# Infor SunSystems Guide to Migrating Data from SunSystems v4

Release 6.3.x

### Important Notices

The material contained in this publication (including any supplementary information) constitutes and contains confidential and proprietary information of Infor.

By gaining access to the attached, you acknowledge and agree that the material (including any modification, translation or adaptation of the material) and all copyright, trade secrets and all other right, title and interest therein, are the sole property of Infor and that you shall not gain right, title or interest in the material (including any modification, translation or adaptation of the material) by virtue of your review thereof other than the non-exclusive right to use the material solely in connection with and the furtherance of your license and use of software made available to your company from Infor pursuant to a separate agreement, the terms of which separate agreement shall govern your use of this material and all supplemental related materials ("Purpose").

In addition, by accessing the enclosed material, you acknowledge and agree that you are required to maintain such material in strict confidence and that your use of such material is limited to the Purpose described above. Although Infor has taken due care to ensure that the material included in this publication is accurate and complete, Infor cannot warrant that the information contained in this publication is complete, does not contain typographical or other errors, or will meet your specific requirements. As such, Infor does not assume and hereby disclaims all liability, consequential or otherwise, for any loss or damage to any person or entity which is caused by or relates to errors or omissions in this publication (including any supplementary information), whether such errors or omissions result from negligence, accident or any other cause.

Without limitation, U.S. export control laws and other applicable export and import laws govern your use of this material and you will neither export or re-export, directly or indirectly, this material nor any related materials or supplemental information in violation of such laws, or use such materials for any purpose prohibited by such laws.

### Trademark Acknowledgements



Publication Information

Release: Infor SunSystems 6.3.x
Publication Date: September 11, 2020
Document code: ss_6.3.x_ss4ss6ng__en-us

# Contents

# About this guide

This guide describes the process of migrating data from versions of SunSystems v4.

This guide does not include the upgrade of SunSystems v5 to SunSystems v6, which is documented in the SunSystems Upgrade Guide.

This guide also includes a checklist at Data Migration Checklist on page 35 which summarizes the steps involved in the migration process, and should be used to ensure all stages have been completed.

### Intended audience

The intended audience of this document are the System Administrators, SunSystems Consultants, external developers and Channel Partner Consultants involved in migrating data and configuring SunSystems.

## Related documents

You can find the documents in the product documentation section of the Infor Support Portal, at http://support.infor.com.

## Contacting Infor

If you have questions about Infor products, go to Infor Concierge at https://concierge.infor.com/ and create a support incident.

The latest documentation is available from docs.infor.com or from the Infor Support Portal. To access documentation on the Infor Support Portal, select **Search > Browse Documentation**. We recommend that you check this portal periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com.

# Chapter 1: Introduction to migrating data from SunSystems 4

This guide covers the migration to SunSystems v6 from the following SunSystems versions:

- 4.4
- 4.3.3
- 4.3.2
- 4.3 (4.3.1)
- 4.2.6
- 4.2.5
- 4.2.4
- 4.2.3
- 4.2.2
- 4.2.1

Before migrating data, ensure that you have planned how you want to set up your Business Unit and analysis of data in SunSystems v6. Also, ensure that you have read the appendices for more details on how data is migrated from SunSystems v4 tables or files into the SunSystems v6 tables.

**Note:** The databases created in SunSystems v4, using the function DB=Database Definition, are referred to as Business Units in SunSystems v6. The Journal Hold File should be empty before you start migrating data, and a new Balance File is created in SunSystems v6. You must create a Business Unit in SunSystems v6 for every Database Definition record in SunSystems v4.

As soon as you start migrating a business unit, referential integrity is automatically switched off. You are required to switch referential integrity back on at the end of the migration process.

You can only migrate one business unit at a time. Repeat the process described in the following sections for each of your SunSystems v4 databases that you wish to migrate.

The migration process consists of the following steps:

- Backup SunSystems v4 data
- Set up a new Business Unit in SunSystems v6
- Create Analysis Dimensions in SunSystems v6
- Run the Data Migration Wizard
- Migrate Users and Groups
- Run Data Migration (DMG)
- Manual corrections to the SunSystems v6 database.

These steps are described in the following chapters.

**Note:** A checklist for data migration is provided in this guide at Data Migration Checklist on page 35.

# Chapter 2: Migrating SunSystems v4 non-English databases

In SunSystems v4, a Language Code is always associated with the data files/tables.

If you are serialized for a single language version of SunSystems v4 in English, you do not need to read this section and can skip to Stage 1 of the migration process.

If you are serialized for a single language version of SunSystems that is not English, there is a Language Code associated with all your data files specific to the language used.

If you are using a multi-lingual version of SunSystems v4, your data files/tables can be associated with any of the different Language Codes you are serialized for. Note that the Language Code associated with your `SSINSTAL` file/table is the language defined as the Base Language of your system; however, the Language Code associated with your database(s) files/tables can be any one of the languages you are serialized for.

**Note:** If you are not sure of the Language Code associated with your SunSystems v4 database(s), create a new operator in OD=Operator Definition leaving the Operator Language field blank. Log in to SunSystems v4 with this operator. In this SunSystems module (main menu), the user interface is the Base Language of your system. However, if you go to any of the other modules, for example, LA=Ledger Accounting, the user interface is in the language of that database.

The Language Code associated with the data files determines the language in which specific pieces of data are stored. The specific data in question is mainly the valid values for flag/parameter type fields, for example, the Account Type for an account record in a French Reference file/table is stored as F (for Fournisseur) and in an English Reference file/table it is stored as C (for Creditor).

