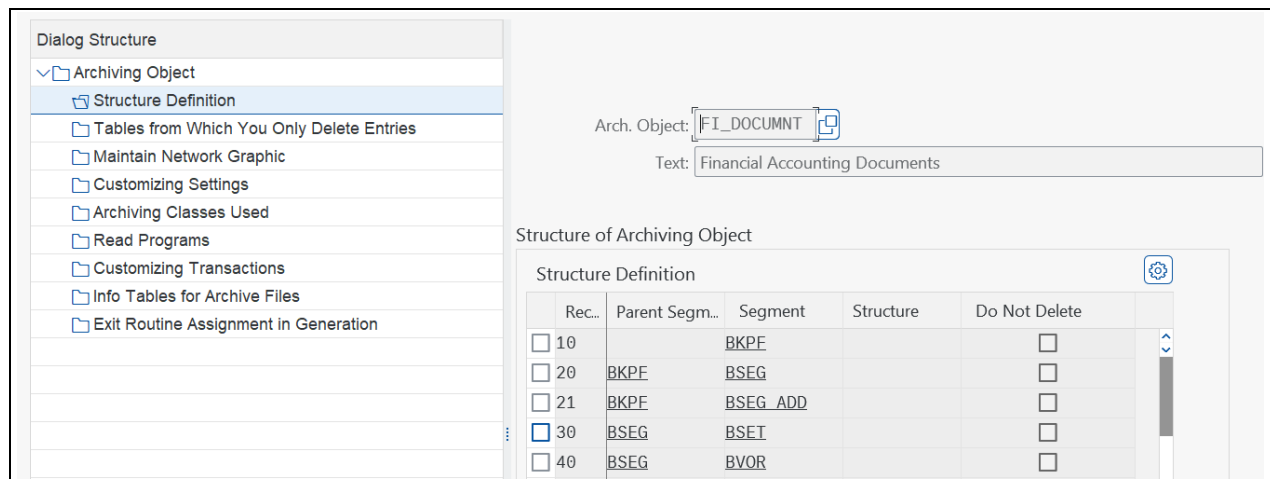


Figure 7.14: Structure of an archiving object (transaction AOBJ)

The residence period is stored in the application-specific customizing of transaction **SARA** (Archive Administration) or defined by using the ILM object FI_DOCUMNT. The retention rules were maintained in transaction **IRMPOL** (ILM rules) (see section 2.5.5, “Differences between residence and retention rules”).

Transaction **SARA** is used to process the object step by step. If the Preprocessing button is visible, a preprocessing program is available for the object and should be executed. For many archiving objects, the data is checked by the preprocessing program and final statuses are set if necessary. For the archiving object FI_DOCUMNT, Figure 7.15 shows the **Write**, **Delete** and **Postprocessing** steps to be executed. There is no preprocessing program.

The write report, which starts the Write step, performs technical checks to ensure that the data can be archived/destroyed and applies residence and retention times. Details on the checks can be found in the documentation of the archiving objects. In the job variant of the write reports, the technical scope is defined, i.e. the parameters, e.g. company codes, document numbers, periods and similar.

