LN Engine User Guide

Data Conversion Adapter 3.4 Suite

Copyright © 2023 Infor

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

The word and design marks set forth herein are trademarks and/or registered trademarks of Infor and/or related affiliates and subsidiaries. All rights reserved. All other company, product, trade or service names referenced may be registered trademarks or trademarks of their respective owners.

Publication Information

Release: LN Engine User Guide

Publication date: April 24, 2023

Document code: U9130G US

Contents

About this guide **8**
Intended audience 8
Related documents 8
Contacting Infor 9
Chapter 1 Introduction **10**
Purpose 10
Scope 10
Definitions, acronyms, and abbreviations 11
Need for the Tool 13
Migration tooling overview 14
Conversion solution 14
Integration of Infor LN implementation and data conversion 15
Test conversion 15
User Profile 16
Authorizations 16
LN Engine Components 16
Different data conversion scenarios 16
Chapter 2 Conversion Scenario **17**
Single Machine Scenario 17
Two Server Scenario 19
With client-server connection 19
Without client-server connection 20
Scenario Selection 21
Chapter 3 LN Engine Basics **22**
LN Engine overview 22

LN Dump Engine 22
LN Load Engine 23

Chapter 4 Migration Procedure 25

Migration Procedure Activities 25
Prerequisites 25
The source engine is BaanIVc4, B50c, or DCA Engine 27
The source engine is LN Engine 31

Chapter 5 DCA3.4 Infor LN Dump Engine sessions details 36

Engine Setup 36
Data Dump Parameters (bcdmp0100m000) 37
Source Companies to be migrated (bcdmp9189m000) 42
Conversion Objects (bcdmp0106m000) 42
Generate obcdll objects (bcdmp0107m000) 44
History/Multiple dumping processes per Target Table (bcdmp8571m000) 45
Selection Range per Source Fields (bcdmp8570m000) 47
Range Values per Source Table per Process (bcdmp8572m000) 48
List of Target Tables to be Migrated (bcdmp9188m000) 49
Related Tables of a M-Object (bcdmp9585m000) 53
Update Priority from last conversion run (bcdmp9288m000) 53
Conversion Codes (bcdmp9190m000) 54
Values by Conversion Codes (bcdmp9191m000) 55
Old Values from a table field (bcdmp9291m100) 56
New Values using a DLL functions (bcdmp9291m200) 56
Maintain Pre-Migration Sessions (bcdmp0103m000) 57
Run Pre-Migration Sessions (bcdmp0105s000) 58
Import Pre-Conversion Session from a ASCII File (bcdmp0203s000) 59
Dump Process 60
Dump Data (Create Data Files) (bcdmp9087m000) 61
Attention points 63
Reset the conversion process 63
Number of dump engines 63
Data Files 63
Dumping Daemon (bcdmp9100m000) for LNUI 64
Number of dump engines 66
Data Files 66
Archive Data Dumping Run (bcdmp5201m000) 66

Check currently logged in users (Unix only) (bcdmp5406m000) 67
Check Environment of present Company (bcdmp6400m000) 67
Check Scheduled jobs (bcdmp5404m000) 68
Print Dumping (data file creation) Status (bcdmp9442m000) 70
Dumping Error logs (Functional) (bcdmp9599m000) 70
Data Dumping Runs (History) bcdmp5500m000 71
Dumping Time Logs per Run (bcdmp5501m000) 72
Error Logs - Archive (bcdmp5503m000) 73
Print Dumping Time Logs (bcdmp5401m000) 73
Reset Conversion Process (bcdmp0200s000) 74
Generate scripts to analyse log files (bcdmp5301m000) 74
Analyse Log Files (bcdmp5300m000) 74
Conversion Simulation (Optional) 75
Simulate Data Conversion (bcdmp8101m000) 76
Simulation Run Details (bcdmp8502m000) 76
Simulation Run Details (History) (bcdmp8503m000) 77
Conversion Script Debugging (Logic Pack Development) 78
To debug DCA Generated Script (bcdmpdebug) 78

Chapter 6 DCA3.4 Infor LN Load Engine sessions details 79

Engine Setup 79
Conversion Parameters (bcgct0100m000) (Mandatory) 80
Build Oracle's data loading command (bcgctoracle) 84
Build MS SQL data loading command (bcgct3101m000) 85
Build Informix loading command (bcgct3102m000) 86
Build DB2 loading command (bcgct3103m000) 87
Build MySQL's loading command (bcgctmysql) 88
Build EDB's data loading command (bcgct3105m000) 88
Target Companies to be migrated (bcgct9189m000) 89
Automation 90
List of Session for Migration Wizard (bcgct7186m000) 90
Migration Wizard (bcgctmigwizad) 92
Company and Job Mapping (bccnv2101m000) 92
Process Jobs per Company (bccnv2201m000) 93
Error Handling (bcgct6100m000) 93
Load Process 94
Data Load Dashboard (bcgcteasym000) 95
Batch Codes (bcgc8188m000) 99

Tables by Batches (bcgct8189m000) 100
Generate table List (bcgct8289m000) 104
Verify List of Tables for Completeness (bcgct8489m000) 105
Set Options to Range of Tables (bcgct9285m000) 106
Copy/Synchronize/Export/Import Batch to Companies (bcgct9289m000) 107
Copy From Another Company (bcgct8283m000) 108
Export/Import Batch details (bcgct8288m000) 109
Conversion Runs (bcgct1550m000) 110
Time Taken for Dumping and Loading (bcgct1552m000) 110
Time Taken for dumping and Loading (bcgct1452m000) 111
Number of records in Conversion Phases (bcgct1551m000) 112
Count number of records in Conversion Phases (bcgct1251s000) 113
Event Viewer (bcrun9599m000) 114
Loading Errors (bccnv5500m000) 115
Get Loading Errors (bccnv5200m000) 116
Loading Errors (bccnv5400s000) 116
Delete Data (bccnv5101m000) 117
Generate (bccnv5202m000) 117
Analyze Current Environment (bccnv5495s000) 118
Details of Users connected (bccnv5406s000) 119
Print Scheduled Jobs (bccnv5402s000) 119
Print Loading Status (bccnv9442m000) 119
Select Table Range for Load only without Dumping (bccnv0205m000) 120
Loading into LN (obcrun90505) 121
Loading Daemon (bcldr9100m000) 122
Log Files (bcrun8588m000) 124
Loading Errors (bccnv5500m001) 125
Load Utilities 125
Generate License Key (bcrun0000m0000) 126
Query Verification (bccnv9190m000) 126
Copy Query Data from Another Company (bccnv9290s002) 128
Customer Critical Data Verification Sessions (bccnv9191m000) 128
Copy Session Data from Another Company (bccnv9291s002) 129
Export/Import of Sessions/Query Data (bccnv9290m000) 130

Chapter 7 Special Migration Features 131

To migrate Text Tables (tttxt001, ttxt002, and tttxt010) 131

Split mapping objects using History/multiple dumping process per table 131

For processing History data 132
For splitting the mapping object in multiple ranges 133
Recoding 134
Unicode Correction 135
UTC conversion 135

Appendix A DCA3.x Differences 137

DCA3.2 137
Enhancements 137
Fixes 138
DCA3.3 138
Enhancements 138
Fixes 139
DCA3.4 139
Enhancements 139
Fixes 139

About this guide

This is the user guide for Data Conversion Adapter Infor Ln Engine. This guide helps you to understand the process to migrate data from Infor BaanIVc4, Infor B50c or any Legacy ERP to an Infor LN environment. The guide also gives describes the LN Engine session components in detail. The LN Engine is one of the components of data migration tooling set called Data Conversion Adapter (DCA) suite.

For best results, read this document carefully before you use the Infor LN Engine in a data migration process. In addition, the contents of this document are easier if you have a general understanding of the overall structure of the sessions, packages, and modules of the Baan applications.

Intended audience

This document is for the users who are familiar with Infor LN, have a technical knowledge of the software architecture, and plan to use the DCA suite (Infor migration tooling) to migrate data from Infor BaanIVc4, Infor B50c, or other legacy ERP systems to an Infor LN environment.

Related documents

You can find the documents in the product documentation section of the Infor Support Portal, as described in "Contacting Infor" on page 9.

Data Conversion Adapter 3.4 - Infor LN Engine - User GuideU9130G US

Data Conversion Adapter 3.4 - Infor LN Engine - Installation GuideU9131G US

Data Conversion Adapter 3.4 - Infor B50c Engine – User GuideU9137F US

Data Conversion Adapter 3.4 - Infor B50c Engine – Installation GuideU9138F US

Data Conversion Adapter 3.4 - Infor BaanIVc4 Engine – User GuideU9149F US

Data Conversion Adapter 3.4 - Infor BaanIVc4 Engine – Installation GuideU9150F US

Data Conversion Adapter 3.4 – Legacy Engine – Installation GuideU9096G US

Data Conversion Adapter 3.4 – Data Conversion Adapter Studio – User GuideU9093F US

Data Conversion Adapter 3.4 – Data Conversion Adapter Studio – Installation GuideU9094F US

Data Conversion Adapter 3.4 – Logic Pack – Installation Guide

Infor Solution License Manager - Installation and Configuration Guide U9200G US

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

This chapter provides information about the scope, the purpose, definitions, and other general information required to use the LN Engine.

Purpose

The purpose of this document is to provide the detailed description of the installation process of the DCA LN Engine of the Infor Migration Tooling for a data migration to Infor LN.

The purpose of this document is to provide the detailed description of data conversion process of the Infor LN Engine of the Infor Migration Tooling (DCA) suite for a data migration to Infor LN.

Scope

The document explains all the sessions relevant to the Infor LN Engine, referred as LN Engine from now. The LN Engine loads the data files created by the DCA Source Engines. The used source engine can be BaanIVc4 Engine, B50c Engine, Legacy Engine, or the LN (dump) Engine.

The document explains how users can use the LN Engine sessions to dump out Infor LN data and how to use the LN Engine loading sessions to load migrated data by the source engine.

For detailed descriptions about the DCA source (dump) engines, except for the LN Dump Engine, refers to the DCA Source Engines user guides.

For a detailed description of the available Logic Packs see the Data conversion guide for the required Logic Pack.

To install the Infor LN engine, see the *Data Conversion Adapter 3.4 – Infor LN Engine – Installation Guide (U9131G US).*

Additional Notes:

- Infor DCA LN Engine is available only in English.
- Own customizations, Infor LN localizations, and extensions are not supported. Changes with impact in the LN engine must be handled.
- This conversion software does not handle integrations with external software.

- The DCA LN Engine contains data dump sessions and data loading sessions. For post-migration process with respect to a specific migration scenario, you must procure the respective Logic pack from Infor. For example, these point-to-point data migration logic packs (solutions) are available:
  - Infor BaanIVc4 SP15 to Infor LN.
  - Infor B50c SP17 to Infor LN.
  - Legacy ERP to Infor LN

  The post-migration processes are specific to the point-to-point data migration Logic Pack.
- For the LN dump capabilities, no special Logic pack or post-migration process is available.

Definitions, acronyms, and abbreviations

Below a list of definitions and abbreviations used with the LN Engine Installation and user guide.

Acronyms	Definitions
4c-<number>	This refers to the activity process number in the BaanIVc4 Engine
BaanIVc4 Engine	Data Conversion Adapter 3.4 – Infor BaanIVc4 Engine. This is used to carry out the data conversion program generated by the DCA-Studio and extract the data from the Infor BaanIVc4 environment.
B50c Engine	Data Conversion Adapter 3.4 – Infor B50c Engine. This is used to carry out the data conversion program generated by the DCA-Studio and extract the data from the Infor B50c environment.
BC	Infor LN Conversion package. All the components of the conversion software are under the “bc” package.
BSE/$BSE	BSE refers to the Baan Shell Environment. By means of the BSE environment variable, a link is established to the directory in which all Baan programs and directories are located.
Conversion	In this document, the terms migration and conversion are synonymous. It stands for the migration of data across various release of ERP.
Data Conversion/ Data Migration	The conversion of all the existing company data to the newly installed ERP environment.
DCA	Data Conversion Adapter. It is a set of tools to assist you migrating data from a legacy ERP or Baan ERP environment to an Infor LN environment.
DCA Logic Pack	A Logic Pack is the complete set of components required to handle a Conversion from a specific ERP source version to an Infor LN release. The Logic Pack consist of these components:

Acronyms	Definitions
	• Rules Pack (DCA Studio Rules, Conversion Rule Objects). • Pre migration Components (Parameters, tables, sessions, and programs) to installed at the source environment. • Post migration Components (Parameters, tables, sessions, and programs) to install at the target environment. For example, Logic Pack of Infor BaanIVc4SP15 to Infor LN 10.4. This logic pack consists of the conversion components to Migrate data from BaanIVc4 Service Pack 15 to the Infor LN 10.4 release.
DCA Object Pack	The (DCA) Object Pack is the collection of conversion rules programs as executables and can be installed at the source environment. They are used by the source engine to extract the source date. You can create the object pack from the DCA studio. With the logic pack delivery also a standard Object Pack is available.
DCA Rules Pack	The (DCA) Rules Pack consists of these components: • DCA Studio Rules. • DCA Object Pack. The DCA studio Rules can be imported into the DCA Studio for making customizations to the standard delivered Logic Pack. The (DCA) Object Pack is used by the source environment engines to extract the source data.
DCA Studio	Data Conversion Adapter 3.4 – Conversion Adapter Studio. A tool to create the Logic Pack Rules Pack. The Rules Pack created by the DCA Studio are used by the source environment Engines to do the actual data extract.
Dump	The process by which you dump the data from your source application.
FAT Client	It is also referred as ‘slave/virtual application server’. It is a stripped version of a master application server, and can run both on Windows and UNIX. It points to the master application server for all resources like, data dictionary, application objects etc.
GMT	Greenwich Mean Time.
KB	Infor Xtreme Knowledge Base Article.
LN Engine	Data Conversion Adapter – Infor LN Engine. This loads the data files created by the source environment Engines to the Infor LN application tables.
LN-<number>	This refers to the activity process number in the LN Engine.
Load	The process by which you load data to your target application.

Acronyms	Definitions
Logic Pack	The conversion rules for a specific point to point data migration written in the DCA Studio and the corresponding Pre-migration and Post-migration sessions. For example, Logic pack of Infor BaanIVc4 to Infor LN, contains a Rules Pack, that can be imported to DCA Studio, and, BC package dumps that can be imported to Infor BaanIVc4 Environment (Pre-Migration sessions) and Infor LN Environment (Post-Migration sessions)
M-Object (M-script)	This is a conversion script. On execution, this creates data files for several Infor LN application tables. So m-object is a group of target tables together, and one table in this group is designated as the 'base' table and all other table as 'related tables'
OS	Operating System of the server/machine
Recoding	Recoding means changing your master data codes. For example, you have created your address codes in InforBaanIVc4, in an unorganized manner, you like to put some standard in address codes, and then during data migration, this kind of cleaning process can also happen.
SE-<number>	This refers to the activity process number in the source engine (BaanIVc4 Engine, B50c Engine, Legacy Engine, or LN Engine)
SLM	Solution License Manager. For details on the Installation of the SLM product see *Infor Solution License Manager - Installation and Configuration Guide (U9200G US)*
Software Conversion	The conversion and upgrade of general or in-house customized software components to a new version of Infor LN.
VRC	The Version Release Customer (VRC) module in Baan Tools is used to manage various versions and releases of a package. These four types of VRCs are available: L: Localizations B: Line of business-specific software C: General customizations O: Customer's own customizations

Need for the Tool

The tool enables the current users to migrate Infor BaanIVc4 data, Infor B50c data, or Legacy data to the Infor LN environment quickly, efficiently, and functionally.

Migration tooling overview

To convert data from a source ERP system to Infor LN, Infor created some migration tools and migration content.

In general, you require a Data dump engine at the source ERP environment called the LN data load Engine. Dependent on the source ERP environment there are some pre migration activities required at the source side and some post migration activities at the LN environment. These last two components are delivered via a Logic pack.

Source ERP	Dump Engine	Load Engine	Logic Pack
Infor BaanIVc4	DCA BaanIVc4 Engine	DCA LN Engine	BaanIVc4sp15 to Infor LN logic Pack
Infor B50c	DCA B50c Engine	DCA LN Engine	B50Csp17 to Infor LN logic Pack
Legacy ERP	DCA Engine	DCA LN Engine	A2LN Logic Pack
Infor LN	DCA LN Engine	DCA LN Engine	None (*)

(*) In this case there is no special logic pack defined as the route is primarily used to move and existing Infor LN version to another environment or database with the same Infor LN version. A Logic Pack is used to capture the migration rules for converting data from an ERP source version to a LN target version.

Infor is delivering a standard Logic pack for a point-to-point data migration. Any customizations required in logic pack can be handled by Infor (certified) migration consultant. Please contact your Infor account manager in case you must customize the standard logic pack or to guide you through the total data migration process.

Conversion solution

This table explains various parts of the conversion solution and the delivery format of the bug fixes delivered for each of the parts:

Part	Delivery format	Central Solution*
Infor LN Engine	Base Engine → IEX dump * Update → Cumulative IEX dump	KB 1167478
Post-Conversion**	Base → IEX dump Update → Cumulative IEX Patch	KB 1168115

* The LN Engine Base can be retrieved via de DCA3.4 release media (from the Infor Download side) or via the KB in Infor Xtreme.

** The content of the post migration processes depends on the used Logic pack.

Integration of Infor LN implementation and data conversion

When customers are looking for a new implementation or reimplementation of Infor LN, data conversion must be addressed. The data conversion must be an integral part of the installation and must be included in the overall implementation plan.

Depending on the type of data migration required and the source ERP system an action plan should include these aspects:

- Hardware installation and configuration
- Installation of Infor LN
- Training of end users on Infor LN tools and applications
- Upgrade of current customizations
- Installation of DCA LN Engine and Logic pack
- Preparation of test conversion
- Test conversion
- Data test
- Approve conversion software and migration procedure
- Go/no-go decisions
- Final conversion
- Stabilization of Infor LN
- Infor LN Tuning.

Test conversion

The test conversion simulates the final conversion with these objectives:

- Tune the conversion environment to customizations (if any):
  If the Infor ERP BaanIVc4/Infor ERP B50c version includes customized table definitions, you can request Infor to deliver customized logic pack. In this case, you can only carry out the test conversion after the customizations have been redefined in Infor LN.
- Provide information about the run time of the actual conversion:
  The plan for the final conversion is based on the run time of the test conversion. An estimate of the actual downtime can be made based on the run log information. For more information about run logs, see the User Guide.
- Provide certainty about the progress of the conversion run:
  The final conversion must take place without errors. If any error occurs during the test conversion, the causes must be traced and cleared so that the test conversion can be repeated without errors before the final conversion.
- Provide certainty about the results of the conversion run:
  The results of the test conversion are assessed based on the data and procedure tests. Because

all functions and procedures in the Infor LN application can be used, based on the data of the user's own company, maximum certainty is achieved with regards to the continuity of business processes.

User Profile

You must have good knowledge of Infor LN and these concepts:

- Package combinations, package VRCs and companies.
- The Infor LN data dictionary structure.
- Database definitions, such as physical/logical tables and table sharing concepts.
- The reconfiguration of tables.

Authorizations

You must have the super user privileges and Infor LN systems Administrator permissions to use this tool. Ensure that you have enough disk-space and write permissions to the "*$BSE/tmp/migration"* directory, and directory where you store the data files. These directories are specified in the "Conversion Parameter" (bcgct0100m000) session. You must create this directory in both the DCA Engine and LN Engine environments, in case this directory does not exist.

LN Engine Components

Infor LN Engine is delivered as "bc" package, in modules "cnv", "gct", "run", "dmp", and "gen".

For a complete list of the "bc" package components see Appendix C of the *Infor LN Engine – Installation Guide (U9130G US).*

Different data conversion scenarios

The complete installation of the source engines and the Infor LN engine may depend on the required conversion scenario. Details about the types of conversion scenarios can be found in the next chapter.

Chapter 2 Conversion Scenario

This chapter describes the conversion scenarios in detail with its architecture. The used source ERP environment refers to Infor BaanIVc4, Infor B50c, or Infor LN. The target ERP environment refers to Infor LN.

The data conversion process depends on the deployment environment of the customer system, such as installation of Infor BaanIVc4 and Infor LN on the same server or different servers. The conversion scenario is specified in the Conversion Parameters sessions and the Two-machine scenarios required some extra setup and configuration of directories.

Single Machine Scenario

In the single machine scenario both the source ERP and the target ERP environments are installed on the same Server.