Only SunSystems v4 backup files that are associated with a Language Code of English (Language Code 01) can be migrated into SunSystems v6. When using FB=File Backup Restore function to backup your data you need to ensure that you are logged in as an operator whose Operator Language is set to 01 (English). The resulting backup file is associated to the Language Code 01 and specific data are converted to English.

If you are using a multi-lingual version of SunSystems v4 that does not include the English language or a single language version that is not English, you should contact your supplier. You will be provided with a temporary multi-lingual serialization, which allows you to create an English operator and backup your data with this operator, as described above.

If you are using a non-English version of SunSystems v4, you are likely to use extended characters in your data; you should read Code Page Conversion on page 9, in the Stage 1 section of the migration process.

# Chapter 3: Stage 1: Backing up SunSystems v4 Data

**Note:** In order to back up your data for the migration, you must be logged in as an English operator. For more details see Migrating SunSystems v4 non-English databases on page 8.

Log in to SunSystems v4, go to OD=Operator Definition and check that the Operator Language field for the Operator Code you use to back up your data is set to 01 (English). Log in to SunSystems v4 with this Operator Code and use the function FB=File Backup/Restore to back up your data files.

Leave the Database Code field blank and back up your Installation File.

Enter the Database Code of the database you want to migrate to SunSystems v6 and then back up the Reference File, the Ledger File, Budget File(s) and Archive File(s).

**Note:** The installation file `SSINSTAL.BAK` must be present.

The default backup folder is `_BACK` under the SunSystems v4 installation folder. However, you can specify an alternative location for your backup files using Operator Setup (OS). Amend the Operator Setup for the operator carrying out the migration and enter an alternative path in the Operator Work Drive field. The path name you enter as the Operator Work Drive must be no longer than 10 characters in length and the path name must end in a backslash. The path must not contain embedded spaces.

For example, `E:\TEMP\` is a valid path name; `E:\TEMP` and `E:\TEMP A\` are not valid.

Copy your SunSystems v4 backup files to your SunSystems v6 backup directory.

All backups must be in the same location, including `PARAM` and `PAY` files (created during Stage 4 of the migration process). If they are not, an 'Open File' error message is displayed when you run Data Migration. Mapped drives included in the path name must be mapped to a location on your hard drive.

If required, you can amend the operator work drive to its original (or another) location after completion of Data Migration.

Refer to your SunSystems v4 documentation for further details on backing up SunSystems v4 data.

## Code Page Conversion

Code Page Conversion may be required for the SunSystems v4 backup files you want to migrate to SunSystems v6, for versions of SunSystems v4 prior to version 4.4. This is necessary if your data uses extended characters, which are typically used in European (non-English) languages.

Your .BAK data file is typically in an OEM Code Page. For example, OEM 850 is for Western European languages. You need to convert the .BAK file to the code page of the destination SQL database.

SunSystems provides a utility called CODEPAGECONVERTER.EXE which can be downloaded from the Infor Support Portal.

### CODEPAGECONVERTER.EXE Utility

The following instructions guide you through the standard code page conversion of an OEM 850 file to a Windows 1252 file for a Western European language system.

**Note:** Systems with Eastern European languages must have their own character sets created for each code page.

When prompted, enter a folder name, that is, the current location of the unconverted backup files; and an output folder name, that is, the location where it stores the converted files. These folder names cannot be the same.

If the SunSystems v6 backup file location of \Infor\Logs\SunSystems\Cobol is not specified as the output folder, the converted files must be moved into that location, or the folder you specified for the operator in User Manager, before proceeding with the next stage of Data Migration.

To run this utility:

1. On the SunSystems v4 installation CD, select CodePageConverter.exe.
2. Input Folder - Select the source folder that contains the files to be converted.
3. Input Code Page - The default is set to convert code page 850. If an alternative code page is required, select the appropriate OEM code page from the dropdown menu.
4. Output Folder - Select an appropriate empty output folder for the converted files. For example, C:\Program Data\Infor\Logs\SunSystems\Cobol.
5. Output Code Page - The default is set to convert to code page 1252. If an alternative code page is required, select the appropriate OEM code page from the dropdown menu.
6. Click Convert to convert all the files in the Input Folder.

A message is displayed to confirm completion. The converted files are then ready for the migration to SunSystems v6.

**Note:** If not already located there, the located files must be moved into the SunSystems default backup file location of \Infor\Logs\SunSystems\Cobol, or the folder you specified for the operator in User Manager.

# Chapter 4: Stage 2: Setting up a new Business Unit in SunSystems v6

You must create a separate business unit in SunSystems v6 for each SunSystems v4 database to be migrated.

Access Business Unit Administration (BUA) and create your new business unit. Enter a Business Unit Code. This code does not have to be the same as your SunSystems v4 Database Code, but it must be 3 characters long, in uppercase and should not start with a number.

**Note:** For single currency, Value 1 must be set to 2 - Must Be Entered, with Values 2, 3 and 4 all set to 0 - Undefined.

For multi-currency, Value 1 must be set to 3 - Calculate If Not Entered, Value 2 should be set to 1 - Only Present if Entered, Values 3 and 4 set to 0 - Undefined.

For more information on setting up business units, see the SunSystems v6 Help >> Administrator Guide.

# Chapter 5: Stage 3: Migrating Users and Groups

There is no automated way to convert Users and Groups from version 4 to version 6. The version 6 Users and Groups have to be recreated using User Manager or Security Console.