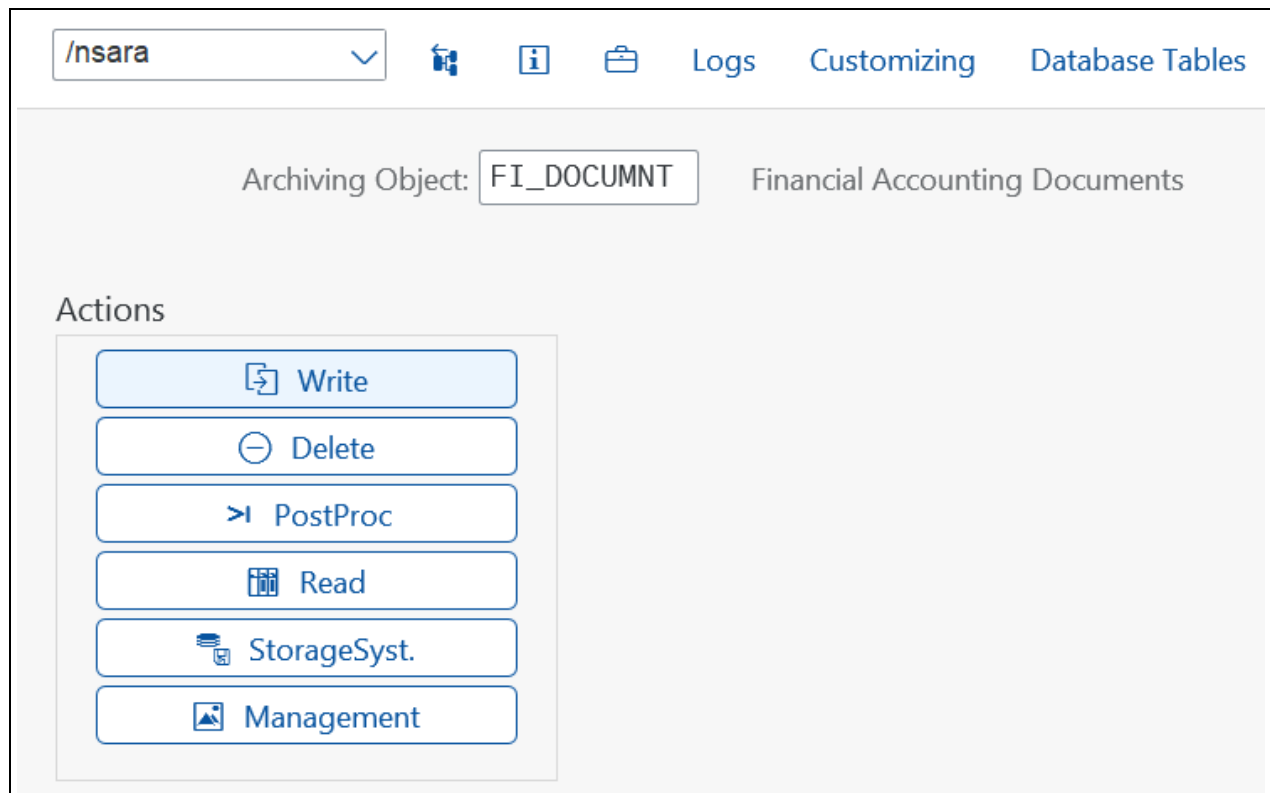


Figure 7.15: SARA initial screen for the FI_DOCUMNT archiving object

Variant Attributes

More

Financial Accounting Documents

Company Codes:

to:

Document Numbers:

to:

Fiscal Year/Period:

2012

To:

2013

99

Document Type:

to:

Selections

Min.No.of Days in the System:

Key Date:

ILM Actions

Archiving:

Snapshot:

Data Destruction:

Processing Options

Test mode

Production mode

Detail Log:

Complete

Log Output:

List and application log

Archiving Session Note:

Figure 7.16: Selection screen of the SARA write report for FI_DOCUMENT

You can see the **ILM actions** group in the selection screen (see Figure 7.16). This indicates that the corresponding ILM object has been assigned to an audit area in transaction **ILMARA**. Select the action **data destruction** data, as in this example the data is not to be archived, but destroyed immediately (see Section 2.3.6, “ILM Actions.”).

The **processing options** group offers you the option of simulating the run in test mode. Then select the level of detail of the log in the **Detail log** field (see Figure 7.17). We recommend that you generate a log that is as complete as possible in the test phase. You should at least check whether this level of detail is necessary in the productive version.

Detail Log:	No Detail Log	▼
Log Output:	No Detail Log	
Archiving Session Note:	Without Success Message	
	Only success messages	
	Complete	▲

Figure 7.17: Selecting the level of detail of the output protocol

Choosing option **complete** of the output log, you receive both messages that a destruction would be successful and information on destruction refusals. These error messages each contain the document numbers concerned and the reason for the non-destruction. The errors issued must be analyzed and rectified if possible/necessary. If necessary, data must then be cleansed or processes adapted.

Note: Dealing with error messages in the write program of the SARA transaction

In every ILM project, error messages are displayed in the log at the latest during the planned destruction runs, indicating that the data does not meet the expected criteria for destruction. This can have many causes, e.g. previous migrations, incompletely executed processes (status not set to final) or similar. If these “inconsistencies” were not a problem in the past, they are now proving to be a major hindrance. It is necessary to check what exactly is causing the error message in SAP ILM and why the data is incorrect. It is then necessary to correct the data and possibly even change the business processes so that the data is of better quality in future and can be destroyed without any problems. Our recommendation is therefore to place a separate data cleansing project alongside the ILM implementation project.