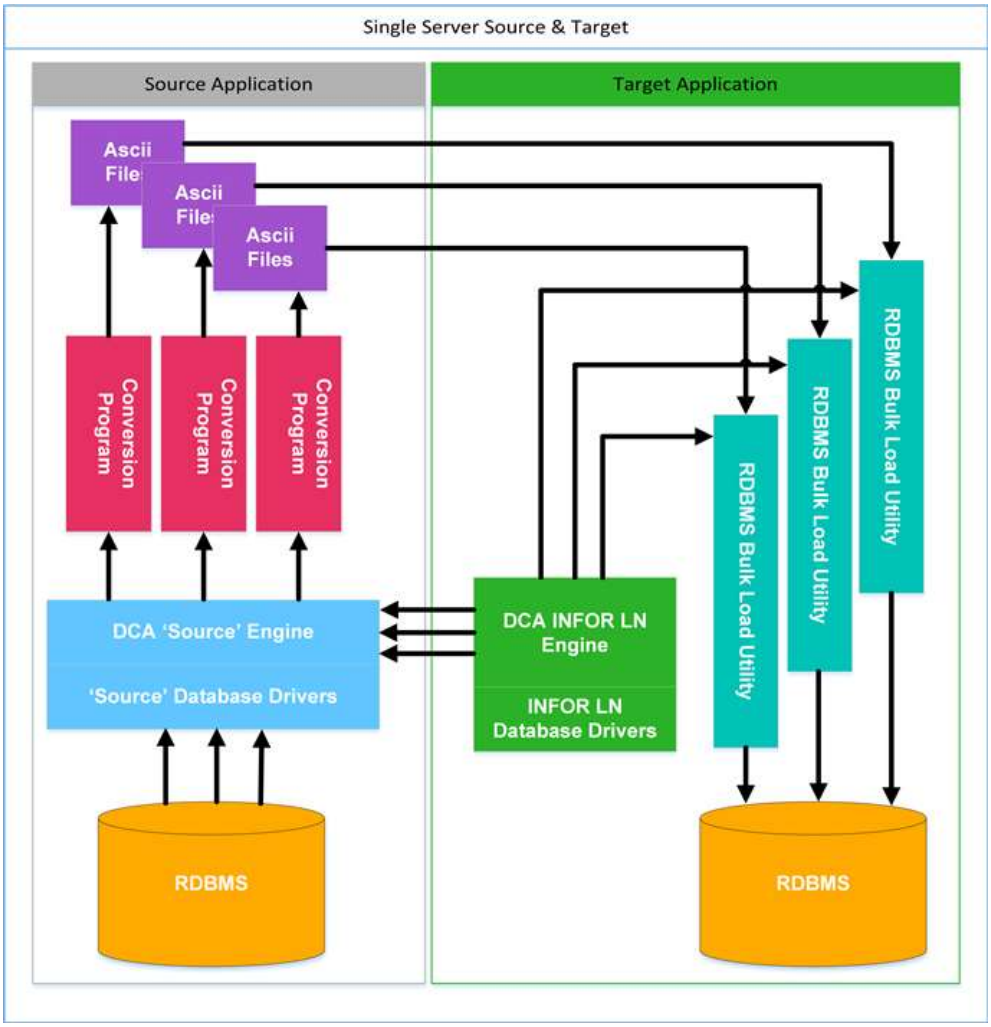

Figure 2-1 Single server scenario

Two Server Scenario

The source and target environment are installed on two separate machines. In this setup these two cases are possible:

- Two server scenario with client-server connection
- Two server scenario without client-server connection

With client-server connection

On the target Environment, a source environment FAT Client is installed. The DCA LN Engine uses this FAT Client to access the resource of actual source environment, to carry out the conversion programs. The BW processes run on the target and the data extraction runs on the source system.

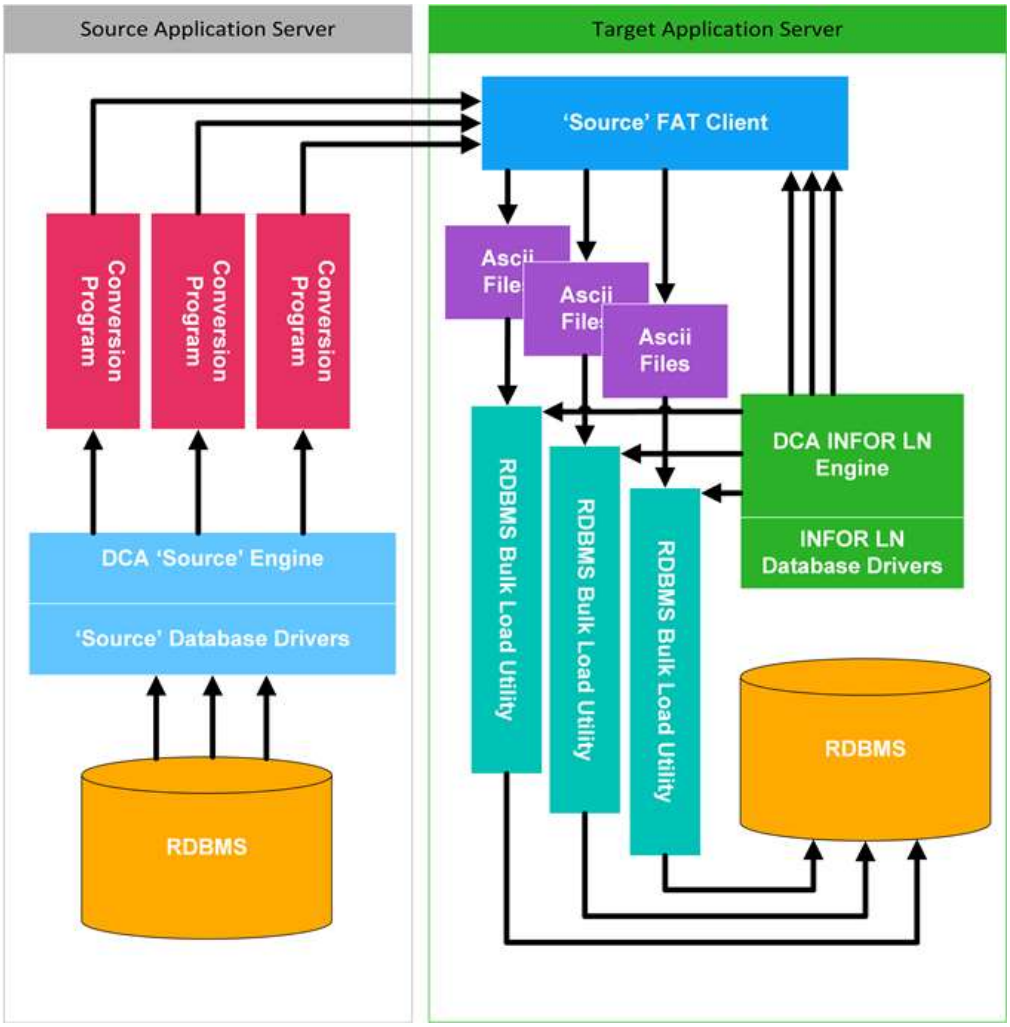

Figure 1-2 Two server with client

Without client-server connection

You can use this conversion scenario when no network connection exists between the source environment server and the target environment server. You can also use this conversion scenario when the “Two server with client-server connection” configuration is not desired or is not possible.

The data dumping process starts in the DCA source environment Engine and creates the data files. After the data dumping is over, these files must be copied or transferred to the target environment server. Start the DCA target engine to load these data files to the database.

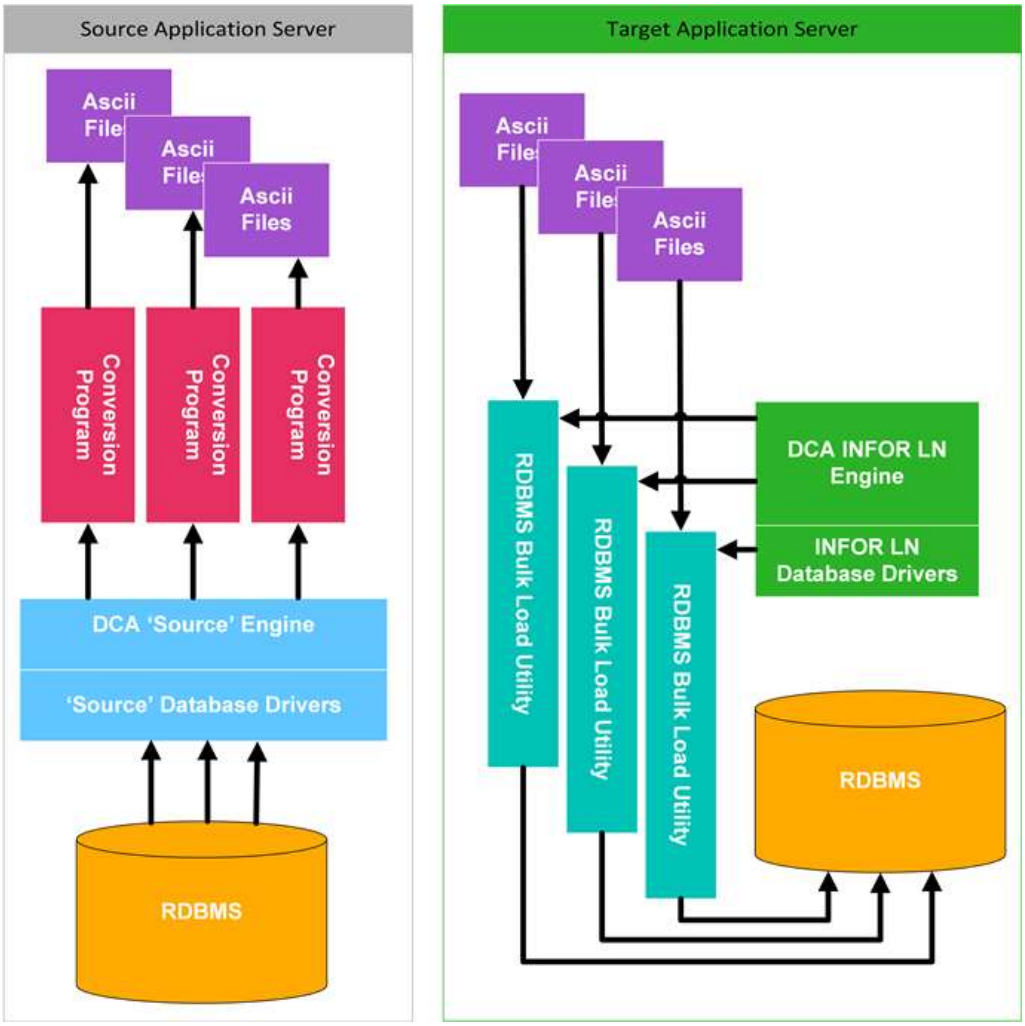

Figure 2-3 Two server without client

By default, in this method, you dump the data in the source environment and later load these files to your target environment. If it is possible to create a shared drive between the two environments a parallel synchronized dumping and loading process of the data is possible. The migration user from source environment and target environment must have read/write permissions of all the files in the shared drive else the synchronized process is not possible.

Scenario Selection

The two conversion scenarios are referred to as the single-machine scenario and the two-machine scenario, respectively and the two machine scenario has two cases: With or without a client-server connection.

Before starting the conversion, you must determine which of the two conversion scenarios best suits the specific situation at the migration customer. You must understand the impact of all the choices presented in the flowcharts, particularly the impact of the customized table definitions for a nonstandard data model (DM).

Pros and Cons of the Two Scenarios

Conversion Scenario	Advantages	Disadvantages
Single Machine	Dumping and loading can be performed in parallel. No network interference Simple Setup.	Because both source and target environments are on the same machine, the server capacity, such as CPU and disk space, can be a bottleneck.
Two Machine (With Client Server Connection)	Dumping and loading can be performed in parallel.	Increased network traffic. Setup involves additional activities, such as FAT client and remote-user connection, which can be complex.
Two Machine (Without Client Server Connection)	Performance advantage. Dumping and loading happen on two different servers. Complete server capacity can be utilized to perform either dumping or loading.	The dump (data) files are moved from the source server to the target server. By default, dumping and loading cannot be carried out in parallel unless a shared drive can be created.

If the source and the target environments are on two different servers, you can also use the single Machine scenario. If the source and the target server OS is also the same and a shared directory is in place, you can run the dump and load processes synchronized. But in this case you must manually alter the Conversion parameters. In the LN engine set the “Source Host Name” parameter to the IP address of the source server and set the “Common” variable in the BWC file of the LN BWC to the shared engine directory.

Note: The installation process of both source environment and target environment migration software remains the same in all scenarios.

Chapter 3 LN Engine Basics

This chapter describes the basic functionality of the LN Engine.

LN Engine overview

The LN Engine is a part of the DCA migration tooling suite. The LN Engine can be used to load data extraction files from the source environments dump Engines such as the DCA Legacy Engine, DCA BaanIVc4 Engine, and the DCA B50c Engine. The LN Engine is also capable of extract data from LN and creates extraction files which can then be loaded by the LN Engine.

Details on the dump and load capabilities of the LN Engine are described in the next sections of this chapter.

LN Dump Engine

The LN Dump Engine is used to extract data from a LN version which can be loaded using the LN Load Engine. You can use the tool to transfer the data to a new database but keep your current LN version or to copy your data to a new environment/server. As the LN version stays the same, just a new database or environment, the dump engine generates the required conversion objects for you.

The dump engine can also be used in combination with the DCA Studio to create the conversion objects. When using the DCA studio you can create mapping objects from one LN version to another or make conversion objects which can extract specific pieces of data. For instance, you want to start a new company with your current active data. When using the DCA Studio you also have access to the recoding functionality. The LN Dump Engine does not replace the DUE process to upgrade data from one LN version to another but in some scenarios is can be used in combination with the DUE logic. In such a case the DCA LN Engine is used to export the data out from the source LN version in a target LN version format. The exported LN data can then be loaded into the LN target version. Running the DUE on the target data completes the transition.

The behavior of the LN Dump Engine is the same as the other DCA source engines. But only the LN engine can generate its own conversion objects.

The first step is to setup the dump engine parameters. Then the required conversion objects must be created by the dump engine or the DCA Studio and loaded into the conversion objects session.

Based upon the available conversion objects the “List of target tables to be migrated” can be created. The “List of target tables to be migrated” and the “Batch list” at the LN Load engine must be synchronized. You can setup the recoding utility if needed.

If the above steps are completed the dump engine is ready to start the LN extraction process. The dump process can be started for the full range or a range specified by the user. The dump process session starts several extraction engines as defined in the parameters and creates the extraction data files. The dump engine stores the progress of the dump process run for comparison with the next run.

If the dumping errors are analyzed and resolved (or accepted) the data files can be passed onto the LN Load engine.

LN Load Engine

The load engine loads the created migration extraction files from the dump engines into the Infor LN. This section describes the main functionalities for the data load process. A detailed description of the process and the specific load engine sessions are described in the next chapters.

The first step in using the LN Load engine is to define the Engine parameters for instance about the migration scenario, LN data base information. The next step for the LN Engine is to define a batch code and specify the tables by batch codes. In the tables by batch codes the user defines which tables must be considered in the data loading process. You can specify a range of tables, or the tables applicable for the package combination, but also just a few specific tables to load. Besides the listing the tables to be used in the loading process you can also define how the tables must be loaded. There are three options of loading.

- Bulk loading
  - The extracted data file is loaded into the database using the data base own bulk load mechanism. All the records of the file are loaded into the database and then the index is constructed. So the loaded presumes the index field in the data file are correct.
- Conventional loading
  - The data in the extracted data file are loaded into the database record for record. The index is created and checked.
- DAL2 loading
  - The records of the data file are loaded via the DAL2 insert mechanism and therefore can use all the checks and features of the DAL2 concept. This loading method is also a record by record loading mechanism. The DAL2 loading feature is only applicable in when the parameter “Direct read from above path” is checked (standard setting for A2LN logic pack load).

If you use the engine to load data into a table which already has content, you must decide if the existing data can be overwritten or if the new records must be added. In the latter case the load option is limited to Conventional/Dal2.

After defining a batch determine if you require storing the conversion run details. For each conversion run you can determine an own run number so you can compare conversion run details.

The conversion process starts with synchronizing the created batch list. The tables you want to migrate as specified in the LN Engine must be synchronized with batch list at the dump Engine. You can do an automatic synchronization when both the source and target environment are connected but in all cases you can also choose the handle the synchronization manually.

Depending on the migration scenario you are now ready to either start the dump and Load process (synchronized environments) or start the Load process. During the load process the extracted data files by the DCA source engine are loaded in to the LN database.

After the loading the loading log files must be checked for loading errors. After resolving the loading errors, the data consistency checks like Data Validation and Check tables are required. After the data is fully loaded and checked the post migration steps of the required logic pack version can be handled.

Chapter 4 Migration Procedure

This chapter explains the activity list and the detailed process flow at the DCA LN Engine for a data migration. For more information on the DCA dump engines see the user guides for the dump engines, see the "**Error! Reference source not found.**" section. For more information on the pre and post migration session and activities see the data conversion guides for the logic pack, see also the "**Error! Reference source not found.**" section. Details of the LN Engine sessions can be found in the next chapters.

Migration Procedure Activities

This activity list provides an overview of the activities mandatory for the conversion from any DCA source dump Engine to Infor LN on a time line. The activities that have the same number can be carried out simultaneously. If the activity processes or maintains data in a session, the details for the LN Engine sessions of these sessions can be found in the next chapter. For the details on the source DCA Dump engine sessions we see the user guides of the dump engines, see the "**Error! Reference source not found.**" section.

This conversion process is applicable for a single company migration scenario.

The delivered migration Logic Pack also consists of a Data migration process checklist Excel file. In this checklist all the data migration procedure activities are listed in sequence from installation of the engines and the logic packs to handling pre migration sessions and post migration sessions.

Prerequisites

Prerequisites

Check 1: Customizing the standard Logic Pack (migration Consultant)

If there are customizations and or localizations present in the source environment and or LN environment, the changes must be handled in the standard Logic Pack. The migration consultant can create a customized Logic pack for the migration customer.

Activities for the migration consultant to create the customized logic pack are:

Load the source and target environment data models into new applications in the DCA Studio. Create a customized migration profile and link it to the new created customized applications. Derive

Prerequisites

this customized profile from the standard migration profile you have received as a part of the Rules Pack. Identify the conversion objects to modify, and copy these objects to your customized profile. Complete the required changes. Use the data model comparison feature in the DCA Studio to identify the changes between standard data model and customized data model. If this step is required also include step: “Check 5“. For more information on how to work with DCA Studio, see the DCA Studio User Guide. See the “**Error! Reference source not found.**” section.

Check 2: Install components

Complete these installations for BaanIVc4 and B50c Logic Pack:

1 Install source engine

2 Install Required Logic pack components (pre-conversion parameters and mapping sessions)

3 Install Infor LN Engine

4 Install Infor LN Logic pack components (post-migration sessions)

Complete these installations for A2LN Logic Pack:

5 Install DCA Engine

6 Install A2LN Logic pack components (Object Pack and A2LN Access Database)

7 Install Infor LN Engine

Complete these installations for using the LN Dump Engine:

8 Install Infor LN Engine

Check 3: Get License

You must obtain a license for your migration project, and configure the license accordingly. To know more about licensing process, refer Infor LN Engine – Installation Guide (U9130G US)

Check 4: Data Conversion Scenario

You must decide on the data-conversion scenario you want to use. For more information on data conversion scenarios, see **Error! Reference source not found.**. This step applies if the source environment is BaanIVc4 or B50c. It is not applicable for the A2LN logic Pack. The below steps are described for the source environment being BaanIVc4 but the same procedure applies for the B50c environment.

If the scenario is Two machines with Client-Server, then you must install the FAT client of Infor BaanIVc4 on the Infor LN Environment, and configure it to connect to the master BaanIVc4 Server, using the user, that is used to perform the conversion run.

If your scenario is Two machines without client-server, and you like to carry out data dumping and loading in parallel, complete the directory setups as below:

- **BaanIVc4 Engine, B50c Engine, LN Engine on Unix**: In this case, you must mount one directory in the LN Machine to BaanIVc4 machine preferably in the same name for easy maintenance (do not mount a directory in BaanIVc4 machine to LN machine). Under this

Prerequisites
directory you can create one directory, say “data” to store the data files and another directory, say “share” for both Engines to communicate. Make sure that the “data” directory has enough space to hold the temporary data files created. • **BaanIVc4 Engine, B50c Engine, LN Engine on MS Windows:** In this case you must share a directory in the LN Machine. Under this directory you can create one directory, say “data” to store the data files and another directory, say “share” for both Engines to communicate. Make sure that the “data” directory has enough space to hold the temporary data files created. • **BaanIVc4 Engine, B50c Engine, LN Engine on different operation systems:** In this case if you can mount a directory in LN to BaanIVc4 machine, the setup is like case 1 above. Otherwise you cannot carry out a dump and load in parallel.
Check 5: Logic Pack build If for the migration customer a customized logic pack created and the changes applied, a Conversion build from the DCA Studio is required. The conversion build compiles the customized Rules pack to the Object pack and stores the object pack at the source engine environment. For more information on how to work with DCA Studio, see the DCA Studio User Guide. See the “**Error! Reference source not found.**” section.

The source engine is BaanIVc4, B50c, or DCA Engine

For a detailed description of the source engine activities see the required source engine user guide, see the “**Error! Reference source not found.**” section.