Because the details and structure of Groups in version 6 is different to that in version 4, the setup of the groups needs to be redesigned for version 6. This is also an opportunity to review all the users, groups and their settings and to eliminate any redundant entries.

# Chapter 6: Stage 4: Creating Analysis Dimensions in SunSystems v6

1. Log in to User Manager with the administrator ID, and create a new user with full access to SunSystems.
2. Log in to SunSystems using the Operator ID you created in step 1 above.
3. Use SunSystems Navigator to access Analysis Dimension (AND).
4. Create as many Analysis Dimensions for your new business unit as you need to transfer your SunSystems v4 Analysis Categories. For each Analysis Dimension that you create, make a note of the Analysis Dimension ID that the system automatically allocates.
5. Use SunSystems Navigator to access Analysis Structure (ANS) and then define how you want to analyze your data, that is, which analysis dimensions are to be applied to which SunSystems v6 data tables.

**Note:** Defining the analysis structure is not strictly part of the data migration. However, it must be done before the new business unit can be used correctly; it is therefore recommended to carry out this step now, that is, directly after defining the analysis dimensions.

You can attach your new analysis dimensions to different analyzable entities, which can be considered as SunSystems data tables:

| Analyzable Entity Table ID | Description |
|---|---|
| 1 | Item |
| 22 | Inventory Issue |
| 31 | Inventory Receipt |
| 122 | Warehouse |
| 132 | Employee |
| 138 | Address |
| 142 | Sales Invoice Line |
| 148 | Supplier |
| 150 | Customer |
| 152 | Chart of Accounts |
| 177 | Purchase Invoice Line |
| 191 | Sales Order Line |

| Analyzable Entity Table ID | Description |
|---|---|
| 204 | Fixed Asset |
| 214 | Purchase Order Line |
| 233 | Movement Order Line |
| 901 | Ledger |
| 931 | Receipt Note Line |

# Chapter 7: Stage 5: Running the Data Migration Wizard

The Data Migration Wizard assists you in creating the two migration parameter files, `PAY-ZZZ.TXT` and `PARAM-ZZZ.DAT` where `ZZZ` is the SunSystems v4 database code. These files are subsequently used in stage 5 of the migration process.

In SunSystems v6, the payment terms are stored in groups called Payment Terms Groups, each requiring its own unique group code. This is done by the Data Migration Wizard, which extracts the payment terms information from the SunSystems v4 backup files, and assists you to enter each unique group code. This information is stored in the file `PAY-ZZZ.TXT`. See Payment Terms File Format on page 32 for more details of this file.

The information you enter in the wizard is also used to generate the `PARAM-ZZZ.DAT` file. This file contains the mapping from SunSystems v4 analysis categories to SunSystems v6 analysis dimensions so that all of the analysis data can be migrated. It also contains the codes you want to substitute for default transaction analysis codes, default bank sub-codes, and default asset sub- codes (when they are blank in SunSystems v4). For more information, see Migration Parameters File on page 33.

To run the Data Migration Wizard:

1. From SunSystems Navigator, select Data Migration Wizard (WDM).
2. Follow the on-screen instructions.

**Note:** Any mandatory fields that are incorrectly entered, or are incomplete when you click Next, are identified by a red '!' exclamation symbol.

# Chapter 8: Stage 6: Data Migration (DMG) in SunSystems

Before running Data Migration (DMG) you should move or copy your SunSystems v4 backup files (and any converted files) to `C:\Program Data\Infor\Logs\SunSystems\Cobol`.

Log in to SunSystems v6 and run Data Migration (DMG).

## The Data Migration (DMG) Form

The following fields are displayed on the Data Migration form:

**Database Code SunSystems v4**
Enter the database code for the SunSystems v4 database that you wish to migrate.
**Note:** The databases created in SunSystems v4 versions, using the function DB=Database Definition, are now referred to as business units in SunSystems v6.

**Business Unit Code SunSystems 6**
Enter the v6 business unit code for the migrated data.

**Static/Payment Terms**
Indicates the number of table rows migrated and not migrated after the Static data migration button has been clicked. The number of Payment Terms table rows is also displayed.

**Transaction**
Indicates the number of table rows that were successfully migrated after the Trans data migration button has been clicked.

**Archive**
Allows you to enter specific, or a range of, years before clicking the Archive data migration button.

**Data Check Button**
Checks for invalid dates in the Ledger backups and for lower/mixed case codes in the Static and Ledger backups and validates bank and asset entries, and analysis code lengths to ensure they are not longer than the SunSystems v6 analysis dimension they are migrated to and address/account relationships.

**Apply RI Button**
This enables referential integrity to be applied to the specified business unit tables and system tables.

**Note:**

- After you have migrated the data, you must run the Payment Terms Update (PYU) utility. This ensures that all the due dates in the ledger file are updated.
- All log files (`.log`) and work files (`.wri`) generated by the system are stored in folder: `C:\Program Data\Infor\Logs\SunSystems\Cobol`
- This folder is normally hidden. If you cannot see them in Windows Explorer, select the 'Show hidden files, folders, and drives' option in Tools >> Folder Options.

## Data Check

Enter the SunSystems v4 database code and your new SunSystems v6 Business Unit Code.
**Note:** If your SunSystems v4 database code starts with a numeric character this causes an error to occur during the migration process. You must rename it before starting the migration process. Once renamed you must then manually open the SSINSTAL file and edit the DBAnnn record, updating the two occurrences of the database code on the record line to the renamed database code.