If the tests are satisfactory, writing can be carried out in production mode. The customizing of transaction **SARA** for the archiving object now decides whether the delete job and the post-processing program are started automatically after writing or whether a manual start is necessary (see Figure 7.18).

The screenshot displays two configuration windows. The top window, titled 'Settings for Delete Program', contains fields for 'Test Mode Variant' (SAP&TEST) and 'Live Mode Variant' (SAP&PROD), each with a 'Variant' button. Below these is the 'Delete Jobs' section, which includes three radio buttons: 'Not Scheduled' (selected), 'Start Automatically', and 'After Event'. The 'After Event' option is accompanied by 'Event' and 'Parameter' input fields. The bottom window, titled 'Settings for Postprocessing Program', features a 'Live Mode Variant' field (SAP&PROD) with a 'Variant' button and a 'Start Automatically' checkbox.

Figure 7.18: SARA customizing/archiving object-specific customizing

You can check the successful destruction of the data for the FI_DOCUMENT object via the functional transaction (e.g. transaction **FB03**), in which the corresponding documents are displayed. The document display should lead to an error message, as the deleted document cannot be found. Alternatively, you can use transaction **SE16** (Data Browser) to check directly in the header table (here BKPF) whether the document number can be found.

Step 3: Destroy master data

In this section, we assume that the master data to be destroyed has already been blocked as described in the section “Blocking master data”. We also assume, that the transaction data that is checked in the EoP check has also been successfully destroyed. The personal data of our master data objects should now be destroyed.

A technical description of how to destroy master data can be found in section 2.7.9, “Examples of deleting master data”. In this section, we would like to give you an overview of the basic procedure. As an example, we will take the data of a customer (archiving object FI_ACCRECV).

When a customer is created in the system, the general data is created first, then the company code data (FI data) and finally the sales data (SD data). These data levels are now also processed one after the other during destruction.

As described above, blocking the customer is a prerequisite that has already taken place in our example and the CVP_SORT table has been filled. In this table, the date of the start of the retention period in the ST_RET_DATE field is filled with the last determined date of each application. SAP ILM uses this date when determining the retention period for each application. In the example below in Figure 7.19 you can see the date 16.06.2016 for the application ERP_SD in table CVP_SORT.

	ID	ID_TYPE	BUKRS	VKORG	APPL_NAME	ST_RET_DATE
<input type="checkbox"/>	0001000010	1	1000	0001	ERP_SD	16.06.2016

Figure 7.19: Table CVP_SORT with Start of date of the start of the retention time in field ST_RET_DATE

In the ILM rule for the retention period (CA_BUPA test area), a period of ten years is defined for customers in Germany (see Figure 7.20).

Application Name	Application Rule Var	Min.Retention Period	Max.Retention Period	Retention Prd Unit	Time Ref.	Time Offset
ERP_CUST		6	6	Month	▼ Start of Retenti...	End of Month
ERP_SD	SD_DE	10	10	Year	▼ Start of Retenti...	End of Year

Figure 7.20: Transaction IRMPOL - Retention period of the ERP_SD application

If we now calculate the destruction date with reference to the retention rule, this means the following for the customer's SD data:

$$16.06.2016 + 10 \text{ years} = 16.06.2026$$

The time offset “end of the year” turns this into 31.12.2026. This means that the destruction can be carried out from 01.01.2027.

The same procedure applies to the other applications. Only when all applications have reached their destruction date can the customer be finally destroyed. These calculations are always carried out at the runtime of the destruction run using the ILM rule that is productive at that time.

To destroy the data, use transaction **SARA** with the archiving object FI_ACCRECV. Here you select the ILM action **Data Destruction** is. In the selection screen, select the named data levels one after the other (from bottom to top), i.e. first the SD data, then the FI data and finally the general data.

After successfully executing the write run and the delete run in transaction **SARA**, check the job log again and the deletion of the table entry in the customer data header table:

- for the SD data in table KNAV
- for the FI data in table KNB1
- for the general data in table KNA1

A corresponding error message should now also be generated when the customer is displayed in transaction **FD03**.