Source Engine Related Activities	Load-Engine Related Activities
BaanIVc4, B50c, and DCA Engine	LN Engine
SE.1. Parameters. Maintain Parameters. For BaanIVc4 Engine and B50c Engine the parameters are maintained in the “Maintain Parameters” (bcgct0100m000) session. In the DCA Engine the parameters settings can be specified while creating a new batch.	**LN.1. Parameters.** Maintain Parameters (bcgct0100m000). Within this session you can general details on the migration such as “Source Release” and “Conversion Scenario”. For A2LN logic Pack you must check the “Directly read from above path” parameter. You can start the Parameter session from the Migration menu or via the “Dashboard” (bcgcteasym000) session.
SE.2 Source Companies. Maintain Source Companies to be migrated (bcgct9189m000) for the BaanIVc4 and the B50c Engine. Not applicable for the DCA Engine.	**LN.2. Batch Code.** Maintain Batch Codes (bcgct8188m000). You can start this session from the Dash board.

Source Engine Related Activities BaanIVc4, B50c, and DCA Engine	Load-Engine Related Activities LN Engine
SE.3. Get Object Pack components. Get DCA Generated Conversion dll Object (bcgct0106m000). This step is only required at the BaanIVc4 and the B50c Engine.	
SE.4. Batch List. Generate target tables that can be migrated from the above list (bcgct9188m000) for the BaanIVc4 Engine and the B50c Engine. In the DCA Engine this step is done by the creating or modification of a Batch.	**LN.3. Batch List.** Maintain the Tables by Batches (bcgct8189m000). **Note**: This session can be started from the Specific menu of the Maintain Batch Code (bcgct8188m000) session.
SE.5 Synchronize the batch. The created batch list must be Synchronized with the target side. This can be handled in the “List of target tables to be migrated” (bcgct9188m000) session from the specific menu. Use the import and export functionality or synchronize in automatic mode from the target side. This step is not applicable for A2LN Manual Synchronization: **SE.5.1** Import the batch details from the target. The list of target tables to be migrated are updated with the Load engine information **SE.5.2** Export the batch details from the source. Continue with **LN.4.2**	**LN.4 Synchronize the batch.** The created batch list at the LN side must be synchronized with the target side. If the user is not using LNUI at the target side and the target-source environments are connected, then the user can choose for the automatic synchronization option. In all migration scenarios the synchronization can be handled manually by export data from the load engine, import this data at the dump engine (records are updated), export from the dump engine and then import back at the load engine. For the A2LN logic pack the synchronization step is not applicable. Manual Synchronization: **LN.4.1** Export the batch details from the Data Load Dashboard” (bcgcteasym000) session via the specific menu option. Continue with **SE.5.1** **LN.4.2** Import the source batch details from the Data Load Dashboard” (bcgcteasym000) session via the specific menu option. The batch list is updated with the dump engine information.
	LN.5. Onpload Informix. Generate OnpLoad Database (bcrun0207m000). This action is required only when your LN Application’s database is Informix. You can start the session from the Specific menu of Batches (bcgct8188m000) or from the main migration menu.
SE.6. Preparation Logic Pack.	**LN.6 Preparation Logic Pack.**

Source Engine Related Activities	Load-Engine Related Activities
BaanIVc4, B50c, and DCA Engine	LN Engine
For the BaanIVc4 Engine and the B50c Engine carry out the pre-conversion process activities of the Logic Pack (pre migration parameters and mapping tables). For A2LN Logic Pack fill the access data base tables with content from the customers Legacy ERP System.	Carry out the Pre-Conversion process activities of the logic pack at the target side.
SE.7 Data Dump For DCA Engine start the dump by processing the created batch. For the BaanIVc4 Engine and the B50c Engine start the “Dump Data” (bcrun9087m00) session. **Note:** If there is a shared drive/directory between the source and the target system, the Dump process and the load process can also be started simultaneously. First start the Data dump at the source Engine and after it is running start the Data Load at the Load engine side.	**LN.7. Data Loading** In all scenarios you can first complete the dump process and then start the load process. This means first handle step **SE.7**, then move the extracted data files to the target environment, in the correct location as specified in the parameters, and then start the “Loading Daemon” (bcldr9100m000) session to load the data. In the case the source environment and target environment are connected you can handle the dump and load process simultaneously. In the case LNUI is used at the target side the user must start the dump process at the source manually (start step **SE.7**). After the dump process is started the load process can be started via the “Loading Daemon” (bcldr9100m000) session. In the case you are not using LNUI the user can start the synchronized dump and load via the “Data Load Dashboard” (bcgcteasym000) session.
SE.8: Dump Errors After the data dump process is finished check for dumping errors and store the dumping results. • Check the “Dumping Error Logs (functional)” (bcrun9599m000) session when the error logging is switched on for the migration run. Analyze the errors and resolve or accept the errors. • Check the “Data Dumping runs (History)” (bccnv5500m00) session to see the Tables dumped, the number of records dumped and the elapsed time. It is advised to print the conversion time log via the “Print History – Conversion Time Log” (bccnv5401m000) session. You can start this session via run	**LN.8: Loading Errors** After the data files are loaded you must check for existing loading errors. The Loading errors can be found using session “Log Files” (bcrun8588m000). Analyze the reported errors and fix the errors and return to step **LN.7** and **SE.7** to reload the tables for the resolved errors. Repeat this loop until all the errors are solved and or accepted. It is advised to store the error results in the “migration process checklist” applicable for the used Logic Pack.

Source Engine Related Activities BaanIVc4, B50c, and DCA Engine	Load-Engine Related Activities LN Engine
program or via the print menu from the “Data Dumping runs (History)” (bccnv5500m00) session. • View the error logs for the dumping run via the Error – Logs Archive (bccnv5503m000) session . You can also start that sessionvia the specific menu from the“Data Dumping runs (History)” session.	The detected loading errors must be communicated to the customer. For instance, via a “Migration Run Data Validation Report”.
	LN.9 Validation Errors The migrated data must be validated before released to the customer and before processing any post conversion activities. There are a few levels of data validation. • Use the “Validate Data Integrity” (ttaad4233m000) session to validate the records in a table against the data definitions for that table in the Infor LN version at the system. • Use the “Check Tables” (ttaad4232m000) session to check if the table consistency is in relation with the data definition for that table in the Infor LN version at the system. • Use the “Reorganize Tables” (ttaad4225m000) session to set the reference counters for the migrated records and to check for undefined references in the table content. Perform this last check always at test run cycles and is optional during the final cutover run. Checking for all the references takes a lot of time. Checking the references is not critical for the data migration process so can also be performed on a suitable moment after the migration. All errors should be evaluated and resolved or accepted. Accepted errors must be approved by the customer. **Note:** Leaving validation errors in the data can result in errors when carrying out LN Business processes and Data upgrade processes.

Source Engine Related Activities	Load-Engine Related Activities
BaanIVc4, B50c, and DCA Engine	LN Engine
	We recommend that you store the detected error results in the “migration process checklist” applicable for the used Logic Pack. The detected validation errors must be communicated to the customer. For example, via a “Migration Run Data Validation Report”.
	LN.10. Logic Pack Post Conversion Perform the Post Conversion process activities as defined per used Logic Pack.

The source engine is LN Engine

When the source engine is the LN Engine that can be used to extract data from an LN Version and reload it in a LN Version. The Load process steps are the same as using another source engine.

This table shows the process overview:

Source Engine Related Activities	Load-Engine Related Activities
LN Engine	LN Engine
SE.1. Parameters. Maintain dumping parameters in the “Data Dump Parameters” (bcdmp0100m00) session.	**LN.1. Parameters.** Maintain Parameters (bcgct0100m000). Within this session you can general details on the migration such as “Source Release” and “Conversion Scenario”. For the “Source Release” choose “LN”. You can start the Parameter session from the Migration menu or via the “Dashboard” (bcgcteasym000) session.
SE.2 Source Companies. Maintain Source Companies to be migrated (bcdmp9189m000) for the LN Dump Engine.	**LN.2. Batch Code.** Maintain Batch Codes (bcgct8188m000). You can start this session from the Dash board.
SE.3. Get Object Pack components. Get DCA Generated Conversion dll Object (bcdmp0106m000). For a dump from an LN version to the same LN version the object pack can be generated using the “Generate” option in the specific menu. In all cases the DCA Studio can be used to create the	

Source Engine Related Activities LN Engine	Load-Engine Related Activities LN Engine
object pack. Via specific menu “Retrieve” the object pack components can be loaded into this session.	
SE.4. Batch List. Generate target tables that can be migrated from the above list (bcdmpt9188m000).	**LN.3. Batch List.** Maintain the Tables by Batches (bcgct8189m000). **Note**: This session can be started from the Specific menu of the Maintain Batch Code (bcgct8188m000) session.
SE.5 Synchronize the batch. The created batch list must be Synchronized with the target side. This can be handled in the “List of target tables to be migrated” (bcdmp9188m000) session from the specific menu. Use the import and export functionality or synchronize in automatic mode from the target side. Manual Synchronization: **SE.5.1** Import the batch details from the target **SE.5.2** Export the batch details from the source. Continue with **LN.4.2**	**LN.4 Synchronize the batch.** The created batch list at the LN side must be synchronized with the target side. If the user is not using LNUI at the target side and the target-source environments are connected, then the user can choose for the automatic synchronization option. In all migration scenarios the synchronization can be handled manually by export data from the load engine, import this data at the dump engine (records are updated), export from the dump engine and then import back at the load engine. For the A2LN logic pack the synchronization step is not applicable. Manual Synchronization: **LN.4.1** Export the batch details from the Data Load Dashboard” (bcgcteasym000) session via the specific menu option. Continue with **SE.5.1** **LN.4.2** Import the source batch details from the Data Load Dashboard” (bcgcteasym000) session via the specific menu option.
	LN.5. Onpload Informix. Generate OnpLoad Database (bcrun0207m000). This action is required only when your LN Application’s database is Informix. You can start the session from the Specific menu of Batches (bcgct8188m000) or from the main migration menu.
SE.6. Preparation Logic Pack. There is no specific LN-LN logic pack available. So no specific pre migration sessions to handle.	**LN.6 Preparation Logic Pack.** There is no specific LN-LN logic pack available. So no specific pre migration sessions to handle.

Source Engine Related Activities	Load-Engine Related Activities
LN Engine	LN Engine
SE.7 Data Dump The data extraction is started via session “Dump Data” (bcdmp9087m00). **Note:** If there is a shared drive/directory between the source and the target system, the Dump process and the load process can also be started simultaneously. First start the Data dump at the source Engine and after it is running start the Data Load at the Load engine side.	**LN.7. Data Loading** In all scenarios you can first complete the dump process and then start the load process. This means first handle step **SE.7**, then move the extracted data files to the target environment, in the correct location as specified in the parameters, and then start the “Loading Daemon” (bcldr9100m000) session to load the data. In the case the source environment and target environment are connected you can handle the dump and load process simultaneously. In the case LNUI is used at the target side the user must start the dump process at the source manually (start step **SE.7**). After the dump process is started the load process can be started via the“Loading Daemon” (bcldr9100m000) session. In the case you are not using LNUI the user can start the synchronized dump and load via the “Data Load Dashboard” (bcgcteasym000) session.
SE.8: Dump Errors After the data dump process is finished check for dumping errors and store the dumping results. • Check the “Dumping Error Logs (functional)” (bcdmp9599m000) session when the error logging is switched on for the migration run. Analyze the errors and resolve or accept the errors. • Check the “Data Dumping runs (History)” (bcdmp5500m00) session to see the Tables dumped, the number of records dumped and the elapsed time. It is advised to print the conversion time log via session “Print History – Conversion Time Log” (bcdmp5401m000) session. You can start this session via run program or via the print menu from the “Data Dumping runs (History)” (bccnv5500m00) session. • View the error logs for the dumping run via the Error – Logs Archive (bcdmp5503m000) session. You can also start that session via	**LN.8: Loading Errors** After the data files are loaded you must check for existing loading errors. The Loading errors can be found using session “Log Files” (bcrun8588m000). Analyze the reported errors and fix the errors and return to step **LN.7** and **SE.7** to reload the tables for the resolved errors. Repeat this loop until all the errors are solved and or accepted. We recommend that you store the error results in the “migration process checklist” applicable for the used Logic Pack. The detected loading errors must be communicated to the customer. For instance, via a “Migration Run Data Validation Report”.

Source Engine Related Activities LN Engine	Load-Engine Related Activities LN Engine
the specific menu from the “Data Dumping runs (History) session.”	
	LN.9 Validation Errors The migrated data must be validated before released to the customer and before processing any post conversion activities. There are a few levels of data validation. • Use the “Validate Data Integrity” (ttaad4233m000) session to validate the records in a table against the data definitions for that table in the Infor LN version at the system. • Use the “Check Tables” (ttaad4232m000) session to check if the table consistency is in relation with the data definition for that table in the Infor LN version at the system. • Use the “Reorganize Tables” (ttaad4225m000) session to set the reference counters for the migrated records and to check for undefined references in the table content. Perform this last check always at test run cycles and is optional during the final cutover run. Checking for all the references takes a lot of time. Checking the references is not critical for the data migration process. You can also perform this on a suitable moment after the migration. All errors must be evaluated and resolved or accepted. Accepted errors must be approved by the customer. **Note:** Leaving validation errors in the data can result in errors during carrying out LN Business processes and Data upgrade processes. We recommend that you store the detected error results in the “migration process checklist” applicable for the used Logic Pack. The detected validation errors must be communicated to the customer. For example, via a “Migration Run Data Validation Report”.
	LN.10. Logic Pack Post Conversion There is no specific LN-LN logic pack available. So no specific post migration sessions to handle.

Chapter 5 DCA3.4 Infor LN Dump Engine sessions details

This chapter explains the sessions in the Infor LN Dump Engine. You can define the start-up menu as **“bcgenconvert**” on the user data (ttaad2500m000) session or start the LN Engine menu from the menu browser. Select **File → Run Program…,** and carry out the ‘Program/Session’ as “**mbcgenconvert**”. ThisLN Engine menu is shown:

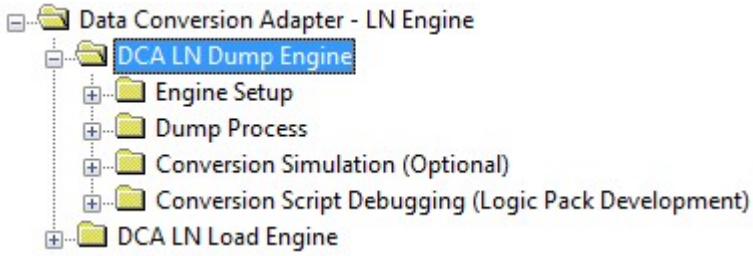

Figure **Error! No text of specified style in document.**-1 LN Dump Engine menu

Engine Setup

Data maintained in the sessions under the Engine setup are crucial to the migration dump process. The sessions under the Engine Setup relate to basic settings of the LN Dump Engine and are shown in this table :

Session	Session Code
Data Dump Parameters	bcdmp0100m000
Source Companies to be migrated	bcdmp9189m000
Conversion Objects	bcdmp0106m000
History/Multiple dumping processes per Target Table	bcdmp8571m000
List of Target Tables to be Migrated	bcdmp9188m000
Conversion Codes	bcdmp9190m000
Maintain Pre-Migration Sessions	bcdmp0103m000

Session	Session Code
General Data Handling Utilities	Submenu
Companies (*)	ttaad1100m000
Create Sequential Dump of Table (*)	ttaad4226m00
Create Table from Sequential Dump (*)	ttaad4227m000
Copy Company Data (*)	tccom0214m000
Create Tables (*)	ttaad4230m000
Delete Tables (*)	ttaad4231m000
Data Validation	Submenu
Validate Data Integrity (*)	ttaad4233m000
Check tables (*)	ttaad4232m000

(*) This is a standard Infor LN Tools session that is not described in detail in this document.

Data Dump Parameters (bcdmp0100m000)

The **Parameters (bcdmp0100m000)** session allows you to maintain the migration parameters. This session controls the overall data migration process. You can maintain only one set of parameters. The session comprises of two tabs: Engine details and options.

You must maintain the Conversion Parameters, in all the source companies, from which to migrate data.

When using LNUI the “Client source Connection Properties” fields are not required and thus not visible in the session.

This table explains the fields in the **Engine Details** tab of the session:

Fields	Description
Group	**General Details**
Source Company to be Migrated	The company for which you are maintaining conversion parameters. This is display field.
Conversion Scenario	Select the conversion scenario for the migration process. For more information on Conversion Scenarios, see **Error! Reference source not found. - Error! Reference source not found.**. The available options: **Single Server**: In this case, both the LN source environment and the LN target environment are deployed on the same machine.

Fields	Description
	Two Servers with client-server architecture: In this case, a client-server connection must be established between the source LN Engine and target LN Engine. This client server connection is achieved by installing an Infor LN FAT client in the LN target engine Server. The LN target Engine can connect to the LN source engine and carry out the data extraction process through this FAT client. **Two Servers without client-server architecture**: In this case the Infor LN source environment and Infor Ln target environment are installed on two different machines. To dump and load data simultaneously you must mount a directory present in the Infor LN source machine to Infor LN target machine.
Directory of data files	Enter the directory to store the created data files. The data files are organized, as <package><module>, under this directory, during the conversion process. You must specify this directory, based on the conversion scenario. For example: **Single Server**: Specify a directory that is present on your Infor LN Server. **Two machines without client server**: Enter the directory, present on your Infor LN source Server. For dump and load data simultaneously, you must mount a directory present in the LN target machine on to LN source machine, preferably of the same name. This example guides you what directory you must enter in this field, depending on Operating System, the name of the directories you can choose by your own. **LN Server is UNIX**: You have a directory “/migration”, in the LN target Server; mount this directory as “/migration”, in the LN source Server. Under this directory you can create two sub-directories “/migration/data” and “/migration/share”. Enter value “/migration/data” in the field. **LN target Server & LN source Server are MS Windows:** You have a folder “D:\migration” in the LN target Server, share this folder, with permission “Everyone have full permission”. Under this folder you can create two sub-folders “D:\migration\data” and “D:\migration\share”. If the machine name of LN target server is “Infor-LN”. You can enter the value “\\Infor-LN\migration\data” in the field. **Two machines with client server**: Enter the directory present on your Infor LN target server. Because, the FAT client installation of Infor LN carries out the data extraction process on the LN target Server. The data files are created on the LN target server itself. **Note**: Ensure that the conversion user has full permission for this directory and enough disk space is available to hold the data files.
Directory of Engine files	This directory is used by the conversion engines to share information and create temporary files. You must specify this directory, based on the conversion scenario. For example:

Fields	Description
	Single Server: Specify a directory that is present on your Infor LN source Server. **Two machines without client server**: Enter the directory, present on your Infor LN source Server. If you are doing a data dumping and loading in parallel, you should mount a directory present in the LN target machine to LN source machine, preferably in the same name. This example guides you what directory you must enter in this field, depending on Operating System, the name of the directories you can choose by your own. **LN Server is UNIX:** You have a directory “/migration”, in the LN target Server, mount this directory as “/migration”, in the LN source Server. Under this directory you can create two sub-directories “/migration/data” and “/migration/share”. Enter value as “/migration/share” in the field. **LN Server and source Server are MS Windows:** Say you have a folder “D:\migration” in the LN target Server, share this folder, with permission “Everyone have full permission”. Under this folder you can create two sub-folders “D:\migration\data” and “D:\migration\share”. Say the machine name of LN target server is “Infor-LN”. You can enter value “\\Infor-LN\migration\share” in the field. **Two machines with client server**: Enter the directory present on your Infor LN target server. Because, the FAT client installation of Infor LN carries out the data extraction process on the LN target Server itself. **Note**: Ensure that the conversion user has full permission for this directory and enough disk space is available to hold the data files.
Path of Time zone files	Specify the path where the time zone directories: “zone info” and “zone rules” are located. Ensure that the application server run time and application components have the correct date and time value. In case of single machine scenario, you can point to the target directory like /b61amig/bse/lib In case of a two machine scenario, copy the zone rules and zone info files from the target $BSE to source $BSE. Set the BAAN_TZ_DIR to configure the time zone directory. **Note:** Enter the path of the parent directory of the “zone rule” directory without a trailing forward slash.
Your e-mail	Enter your e-mail address if you want to receive dumping related error messages in case the migration scenario is a two machine scenario.
Group	**Client source Connection Properties (Not for LNUI)**
Installed directory path (Local)	The Infor LN Windows installation directory in the local machine, from which you start the conversion process. You must enter this directory in the MS-DOS naming convention. For example, if you have installed Infor LN Windows in " C:\Program Files (x86)\Infor\BW\ERPLN"; the

Fields	Description
	corresponding DOS naming convention can be something like: "C:\PROGRA~2\Infor\BW\ERPLN". **Note:** The DOS naming convention is machine dependent. To know the DOS name for a specific folder, start the command prompt, go to the parent folder Type the command "dir. /x" to view the DOS naming convention for its sub-directories.
Bshell name	The Bshell name of the LN source environment configuration.
Protocol	The Protocol used by the LN windows to connect to the Infor LN source environment. The available options are: Rexec, Baan Login, and SSPI Login.
Port number	The Port number used by Infor LN windows to connect to the Infor LN source environment.
User name	Specify the user name used to connect to the Infor LN source environment. Ensure that this user is linked to the correct package combination, and has permission to retrieve data from required companies and full permission to all directory structures.
Password	Specify the password for the above specified migration user.
Group	**Target Environment Details**
Database of Target environment	Select the database used by your Infor LN target application. The database options are: • Oracle • MS SQL Server • DB2 • Informix • My SQL • EDB – Postgres • Not applicable
Target's company's Time-zone	Select the time zone of your Infor LN target environment.
Cross Company	Specify this field when your LN target company is different from the LN source company. This is referred to as 'cross-company migration' in this document. **Note**: If you are copying data to a different Infor LN target company, you must map the source company to the respective target company, in the Maintain Codes for re-coding data (bcdmpt9190m000) session. Use the Code "COMP", for this purpose. Also use the Re-coding utility in the DCA Studio, to correct the Rules Pack, because some data store the company number.