Before migration takes place, all files need to be checked. Select Data Check and enter the Business Unit code. Data Check:

- searches for invalid dates in the ledger backups.
- searches for lower/mixed case codes in the Static and Ledger backups.
- validates that all default analysis codes within the `PARAM-ZZZ.DAT` file have an equivalent default code of spaces in the SunSystems v4 Reference file backup.

Any errors are reported in the `DATACHECK-ZZZ.LOG` file.

**Note:** These folders are normally hidden. If you cannot see them in Windows Explorer, select the 'Show hidden files, folders, and drives' option in Tools >> Folder Options.

**Note:** If the rules for automatic conversion of invalid dates during Actual Ledger and Budget Ledger data migration (Step 0) are acceptable, invalid dates within the ledger backups may be left.Correct any errors reported and re-run the Data Check to ensure all errors have been corrected.

## Upper Case Conversion of Codes

The entry of upper case codes is strictly enforced in SunSystems v6. Where a code in SunSystems v4 can be in upper case, lower case or mixed case, Data Migration ensures that it is converted to upper case in SunSystems v6.

The Data Check process checks whether any lower or mixed case codes are found in the static or ledger files and reports them to a file called `LCASEFIELDS-ZZZ.WRI`.

If required, edit `LCASEFIELDS-ZZZ.WRI` and amend the upper case equivalent code for the lower/mixed case codes reported.

**Note:** When converting lower case to upper case characters, a conflict could be caused by non- unique codes, in cases where two codes are only differentiated by the case of the characters. For example,

in SunSystems v4 `acct1` and `ACCT1` are allowable as two different account codes; however, as they are both upper case in SunSystems v6 they are not separate codes.

Entries can be left as they are if each upper case code is unique. For example:

```
ACC,ACC,BANK1 ,Bank1 ,Y,Account Code on Chart Of Account record
ACC,ACC,BANK2 ,bank2 ,Y,Account Code on Chart Of Account record
ACC,ACC,BANK3  ,bank3  ,Y,Account Code on Chart Of Account record
```

These entries do not need to be changed because each upper case code is unique.

If two or more codes are not unique, (that is, they have the same code when converted to upper case) you have two alternatives:

- you can amalgamate the multiple codes into one when migrating

or

- you can change the codes in order to migrate all of the multiple codes with the same uppercase equivalent.

## Amalgamating Codes

To amalgamate the codes to migrate only one of the multiple entries:

Decide which occurrence is to be migrated to the table in SunSystems v6, and delete the '`Y`' in the Preferred Occurrence column of all other occurrences.

For example, entries as first reported:

```
ACC,ACC,ACCOUNT1 ,account1 ,Y,Account Code on Chart Of Account record
ACC,ACC,ACCOUNT1 ,Account1 ,Y,Account Code on Chart Of Account record
```

Entries as amended:

```
ACC,ACC,ACCOUNT1 ,account1 ,Y,Account Code on Chart Of Account record
ACC,ACC,ACCOUNT1 ,Account1 , ,Account Code on Chart Of Account record
```

**Note:** The only difference is that the '`Y`' has been deleted from the row that is not to be migrated. In this instance the SunSystems v4 chart of accounts record for code `account1` is migrated, but the record for code `Account1` is not migrated. Thus, all instances of `account1` and `Account1` (for example on ledger lines being migrated etc.) are converted to `ACCOUNT1` in SunSystems v6.

## Maintaining Unique Codes

To migrate both lower and mixed case codes, maintaining separate records:

Modify at least one of the reported entries so that the code to be used in SunSystems v6 (upper case) is unique.

For example, entries as first reported:

```
ACC,ACC,ACCOUNT1 ,account1 ,Y,Account Code on Chart Of Account record
ACC,ACC,ACCOUNT1 ,Account1 ,Y,Account Code on Chart Of Account record
```

Entries as amended:

```
ACC,ACC,ACCOUNT1 ,account1 ,Y,Account Code on Chart Of Account record
ACC,ACC,NEWNAME1 ,Account1 ,Y,Account Code on Chart Of Account record
```

**Note:** In this case both 'Y's have been retained, but one of the upper case codes has been amended to NEWNAME1 so this will be used as the account code in SunSystems v6. You can amend either or both upper case codes, but each code amended must be unique and cannot be the same as an existing code (of the same record type) in the SunSystems v4 data being migrated.

**Table 1: Fields Converted by Data Migration**

| All Occurrences of | Key Fields |
|---|---|
| Account Codes (ACC) | Account Code on Chart of Account Record<br>Account Code on Bank Details Record<br>Account Code From on Daily Conversion Rate Defn<br>Account Code From on Conversion Code Defn Record<br>Account Code on Payment Reference Record<br>Account Code on Ledger Transaction Record (part of) |
| Analysis Codes (ANV) | Analysis Code on Analysis Record |
| Address Codes (ADB) (ADD) | Address Code on Address Analysis Record<br>Address Code on Address Record |
| Bank Details (Account Code) (BKA/ACC) | |
| Bank Details Transaction Reference (BKA) | |
| Tax Details (Analysis Code) (TAX/ANV) | Transaction Analysis Code on Tax Details Record |
| Budget Check (Account Code) (ACC) | |
| Standard Text (ESR/ACC) | |
| Analysis Layouts (GA1/GAN) | Layout Code on Analysis/Statement Layout Record |
| Statement Layouts (GS1/GAN) | |

| All Occurrences of | Key Fields |
|---|---|
| Statement Line Contents (SLN) | Line Content Code on Statement Line Content Record |
| Table Formats (FTF) | Table Code on Table Format Record |
| Table Rows (FTR) | Table Row Code on Table Row Record |
| Table Columns (FTC) | Column Content Code on Table Columns Record |

In SunSystems v4 there is no validation to ensure that the Address Code on the Chart of Account record (when Chart of Account record is a Debtor, Creditor or Client type) is a valid Name and Address record. In SunSystems v6 this is a foreign key relationship. This lack of validation causes Referential Integrity problems in Data Migration. Data Check reports the offending Account/Address codes in the `DATACHECK-ZZZ.LOG`. You are advised to correct the SunSystems v4 data before migration, rather than correct the SunSystems v6 data after migration.