7.4.6 Special Cases and How to Implement Them with SAP ILM

This is a new section, not present in the original edition. It follows the previous new section.

SAP ILM can be customized to effectively handle some special cases with appropriate creation of retention rules and BADl implementations. Below we list some examples of such special cases.

Erasure On Request - The Right To Be Forgotten

One of the most urgent tasks for data controllers is the processing of ad hoc erasure requests from data subjects. According to the General Data Protection Regulation, the data controller must take measures to comply with such erasure requests. In the context of ILM implementation, this means that the personal master data records of the data subjects must be blocked and destroyed.

Generally speaking, the GDPR is a subordinate law. This means that there must be absolutely no legal or contractual reasons against destruction if a customer is to be blocked or destroyed. Here are two examples of possible scenarios:

- A customer has bought something and is annoyed about whatever and wants to be deleted. Under no circumstances can this customer be blocked in the current financial year. The data can be destroyed at the earliest after the end of the statutory retention period. When he can be blocked also depends on his purchase, for example. At the very least, it should not be blocked before the end of the statutory warranty period.
- A customer has registered with a company portal to view offers. The offers are not of interest to him, so the customer wants his data to be destroyed. This should be done immediately at his request, i.e. within 30 days.

What else needs to be considered? Customers, suppliers and business partners should be blocked if there are no open transactions and no business has been conducted for a long time. Here is another example: A customer buys goods once and the statutory warranty period is five years (e.g. warranty under the German Construction Contract Procedures, VOB). In this case, the customer should not be blocked for at least five years.

In general, the following special cases apply to customers, suppliers or business partners:

- There are customers, suppliers or business partners who are created in the system but for whom no business is generated. The following should be noted here:
 - This master data should be blocked after a certain period of time - which may well be two to three years - and destroyed immediately afterwards.
 - If a customer, supplier or business partner wishes to be deleted, a field to be defined is set. The master data can then be blocked and destroyed the next day. To do this, you need a field in which you can note the customer's request. Normally, this would be the deletion flag for the customer or supplier. However, this field is usually used in a different context. Therefore, the only remaining option is to add a customer-specific field and implement a new condition field for the three master data or ILM objects CA_BUPA, FI_ACCPAYB and FI_ACCRECV.
 - It is also possible that the person is an agent or has been assigned to a project as a supervisor, project developer or similar. This does not result in a transaction directly assigned to this person. However, the personal data may neither be blocked nor destroyed. Here, for example, the

NODEL indicator can be set for the customer. This means that no EoP check is carried out for this person, so they are neither blocked nor destroyed.

- Finally, there is master data for which transactions have been posted at some point. Here, the retention periods of the transaction data determine when the customer is destroyed. In this case, the blocking of a customer also depends on the transaction data. For example, the master data object should not be blocked during the term of a warranty period.

Regardless of the general, regular and recurring blocking runs, the blocking program for personal master data can also be part of a company process to implement the “right to be forgotten” of the General Data Protection Regulation. Either such a block is possible directly, or information is provided about which application in which system objects to a block. This is the result of the application-specific EoP check and can in turn be important information to pass on to the requester who wishes to exercise their “right to be forgotten”, or can also lead to follow-up activities in the respective application.

The following steps must be carried out so that the corresponding master data can be blocked immediately and then destroyed:

1. In the case of a customer request for deletion, the indicator to be defined must first be set for this customer (or supplier or business partner). It would be very helpful if the corresponding deletion flags were also set for the master data. A special feature would be if customers or suppliers are only to be destroyed in individual company codes. You should clarify with your data protection officer whether this is a data protection requirement.
2. To automate this process, it would be helpful to create an implementation for the BAdI BUP_PARTNER_EXLIST (business partner) or the BAdI CVP_EOP_MODIFY_SELECTION (for customers or vendors). This means that not all master data is selected and checked, but only the master data for which the associated persons also want to be destroyed. Of course, these filters must then be used when creating job variants for blocking at the customer's request.
3. If the customer, supplier or business partner has been successfully blocked, you can destroy the master data. For technical reasons, destruction is only possible on the day after the blocking at the earliest.