This table explains the fields of **Options** tab of the session:

Fields	Description
Group	**Performance Options**
Number of bshells	Specify the number of processes to be started to perform the data extraction process. A higher value improves the overall performance of the data conversion. However, examine the hardware profiles (Number of CPUs, RAM available, etc.) before you specify a higher value and take the conversion scenario and the dump and load process into account. **Note:** Take care of not using too much capacity of the system in case the capacity is shared with normal production (virtualization) **Caution:** The system creates individual shells for each process that it starts. Ensure that you have sufficient user licenses for each process that you start. For example, if you start 6 processes, then ensure that you have 6 free user licenses available before you start the conversion run.
Display Record Count(UI)	Specify whether you want to see the number of records dumped. **Note**: If you set the field to Yes, the performance of data extraction reduces. You must also specify a higher value in the Step Size to Display Record Count field.
Step Size to Display Record Count	This field is enabled only when the Show record counts while showing progress? Field is set to 'Yes'. This value indicates the number of records the system dumps before the interface is refreshed to indicate the progress. **Note**: Specify a higher value for this field to improve the performance. For example, if you specify the value as 1000, each process after extracting 1000 records, updates the interface.
Store Dumping details	Select Yes to store these details per target table: Duration of the data extraction process per target table. Number of records extracted per target table.
Group	**Miscellaneous**
Current Batch	This is a display field that provides information about the Batch you have synchronized from your LN Load Engine. This field is populated when you synchronize your batch details from LN Load Engine.
Master Data Only?	This is a display field that indicates whether the current batch contains only the master data. This value is set in LN Engine and is populated during the synchronization.

Fields	Description
Archive time log after conversion	Set the field to Yes, if you want to archive the time log automatically after data conversion. **Note**: if you set the field to No, you can use the Archive Data dumping run (bcdmp5201m000) session to archive the time log after the data conversion.
Is Source environment Multibyte?	Set the field to Yes, when the Infor LN source environment is multi-byte.
Is Source Environment Unicode	Set the field to Yes, when the Infor LN source environment is a Unicode environment.

Source Companies to be migrated (bcdmp9189m000)

This session allows you to maintain the list of source companies that must be migrated to the Infor LN target environment. In this session, you can maintain companies that are linked to your current package combination only.

Note: You must maintain this information in the company (your **current working company**) from which you start the data dumping process. You must not maintain this information in any other source companies you want to migrate.

The data maintained in this session is not cleared either after you perform dump only or dump and load. Therefore, prior to the conversion run, ensure that data is properly maintained in this session.

Fields	Description
Source Companies	Enter the number of the company that you like to migrate. This field is enabled only when you add a new record. The corresponding company description is shown.
Commands	Description
Generate	Click “Generate” or select this option from the Special menu, to automatically add all the companies linked to your package combination.

Conversion Objects (bcdmp0106m000)

With this session you can notify the LN Dump Engine about the available conversion script objects available. The conversion object can be generated by the DCA Studio via a created LN to LN Logic

pack or generated via the “Generate obcdll objects” (bcdmp0107m000) session from the specific menu. You cannot maintain conversion object data manually.

Select the **Retrieve** option, to automatically scan for the conversion objects in the vrc structure of the package combination. All the objects in the bc DLL module within the structure are made available. Therefore, ensure that you do not create any test scripts in the DLL module. Before using this session, you must generate the conversion objects from the DCA studio or via the **Generate** option in the specific menu.

Select the **Generate** option to generate the conversion object based upon the current available RDD information. You can use this option if you want to export the current LN version to the same LN version but on another server and/or into another database. For more details, see the description of the “Generate obcdll objects” (bcdmp0107m000) session. After generating the conversion object, the user must use the **Retrieve** option to list the available generated conversion objects in the current session.

Select the **Delete All** option to purge all currently listed conversion objects from the session.

Note: You must maintain this information, in the source companies, from which to migrate data.

This table explains the fields of the session.

Fields	Description
Package Combination	The package combination for which the objects DLLs are displayed.
Sequence	This is a display field. It links a sequence number to the conversion dll objects.
Object DLL’s	The conversion scripts currently available in the vrc structure of the package combination in the system (located in the obcdll directories). A conversion dll has this naming convention: odll<t><pp><mmm><nnn> t = Type of the script f-script → Populate one target table m-script → Populate n target tables pp = package of the base target table mmm = module of the base target table nnn = table number of base target table **Example f-script: odllftcmcs045:** This is the migration mapping object for target table tcmcs045 **Example m-script: odllmtcmcs003:** This is the migration mapping object for target tables tcmcs003 and whwmd200. In this case tcmcs003 is the base table.
Commands	Description
Retrieve	Use this option to retrieve the available conversion objects within the vrc structure your package combination.

Fields	Description
	Before the list is retrieved, fist the table is cleared.
Generate	Use this option to generate the conversion objects based upon the RDD information in the system for the current package combination. This option created obcdll scripts without the use of the DCA Studio. This option starts the “Generate obcdll objects” (bcdmp0107m000) session.
Delete All	Use this option to clear the table.
Cancel	Closes the session.

Generate obcdll objects (bcdmp0107m000)

Use this session if the user wants to:

- Create a copy of the current LN environment onto another server
- Create a copy of the current LN environment to another database.

In previously versions of the bc package this functionality was located in the bc tls module.

The program generates conversion objects based upon the RDD information for the current package combination without using the DCA Studio. Therefore, it can only be used when the source and target LN version remains the same.

The RDD information of the current package combination is retrieved from the $BSE/dict directory. Therefore, during the generation of the conversion objects (obcdll scripts); also customizations to tables are considered. Examples of customization to tables outside direct table definition changes are:

- Customer defined fields (“Customer Defined Fields” (ttadv4591m000))
- Timestamp definitions (“Table Timestamp Definitions” (ttadv4136m000))
- Multi Language fields (“Registered Tables with Multi Language fields” (ttadv4137m000))

The program creates a standard 4GL program script, compiles the script and stored the compiled version in the bc obcdll directory.

Via the ranges by package, module, and table the user can define for which range the scripts must be generated. During the process a log file is kept to track the progress and view the carried out steps. The user can specify the location for this log file. The same directory is also used to temporary store the generated scripts. After the compilation step is completed the scripts are removed from the directory. In case a log file is already present then a backup of these log files are created.

This table explains the fields of the session:

Fields	Description
Package	Use the “From” and “To” fields to define a range based upon the packages.
Module	Use the “From” and “To” fields to define a range based upon the Modules.
Table	Use the “From” and “To” fields to define a range based upon the tables.
Log File Path	Specify the location of the log file created by the generate obcdll object program. Within the log file the user can view which steps are handled. The standard location of the log file is defaulted like: $BSE/tmp/migration/bcdmp_generate
Include 2GB check	This option can be used when the operating system of source LN environment is Unix. By setting this option the generated obcdll scripts creates data dump files without exceeding the 2GB limit per data file. If more records must be dumped for the table an additional file is opened.
Commands	Description
Generate	This option starts the generation process. **Note**: this option does not close the session.
Close	This closes the session without generating the scripts and returns to the calling program session.

History/Multiple dumping processes per Target Table (bcdmp8571m000)

This session and the related sub-sessions are used to split mapping objects in these cases:

- Migrate history/static data before live migration. It reduces the system downtime during the live migration. First migrate the static data of a migration object up to a specific date and during the live migration only run the migration object from that specific date up to current date. Split the conversion object in 2 ranges.
- Migrate a source table which contains a large amount of data. To lower the total execution time of the mapping object it can be split into multiple objects where each object migrates a specified range of the main source table.

For more detailed information on how to set up this functionality, see **Error! Reference source not found.**, “Split mapping objects using History/multiple dumping process per table”.

You cannot add any information in this session. The **history/multi-process per table** setup is initially performed in the DCA Studio. During the conversion build you must enable the **Consider**

History and Multi process setup flag in the **Settings** option of your migration profile. When you perform a conversion build, a **splitinfo** file is created in your source Engine environment, where the dll objects are located (By default *$BSE/application/<Your VRC>/obcdll*). Therefore, after starting this session, you must import information from this file by clicking ***Specific → Import from DCA generated file***.

This table shows the fields and commands of the History/Multiple dumping processes per Target Table (bcdmp8571m000) session:

Fields	Description
Target Table	The target table to which you migrate the history data. This is a display field.
Number of Bshells	This is a display field, and indicates the number of processes used to migrate data to the target table. To change the number of process, use the option Specific → Change No. of Processes For migrating the static history data use 1 process and for splitting a mapping object use a value greater than 1. **Note:** By default, during import this field gets the number of processes you have maintained in Conversion Parameters (bcdmp0100m000) session
Source Table	This is a display field. And provides the information about the source table from which data is being migrated. You can compare this with the migrate <source table> statement in your conversion objects maintained in DCA Studio.
Split data on fields	This is a display field. Shows the fields on which you are going to split or group the source data. **Note:** While splitting data, you can use the primary index of the source table, or some date fields. This setup is initially performed at the DCA Studio and you cannot modify it.
Criteria?	The tab(s) are enabled only if you are migrating history data and want to use multiple processes. **Note**: The scenario is not advisable, if your split criterion is not based on a unique index. You must specify the range to consider. These values are used for generating the boundary values automatically for the processes.
Commands	**Description**
Import from DCA generated file	This is the first action you must perform, when you start this session. The multi-process or history data migration setup you carry out in the studio during a conversion build generates a splitinfo file. This option reads the splitinfo file and populates the data in this session.
Change Number of processes	Use this option, if you want to use multiple processes during data dumping for a target table. When you increase the number of processes, ensure that you have adequate hardware resources to support the load.

Fields	Description
Open Selection Criteria.	This option starts the “Selection Range per Source Fields” (bcdmp8570m000) session
Generate Range values per processes	This option calculates the range values for each of the processes, if you are using multiple processes to dump data. This action counts the number for records in the selection criteria specified in the “Selection Range per Source Fields” (bcdmp8570m000) and calculates the boundary values for each process. **Caution**: This action takes long time if the source tables contain many records. Instead of generating you can also maintain the rage values in Range values per processes session.
Open Range values per Processes	This option starts the “Range Values per Source Table per Process” (bcdmp8572m000) session. Within this session the user can maintain the ranges per source table.

Selection Range per Source Fields (bcdmp8570m000)

Use this session to list the selection range for each source field of the target table.

You must enter ranges per selected source field in this session, if you are migrating history data. The values are mainly used during the generation of the boundary values for processes.

This table eshows the fields and actions in the session:

Fields	Description
Target Table	This is a display field that shows the target table to which you are migrating the history data or using multiple processes to dump data.
Source Table	This is a display field that shows the information about the source table from which data is being migrated. You can compare this with the migrate <source table> statement in conversion objects of your DCA Studio.
Field Order	This is a display field that indicates the number of source table fields, based on which you want to split the data.
Field Name	This is a display field. Shows the source table fields, based on which to split the data or decide the up to values for history data migration.
Value from	Enter the value for the above source field from the point you want to migrate data from your source table.
Value to	Enter a value that indicates the point to which you like to migrate the source table data.
Action	Description

Open Range values per processes	This option starts the “Range Values per Source Table per Process” (bcdmp8572m000) session, which is handled by each process. You must maintain data in this session, in case you are using multiple processes to dump the source data.

Range Values per Source Table per Process (bcdmp8572m000)

In this session you can define the “from” and “to” ranges for each data dumping process. It is mandatory to maintain the range of values for each process. To automatically generate the values, select “Generate Selection Range Values” from the Specific menu. The automatic generation takes time to scan the entire source table data and split across the number of process. It is also possible to manually maintain data in this session.

Currently you cannot manually maintain the selection criteria for each process (you must use GTM in case you want to manually maintain the data). If the data is maintained manually and you use multiple processes to dump the data, the duplication of values can occur in the created data files. Different processes can dump the same source records in their respective data files. These duplicate values can stop loading of the data to Infor LN target environment. To resolve this problem, you can generate the values during the test migration run, and use the values for manual entry when you actually carry out the migration process.

This screen displays the range values specified to extract data from source table *tfacp200 (Open Items),* when you use **two parallel processes**. During run-time, the range values are taken as follows:

For example, if you must split the tfacp200 mapping object. Assume you set the number of processes to 2 and you used the full primary index.

In the range values per process you specify 3 values:

{‘’,’’, 0, 0, ’’, 0, 0}

{‘ 1001’, ‘ACP’, 19960358, 0, ‘’, 0, 0}\

{‘YASH02’, ‘ACP’, 19961449, 0, ‘’, 0, 0 }

This leads to these ranges for the 2 processes:

1st Process→ Take values between {‘’,’’, 0, 0, ’’, 0, 0} and {‘ 1001’, ‘ACP’, 19960358, 0, ‘’, 0, 0}

2nd Process → Take values between > {‘ 1001’, ‘ACP’, 19960358, 0, ‘’, 0, 0} and {‘YASH02’, ‘ACP’, 19961449, 0, ‘’, 0, 0 }

This table shows the fields and commands in the session:

Fields	Description
Target Table	This is a display field, and indicates the target table to which you migrate the history data or use it to split the object over a number of processes to dump the data.

Source Table	The source table.
Split on Fields	The fields, on which the source table must be split. This is defined in the DCA studio and imported via the “splitinfo” file
Process	This display indicates the processes number for which the starting range value can be set.
Range Values per Processes	The range values for each of the processes. For example, if you use two processes to dump the source table, you can see the range values as follows: 1 {} |* The from value for 1st process 2 {up to value for 1st process} 3 {up to value for the 2nd process} **Note**: You can generate these values using the option Special → Generate range values per processes.
Commands	Description
Generate Selection range values	Use this option to automatically generate the range values. The process takes time, because it must scan the entire source table data split across the number of process.

List of Target Tables to be Migrated (bcdmp9188m000)

In this session you can maintain the “list of tables to be migrated” which is used by the dump engine. The first step is to generate the list based upon the mapping objects defined in the vrc structure of the package combination. See also session “Conversion Objects” (bcdmp0106m000). The session uses the “mscripts” file which is delivered as part of the logic pack or created during a conversion build from the DCA studio. This file is used to also show the related tables which are part of a m-script type mapping object.

The second step is to synchronize the “list of tables” with the “tables by batches” list from the LN Engine. You can synchronize automatically or manual.

All the object with the field “Skip” set to “No” are used in the dumping process. The field can only be set at the LN engine in the “tables by batches” which is synchronized with the current session. During the synchronization tables can also be imported into the list for which no conversion object is present. In that case the Skip field is automatically set to skip. After a manual import the data must be exported and imported at the tables by batches in the LN Engine.

Within this session you can also specify a priority for a mapping object. By setting the priority you define the sequence how the mapping objects are assigned to the dump engine processes. A priority can also be set based on elapsed dumping times from a previous dumping run or based on a record count of a simulation run. Both options can be triggered from the specific menu.

Note 1: You must maintain this information in all the source companies from which you want to migrate data.

Note 2: If you plan to use history data migration or number of parallel dumping processes for each target table (when source data size is large), you must perform the respective setup (History/Multiple dumping Process per Target Table (bcdmp8571m000)), before you generate information in this table.

This table shows the fields and actions in the session:

Fields	Description
Company No.	The current company number for which you are generating tables to be migrated. This is a display field. **Note**: The data dumping process copies the data from other companies when you migrate multiple source companies to your current company from which you start the dumping process.
SrciptName	The target table name. The standard format used to represent target table is: ppmmmnnn Wherein: pp represents Package mmm represents Module nnn represents Table number Example: tccom000, tttxt010
Skip?	This is a display field indicating if a target table must be migrated or skipped during dumping of the data. The field is set during the synchronization process with the LN target Engine. At the LN target Engine, you can specify which tables must be skipped or not. If the table is selected (not skipped) at the LN target engine but at the LN source engine there is no Mapping object, then the flag “Skip” is set to “Yes” by the LN source engine. **Note**: For m-Objects the “Skip” flag is also visible for all related tables in the related table session (bcdmp9585s000). For more information, see the Related tables below.
Object Present?	Indicates whether a corresponding conversion script is present in the system for the current package combination. During the “batch list” synchronization process this “skip” flag is set to “Yes” when there is no conversion object in the system.
Priority	You can specify the priority to indicate the order of the dumping. If you do not indicate the priority, the dumping process first dumps the tables having Duration field value longer and then the tables with Duration field value Shorter. When you do not specify the priority, the tables with longer durations get the precedence over the shorter durations.

Fields	Description
Duration	This field gives the information about the data size of a target table. Following are the possible values: • Shorter: The data file is small in size. • Longer: The data file is big in size. You can set this value in LN Load Engine, and it is made available during synchronization process between the source and the target Engine. The dumping process first selects the tables with longer duration.
Related Tables	Indicates whether a target table is an M-Object (See Terms & Abbreviations). The target table is an M-Object, when the value of this field is greater than one, and the current target table is the base table. To view the related tables, on the Specific menu, click Open Related Tables. This option is only enabled when the number of related tables is greater than one.
Logical Table?	Indicates whether the target table is a logical table in the current LN company for which you are performing the migration. For logical tables, the Skip field is set to Yes automatically. This information is synchronized from LN Load Engine.
Instance Number	In case you use multiple instances (processes) to create data dumps for a target table, that target table occurs as many times as the number of instances. For example, you have planned to use 4 instances/processes to create data files for the tfgld100 table, then you find 4 records for table tfgld100, each one with instance numbers 1, 2, 3, 4.
Actions	Description
Generate	Use this option to populate the information from the “Conversion Objects (bcdmp0106m000) session. Additionally, this option inserts the list of object DLLs in the bcdmp988 table. It scans the directory obcdll in all the Package VRCs. The scanning starts from the current VRC of the package combination that pertains to the login and progresses to all the VRCs that are derived from the current Package VRC. For M-Scripts, a file with the name mscripts must be present in the obcdll directory. This file is delivered with the installed logic pack or can be created by performing a conversion build from the DCA Studio. Based on the M-script file the related child tables per base conversion object table are stored in the “Child Tables for repeated I/O” (bcdmp985) table. **Note**: This is the first step you must carry out in this session.

Fields	Description
Open Related Tables	This option starts the “Related tables of a M-Script” (bcdmp9585m000) session, which displays the child (related) tables for the M-script mapping object.
Import Target Batch Details	This option can be used to synchronize the batch list from the target engine side. The option is available for all scenarios. But in case of “Single machine” or “Two Machine with client server connection” the batch list can be synchronized also automatically from the LN Load Engine. Before importing the target batch details it must be exported from the LN Load Engine. The export from the LN Load Engine creates a <compnr>.logical.988.info file in the defined “Directory for Engine Files”. In case this directory is not shared, between the source and the target engine, this file must be copied to the source environment’s “Directory for Engine Files”. The file contains information about target tables to be migrated, the logical/physical mapping, bypass(skip), duration, mode of loading information which is imported into the source environment.
Export Source Batch Details	This option can be used to synchronize the batch list from the source engine side to the target engine side. The option is available for all scenarios. But in case of “Single machine” or “Two Machine with client server connection” the batch list can be synchronized also automatically from the LN Load Engine. You can export the source information after you performed the generation of the list and imported the batch details from the target engine. This action creates an export file in the “Directory for Engine Files” specified in the parameter (bcdmp000) table. The name of the file is “<company>.objects.info file”. If the “Directory for Engine Files” is not shared, between the source engine and the target engine, you must move the file manually tot the “Directory for Engine Files” specified for the target engine side. Within the LN Load Engine you can import the file from the ‘Data Load Dashboard” (bcgcteasym000) via the specific menu option.
Update priority from last conversion run	Use this option to update priority from the last conversion run. The target table with the highest elapsed dump time, gets priority one. The next table with the highest elapsed dump time, gets the previous priority plus one. This is handed for all conversion tables in range. The priority one is the highest and priority 999 the lowest. The option starts the “Update Priority from last conversion run” (bcdmp9288m000) session.
Generate priority from simulation run	If you performed a data conversion simulation (record counting), then the priority of data dumping can be calculated from this information.

Fields	Description
	The target tables that have more records to be migrated get the highest priority.

Related Tables of a M-Object (bcdmp9585m000)

Use this session to view the related tables linked to the parent (base) migration object. Migration M-objects are related to a base target migration table but dump also data for other target tables. When the objects are retrieved via the generate option in the “List of Target Tables to be Migrated” (bcdmp9188m000) session, the content of the mscript file is analyzed to get the related table information.

This table shows the fields of the session.