## Static Data

Enter the SunSystems v4 Database Code and your new SunSystems v6 Business Unit Code. Click Static.

Once the migration has finished, the number of rows that were successfully migrated is displayed on the form, as well as the number of Payment Terms rows.

Check if any errors have occurred, by opening the file called `STATREP-ZZZ.LOG`.

## Actual Ledger and Budget Ledger Data

Click Trans (translation).

If any invalid dates exist within the ledger backups, these are converted using the following rules:

- If a corrupt Entry Date is found, then the System Date is substituted.
- If a corrupt, or zero, Transaction Date is found and the Entry Date is not zero, then the Entry Date is substituted.
- If a corrupt or zero Transaction Date is found and the Entry Date is zero, then the System Date is substituted.
- If a corrupt Posting Date is found, then the Entry Date is substituted.
- If a corrupt Due Date is found then the date is set to Null values.
- If a corrupt Allocation Date is found then the date is set to Null values.

Once the migration has finished, the number of rows that were successfully migrated are displayed on the form, as well as the number of Payment Terms rows.

Check details of any automatic date conversions and if any errors have occurred, by opening the file called `LDGREP-ZZZ.LOG`.

## Archive Data

Before migrating archive data, use SunSystems Navigator to access Ledger Setup (LES) and ensure the Ledger Archiving check box, within the Options group on the Transaction Rules tab, is checked.

Enter a range of years using the Archive Year From and To fields, to migrate all your archive data within that range. Alternatively, enter specific year(s) in the Archive Year(s) fields.

Click Archive.

Once the migration has finished, the number of rows that were successfully migrated is displayed on the form.

Check if any errors have occurred, by reviewing the contents of file `ARCHREP-ZZZ.LOG`, where `ZZZ`

is the SunSystems v4 database (business unit).

## Switching on Referential Integrity

Enter the Business Unit code of the SunSystems v6 business unit you wish to apply the referential integrity to.

Click RI. This enables referential integrity to be applied to the selected business unit tables and the system tables.

The application of Referential integrity is a two-stage process. Firstly the specified business unit and system tables are checked for referential integrity errors. If errors are found these are reported and processing is terminated. If no errors are found referential integrity is applied to the tables.

If referential integrity errors are reported these must be corrected before referential integrity is rerun. The errors are held in the table `RI_ERR`. Correct the errors and click Apply RI. See the section below for an explanation of how to interpret the entries in the `RI_ERR` table. Click Cancel to end the Data Migration session before accessing the `RI_ERR` table.

If no errors are reported in the table `RI_ERR`, the referential integrity is applied for the specified business unit and system tables.

Check if any errors occurred during the application of referential integrity by opening the file called `REFINTEGRITY-ZZZ.LOG` (where `ZZZ` is the SunSystems v4 business unit).

## Correcting Referential Integrity Errors

Referential Integrity in SunSystems v6 is much stricter than in SunSystems v4, hence there may be referential integrity errors reported after a migration, as certain data that is acceptable in SunSystems v4 is not allowed in SunSystems v6.

All referential integrity errors occur because a value in a foreign key column on a child table does not occur as data in the primary key column of the parent table. For example, an address record may have

a reference to a chart of accounts code that does not exist in SunSystems v4; however, in SunSystems v6 the chart of accounts record must exist in order to create the reference in the address.

You must correct these errors manually. There are two ways of correcting these errors.

A row must be inserted into the parent table with the data from the foreign key in the primary key.

The foreign key data or row containing the foreign key in error must be deleted from the child table.

The rows in the `RI_ERR` table give the user the information on the parent table, the child table, the child Table column affected and the data missing from the parent table primary key.

The entries in the `RI_ERR` table do not report the primary key value of the child rows in error.

There are various methods of finding these:

- If the number of rows in the child table is small, access the static data maintenance function that maintains the child function e.g. `ZZZ_SUPP` is the Supplier Table, so the static data function is Suppliers Setup (SUS). Click next through the rows until you find the reported value in the reported column.
- For larger sets of data it may be more convenient to access the tables directly using SQL Server.

Use SQL Server Management Studio to display a grid of data for the child table as follows:

Open the server running SunSystems by selecting the server type and name from the drop-down menus. Select the required Authentication and click Connect. Double-click on Databases >> the SunSystems database (for example SunSystemsData) >> Tables. Scroll down the left pane to the child table in error. Right-click on the child table to be interrogated and select Edit Top 200 Rows.

The rows in the table are displayed in a grid. Search for the invalid values in the column reported. The primary key is usually the first column in the grid.

If the table has many rows, use SQL Server Management Studio to query the table and bring back a set of results. To do this, click New Query.

Select the Database from the drop-down list in the menu bar.