Example: Destroy Customer on Request

Here is an example of the procedure for destroying a customer: In transaction **SARA** (archive administration), create a variant for the archiving object FI_ACCRECV with the parameters from Figure 7.21. First destroy the SD data, then the FI data and finally the general data (observe the sequence). If the deletion indicator is set, the runtime can be significantly shortened. To do this, use the option **Consider Deletion Indic.**




Data To Be Archived: SD Data		
Customer Master Data		
Customer: <input type="text"/>	to: <input type="text"/>	
Company Code: <input type="text"/>	to: <input type="text"/>	
Sales Organization: <input type="text"/>	to: <input type="text"/>	
Selections		
Min.No.of Days in the System: <input type="text"/>		
Options		
FI link validation off: <input type="checkbox"/>		
SD document validation off: <input type="checkbox"/>		
Consider Deletion Indic.: <input checked="" type="checkbox"/>		
ILM Actions		
Archiving: <input type="radio"/>		
Snapshot: <input type="radio"/>		
Data Destruction: <input checked="" type="radio"/>		
Processing Options		
Test mode: <input type="radio"/>		
Production mode: <input checked="" type="radio"/>		
Detail Log: Complete		
Log Output: List and application log		
Archiving Session Note: <input type="text"/>		

Figure 7.21: Destroying customer master data - data level and deletion flag

4. The business partner can now be destroyed with the archiving object CA_BUPA. This object checks whether the corresponding customer or supplier master data has already been destroyed.

Archiving Without Retention Rule

The definition of retention rules is a necessary step for the use of ILM data destruction. The retention rules are evaluated not only for the correct business decisions, but also for the technical requirements. If no applicable retention rule can be identified, SAP ILM ensures that processed data objects are not destroyed. This system behavior occurs both with direct ILM data destruction and with archiving used by SAP ILM if the ILM object of the archiving object has been assigned to an audit area.

Occasionally, the data archiving process can be impaired. For example, country-specific retention rules should be maintained for financial postings (ILM object FI_DOCUMENT). In some countries, however, the retention rules have not yet been finalized. Nevertheless, the data archiving procedure is to be resumed or continued in order to control the data volume. SAP Note 3156148 (ILM: Set expiration date of archive files

from infinite value to finite) offers a practicable solution. This note describes that the end of the minimum retention period (expiration date) can be marked as “unknown” (“9999”) during the archiving process if the applied rule for the minimum retention period has an empty value.

Archived data (ADK files) with an unknown expiry date can be converted using the ILM conversion report RSARCH_CONVERT_TO_ILM. As a result, the expiry date can be recalculated according to the updated ILM retention rules. Alternatively, transaction [ILM_CHANGE_RET](#) (change expiration date) is provided to set the end of retention directly.

Note: No shortening of the retention period

The ILM conversion report cannot shorten the end of the retention period. It is therefore strongly recommended to test the retention rules so that archive files with an unknown expiry date are created, if the final retention period is not yet known. In this context, also read SAP Note 3156148 to understand when the expiration date is considered “unknown”.

Blocking Transactional Data

In this chapter, we have focused on the blocking and destruction of personal master data. Blocking them means that the personal information of the blocked personal master record, including the personal information used in the transition documents, cannot be made accessible to end users. However, a personal master record can only be blocked when it is no longer required in all relevant applications.

If the customer, supplier or business partner repeatedly generates a new transaction, the personal master record cannot be blocked in the foreseeable future. This means that all historical transaction data is still visible in the system.

The ILM_BLOCKING business function is a practical solution for restricting access to historical transaction data, even if the person master record is still “active” and the current transaction data can be displayed. However, the introduction of this functionality should be carefully considered. The application is based on the archiving of transaction data and requires corresponding ILM customizing. Appropriate authorizations are required to access this archived and blocked data. Before implementing such a scenario, you should consider very carefully whether a customer/supplier/business partner has an interest in such data not being blocked. For example, it can be very helpful for a customer to be able to view all orders from the last few years. A data protection-compliant implementation of a very large American online retailer allows the customer to access all orders since at least the year 2000.

Example: Payment batch

Transfers to a bank account are usually imported into the SAP system and summarized there as a payment batch. After processing, e.g. after posting, the data is no longer required. Such payment batches can now be destroyed promptly or archived for security, blocked and only destroyed later. If you opt for archiving, you can protect this data from general access.