Fields	Description
Parent Table	This field displays the parent (base) conversion object table.
Company No.	This field displays the current company number.
Child Table	This field display which child table is related to the parent (base) conversion object table. During carrying out the Parent (base) conversion object also data is dumped out for the related child tables.
Skip?	This is a display field indicating if a child table must be migrated or skipped during dumping of the data. For more details, see the “Skip?” field in the “List of Target Tables to be Migrated” (bcdmp9188m000) session.

Update Priority from last conversion run (bcdmp9288m000)

Use this session to update the priority field in the “List of Target Tables to be Migrated” (bcdmp9188m000) session. The priorities are set based upon the elapsed dumping times from a previous conversion run.

The table with the longest elapsed dumping time gets the highest priority (0) the next one gets the previous priority plus one and so on.

You can only use this option when dumping details are stored per conversion run.

This table shows the fields of the session.

Fields	Description
Company	This field displays the current company number.

Fields	Description
Run Code	Use this field to specify which previous run number shall be used to set the priorities.
Commands	Description
Continue	This option sets the priorities.

Conversion Codes (bcdmp9190m000)

Use this session to maintain Codes to recode your master data during migration. For more detailed steps on 'how to recode your data', see **Error! Reference source not found.**: "Recoding".

Note 1: There are two conversion codes which are standard used and cannot be used for any other recoding purpose.

- "*COMP"* code is reserved for cross company migration.
- *"UNIC"* code is reserved for the Unicode correction feature.
- *"ITEM"* code is reserved for the Item recoding feature.

Note 2: To migrate several source companies, you must maintain the recoding information in all the companies, even if the master data is stored in one company. Alternatively, you can share the bcdmp990 and bcdmp991 tables in your current working company

This table shows the fields and commands in the session:

Fields	Description
Code	You can specify a code, to identify your recoding scenario. The codes "COMP", "UNIC" and "ITEM" are reserved of which conversion code "COMP' is created by default.
Description	Specify the description that explains your recoding scenario.
Values Present	Indicates whether the conversion code has at least one applicable value present in the "Values by Conversion Codes" (bcdmp9191m000) session.
Action	Description
Open values by Conversion Code	This option starts the "Values by Conversion Codes" (bcdmp9191m000) session. In this session you maintain the mapping between your existing value in the LN source Environment and new value in the LN target environment.

Values by Conversion Codes (bcdmp9191m000)

You can start this session from the Special menu of “Maintain Conversion Codes” (bcdmp9190m000) session. In this session you can maintain the mapping between the old codes in your Infor LN source environment and the new conversion codes in your new Infor LN target environment.

The old value and new values can be created manually, retrieved via the source master data table or via a dll function, in case you have a specific algorithm to generate the new value from the old value.

To remove all the values by conversion codes you can delete the conversion code.

This table shows the fields and commands in the session:

Fields	Description
Code	This is the conversion code as specified in the conversion codes session for which to maintain the old and new values.
Old value	Enter the existing codes as present in your source environment. For example, if you are recoding your warehouses, you must enter the existing warehouse codes that you like to recode.
New value	You must mention the corresponding new values for the old codes. It is required to specify the new values, for the all the old codes mentioned in the Old Value field; otherwise, you encounter data corruption problems after the data conversion.
Apply	Indicates whether the Old Value must to be replaced with the New Value. If the New Value is not empty, this is read-only and always Yes. If the New Value is empty, this field is editable and can be changed to Yes, which means that the Old Value is replaced with an empty value or changed to No.
Commands	**Description**
Fill Old Values from a Table Field	Use this option to fill the codes automatically if you know the master data table where the codes are located. Entries are created with an empty New Value and Apply set to “No”. Replace the empty New Value with a desired New Value. This automatically changes Apply to “Yes”. To replace the Old Value with an empty value, leave the New Value empty and change Apply to “Yes”. This option starts the “Old Values from a Table Field” (bcdmp9291m100) session.
Fill New Values Using a Dll function	If you have some specific algorithm to populate the new values, you can code that function in a DLL. Use this option to call the dll to populate the new values. For new values that are not empty, Apply is defaulted to “Yes”. This option starts the “New Values using a DLL Functions” (bcdmp9291m200) session.

Old Values from a table field (bcdmp9291m100)

You can start this session from the Special menu from the “Values by Conversion Codes” (bcdmp9191m000) session. Use this session to match the old codes in the existing source environment, to the corresponding conversion code. The values are retrieved from your current company and package combination.

This table shows the fields and commands in the session:

Fields	Description
Table Field Name	Enter the table field, from which you want to fill the old values. The coding standard for the table field is ‘ppmmmnnn.ffff’ pp → Package mmm → Module nnn → Table number ffff → field name Example: tcmcs003.cwar
Filter Criteria	Specify the filter criteria for the value before it is added to this table. For example, to recode only the warehouse codes starting with “AB”, so you can specify the criteria as tcmcs003.cwar like “AB.*”.
Commands	Description
Continue	Click this button to import the old values from the table field. Existing values are retained.

New Values using a DLL functions (bcdmp9291m200)

This session automatically generates the new values for the corresponding old master data codes. It is useful when you have some logic to generate the new value. In that case, you must create a dll function, in which you can program your logic. By default, LN Dump Engine provides the bccnvdll9000 dll for this purpose. You can create your own dll, if you do not have source codes. The function in this dll can have any name, but the signature of the function should be as follows:

function extern domain bcstr30 bccnvdll9000.fill.cwar (

domainbcstr30old-value)

The naming convention:

<your dll name>.fill.<your conversion code>

A function example:

function extern domain bcstr30 bccnvdll9000.fill.cwar(

```
        domain bcstr30 old-value)
{ |* Write your logic here. Here in all the old values,
|*in front the letter “A” is added
return ("A"& old-value)
}
```

Caution: Ensure that you compile this dll, without any errors.

This table shows the fields and commands in the session:

Fields	Description
Dll Name	Enter the dll name in which you have written the function to construct the new codes. This naming convention is used for the dll: ppmmm<dll name> pp → Package mmm → Module <dll name> → any user defined name Example: bccnvdll9000
Function Name	The function in the above dll where the actual logic is coded. When you specify the function, you need not specify any arguments. At the runtime the old values get passed to the function, one by one. **Note**: Ensure that you compile your above dll and created the object.
Commands	Description
Continue	Click this button, to call the function that constructs and populates the new value for each old value present in the Values by Conversion Codes.

Maintain Pre-Migration Sessions (bcdmp0103m000)

Use this session to hold pre-conversion sessions as developed by Infor via the standard Migration Tooling and Standard migration Logic Pack. You can manually insert bc sessions or import the sessions via a created bc session component file according to a specified format.

After the list is created the user can exclude sessions, reorder the sessions and carry out the sessions in the specified order.

To delete all the records in the list, use the standard “Delete Range…” option from the Edit menu.

Note: After you start a session from this list, the system sets the “Started?” field to “Yes”. You can also use this session to hold your custom pre-conversion sessions if any.

This table explains the fields and commands in this session:

Fields	Description
Session Name	Enter the code for the pre-conversion (BC session or application) session you must run as a pre-conversion activity. These sessions require a manual trigger and are not started automatically. This session is used to organize the pre-conversion sessions or you can consider it as a container holding all your pre-conversion sessions.
Description	The description of the pre-conversion session.
Sequence	Displays the order in which the sessions must be started.
Select Yes/No	Set the field to YES, in case you like to skip the execution of this session.
Started Yes/No	This field indicates whether the session is started or not. When you start a session from this list, the field is automatically set to YES. It does not guarantee that the session is handled correctly, because the user can cancel the session.
Commands	Description
Start Session	This option starts the selected session.
Start Sequence	Use this option to start the sessions in the specified sequence. This option starts the “Run Pre-Migration Sessions” (bcdmp0105s000) session.
Import from File	With this option you can import a list of session from an ascii file. The file can be created via the pre-migration session list within the Migration process checklist or can be created from the standard tools session “Sessions” (ttaad2500m00). **Note**: The file structure is pre-defined. The file structure is shown in the “Import Pre-Conversion Session from an ASCII File” (bcdmp0203s000) session and must be” <Session Code><Separator><Session Description>”.
Reorder	Use this option to arrange the sessions in multiples of five according to the current specified sequence.

Run Pre-Migration Sessions (bcdmp0105s000)

Use this session to start the listed pre-migration session in the specified sequence.

When the session is started the first not handled session from the pre-migration list is shown. The user can start the listed session or go to the next, last, previous, or first not started (handled) session.

After a session is started the status of the session is changed in the Maintain Pre-Migration Sessions list.

Note: Click**Close** to exit the session.

This table explains the fields and commands in this session:

Fields	Description
Sequence	This field displays the specified sequence of a not started session from the Maintain Pre-Migration Sessions list.
Session	This field displays a not started session from the Maintain Pre-Migration Sessions list.
Description	The description of the listed session.
Commands	Description
Start Session	This option starts the listed session.
Next Session	This option goes to the next not started session from the list.
Last Session	This option goes to the last not started session from the list.
Previous Session	This option goes to the previous not started session from the list.
First Session	This option goes to the First not started session from the list.
Close	This option closes the session.

Import Pre-Conversion Session from a ASCII File (bcdmp0203s000)

Use this session to import a session list from an ASCII file. The session list must be in this format: <Session Code> <Separator> <Session Description>.

The user must create the list. As input the user can print the sessions from the standard tools session: “Sessions” (ttaad2500m000) or via using the pre conversion list from the Migration process checklist.

This table explains the fields and commands in this session:

Fields	Description
File Path	Use this field to specify the file name and the file location of the import file.
Separator	Use this field to specify the used separator in the import file.
Format	This field displays in which format the input file must be.

Fields	Description
Example	This field displays an example of the format which must be used in the input file.
Commands	Description
Import	This option starts the import of the list.

Dump Process

Under this menu item you find sessions that help you to start the data dumping /extraction process. Check the sessions that you must carry out before conversion, and some utilities can be used during data dumping process.

Session	Session Code
Procedure	Submenu
Dump Data (Create Data Files)	bcdmp9087m000
Dumping Daemon	bcdmp9100m000
Archive Data Dumping Run	bcdmp5201m000
Check-list	Submenu
Archive previous Data Dumping Run	bcdmp5201m000
Check currently logged-in users (Unix only)	bcdmp5406m000
Check Environment of present Company	bcdmp6400m000
Check Scheduled Jobs	bcdmp5404m000
Validate Data Integrity (*)	ttaad4233m000
Reports	Submenu
Print Dumping (data file creation) Status	bcdmp9442m000
Dumping Error Logs (Functional)	bcdmp9599m000
Data Dumping Runs (History)	bcdmp5500m000
Utilities	Submenu
Reset Conversion Process	bcdmp0200s000
Generate List of Log Files to Analyze	bcdmp5301m000
Analyze Log Files	bcdmp5300m000

Dump Data (Create Data Files) (bcdmp9087m000)

Run this session to start the data dumping process creating the data files, which can be loaded to your Infor LN target Application. Before running this session all the required pre migration activities as described in the data conversion process checklist must be performed.

Before starting the actual dump process the “Range of Tables for Dumping” (bcdmp4220m000) session is started. In this session you can specify which range of tables must be dumped. Based upon the entered Range the dump process looks into the “List of Target tables to be converted” (bcdmp9188m000) session. All tables falling within the range and have the “Skip” flag to “No” are taken into the dump process.

After you specify the range per package, module and table click **OK** to proceed with the dumping process. Click **Cancel** to exit the dumping process.

After proceeding with the dumping process a control question is asked if you are really sure to continue with the process. Click **No** to cancel the process or **Yes** to proceed.

If the dumping process was already started, the user gets a question whether a “fresh Conversion Run” is required or if the user wants to continue with the “Previous Run”. The latter option can be used when a dumping process was aborted but the user does not want to restart the complete conversion run.

If the dumping process was already started also the required dumping directory for the exported data files is already present. A question is asked if you want to keep the current directory and the contents or delete it and create it again. If you keep the directory it overrides only the existing data files if the files (tables) are part of the current dumping process. The location of the data files is determined by the “path of the data files” specified in the “Data Dump Parameters” (bcdmp0100m000) session. The path is extended with the current company number. For example, when your data files path is *$BSE/tmp/migration*, and the source company is *123*, then it creates a directory, *$BSE/tmp/migration/123*.

After you made a choice the dumping process continues and the main dumping process screen is shown.

The main reason of the session is to monitor the dumping process. You can see how many dump engines instances are running and which Mapping object is running in the engine. After an engine is ready processing one mapping object it gets a new mapping object assigned until all mapping objects in the defined dump range are processed. The session starts with the number of engine dump instances as specified in the “Data Dump Parameters” (bcdmp0100m000) but depending on the performance and capacity of the system the user can extend or lower the amount of dumping instances.

The displayed progress indicator does not represent the progress in terms of volume of data dumped but, indicates the progress in terms of the number of tables dumped.

If all the specified conversion objects for the run are carried out a “Dumping Complete” message is displayed.

This main dumping process screen displays this information:

Information	Description
Tables to dump	This field displays the total number of tables in the selection range you entered. Taken into account the ones within the range which are set the be skipped.
Tables dumped	This field displays the number of tables for which the data files are created.
Dumping Tables (Column)	This column displays the Mapping object running in the dump engine instance. For the standard objects this is also the dump for the table. In case of an M-object the table specified is the main table. You do not see the related tables for the m-script mapping object. The number of rows corresponds to number of dumping instances/processes. (up to 19 engines)
Company No. (Column)	This column displays the number of the company to which the data is being dumped. The number of rows corresponds to number of dumping instances/processes. (up to 19 engines)
Records dumped (column)	This column displays the records dumped so far into each target table in the current dumping instance and taking into account the refresh frequency for number of records as specified in the “Parameters” (bcdmp0100m000). The number of rows corresponds to number of dumping instances/processes. (up to 19 engines).

Following are the possible actions you can perform from this screen. You must use the respective key combinations to carry out these actions.

Actions	Descriptions
Cancel dumping [Ctrl + A]	This action stops the dumping. When you use this key combination, the system prompts you for the confirmation. Click Yes to stop the process. The system waits for the current dumping processes to complete.
Invoke bshell [Ctrl + P]	Use this key combination to start additional dumping processes.
Revoke bshell [Ctrl + N]	Use this key combination to reduce the number of processes in case of system overload.

Attention points

Reset the conversion process

When you start the dump process session, a message may appear “Dumping started already by <user> in company <company number>”. The message is displayed when somebody else has started the conversion run or the conversion run you have started earlier got killed. Ensure that no user is performing a conversion and the last conversion is terminated. You can reset the lock to start the conversion run again by using the session, *Dump Process →Dump Utilities → Reset Conversion Process (bcdmp0200s000)*

After you reset the conversion process you must restart the dumping process and a message is displayed if you want to start a new conversion or continue processing of the not dumped tables (completed mapping objects) of the previous run. When you choose the “Fresh Run” option you can specify the new range of tables to dump.

After resetting the conversion, you must archive the dumping history manually in case you want to keep the dumping history.

Number of dump engines

The number of dumping processes that you can invoke is the number of user licenses you have minus 1. The maximum number of dumping engines is 19. Always evaluate the system resources to set the amount of dumping engines. Too much dumping engines can cause to a lower performance and increases the total execution time.

Data Files

During the dumping process, data files (.dat) are generated for each of the Infor LN tables within the range you have specified. If the target data base is MSSQL and the dump engine is ready with processing a mapping object all the dumped data files are transformed to *.da2 files.

Data files in ASCII format are created in the path specified for Directory of data files field in the “Data Dump Parameters” (bcdmp0100m000) session. For example:

```
 <Directory of data files>/<company number>
<package+module>/<tablename.dat>...
```

```
The data file for the tcccp000 table is available in the following location:
```

```
< Directory of data files>/<company number for which conversion is being
run>/tcccp/tcccp000.dat
```

Dumping Daemon (bcdmp9100m000) for LNUI

This session replaces the session: “Dump Data (Create Data Files)” (bcdmp9087m000) when using LNUI.

Run this session to start the data dumping process which creates the data files. The created data files can be loaded to your Infor LN target Application. Before running this session, run all the required pre migration activities as described in the data conversion process checklist.

The main reason of the session is to monitor the dumping process. You can see how many dump engines instances/processes are running and which Mapping Object is running in the engine. Each mapping object has a main table to process and only the main mapping object table is shown in the session. If an engine is ready processing one mapping object it gets a new mapping object assigned until all mapping objects in the defined dump range are processed. The user can start, stop or resume the migration process using the buttons on the screen. The dumping process starts with the number of dump engine instances as specified in the “Data Dump Parameters” (bcdmp0100m000). Depending on the performance and capacity of the system the user can add or remove dumping instances. In the progress of the dumping process is shown by the number of mapping objects still to process, the number of mapping objects processes. Based upon these numbers a percentage complete is calculated.

Using the selection range in the session the user can specify which data must be migrated. Based upon the entered selection range the dump process looks into the “List of Target tables to be converted” (bcdmp988) table. Only tables falling within the range and have the “Skip” flag set to “No” is taken into account for the dump process.

The session shows the location of the dump engine files and also the location of the data files as specified in the migration dump parameter table (bcdmp000). The user must specify if the files in the dump directory needs cleared or not. With the option “Clear Directory Content” selected all the files and subdirectories are removed from the data dump directory. A new sub directory structure is created and data files are created. With the option to “Clear Directory Content” not selected the current sub directory structure is maintained. Only files created by the dump process are overwritten. The directory path is extended with the current company number. For example, when your data files path is *$BSE/tmp/migration*, and the source company is *123*, then the created directory is *$BSE/tmp/migration/123*.

This dumping process screen displays this information:

Fields	Description
Group	**Selection Range**
Package	Use this field to set a range of packages for the dump process.
Module	Use this field to set a range of modules for the dump process.
Table	Use this field to set a range of tables for the dump process.
Group	**Progress**

Tables incompletely dumped	Displays the number of tables for which the dump partially succeeded.
Tables to Dump	Displays the number of tables to dump.
Dumped	Displays the number of tables dumped.
Percentage Complete	Displays the percentage of the tables dumped.
# Processes	Displays the number of processes used for dumping.
Group	**Batch and Run Details**
Directory for Engine Files	Displays the directory where the Engine communication files are located. The Directory is set in the dump parameter table (bcdmp000).
Path of Data Files	Displays the location where the data files are stored. The directory is set in the dump parameter table (bcdmp000).
Clear Directory Content	Select this check box if the content of the data dump directory must be removed. Uncheck this option when you only want to override the dump files for the current dump process.
Start	Displays the start date and time for the dump process.
Finish	Displays the end date and time for the dump process.
Grid	
Process Number	Displays the dump Engine process number.
Main Table	Displays the main table and description of the mapping object which is currently dumping in the dump engine process.
Records Dumped	Displays the number of records dumped by the mapping object.
Company Number.	Displays the company number.
Status	Displays the loading status. The possible loading statuses are: • **Started** – Dump engine process is started. • **Dumping** – Engine process is dumping data. • **Cancelled** – Engine process is cancelled. • **Completed** – Engine process is completed. There are no more migration mapping objects to process.
Command	Description
Resume Dumping	Use this option when the dumping process was stopped.
Start Loading	Use this option to start the dumping process for the selected range.
Stop Loading	Use this option to stop the dumping process.

Add Process	This option adds a dump process engine.
Remove Process	This option removes a dump process engine.
Reset Process	Use this option from the specific menu to reset the migration process after a previous dump process is aborted or dump processes killed.

Number of dump engines

The number of dumping processes that you can invoke is the number of user licenses you have minus 1. The maximum number of dumping engines is 19. Always evaluate the system resources to set the amount of dumping engines. Too much dumping engines can cause to a lower performance and increases the total execution time.

Data Files

During the dumping process, data files (.dat) are generated for each of the Infor LN tables within the range you have specified. If the target data base is MSSQL and the dump engine is ready with processing a mapping object all the dumped data files are transformed to *.da2 files.

Data files in ASCII format are created in the path specified for Directory of data files field in the “Data Dump Parameters” (bcdmp0100m000) session. For example:

```
 <Directory of data files>/<company number>
<package+module>/<tablename.dat>...
```

```
The data file for the tcccp000 table is available in the following location:
```

```
< Directory of data files>/<company number for which conversion is being
run>/tcccp/tcccp000.dat
```

Archive Data Dumping Run (bcdmp5201m000)

Use this session to archive the details of the last dumping process. In case you do not archive the details, the next conversion run overwrites the present information. Hence you must archive the dumping information, after data dumping or resetting the conversion process.

In Conversion Parameter (bcdmp0100m000) session, the ‘Archive time logs after conversion?’ field must be set to “Yes”, to archive the time logs automatically, if the conversion run ends smoothly. If the conversion process got killed, then archiving of the conversion logs must be handled manually.

In the session you can specify any reference information about the last conversion run. You can use this information to interpret the archived information. Click “Continue” to start the archiving of the dump information.

This table explains the fields and commands in this session:

Fields	Description
Enter the reference Text, if any	Use this field to add a reference text to for the archive dumping time log record.
Commands	**Description**
Archive	This option stores the Dumping time log data.

Check currently logged in users (Unix only) (bcdmp5406m000)

Use this session to see which users are currently connected to the system and a specific package combination. The list is presented in a report. The session does not have a screen but only results in a report. The user can select a device to print the report.