Enter the following query:

```
SELECT * FROM AAA
WHERE BBB = CCC
```

where `AAA` = Child Table Name, `BBB` = Child Column Name, `CCC` = The Foreign Key Value.

**Note:** Columns that allow alphanumeric values must have the value in single quotes. Columns that only allow numeric values must not have the value in quotes.

The primary key is usually the first column in the returned data set.

**Note:** You can also right-click on the child table to be interrogated, select Top 1000 Rows and edit the query in the right pane.

# Chapter 9: Manual Corrections to the SunSystems v6 Database after Migration

If, when migrating, the SunSystems v4 business unit name is different to the SunSystems v6 business unit name, some manual adjustments are required. This is because Data Migration cannot convert the Business Unit codes that are held on the following static data: Corporate Allocation Target (CAT) and Financial Analysis (FNA) >> Consolidation. You must amend the business unit codes held on these database records manually using the associated static data maintenance function.

If you are going to be using memo accounts with journal types created in SunSystems v4, you must amend the relevant journal types using the special Posting Overrides version of the Journal Types form, which has additional options on the Security tab. If you do not have access to the Journal Types Posting Override form, you must first enable access to this form in User Manager and then add it to the user group menu using User Group Menu Designer. Detailed instructions for this are available in the Enabling Access to the Journal Types Posting Override Form section of Administrator Help >> Financials Administration >> Setting Up and Controlling Ledger Entry >> Journal Types >> Overriding the Posting Rules for a Journal Type.

On the Security tab of the Journal Types Posting Override form, the Memo Post Rule Override field is set to 0-Undefined; amend it to the required setting. If you do not amend this field, and you use ledger import to import memo amounts, the values are set to zero regardless of how you have set up the business unit.

## Journal Headers

Journal header processing is new functionality in SunSystems v6. Data migration does not create journal headers for migrated transactions. This causes journal listings to display a blank journal type for transactions that were migrated from SunSystems v4 to SunSystems v6. The assumption is that journal headers are only set up for journals created in SunSystems v6.

## Payment Terms Update

It is also necessary to run Payment Terms Update (PYU) for all your account codes after migrating SunSystems v4 data.

This enables the system to calculate and store the due dates of your migrated transactions.

**Note:** In the following circumstances, Due Dates are not migrated:

- If the Due Date is not present in the `SALF-XXX.BAK`.
- If the Account Code is of the following types: Memo, Balance Sheet or P&L.
- If the Allocation Marker is of the following types: Correction, Reconciled, Allocated or Paid.

For further information on Payment Terms Update refer to the Administrator Help.

# Appendix A: Data Migrated from SunSystems v4

Financials Static Data (Per Database)

| SunSystems v4 Record (Reference File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| ACC | Chart of Accounts (Accounts, Suppliers, Customers Payment Terms) | ZZZ_ACNT<br>ZZZ_ACNT_ANL_CAT<br>ZZZ_SUPP<br>ZZZ_SUPP_ANL_CAT<br>ZZZ_SUPP_TAX<br>ZZZ_CUST<br>ZZZ_CUST_ANL_CAT<br>ZZZ_CUST_TAX<br>ZZZ_PYMT_TERMS_GRP<br>ZZZ_PYMT_TERM | Supplier tables updated if account type is C-Creditor or T-Client.<br>Customer tables updated if account type = D-Debtor.<br>Payment terms will only be present on Creditor, Client or Debtor accounts.<br>When migrating, the system will find the payment term code that matches the details on ACC record. |

| SunSystems v4 Record (Reference File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| ADD<br>ADB | Names and Addresses<br>Address Analysis | ZZZ_ADR<br>ZZZ_ADDR_ANL_CAT<br>ZZZ_CUST_INV_ADDR<br>ZZZ_CUST_DELV_TO_ADDR<br>ZZZ_SUPP_ORDER_ADDR<br>ZZZ_OWN_CO_ADDR<br>ZZZ_OWN_CO<br>ZZZ_CONTACT<br>ZZZ_SUPP_ADDR_CONTACT<br>ZZZ_CUST_ADDR_CONTACT | If D-Debtor address, `CUST_INV` and `CUST_DELV_TO` addresses will be created for each input record.<br>If C-Creditor or T-Client address, `SUPP_ORDER_ADDR` will be created.<br>If input address is 0000000000 this will be mapped to `ZZZ_OWN_CO` where `OWN_CO_CODE` = 1.<br>**Note:** Address code on ACC records is the registered address. |
| FVR<br>FDR<br>FAA<br>FBR | Asset Record<br>Asset Analysis<br>Asset Diary<br>Asset Budget | ZZZ_ASSET<br>ZZZ_ASSET_ANL_CAT<br>ZZZ_ASSET_NOTE<br>ZZZ_DEP_PSET<br>ZZZ_ASSET_BDGT | FAA records map to `ZZZ_DEP_PSET` table. |

| SunSystems v4 Record (Reference File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| AND<br>(Analysis Code Lengths are stored on the Data Dictionary) | Analysis Definitions<br>(Analysis Categories, Financial Reporting Sub-categories) | ZZZ_ANL_CAT<br>ZZZ_ANL_ENT_DEFN<br>ZZZ_ANL_SUBCAT | Users should manually set up Analysis Categories (so that duplicates can be eliminated). This will create ZZZ_ANL_CAT and ZZZ_ANL_ENT_DEFN rows. As part of migration input parameters, users must map old Analysis definitions with the new Analysis Dimensions so that ZZZ_ANL_SUBCAT and ZZZ_ANL_CODE and any other ZZZ_ANL_CAT rows can be written.<br>ZZZ_ANL_SUBCAT is created as part of the migration process.<br>The following are created as part of the installation of the system:<br>• ANALYSABLE_ENTITY<br>• ANALYSABLE_ENTITY_DDD<br>• ANL_DIR<br>• ENT_USEABLE_ANL |
| ANV | Analysis Codes | ZZZ_ANL_CODE | All codes to be migrated including '*' asterisks and suspended ones. It is an error if the defined length in v6 is less than in SunSystems v4. |

| SunSystems v4 Record (Reference File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| BKA<br>BK2 | Bank Details<br>Bank Details Extension | ZZZ_BANK_DETAILS<br>ZZZ_BANK_DE-TAILS_EXT | Records with the following format codes are automatically migrated:<br>• BB - Belgium<br>• EB - Spain<br>• ED - Austria (EDIFACT)<br>• FB - France<br>• GB - Germany<br>• IB - Ireland<br>• IT - Italy<br>• NB - Netherlands<br>Records with other format codes require manual migration. |
| CVD<br>CND<br>CNV | Conversion Definitions<br>Daily Conversion Rates<br>Conversion Tables | ZZZ_CURR<br>ZZZ_CURR_DAI-LY_RATE<br>ZZZ_CURR_PERD_RATE | These must be set to convert from Base Currency to the currency specified in the SunSystems v4 record. This means the signs need reversing, that is, reverse the multiply and divide. |
| LDG | Ledger Definitions and Asset Register Definition | ZZZ_LDG_DEFN | |
| AJD | Journal Definition | ZZZ_JNL_DEFN | |
| AJP | Journal Presets | ZZZ_JNL_PRESETS | |
| SQN | Sequence Number Definitions | ZZZ_LDG_SEQUENCES | |
| BUD | Budget Definitions | ZZZ_BDGT_DEFN | |
| CON<br>GA1 | Financial Analysis Consolidation<br>Financial Analysis Layouts | ZZZ_FIN_RPT_ANL<br>ZZZ_CONSOLI-DATE_PARAMS | |
| GS1<br>SLN | Financial Statement Layouts<br>Financial Statement Rows | ZZZ_FIN_RPT_STMNT<br>ZZZ_FIN_RPT_STM-NT_LINE | |

| SunSystems v4 Record (Reference File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| FTC<br>FTF<br>FTR | Financial Table Columns<br>Financial Table Formats<br>Financial Table Rows | ZZZ_FIN_RPT_COL_SET<br>ZZZ_FIN_RPT_TBL_FMT<br>ZZZ_FIN_RPT_ROW | |
| COL | Column Headings | ZZZ_FIN_RPT_COL_HDGS | |
| PPF | Payment Profiles | ZZZ_PYMT_PROF | |
| TAX | Tax Details | ZZZ_TAX_ACNT | |
| DAY | Daybook Definitions | ZZZ_DAYBOOK | |
| DPH<br>DPT | Dep.Tables Headings<br>Depreciation Tables | ZZZ_DEP_TBL_HDG<br>ZZZ_DEP_TBL_COL | |
| ALD<br>ALR<br>ALS<br>ALT<br>ALC<br>ALX | Allocation Definitions<br>Allocation Ratios<br>Allocation Sources<br>Allocation Targets<br>Calendar Definitions<br>Change in Balance details | ZZZ_CORP_ALL_DEFN<br>ZZZ_CORP_ALL_DEFN_STEP<br>ZZZ_CORP_ALL_RATIO<br>ZZZ_CORP_ALL_SRCE<br>ZZZ_CORP_ALL_TRGT<br>ZZZ_CORP_ALL_CALN<br>ZZZ_CORP_ALL_BAL_CHNG | |

| SunSystems v4 Record (Reference File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| ESR | Payment information | ZZZ_NOTE<br>ZZZ_NOTE_DETAIL<br>ZZZ_NOTE_LDG | For each v4 ESR record that matches a Journal line in the Actuals ledger, a new Note is generated on ZZZ_NOTE. The ESR text is stored in ZZZ_NOTE_DETAIL with the related Journal line key recorded in ZZZ_NOTE_LDG. They are all linked using a unique NOTE_ID. The Account Code, Journal Number, Journal Line and Transaction Reference must match to qualify for ESR conversion to Journal Notes; unmatched or lost ESR entries are detailed in the ESRREP-ZZZ.LOG |
| SPR | Spread Ratios | ZZZ_TIMING_RULES | |

**Table 2: Financials Transactional Data (Per Business Unit / Database)**

| SunSystems v4 Record (Ledger File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| LEDGER-BACKUP-CONTROL-RECORD | Ledger control | SALFMSC | Required for last journal number. |
| LEDGER-BACKUP-RECORD | Ledger transaction | ZZZ_Z_SALFLDG | _Z in table name represents actual ledger code - A for Actuals.<br>The ledger extension table will not be updated by the migration, but will be done later as required by posting programs. |

| SunSystems v4 Record (Ledger File) | Description | SunSystems v6 Table | Comment |
|---|---|---|---|
| LEDGER-BACKUP-RECORD | Budgets - as per ledger | ZZZ_Z_SALFLDG | _Z in table name represents budget ledger code - B-K for Commitments. |
| LEDGER-BACKUIP-RECORD | Archives - as per ledger | ZZZ_A_SALFLDG_A<br>ZZZ_A_SALFLEX_A | |