You can use this session if you want to know which users are connected to the system at the moment

Check Environment of present Company (bcdmp6400m000)

Before you perform the conversion run, you must check the hardware and software configurations at the source. Use this session to retrieve the information from the system. The information is shown in a report.

The session provides this information:

- User environment variables on the source system.
- Source Porting set information.
- Data size of the source company.
- Free space available for data dumps.
- Information about the creation of all the BC tables.

This table explains the fields and commands in this session:

Fields	Descriptions
Check Porting Set?	Select this check box to display the porting set details of the current Infor LN source environment.
Check User Details?	Select this check box to display the user details of the conversion user in a report.
Check BC Package Tables?	Select this check box if you want the system to check whether the BC package tables are created properly.
Check current Company size?	Select this check box to calculate and report the current company size to the user. This process is time consuming, depending on the data volume.
Check free space on data file directory?	Select this check box to report the free space of your data files directory.
Target Infor LN10.x Server Name	This field is enabled only when your conversion scenario is **Two machines with client server** and the operating system of Infor LN source environment is **UNIX**. This information is used to obtain the FAT client details.
Path of Infor LN FAT client	This field is enabled only when your conversion scenario is **Two machines with client server** and the operating system of Infor LN source environment is **UNIX**. This information is used to obtain the FAT client details.
Actions	Description
Print	Generates reports according to your selections.

Check Scheduled jobs (bcdmp5404m000)

The session is intended to give a report of scheduled jobs along with the next execution date and time. You can use this session to check if there are jobs scheduled to run during a scheduled dumping process. Jobs could lower the performance of the dumping process. The results are displayed as a report.

Analyze Scheduled Jobs is developed on these versions of UNIX:

- HP UNIX : Version HPUX10.20 & Version HPUX11.00
- Sun Solaris: Version 2.5.1

Note: The smooth functioning of this utility on other flavors of UNIX is not guaranteed due to difference in the methodology of each UNIX flavor. This utility works well on Windows. For more details, see these manuals:

- UNIX Manual for Crontab and AT commands

- NT Manual for Task-Scheduler and AT commands
- Job Manual for Baan

This table explains the fields and commands in this session:

Fields	Descriptions
Time (From <> To)	Enter the time range for which you like to check for the execution time of Job. During the data conversion it is preferred that no other scheduled jobs run as they reduce the system's performance, you must enter the time in hours' format.
Report Operating System Jobs?	Select this check box to analyze the operating system- level jobs scheduled in the specified time frame.
Report Baan Jobs?	Select this check box to report the jobs scheduled in the Infor LN source environment.
Report Details of Jobs in all companies?	Select this check box to scan and report jobs scheduled in all the companies of Infor LN source application.
Action	Description
Print	Click this button, to scan and report the scheduled jobs in the specified time frame.

The output report "Analyze Scheduled Jobs", reports the jobs that are scheduled in a specific environment. The jobs at Infor LN10.x level are reported and the details are obtained from the job data. The details of the scheduled jobs at the operating system level are case dependent.

In the UNIX environment, the jobs scheduled using CRONTAB files and AT command are reported. In case of Windows, tasks scheduled using Task Scheduler are reported.

Note: To obtain a report of scheduled jobs, you must have bc_conv.exe in $BSE\tmp directory.

The format of the report generated is displayed below:

- Job name
- Company
  The company number is zero for tasks scheduled at the operating system level.
- Description Source
- Next execution Date and time
- Status
- Periodicity
  Periodicity for OS – tasks for each Month/Day/Week.

UNIX Specific:

- For jobs scheduled using Cron, the Job name is Cron and the description contains the command.
- For AT jobs, the Job name is AT and the description is the name of the AT command. The next execution date is provided and is obtained from the at -l command.

Windows Specific:

- The periodicity is not available for the tasks scheduled through Task Scheduler.

Print Dumping (data file creation) Status (bcdmp9442m000)

The purpose of this session is to print the table details (progress), during the data dumping process. Use this session to find out the target tables' data files that are created, and are in progress.

The result is displayed as a report. When using the print command, the user can select the output device for the report.

This table explains the fields and commands in this session:

Fields	Description
Print	Use this field to make a selection between tables dumped or tables to be dumped.
Tables (From <> To)	Enter the range of target tables for which you want to know the details.
Actions	**Description**
Print	Click the button to generate report that indicates the progress of the dumping process.

Dumping Error logs (Functional) (bcdmp9599m000)

Use this session to view the functional errors such as data inconsistencies. The dumping errors are created by the migration conversion objects or the mapping tables during the data conversion. You can view these errors after data dumping via this session. For example, Data file cannot be opened error…. .

Note: This session only shows logged issues during handling the conversion objects. So only logic pack related log messages.

The fields and actions in this session are explained below:

Fields	Description
Company Number	The company number for which the error has occurred. This is a display field
Table	The (conversion object) target table in which the error occurred. This is a display field
Process Number	The number of the process in which the error occurred. This is a display field
Sequence	This field shows the order in which the errors have occurred.
Error Description	This is the description of the encountered error during the dumping process.
Actions	Description
Clear data…	Use this option to clear all the logs created.

Data Dumping Runs (History)bcdmp5500m000

Use this session to see the run codes with the start and end times of the dumping process per run. After selecting a specific run number, you can use the specific menu to drill down to the Conversion Time Logs per run and the archived error message for the selected run.

If you want to clear the Data dumping runs you can select the run and delete it. All related data is deleted as well.

Note: This session only shows data when the conversion run is archived via the “Archive Data Dumping Run” (bcdmp5201m000) session.

This table explains the fields and commands in this session:

Fields	Description
Run Code	Each time a new dump process is started also a new run code is created.
Start Date	The start date and time of the dumping process.
End Date	The end date and time of the dumping process.
Elapsed time	The total elapsed time of the dumping process.
Comments	The information entered by the user during archiving the data dumping conversion run details. The user must enter some valid information to indicate the purpose of the conversion run.
Commands	Description

Open Dumping Time Logs	Use this option from the specific menu to start the “Dumping Time Logs per Run” (bcdmp5501m000) session. With this option the dumping times per run can be viewed. The option is only available when a run is selected.
Open Error logs	Use this option from the specific menu to view the archived error log messages for the selected run. This option starts the “Error Logs - Archive” (bcdmp5503m000) session.
Print	Using this option starts the “Print Dumping Time Logs” (bcdmp5401m000) session.

Dumping Time Logs per Run (bcdmp5501m000)

Use the session to view the data files created during the dumping run and their corresponding details such as the duration for each table and the number of records dumped.

Using the group buttons you can navigate to other archived conversion runs.

This table explains the fields and actions in this session:

Fields	Description
Run Code	The archived run number and the description of which you are viewing the details. This is a display field
Company	The company number for which the dumping logs are created.
Table	The target table for which the data is stored during this dumping run.
Start Date	The start date and time for the target table specified in the Table field.
End Date	The end date and time for the target table specified in the Table field.
Elapsed Time	Shows the duration of the data dumping for the target table specified in the Table field.
Rec. Count	Shows the number of records saved to the data file for the target table specified in Table field.
Process Number	Shows the dump process number for the target table for each conversion run.
Commands	Description

Error Logs - Archive (bcdmp5503m000)

Use this session to view the archived functional error log message for a dumping run.

This table explains the fields and commands in this session:

Fields	Description
Run	The archived run code for which you want to see the conversion time log details.
Company	The company for which the run is processed.
Table	The name of the script for which the error message is generated.
Error Description	The archived functional error message logged for the processed script within the dumping process run.
Process Number	The Process number in which the data for the table was running. Used when the migration mapping object is split over multiple dumping (process) engines.
Creation Time	This field displays the date and time when the functional error message was logged for the conversion run.

Print Dumping Time Logs (bcdmp5401m000)

Use this session to print all the conversion run details of a specified range. The conversion time log can be printed in the order of the tables or in the order off processing time for the table. The results are displayed in a report.

The conversion run can be used to compare conversion runs or can be used to set the priorities for migration mapping objects with long processing times.

Note: It is recommended for each complete conversion run to print the conversion time (to file).

This table explains the fields and commands in this session:

Fields	Description
Run Code	Specify the Run code from and to range for which the conversion time log details must be printed.
Table	Specify the Table from and to range for which the conversion time log details must be printed.
Company	Specify the Company from and to range for which the conversion time log details must be printed.
Sort mode	The conversion time log can be printed in two sort modes: • According the table names in alphabetical order

	• In an elapsed time order with processing time of zero is at the top.
Sort Order	Use this field to specify if the report must be printed in ascending or descending order.
Commands	Description
Print	This command prints the entered selection on a report. The user is able to select an output device.

Reset Conversion Process (bcdmp0200s000)

The data dumping process locks the conversion parameters so that no other conversion run can be handled for a company at the same time. However, there are situations when, you encounter some problems such as non-availability of disk space, performance/memory overload, restart of the environment/database, the conversion script in debug mode and so on. These problems make the data dumping process non-responsive. The only solution to this situation is to kill the dumping process. However, the locks on the conversion parameters are not removed.

Use this session to release the lock on the conversion parameter. If the lock is present, the Continue button is enabled. Click Continue to release the lock.

Generate scripts to analyse log files (bcdmp5301m000)

Use this session to create files used by the “Analyze log files” (bcdmp5300m000) session. This utility is available only if LN source environment operating system is Unix.

This table explains the fields and commands in this session:

Fields	Description
Bshell name	Use this field to specify the bshell name of the source engine environment as specified in the configuration file (BWC file).
Commands	Description
Continue	Use this option to generate the log files.

Analyse Log Files (bcdmp5300m000)

Use this session to analyze the log file generated by the LN source engine, compile the information in a file and mail it to the user’s mailbox, specified in the parameters (bcdmp000). This utility is only available if the LN source engine is installed on a Unix machine and only for a two server scenario without a client connection.

This session takes some processor time, to perform the periodical checks. You must not use this utility during the Go-Live migration run.

This table explains the fields and commands in this session:

Fields	Description
Analyze Log Files already started?	The field displays whether this utility is running or not. The field contains one of the following values: Yes: The Analyze Log File utility is activated. No: The Analyze Log File utility is not activated.
Commands	**Description**
Start Analysis	Use this option to start the utility. A background process that starts analyzing the log files is initiated.
Stop Analysis	Use this option to stop this utility. The background process stops.

Note: To create the files used by this session, you must first run the “Generate scripts to analyse log files” (bcdmp5301m000) session.

Conversion Simulation (Optional)

Under this menu item you find sessions to run a simulation of the migration mapping objects. The simulation process uses special migration object version to start a record count. The record count and the estimate process time can be used to set the conversion priorities.

Prior to running these sessions, a migration build from the DCA must be generated. Within the Generation settings of the DCA studio the objects must be generated in simulation mode. This adds the record counting code to the migration mapping objects.

The special generated objects are stored in the vrc path in the “/obcsml” directory. The new generated objects do not impact the installed “/obcdll” objects during the installation of the logic pack.

This table explains the available sessions in this section:

Session	Session Code
Source Companies to be Migrated (*)	bcdmp9189m000
Simulate Data Conversion	bcdmp8101m000
Simulation Run Details	bcdmp8502m000
Simulation Run Details (History)	bcdmp8503m000

(*) This session is described in detail earlier in this chapter.

Simulate Data Conversion (bcdmp8101m000)

Use this session to retrieve all the generated simulation objects by the DCA Studio. Per simulation object you can set the “Skip” flag as to mark to object to be used in a simulation process or not. Within the session you can also set a range of objects to “Skip” or not. Once you specified the simulation objects you can start the simulation.

This table explains the fields and commands in this session:

Fields	Description
Company number	The company for which a simulation run is required.
Target Table	The target table for which you want to a simulation.
Bypass Yes/No	Is this target table to be considered in the simulation run or not.
Commands	Description
Generate tables	This option retrieves all generated simulation objects by the DCA Studio from the “/obcsml” directory.
Set Skip Yes/No	The option starts a sub session “Set Skip Yes/No to Range of tables” (bcdmp8201s000). Within the session you can set the “Skip” flag for a range of target tables. You can enter a selection range based upon package, module and table.
Run Simulation	This option starts the simulation run. It takes only the target tables into account that are marked with the “Skip” flag set to “Yes”.

Simulation Run Details (bcdmp8502m000)

In this session you can view the simulation results generated by the previous described session. The user can view the record counting for a source table in a specific company. You can view the start date and end date and the processing time for the simulation.

The current simulation results can be stored in a history table to compare different simulation results. If the data is not stored in history the data is deleted and recreated for the new simulation.

This table explains the fields and commands in this session:

Fields	Description
Company number	The company for which a simulation run is done.
Start Date	The start date and time of the simulation.
Target Table	The target table a simulation run is done.

End Date	The end date and time of the simulation run.
Source table	The used source table.
Number of Records in Source	The number of records in the source table.
Number of migrated records	The number of records that are migrated to the target table.
Commands	Description
Move to History	This option stores the current simulation results in a History table.

Simulation Run Details (History) (bcdmp8503m000)

In this session you can view the simulation History results archived by the previous described session. The user can view the record counting for a source table in a specific company. You can view the start date and end date and the processing time for the simulation and do a compare between other simulation runs. The current simulation results can be deleted from the history table

This table explains the fields and commands in this session:

Fields	Description
Simulation Run number	This display field shows the current Simulation Run Number.
Company number	The company for which a simulation run is done.
Start Date	The start date and time of the simulation.
Target Table	The target table a simulation run is done.
End Date	The end date and time of the simulation run.
Total Records to be Migrated	This field shows the total number of records to be migrated for the current target table.
Source table	The used source table.
Records in source	The number of records in the source table.
Number of migrated records	The number of records that are migrated to the target table.
Commands	Description
Delete	This option deletes the current simulation results from the History table.

Conversion Script Debugging (Logic Pack Development)

Data maintained in the session under the Conversion Script Debugging is used to enable debugging of conversion scripts.

Session	Session Code
Debug DCA Studio generated Conversion Script	bcdmpdebug

To debug DCA Generated Script (bcdmpdebug)

This session is used to start the DCA Studio generated conversion scripts in debug mode.

This table explains the fields and commands in this session:

Fields	Description
Program Script	Specify the DCA generated mapping object to debug. To start a conversion object in debug mode use this legend: <t><mapping object name> <t> = script type; “f” = for normal script and “m” for M-Script The mapping object name consists of: pp = Package (target) mmm = Module (target) xxx = table (target) Example: ftcmcs001 or mtcmcs003 **Note:** For debugging a m-script mapping object the base conversion mapping table must be entered.
Commands	Description
Debug	Use this option to start the debugging process of the specified conversion object.

Chapter 6 DCA3.4 Infor LN Load Engine sessions details

This chapter explains the sessions in the Infor LN Load Engine. You can define the start-up menu as **"bcgenconvert**" on the user data (ttaad2500m000) or start the LN Engine menu from the menu browser. Select **File → Run Program…,** and carry out the 'Program/Session' as "**mbcgenconvert**". This LN Engine menu appears:

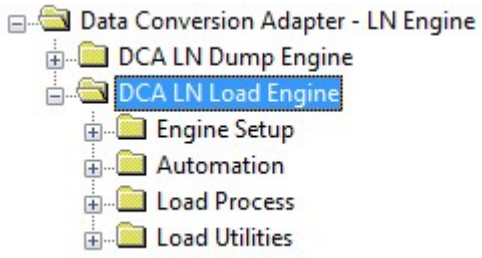

Figure **Error! No text of specified style in document.**-1 LN Load Engine menu

The following sections explain each of the sessions under this sub-menu. In this chapter no screenshots of the separate sessions are added but just the explanation of the fields in the sessions. To view the session open the session via the specified menu or via run program.

In some cases, the conversion menu contains standard LN sessions. These sessions are listed but not described in detail.

Engine Setup

Data maintained in the sessions under the Engine setup are crucial to the migration load process. The sessions under the Engine Setup are listed in the below table and relate to basic settings of the LN Load Engine.

Session	Session Code
Data Load Parameters	bcgct0100m000
Target Companies to be migrated	bcgct9189m000
Generate Onpload Database	bcrun0207m000
General Data Handling Utilities	Submenu
Companies (*)	ttaad1100m000
Create Sequential Dump of Table (*)	ttaad4226m00
Create Table from Sequential Dump (*)	ttaad4227m000
Copy Company Data (*)	tccom0214m000
Create Tables (*)	ttaad4230m000
Delete Tables (*)	ttaad4231m000

(*) This is a standard Infor LN Tools session which is not described in detail in this document.

Conversion Parameters (bcgct0100m000) (Mandatory)

In this session, you must maintain the conversion parameters that control the overall data migration process. You must maintain only one set of parameters for your current working company.

Note: You must maintain Migration Parameters in all the target companies to which you load data. To perform this, you must change your working company and maintain parameters.

When using LNUI the “Client source Connection Properties” fields are not required and thus not visible in the session.

This table explains the fields in the **Source Engine Details** tab of the session:

Fields	Description
Group	**General Details**
Source Release	You must enter your source environment. These values are possible in this filed: **BaanIV**: Select this option if you are migrating data from Infor BaanIVc4 series. **BaanERP**: Select this option if you are migrating data from Infor B50c series. **Legacy**: Select this option if you are migrating data from systems other than the listed Infor ERP Baan applications.

Fields	Description
	LN: Select this option if you are migrating data from an Infor LN environment.
Conversion Scenario	The conversion scenario you want to follow for the migration process. These values are possible for this field: **Single Server**: Both the source environment and the target environment exist on the same server. **Two Machine with client-server**: The source environment and the target environment exist on two separate servers. You can perform the conversion with a client-server connection between the source and the target server. **Two Machine without client-serve**r: The source environment and the target environment exist on two separate servers, you can perform the conversion without a Client-Server connection between the source and the target server (which means dumping is performed independently on the source server, ASCII files are copied to the target server, and then loaded). For more details on the conversion scenario see Chapter 2 **Error! Reference source not found.**.
Group	**Source Environment Details**
Source hostname	Specify the host name of the source environment system. **Single Server Scenario**: You must specify the current Host name, defaulted by the system. **Two Machine with client-server**: When a Fat Client is installed you must specify the current host. When using a shared drive to communicate between source and target then you must use GTM to set the source host name. This field is not applicable for “Source Release” set to “Legacy “.
BSE of Source Env	Specify the BSE directory of the source environment system. This field is not applicable for “Source Release” set to “Legacy “.
BSE_TMP of Source Env	Specify the temp directory of the source environment system. This field is not applicable for “Source Release” set to “Legacy “.
Path of Time zone files	Specify the path where the time zone directories: “zone info” and “zone rules” are located. These directories should always be under the parent directory “/lib”. Ensure the application server runtime and application components have the correct date and time value. For a normal LN installation, the path of the time zoned files are: $BSE/lib/zoneinfo and $BSE/lib/zonerules In the above case the „Path of the Time zone Files“ is: $BSE

Fields	Description
	You can also store the engine files in a shared directory for the source and target engine. For instance, /migration/enginefiles/lib/zoneinfo and /migration/enginefiles/lib/zonerules. In this case the „Path of the Time zone Files“ is: /migration/enginefiles In case of single machine scenario, you can point to the target directory. In case of a two machine scenario, copy the zone rules and zone info files from the target $BSE to source $BSE. Set the BAAN_TZ_DIR to configure the time zone directory. **Note:** Enter the path of the parent directory of the “zone rule” directory without a trailing forward slash. This field is not applicable for “Source Release” set to “Legacy “.
Group	**Client Source Connection Properties**
Installed Directory	The source engine Windows installation directory in the local machine, from which you start the conversion process. You must enter this directory in the MS-DOS naming convention. For example, if you have installed Infor BaanIVc4 Windows in "C:\Program Files\Baan\Baan Windows"; the corresponding DOS naming convention can be "C:\Progra~1\Baan\BaanWI~1". **Note:** The DOS naming convention is machine dependent. To know the DOS name for a specific folder, start the command prompt, go to the parent folder Type the command "dir. /x" to view the DOS naming convention for its sub-directories. This field is not applicable for “Source Release” set to “Legacy “.
Protocol	The Protocol used by the source engine to create a connection to the source engine environment. The available options are: “Rexec”, “BaanLogin”, “SSPI Login”, and “NA”. This field is not applicable for “Source Release” set to “Legacy “.
Port No.	The Port number used by source engine to connect to the source engine environment. This field is not applicable for “Source Release” set to “Legacy “.
User	Specify the user name used to connect to the source engine environment. Ensure that this user is linked to the correct package combination, and has permission to retrieve data from required companies and full permission to all directory structures. This field is not applicable for “Source Release” set to “Legacy “.
Password	Specify the password for the above specified migration user. This field is not applicable for “Source Release” set to “Legacy “.

Fields	Description
Bshell name	The Shell name of the source engine environment configuration. This field is not applicable for “Source Release” set to “Legacy “.