# Appendix B: Payment Terms File Format

The following table describes the format of the Payment Terms file (`PAY-zzz.TXT`) file:

| Position | Data |
|---|---|
| Character 1 to 3 | PTD |
| Character 5 to 7 | Payment Terms: number of days |
| Character 9 | Payment Terms: type (blank, M-Monthly or N-Net) |
| Character 11 to 13 | Discount 1: Days |
| Character 15 to 19 | Discount 1: Percentage |
| Character 21 to 23 | Discount 2: Days |
| Character 25 to 29 | Discount 2: Percentage |
| Character 31 to 45 | Edit this value: type a Payment Terms Group Code (mandatory) |
| Character 47 to 76 | Edit this value: type a description for the Payment Terms Group Code |
| Character 78 to 92 | Edit this value: type a short description for the Payment Terms Group Code |
| Character 94 to 108 | Edit this value: type a Lookup Code |
| Character 110 to 159 | Edit this value: type a Description for Document |
| Character 161 to 210 | Edit this value: type a Description for Document |
| Character 212 to 261 | Edit this value: type a Description for Document |
| Character 263 to 312 | Edit this value: type a Description for Document |

**Note:** If you open the file with a text editor, be careful not to delete or move any of the commas. Do not leave any extra spaces or lines at the end of the file.

# Appendix C: Migration Parameters File

The information you enter in the Data Migration Wizard is used to create a flat text file (one for each of the SunSystems v4 databases) called `PARAM-ZZZ.DAT`, where `ZZZ` is the SunSystems v4 Database Code. The file is generated in the SunSystems v6 backup folder that you specify in the Data Migration Wizard. This is then used in the Data Migration (DMG).

## Analysis Mapping

The `PARAM-ZZZ.DAT` file contains a record for each SunSystems v4 Analysis Category you want to migrate to SunSystems v6:

- Account Analysis Category A0 to A9
- Analysis Category Asset F0 to F9
- Transaction Analysis Category T0 to T9
- Movement Analysis Category M0 to M9
- Item Analysis Category I0 to I9
- Address Analysis Category C0 to C9

**Note:** You can migrate the SunSystems v4 SunBusiness analysis codes to SunSystems v6 analysis dimensions, however, SunBusiness transactions are not migrated to SunSystems v6.

Each record has a maximum of 19 characters, as follows:

- The first two characters indicate the SunSystems v4 Analysis Category to migrate, for example A0, A1 etc.
- The next two characters indicate the Analysis Dimension Id that the SunSystems v4 Analysis Category is to become in SunSystems v6.
- The next 15 characters indicate the default Analysis Code for SunSystems v6, which replaces the default (blank) Analysis code that existed in SunSystems v4. This code must be in upper case.

**Note:**

- You must ensure that you only have default entries for analysis categories that have a ‘spaces‘ analysis code in your SunSystems v4 backup. Setting defaults for codes that do not exist in your SunSystems v4 `ssrf` backup will cause errors to be reported in the error log produced during Data Check within Stage 6 and, if left uncorrected, will cause the migration to fail.
- There is no comma delimitation in this file. Do not leave any extra spaces or lines at the end of the file.

# Blank Transaction Analysis Codes

There is a record in the PARAM-ZZZ.DAT file to indicate the error suspense account code in SunSystems v6.

Characters 1-6 are LEDGER.

Characters 7-16 indicate the error suspense account for SunSystems v6. Data Migration checks that this is a valid account code in the version 6 Chart of Accounts table. This implies that you must run static data migration before running the ledger transaction migration or set the account up manually in SunSystems v6.

# Blank Bank Sub-Codes

There is a record in the PARAM-ZZZ.DAT file to indicate the default bank sub-code in SunSystems v6 that replaces the default (blank) bank sub-code in SunSystems v4.

Characters 1-4 are BANK.

Characters 5-9 indicate the default sub-code for SunSystems v6, which replaces the default (blank) Bank sub-code that existed in SunSystems v4. Enter spaces to migrate the space codes in SunSystems v4 to SunSystems v6 without the codes changing. If you are changing these codes from spaces, then all five characters must be non-spaces. All alpha characters must be in upper case.

# Blank Asset Diary Codes

There is a record in the PARAM-ZZZ.DAT file to indicate the default Asset Note Code in SunSystems v6, which replaces the default (blank) Asset Diary Note Code that existed in SunSystems v4.

Characters 1-5 are ASSET.

Characters 6-10 indicate the default Asset Note Code for SunSystems v6. This replaces the default (blank) Asset Diary Code that existed in SunSystems v4. Any alpha characters must be in upper case, you need only supply a default Asset Note Code if you are serialized for Asset Register in SunSystems v6.

To cater for the very occasional instances of corrupt data, that is, where a ledger transaction has been found with no account code it is necessary for the user to supply the error suspense account so that Data Migration can substitute this for any corrupt account codes found during the migration. All transactions changed in this way are reported in the LDGREP_ZZZ.log

# Appendix D: Data Migration Checklist

The following checklist should be used to ensure all the required steps in the process have been completed.

| Process | ✓ |
|---|---|
| Stage 1: Back Up SunSystems v4 Data | |
| Stage 2: Set up a new Business Unit in SunSystems v6 | |
| Stage 3: Create Analysis Dimensions in SunSystems v6 | |
| Stage 4: Run the Data Migration Wizard | |
| Stage 5: Migrating Users and Groups | |
| Stage 6: Run Data Migration (DMG) | |
| Post Migration: Manual Corrections to the SunSystems v6 Database | |