This table explains the fields in the **LN Engine Details** tab of the session:

Fields	Description
Path of Data files	You must enter the directory where the data files are located for data load.
Directory read from above path	If you stored the data dump files without using the <package>/<module> sub-directories, then you can select this check box. This field is mostly “Yes”, if you are migrating data using the legacy source engine. For the other source engines, the field is mostly not selected as it is organized under <package>/<module> sub-directories. Selecting this option enables also the loading method via the Dal You can specify this in the tables by batch sessions.
Directory of Engine files	Specify the directory path for engine files. This the directory used by Engines to share information between the source and target engine and create temporary process files. For example: /migration/engine_files.
Company of EMM	You must specify the company in which Enterprise Modelling data is stored. During the migration of the source data, the EMM data is loaded into the tables of that company. **Note**: This field is mostly applicable in case you are migrating data from Infor ERP BaanIVc4 systems.
Number of Loaders	In this filed you must enter the number of parallel loaders to be started. A higher value here improves the overall performance of the data conversion process. But before giving a higher value, examine the hardware profiles (Number of CPUs, RAM available, etc). The number of loaders can be changed during the conversion run. Click the + button on the loading window to invoke another loader. If you LN database is DB2, the number of loaders is always one and therefore this field is disabled. The maximum number of loaders that can be invoked at the same time depends upon the power of the CPU and other system resources available on the server.
Use bulk loading method	Select this check box, if you use the RDBM’s bulk loading utility. This is the most preferred way in terms of performance.

Fields	Description
	Note: If you are not using bulk loading, then the system uses the Infor LN tool function, “db.insert()”, to load the data, so it like a row by row insert and referenced are checked during inserts. So you should do a create to runtime with disabled references. After migration you can enable the references.
Database	Select the current company’s database in which the date is loaded form the list.
Data files loading command	Shows the data file loading command for specific database. For example, in case of Oracle: /usr/oracle11/homeb4/bin/sqlldr parfile=/data/migration/engine/990/parfile, control=/data/migration/0990/bpmdm/bpmdm001.ctl, log=/data/migration/0990/bpmdm/bpmdm001.log
Commands	Description
Build	You must use this button to specify the database loading command details. Depending on the “Database” selected, these sub sessions are started: • Build Oracle’s data loading command (bcgctoracle) • Build MS SQL data loading command (bcgct3101m000) • Build Informix loading command (bcgct3102m000) • Build DB2 loading command (bcgct3103m000) • Build MySQL’s loading command (bcgctmysql) • Build EDB’s loading command (bcgct3105m000)

Build Oracle’s data loading command (bcgctoracle)

This table explains the fields in the loading command session for Oracle:

Fields	Description
Loader command	The value for this field is sqlldr.
Oracle_SID	The Oracle system Identifier. This variable holds the database name of the Infor LN application, to which you want to migrate data.
Oracle_Home	This variable holds the installation directory of Oracle Database. For example: /usr/oracle11/homeb4
Schema	Enter the oracle schema in which the Infor LN Tables are present.
Oracle user	The Oracle user. Ensure that the user has necessary privileges to load data.

Password	Specify the password for the Oracle user.
Maximum errors per table	Specify the maximum number of errors allowed for each table, during data loading. The loader stops loading data to a table, when errors exceed the value specified in this field.
Date Format	Enter the format of date values in the data files. **Note**: When you use DCA Engine to extract data from source application, generally the date format is SYYYY-MM-DD HH24:MI:SS
Character set	Specify the character set of your data files.
Field delimiter	Specify the field delimiter used in the data files. You can enter the field delimiter as a Keystroke or via the ASCII value. **Note**: The Ctrl+A is the default value for this field.
Varchar database	Select this option if the database is a varchar database or deselect if it is not a varchar database. Depending on the setting the format of the required format file for the bulk loader (control file per table) is different. For a varchar database the trailing spaces per field in the data load file is ignored.
Additional Options	Specify additional command line arguments for sqlldr, if required.
Command	Description
Show	Shows a sample of Par file entry.
Sample	Shows a sample control file.
ASCII =	Invokes the ISO-8859/1 Character Table.

Build MS SQL data loading command (bcgct3101m000)

This table explains the fields in the loading command session for MS SQL:

Fields	Description
Loader command	The loader command for MS SQL is bcp.
MS SQL Installation Directory	Specify the installation directory where MS SQL server utility's binaries (executables) are stored. The data loading process uses the binary called bcp.exe. **Note**: You must specify this directory in DOS naming convention. For example: `C:\PROGRA~1\MICROS~1\100\Tools`
Database	Enter the Infor LN Database.

Fields	Description
Server[\Instance] (-S)	Specify the SQL server host. This is the machine on which the SQL Server database runs. **Note**: In case the database runs under a specific instance name, you must also specify the instance name along with the server. For example, **`LNSERVER\LNINSTANCE`**
MS SQL User (-U)	The MS SQL user. **Note**: Ensure that the user has necessary privileges to load data.
Password (-P)	Specify the password for the user.
Maximum errors per table (-m)	Specify the maximum number of errors allowed for each table, during data loading. The loader stops loading data to a table, when errors exceed the value specified in this field.
Code page (-C)	Specify the code page of the data in the data file.
Batch Size (-b)	Specify the number of rows for each batch of data copied. Each batch is copied to the server as one transaction. SQL Server commits or rolls back the transaction for every batch, in the case of a failure. By default, data in the specified data file is copied in one batch.
Field terminator (-t)	Specify the field delimiter used in the data files. You can enter the field delimiter as a Keystroke or via the ASCII value. **Note**: The Ctrl+A is the default value for this field.
Hint	Specify the hint(s) to be used during a bulk copy of data to a table. The defaulted value is: “TABLOCK” **Note**: This hint is applicable to all the data files.
Additional Options	Specify additional command line arguments for bcp, if required.
Command	Description
ASCII =	Invokes the ISO-8859/1 Character Table.

Build Informix loading command (bcgct3102m000)

This table explains the fields in the loading command session for Informix:

Fields	Description
Loader name	Shows the loader name onpload for INFORMIX database.
INFORMIXDIR	The INFORMIXDIR environment variable specifies the directory that contains the subdirectories in which your product files are installed.

INFORMIXSERVER (-S)	Specify the INFORMIX server host name. This is the machine where the INFORMIX Server database runs.
Database Name	Specify the Infor LN database.
Maximum errors per table (-e)	Specify the maximum number of errors allowed for each table, during data loading. The loader stops loading data to a table, when errors exceed the value specified in this field.
Mode of loading (FCL/FVL)	Indicates the mode of loading the data. These values are possible for this field: FCL or FVL
Log file path (-l)	Specify the path where the log files are to be stored.
Field Terminator	Specify the field delimiter used in the data files. You can enter the field delimiter as a Keystroke or via the ASCII value. **Note**: The Ctrl+A is the default value for this field.
Additional Options	Specify additional command line arguments for bcp, if required.
Command	Description
ASCII =	Invokes the ISO-8859/1 Character Table.

Build DB2 loading command (bcgct3103m000)

This table explains the fields in the loading command session for DB2:

Fields	Description
Loader name	Shows the loader name for DB2 database.
DB2DIR	The directory for DB2.
DB2INSTANCE	Specify the DB2 server host. This is the machine name where the DB2 Server database runs.
Database	The database name in which the specified table or view for DB2 exists. If not specified, this is the default database for the user.
Turned On	The Value V in this field echoes the current command.
Termination Character	Statement termination character. **Note**: T value in this field turns on/off the current statement.
Input From File	Indicates the input for the data loading. **Note**: F in this field indicates the input for loading the data to the target is a file.
Delprioritychar	Specify the current default priority for delimiters, such as character delimiter, column delimiter etc. For example, coldel0x01chardel0xFF.

Fields	Description
Date Format	Specify the valid date elements in the DD-MM-YYYY format.
Time Format	Specify the valid date elements in the HH:MM:SS format.
Time stamp	Specify the valid time stamp format in the source file. Valid time stamp elements can be in the dd-mm-yyyyhh:mm:ss format.
Warning Count	Specify the maximum number of warnings allowed for each table, during data loading. The loader stops loading data to a table, when errors exceed the value specified in this field.
ColumnDelimiter	Specify the field delimiter used in the data files. You can enter the field delimiter as a Keystroke or via the ASCII value. **Note**: The Ctrl+A is the default value for this field.
Additional Options	Specify the additional options, if required for the data loading method for DB2
Command	Description
ASCII =	Invokes the ISO-8859/1 Character Table.
Sample LoadDB2 file	Shows an example of load DB2 file for a company

Build MySQL's loading command (bcgctmysql)

This table explains the fields in the loading command session for Mysql:

Fields	Description
Loader command	The loader command for Mysql is mssqlimport.
MySQL installation path	Specify the installation directory where Mysql server utility's binaries (executables) are stored.
Schema	Enter the Mysql schema in which the Infor LN Tables are present.
MySQL user	The Mysql database user. Note: Ensure that the user has necessary privileges to load data.
Password	Specify the password for the user.
Field Delimiter	Specify the field delimiter used in the data files. You can enter the field delimiter as a Keystroke.

Build EDB's data loading command (bcgct3105m000)

This table explains the fields in the loading command session for EDB:

Fields	Description
Loader command	This field displays the native database bulk loader command. For this EDB it is called “edbldr”.
EDB Installation Dir.	Enter the installation directory of the EDB Database.
EDB Database Name	Enter the name of the EDB Database in which the Infor LN Tables are present.
EDB user	The EDB user. Ensure that the user has necessary privileges to load data.
Password	Specify the password for the EDB user.
Maximum errors per table	Specify the maximum number of errors allowed for each table, during data loading. The loader stops loading data to a table, when errors exceed the value specified in this field.
Date Format	Enter the format of date values in the data files. **Note**: When you use DCA legacy Engine to extract data from source application, generally the date format is SYYYY-MM-DD HH24:MI:SS
Character set	Specify the character set of your data files. **Note**: For EDB the standard character set is UTF8
Field delimiter	Specify the field delimiter used in the data files. You can enter the field delimiter as a Keystroke or via the ASCII value. **Note**: The Ctrl+A is the default value for this field.
Additional Options	Specify additional command line arguments for edbldr, if required.
Any information	Displays the database website for more information on calling the native database loader and its settings.
Command	Description
Show	Shows a sample of Par file entry.
Sample	Shows a sample control file.
ASCII =	Invokes the ISO-8859/1 Character Table.

Target Companies to be migrated (bcgct9189m000)

In this session you can list all the target companies required in the migration process. You can only add companies applicable to the current package Combination of the user.

Fields	Description
Company	The target Company which is filled with migrated data. The description of the company is also displayed.
Command	Description
Generate	Use this command to retrieve all the companies linked to the current package combination of the logged on user.

Automation

The sessions under the menu automation are used to specify information to automate the required migration process sessions.

You do not require handling the sessions in this menu if you are manually handling the process of the migration process checklist.

Session	Session Code
Migration Wizard	Submenu
List of Session for Migration Wizard	bcgct7186m000
Migration Wizard	bcgctmigwizad
Company and Job Mapping	bccnv2101m000
Process Jobs per Company	bccnv2201m000
Error Handling	bcgct6100m000

List of Session for Migration Wizard (bcgct7186m000)

Use this session to add a list of pre-conversion and/or post-conversion sessions to be carried out through the Migration Wizard (bcgct7286m000).

After you start the Migration Wizard, the data in this session cannot be altered or no new data can be entered. Only way out is to cancel the Migration Wizard (bcgctmigwizad) session and then clear the entire data of the List of Sessions for Migration Wizard (bcgct7186m000) session. The technical sessions for source and target side are hard coded and ordered. Custom sessions (sessions manually added by you) can be reordered using Reorder Sequence.

This table explains the fields in the session:

Fields	Description
Sequence Number	Enter the sequence number to indicate the order in which the pre-conversion or post-conversion session should be carried out. It is defaulted by the multiples of 10 but you can change to any value.
Categories	Select the category of the session such as, whether the session must be carried out as a part of pre-conversion or post-conversion process.
Session Code	Use this field to specify the session code of the session to be carried out.
Session Description	Use this field to specify the session description corresponding to the session code.
Environment	Select the environment such as source environment or target environment in which the session must be carried out.
Package	Specify or select the implemented package of the target environment for which the session is carried out. The package name is defaulted with first 2 letters of the session code.
Selected	Select **Yes** or **No** to indicate whether the session is selected to be carried out by the Migration Wizard (bcgct7286m000) session.
Executed	Indicates whether Migration Wizard (bcgct7286m000) session has carried out the session or not.
Mandatory	Indicates whether session execution is mandatory or not for the successful conversion. This is only an information field and does not change the functionality of Migration Wizard (bcgct7286m000) session in any way.
Command	Description
Start Wizard	This commands start running the wizard session.
Clear Data	This command clears the data in the session.
Reorder Sequence	Click Reorder Sequence to reorder the run sequence of the sessions in preconversion or postconversion category. To reorder the sequence: Change the sequence number of the session as per the requirement. For example, if you want the session with current sequence number 30 to become 20, then change the sequence number to any number between 11 and 19. Save the data for the session. Click Reorder Sequence.

Migration Wizard (bcgctmigwizad)

Use this session to start the Migration wizard. Run all the session specified in the “List of sessions for migration wizard” (bcgct7186m000) session in the specified sequence.

It is not required to use the migration wizard. You can always run the migration session in sequence manually. Follow the migration process checklist provided for the Migration Logic Pack.

The steps to run the migration Wizard:

1 When executing the Migration Wizard (bcgct7286m000) session without completing the entire migration process the last time. Aconfirmation dialog box is displayed from the stage where Run Migration Wizard was stopped previously.

2 Click **Yes** to continue with the migration process from the session, where it was terminated last time. Click **No** to start the fresh migration process. Reset the value in the “Executed” field to **No** for all the sessions maintained using the “List of Sessions for Migration Wizard” (bcgct7186m000) session.

3 To verify the list of sessions maintained using “List of Sessions for Migration Wizard” (bcgct7186m000) session click **Check**.

4 After verifying the session list, close the List of Sessions for Migration Wizard (bcgct7186m000) session. Click **Next** from Migration Wizard (bcgct7286m000) session - Screen II. Now the first parameter session you have selected is carried out.

5 After you close the first parameter session, you can continue with the next parameter session. Click **Next**.

6 You can terminate the migration process for the time being by clicking **Cancel**.

7 You can subsequently resume the process by executing the Migration Wizard (bcgct7286m000) session. Click **Yes** in the Run Migration Wizard from the stage where it was stopped previously confirmation box.

8 Click **Back** to go back to the previous session carried out by the migration wizard.

9 After executing source and target side Conversion Parameters (bcgct0100m000) session, manually run the Generate List of Target Tables Being Converted (bcgct9188m000) session at the source side. Then continue with the Migration Wizard (bcgct7286m000) session.

10 When the last post-conversion session is handled, click **Finish** to end the migration wizard.

Company and Job Mapping (bccnv2101m000)

Use this session to list the jobs required during the data load process. The added jobs to this session can be handled via session “Process Jobs per Company” (bccnv2201m000).
This table explains the fields in the session:

Fields	Description
Company	The Company too which the job belongs.

Job Name	The name of the job. The field has a zoom option which starts the Job Data (ttaad5500m000) session.

Process Jobs per Company (bccnv2201m000)

Use this session to handle all listed jobs in the “Company and Job Mapping” (bccnv2101m000) session. The jobs are carried out through OS level commands written into the file dojob in UNIX and dojob.bat in Windows.

This table explains the fields in the session:

Fields	Description
BW Path	Specify the BW path where the bw.exe is located. This is only required when installed on Windows and not required when running under UNIX.
Path+bwc File name	Specify the path and the configuration (bwc) file to start the application and run the Job. This is only required when installed on Windows and not required when running under Unix.
Command	Description
Continue	Use this command to run the listed Jobs in Company and Job Mapping (bccnv2101m000) session.

Error Handling (bcgct6100m000)

Use this session to automate the error handling process. During running migration process sessions, the user might encounter problems. This could be process related or loading related. This session can monitor the errors and if an error is detected notify a list of users via sending an email.

This table explains the fields in the Error Handling (bcgct6100m000) session:

Fields	Description
Mail run time errors	Specify if you want to mail the run time errors, encountered during the migration process, to the user list. You can set this parameter to “Yes” or “No”.
Alert Message	If an error is encountered an email is send as notification. Within this field you can specify the alert message to be displayed in the e-mal.
Event Viewer	This check box is checked if there are records present in the Maintain Event Logs (bcrun999) Logs table. When there are no records in this table the check box is not selected.

	This is a display field.
Loading Errors	This check box is checked if there are records present in the Maintain Error Report (bccnv500) table. When there are no records in this table the check box is not selected. This is a display field.
Is this utility already Running?	When the start and stop button are used or the session is restarted this field displays the monitor status. “No” if monitoring is not started and “Yes” if the monitoring is started.
Check error logs for every	In this field the user can specify the interval for checking the logs files during the migration process.
Name	The name of the user to which the error log mail must be sent.
E-Mail	Specify the email address of the user.
Command	Description
Mailing Options	Specify the mailing system to be used.
Send	Sends a test mail to the list of users.
Event Viewer	View the Event viewer messages. This option starts the “Event Viewer” (bcrun9599000) session.
Loading Errors	View the loading errors. This option starts the “Loading Errors” (bccnv5500m000) session.
Start	Starts monitoring the process error detection.
Stop	Stops monitoring the process error detection.

Load Process

The session listed under the menu Load process are all kinds of session related to loading the data in the target LN environment.

Session	Session Code
Data Load Dashboard	bcgcteasym000
Loading Daemon	bcldr9100m000
Data Validation Utilities	Submenu
Log File	bcrun8588m000
Validate Data Integrity (*)	ttaad4233m000
Check Tables (*)	ttaad4232m000

Session	Session Code
Reorganize Tables (*)	ttaad4225m000

(*) This is a standard Infor LN Tools session that is not described in detail in this document.

Data Load Dashboard (bcgcteasym000)

This Dash board organizes all the necessary sessions in the LN Load Engine at one place. Using this session, you can start the parameters, handle the Batch lists and conversion runs, initiate the dump and/or load process, checking the error handling and access some utilities and reports.

A standard procedure in using this session is to:

- Set up the parameters.
- Define and select a batch to process.
- Define and select the conversion run, if required.
- Synchronize the Batch details with the source engine (automatic or manual)
- Select to start the dump and load process simultaneously or start the load only process.
- Verify the loading results.

This table explains the fields in this session.

Fields	Description
Group	Parameters
Parameters	This display field shows the existence of conversion parameter. It is mandatory to maintain conversion parameter before starting the dash board.
Conversion Scenario	The value in this display field indicates the kind of connectivity available between the source and the target environments. The field can be maintained in the “Data Load Parameter” (bcgct0100m000) session.
Database	The value in this display field indicates the database to which the current company of the target environment is linked. The field can be maintained in the “Data Load Parameter” (bcgct0100m000) session.
Maintain	Starts the “Data Load Parameter” (bcgct0100m000) session for viewing or maintaining the parameters.
Group	Select your Batch
Batch	Specify the batch to use in the conversion run. A Batch holds a list of Infor LN tables to consider in the migration conversion run. You

Fields	Description
	can enter the Batch or select a batch using the zoom option on the field. For the selected field also the description is shown.
Maintain	Starts the “Batch Codes” (bcgct8188m000) session for viewing, creating or maintaining the Batch. From this session you can also access the “Tables By Batches” (bcgct8189m000) session
Group	Store Conversion Run details
Store Conversion Run details?	If this check box is selected, the conversion run details is stored for the conversion run. The fields in this group are enabled. If the check box is cleared the fields in this group are disabled.
Dump and Load time?	Select “Yes”, if you want to store the used dump time and load time per table in the Batch for the conversion run. Select “No” if you do not want to keep the dump time and load time. The field is enabled when the “Store Conversion Run Details” check box is selected.
Count records after conversion	Select “Yes”, if you want to count records after the conversion run and store the results. Select “No” if you do not require the record count per table in the batch of the conversion run. The field is enabled when the “Store Conversion Run Details” check box is selected.
Run number	Specify the run number under which the run details must be stored. Use the Maintain button to create a new Run. The field is enabled when the “Store Conversion Run Details” check box is selected.
Maintain	Starts the “Conversion Runs” (bcgct1550m000) session for viewing, creating or maintaining the conversion run. From this session you can also access the details of previously stored conversion runs like dump and load time per table in the conversion run and the records loaded per table in the conversion run.
Group	Error Handling
Alert & Analyze	Click this button to open the “Error Handling” (bcgct6100m000) session. This session allows you to configure the error handling details, such as mailing options, alert message etc. for the conversion run.
Event Logs	Click this button to open the “Event Viewer” (bcrun9599m000) session. Event logs contain more information about errors encountered during conversion runs. It also contains information about when a specific process is carried out.

Fields	Description
Loading Errors	Click this button to open the “Loading Errors” (bccnv5500m000) session. This session lists the loading errors occurred during a data conversion run. After a conversion run, use this session to check for data failed to load. After starting this session, you must carry out the “Get loading errors” option from the Specific menu to scan the log files and get the errors if any.
Group	Utilities
Delete data (from database)	Click this button to open the “Delete Data” (bccnv5101m000) session. Use this session to delete data from specific tables, before or after the conversion. The data is deleted directly from the underlying RDBMS, therefore no reference checks are performed before deleting the data. After you start the session, you must carry out the ‘Generate… list of tables option from the Specific menu, to view the list of application tables for each company. Then you must select the tables you want to delete. Select the Delete… option from the Specific menu to remove data directly from the database.
Group	Reports
System Details	Opens the Analyze Current Environment (bccnv5405s000) session that allows you to retrieve the information about: Logged on user details Database details (Currently this details are not implemented for MS-SQL) Tables and existing data
Logged in Users	Opens the Details of User Connected (bccnv5406s000) session that allows you to print the number of users connected to the Infor LN environment. This option is available only when your LN server is UNIX.
Scheduled Jobs	Opens the Print Scheduled Jobs (bccnv5402s000) session to generate a report of the scheduled jobs along with next execution date and time.
Loading Status	Opens the Print Loading Status (bccnv9442m000) session. You can use this session during conversion process to know the number of tables loaded and the number of tables pending.
Commands	Description
Synchronize with source	Use this option in case you want to synchronize Batch details with the source engine.

Fields	Description
	For each conversion run you must synchronize the details using this option or manually by maintaining the table by batch on the source and target engine. There are three options: Automatic Import Source Batch Details Export Target Batch Details Automatic: This option synchronizes the details automatically with the source engine. Import Source Batch Details: This option is not applicable in the case of 'Only Load'. This option is applicable only when you manually synchronize the Infor ERP source engine details with your Infor LN Engine. Before you perform this, you must export the Infor ERP source engine details, and copy that file to Infor LN target server. Export Target Batch Details: This option is not applicable in the case of 'Only Load'. This option is applicable only when you manually synchronize the Infor LN Engine details with your Infor ERP Baan source Engine. This action creates a file; you must copy it to your source Infor ERP Baan server and perform an import.
Dump and Load	This feature is used to dump data in the Infor ERP Baan source Engine and load it to Infor LN engine simultaneously.
Only Load	This feature is used when you migrate data from legacy systems to Infor LN. It assumes that you have already created the data files using DCA Engine and these files are copied to the Infor LN machine.

From the Data Load Dashboard session, this session can be started.

Session	Session Code
Data Load Parameters (*)	bcgct0100m000
Batch Codes	bcgct8188m000
Conversion Runs	bcgct1550m000
Error Handling (*)	bcgct6100m000
Event Viewer	bcrun9599m000

Session	Session Code
Loading Errors	bccnv5500m000
Delete Data	bccnv5101m000
Analyze Current Environment	bccnv5495s000
Details if Users connected	bccnv5406s000
Print Scheduled Jobs	bccnv5402s000
Print Loading Status	bccnv9442m000
Select Table Range for Load only without Dumping	bccnv0205m000
Loading Daemon	Bcldr9100m000

(*) This LN Engine session is described elsewhere in this chapter.

Batch Codes (bcgc8188m000)

Use this session to define a Batch code. Within a batch code you can link tables you want to migrate. For each migration run you can use one batch code.

This table explains the fields in the session:

Fields	Description
Batch	Use this field to specify a batch code. The Batch code is used within the dashboard session for a migration run.
Batch Description	Use this filed to specify the description of the Batch code
User Login	This field shows the user name who created the Batch. This is a display field and is defaulted with the logged on user on the moment of the batch creation.
Creation Date	This field shows the date of the batch creation. This is a display field and is defaulted with the actual date and time on the moment of the Batch creation.
Master Data only?	Use this field to specify if a batch is only for master data tables or not. This field is synchronized with the source dump engine and the setting can be used within the source engine dump process by the migration mapping objects.
Load Via DAL2	With this field the user can determine how the linked tables for a Batch must be loaded. By default, this value is not selected. When selected all tables linked to the current batch are loaded via triggering a DAL2 insert instead of directly presenting the data to the native loaders.

	This option is only visible when the “Directly read from above path?” variable is selected in the “Data Load Parameters” (bcgct000).
Text	This field shows if Text is added to the Batch Code record.

Command	Description
Tables by batches	Use this command to specify the tables linked to a specific Batch. This command starts the “Tables by Batches” (bcgct8189m000) session.
Copy/Synchronize/Export/Import Batch to companies	Use this command to migrate data to multiple companies and to use the same batch settings as defined in current company. Before using this command first specify the target companies in the “Target Companies to be Migrated” (bcgct9289m000) session. This command starts the “Copy/Synchronize/Export/Import Batch to Companies” (bcgct9289m000) session.
Copy Batch details from another Company	Use this command to copy the Batch details from another company. This command starts the “Copy From Another Company” bcgct8283m000) session.
Export/Import Batch details	Use this command to synchronize the Batch details manually. This command starts the “Export/Import Batch details” (bcgct8288m000) session.
Print	Use this command to print the selected batch and the Table by batches linked to the batch. The output is printed send to a reported and the user must select a device where to send the report to.

Tables by Batches (bcgct8189m000)

Use this session to link tables to a Batch. In this session you can also specify how the table for a batch code should be loaded into the LN database.

In case of a migration Dump and Load process the batch list between the target and the source engine must be synchronized. During the synchronization process, which can be started from the Batch Codes (bcgc8188m000) session, fields are synchronized.

This table shows the synchronized fields:

Field	Synchronized to source	Synchronized from source
Batch Code	Yes	No
Table Name	Yes	No

Field	Synchronized to source	Synchronized from source
Skip	Yes	Yes
Script Present	No	Yes
Logical Table	Yes	No
Duration	Yes	No

This table explains the fields in the session:

Fields	Description
Batch Code	This field shows the selected Batch Code.
Table Name	This field shows the table code and the description linked to the Batch Code.
Priority	Use this field to define the load priority of a table to be processed. The “0” value is the highest priority and the “9999” (default) value the lowest. With setting the priority you can set the load sequence of a table (dependency) which is required for a batch code with DAL2 Loading. This option is only visible when the “Directly read from above path?” variable is selected in the “Data Load Parameters” (bcgct000).
Skip?	The value in this field indicates whether the table is considered for migration or not. If this check box is selected, the table is bypassed (not considered) during the data load process. If this check box is cleared, the table is considered for migration. The value is synchronized with the source engine. If the field is set to skip at the LN side it gets the same setting in the source engine. If the Skip flag is not selected at the LN side the value can be determined at the source engine depending on the availability of a Migration Script object.
Script Present?	This display fields indicates if for the table a Migration script object is present within the source Engine. This field is synchronized from the source engine. If there is no script present, the skip flag is selected.
Logical Table?	This display field indicates if the table is a logical or physical table in the current company. If the table is a logical table, the skip flag is defaulted to yes.

Fields	Description
	Within a company only the physical tables are populated with migrated data.
Duration	With this field you can group a table according to the expected dump time of the table. Select Longer if the table is extremely large and select Shorter for the other tables, which is also the default value. This field is synchronized with the source dump engine. While dumping, all the tables marked as Longer are dumped first, followed by tables marked as Shorter. This helps in load balancing, and ensures that dumping the large tables does not start at the end and take a long time when all the other Bshells are already completing the dumping process. Tables that usually contain a large amount of data are marked as Longer. You can choose a list of tables manually, as per your requirements. During the migration process you must address the Load balancing. It is a technique to improve the overall performance and runtime of the migration process (dump and load). Another load balancing technique is setting priorities within the dump engine.
Load Option	The value in this field indicates if the current present data in the target table must be overwritten or retained during migration. The allowed values are: Truncate (default value) If the table on the target side already has some data, the data is deleted and data from the table on the source side is added to the table on the target side during migration. Append The data from the table on the source side is appended to the already existing data in the table on the target side. The Truncate option is not available in the Informix-specific loader. If the load option is Truncate, the table rows are manually deleted with SQL, and the Append option is the default option.
Mode of Loading	The value in this field is used during the generation of the control file. The setting determines how the native database loader inserts the records into the database. The allowed values: Automatic (default value) All records are added (batch wise) and afterwards the index check is done (Bulk load). Conventional

Fields	Description
	The records are added record for record and after each record an index check is done (comparable to a db.insert) Below are two examples. The Oracle SQL* Loader has two load modes. The two modes are Direct and Conventional. If you set the load mode to Automatic, and the size of the data file is greater than 10,000 bytes, the load mode of SQL* Loader is set to Direct. If you set the load mode to Conventional, the load mode of SQL* Loader is also set to Conventional. Informix HPL has two load modes. The two modes are Express and Deluxe. If you set the load mode in Informix to Automatic, and the size of the data file is greater than 10,000 bytes, the load mode of Informix HPL is set to Express. Otherwise, the load mode is Deluxe. If you set the load mode to Conventional, the load mode of the Informix HPL is set to Deluxe.

Command	Description
Generate	Use this option in the Specific menu to generate the list for a range of tables under a batch. It starts the “Generate table List” (bcgct8289m000) session. For the session details see the next section.
Verify List of tables for completeness	Use this option in the Specific menu to generate a list of expected/missing tables in the batch. This option start the “Verify List of Tables for Completeness” (bcgct8489m000) session. For the session details see the next section.
Set Options to Range of Tables	Use this option in the Specific menu to update the values in the session for a range of tables simultaneously. Options which can be set are: Skip? Load Option Mode of Loading Duration This option starts the “Set Options to Range of Tables” (bcgct9285m000) session.
Update Table sharing Information	This option resets the “Logical Table” field. When after a batch is generated or created the table sharing is changed then you can

Fields	Description
	use this option to reset the “Logical Table” field according the current table sharing information in the system.

Generate table List (bcgct8289m000)

Use this session to update the values optionsfor a range of tables simultaneously for a specified Batch. After running the session, the value of the option is changed for all tables falling in the specified range of the batch.

These options are updated:

- Skip?
- Load Option
- Mode of Loading
- Duration

For a detailed description of the possible values of the option see the field description of the “Tables by Batches” (bcgct8189m000) session.

Use this session when the batch is already generated and synchronized with the source engine and the user updated the Batch List either on the target engine side or the source engine side.

This table explains the fields in the session:

Fields	Description
Generate Tables from	Select one of the options from which source the tables for the batch must be generated.
Runtime Data Dictionary (RDD)	For the specified selection range (Package, module and table) the target tables list to be loaded is generated from the target data dictionary.
Tables with errors from previous Run	For the specified selection range (Package, module and table) it picks up all the tables which were not properly migrated in the previous run. Before using this option run the “Generate Error Report” (bcgct5200m000) session. If there are no errors in the previous run detected, then the message “No data within selection” is shown.
Conversion DLL’s generated by DCA	This option generates the target tables list to be loaded from source conversion DLLs generated by DCA Studio. To use this option first, the List of target tables must be exported at the source engine and stored in the location as specified in the Data Load Parameter in field “Directory of Engine Path”.

Fields	Description
	Note: When using this option, the specified ranges are ignored but a complete batch list is generated upon all available conversion scripts at the source engine. This option is not applicable for the conversion scenario" two Machine without Client/Server"
Data Files from the path Below	This option generates a batch list upon the available exported data files in the specified directory. This option is only applicable when the source release in the conversion parameters is set to "Legacy". When this option is selected also the "Path" field is enabled to specify the directory location of the data files. Note: When using this option, the specified ranges are ignored. A complete batch list is generated upon all available data files in the specified directory.
Package	Specify the "from" and "to" package for the range.
Module	Specify the "from" and "to" module for the range.
Table	Specify the "from" and "to" Table for the range.
Command	Description
Generate	Use this command to generate the tables for the current Batch with the specified generation option and for the specified Range.

Verify List of Tables for Completeness (bcgct8489m000)

Use this session to print a list of expected tables in the batch. The session checks the references of a specific table and sees if the referenced tables are part of the batch or not.

Based upon the results the user can decide to expand the Batch list with the missing or expecting tables. If this action is needed depends on the how the batch code is used.

The session puts the results on a report which can be printed/saved/viewed according to the user's Device Selection.

On the report the table in the batch is shown with all the referred tables. Per referred table you can see the reference mode, if it is a physical or logical table, and on which level the referred table exists. The current table can refer to another table (level 1) which in turn can refer to yet another table (level 2).

The Reference mode field in the report shows one of these two values:

- **Mandatory:**
  Must be added to the list of tables present in "Tables by Batches" (bcgct8189m000) session or the data of the referred table is already present or is loaded later.
- **Mandatory unless empty:**

Should be added to the list of tables present in “Tables by Batches” (bcgct8189m000) session unless those are empty tables or the data of the referred table is already present or is loaded later.

- **Not Mandatory:**

  Not required to be present in the list of tables in “Tables by Batches” (bcgct8189m000) session.

This table explains the fields in the session:

Fields	Description
Checking Batch	This display field show the batch code which is checked for completeness.
Command	Description
Print	Use this option to print the verify completeness list for the specified Batch.

Set Options to Range of Tables (bcgct9285m000)

Use this session to update the values of the below options for a range of tables simultaneously for a specified Batch. After running the session, the value of the option is changed for all tables falling in the specified range of the batch.

- Skip?
- Load Option
- Mode of Loading
- Duration

For a detailed description of the possible values of the option see the field description of the “Tables by Batches” (bcgct8189m000) session.

Use this session when the batch is already generated and synchronized with the source engine and the user updated the Batch List either on the target engine side or the source engine side.

This table explains the fields in the session:

Fields	Description
Set value for “Skip”?	Select this check box if you want to set the “Skip?” flag for a range of tables in the batch. If the option is selected specify the required value, “Yes” or “No”, using the radio button.
Set value for “Load Option”?	Select this check box to set the “Load Option” flag for a range of tables in the batch. If the option is selected specify the required value, “Truncate” or “Append”, using the radio button.

Fields	Description
Set value for “Mode of Loading”?	Select this check box to set the “Skip?” flag for a range of tables in the batch. If the option is selected specify the required value, “Automatic” or “Conventional”, using the radio button.
Set value for “Duration”?	Select this check box if you want to set the “Duration” flag for a range of tables in the batch. If the option is selected specify the required value, “Shorter” or “Longer”, using the radio button.
Package	Specify the “from” and “to” package for the range.
Module	Specify the “from” and “to” module for the range.
Table	Specify the “from” and “to” Table for the range.
Command	Description
Continue	Use this command to update the tables of the current Batch with the specified value of the option, if the option is selected.

Copy/Synchronize/Export/Import Batch to Companies (bcgct9289m000)

When there are multiple target companies to be migrated as specified in the “Target Companies to be Migrated” (Bcgct9189m000) session, then this session can be used to:

- Copy a batch and the linked table to the batch to another target company.
- Synchronize the batch for the target companies to consider with the source engine.
- Export the Batch list information for the source engine, for the target companies to consider.
- Import the Batch list information from the source engine, for the target companies to consider.

Based upon the action chosen in the calling session one of the above activities is enabled.

Using this session, the user does not have to invoke the action in each of the target companies.

This table explains the fields in the session:

Fields	Description
Batch + Description	This field displays the batch code and the description on which the activity applies.
‘Skip’ tables which are having data?	When this option is selected, for the activity, the program checks if in the target company the table contains data or not. If so the field “Bypass” is set “yes”.

Fields	Description
‘Skip’ tables which are logical in the target company?	When this option is selected for the activity, the program checks if in the target company the table is a logical table or not. If so the field logical is set to “Yes” and the “Bypass” field is set “yes”.
Copy Parameter too?	When this option is selected, for the activity, the program copies the parameter table. This option is ignored for the synchronization, export, and import process.
Command	Description
Copy	Starts the copy process to the “to be considered” target companies. If the table is a logical table in the target company, the table is skipped. During a copy batch action this is always checked and therefore the “Skip tables which are logical in the target company?” check box is selected and disabled. Command is enabled based upon the action in the called from session.
Synchronize	Starts the synchronization process between the “to be considered” target companies and the related source engine companies. Command is enabled based upon the action in the called from session.
Export	Starts the export process for the “to be considered” target companies. Command is enabled based upon the action in the called from session.
Import	Starts the import process to the “to be considered” target companies. Command is enabled based upon the action in the called from session.

Copy From Another Company (bcgct8283m000)

Use this session to copy the linked table related data for the currently selected batch code. The data is copied from the company number and batch code specified.

Once the table related data copy process is over, the Process completed message box is shown.

Note

If the Tables by Batches (bcgct8189m000) session already consists of table related data for the currently selected batch code, a message box is displayed.. The message box provides these buttons:

- **Append**: To add the copied data to the existing data in Tables by Batches (bcgct8189m000) session.
- **Overwrite**: To delete the existing data and to store only the copied data in Tables by Batches (bcgct8189m000) session.
- **Cancel**: To cancel the action.

Before you can use this session, the Maintain Batch Codes (bcgct8188m000) session should be running. The Copy From Another Company (bcgct8283m000) session can be invoked from the Specific menu of Maintain Batch Codes session.

This table explains the fields in the session:

Fields	Description
From Company	Use this field to specify the company number from which the table related data must be copied.
Batch	Use this field to specify the batch code from where the table related data must be copied.
Command	Description
Copy	Click this button to start the copy batch process.

Export/Import Batch details (bcgct8288m000)

This session allows the user to export or Import the table by batches information to an ascii file. The user can specify a range of batch codes used in this export or import process.

During the export all the tables for a batch code, in the specified batch code range, are stored in an ascii file called: batches.lst.

During the import all the tables for a batch code, in the specified batch code range, stored in the ascii file called: batches.lst, are imported in the current company.

This table explains the fields in the session:

Fields	Description
Action	Set the action to Export or to Import the Batch list details.
Path of the Batch file	Specify the location where the ascii file must be stored.
Batch code	Specify the Batch code from and to range.
Command	Description
Continue	This command starts the export of import of the batch details to or from an ascii file for all batch codes falling in the specified batch code range.

Conversion Runs (bcgct1550m000)

Use this session to maintain run numbers for conversion processes.

This session can be used to create new run numbers or delete existing run numbers or edit description of an existing run number, for the conversion process.

A conversion run number is used to store conversion process information like table dumped, loaded, amount records converted and time taken. By using different run numbers for each conversion the user can store the conversion run information for each run and is able to make compare run numbers.

This table explains the fields in the session:

Fields	Description
Run	This field displays the unique run number for a given conversion run.
Description	Use this field to add a description to the run number.
Date	This field displays the date and time when the run number was created.
Command	**Description**
Time Take for Dumping and Loading	Starts the “Time Taken for Dumping and Loading” (bcgct1552m000) session
Number of records in Conversion Phases	Starts the “Number of records in Conversion Phases” (bcgct1551m000) session. This session allows the user to track the record count for migrated tables in the different phases of the migration process.

Time Taken for Dumping and Loading (bcgct1552m000)

Use this session to display the time taken for dumping and loading a table for a specific company and a run number.

To display the data, use the Find button and specify these details:

Run Number:
Specify the Run Number for which the time taken to dump and load a table for a specific company, must be displayed. Zoom session button can be used to select the Run Number from the separate session.

Company Number:
Specify the Company Number under which, the time taken for each table must be displayed.

Table Name:
Specify the Table Name, for which the time taken must be displayed. This field can be used to display the time for a single table else all the tables for the specified company are displayed.

Before you can use this session, the dumping time information must be retrieved from the source engine and the loading time from the target engine. The user can initiate the retrieval of the dump and load information via the “Retrieve” option in the specific menu.

This table explains the fields in the session:

Fields	Description
Run number	The run number and the description of the run number for which the data is displayed.
Company	This field displays the company number for which the data is displayed.
Table Name	This field displays the table code.
Dumping Start	This field displays the start date and time for the dump process of the specified table at the source engine.
Dumping End	This field displays the end date and time for the dump process of the specified table at the source engine.
Dumping Elapsed	This field shows the dumping time taken for the specified table.
Loading Start	This field displays the time when the dumping of the table, on the source side was completed.
Loading End	This field displays the time when the loading of the table, on the target side was completed.
Loading Elapsed	This field shows the loading time taken for the specified table.
Command	Description
Retrieve	Retrieves the dumping and loading information for the specified Run and company number.
Find	Use the find command to zoom to a specific Run and Company number.
Print	Starts the “Time Taken for dumping and Loading” (bcgct1452m000) session.

Time Taken for dumping and Loading (bcgct1452m000)

Use the session to print the dumping and loading time for a specified range to a report.

You can set the ranges by: Run number, company number, and Table. The data to show on the report can be sorted by: Table, Elapsed dumping time, or Elapsed loading time.

This table explains the fields in the session:

Fields	Description
Run number	Specify the range of Run numbers for the report using the “from” and “to” field.
Company	Specify the range of Company numbers for the report using the “from” and “to” field.
Table	Specify the range of Table codes for the report using the “from” and “to” field.
Sort Mode	Specify how the data on the report must be sorted. Possible values are: Sort by Table Sort by Elapsed dumping time Sort by Elapsed loading time
Command	Description
Print	Creates and sends the report data to a specified device.

Number of records in Conversion Phases (bcgct1551m000)

Use this session to see the number of records for a specific Run Number and conversion phase. The display is classified as number of records per table per company and conversion phase.

To display the data, click **Find** and specify this information:

Run Number:
Specify the Run Number for which the time taken to dump and load a table for a specific company, must displayed. Zoom session button can be used to select the Run Number from the separate session.

Conversion Phase:
Select the Conversion Phase from the drop down list. The options provided are as follows:

- Pre-conversion: Before the data is loaded into target environment.
- After Conversion: After conversion run and before the post migration sessions are carried out.
- After Post-Conversion: After all the post migration session are carried out.
- After Reorganize Tables: After the session ttaad4225m000 is carried out.

Company Number:
Specify the Company Number under which, the time taken for each table must be displayed. If the company number is not specified, all the companies for the specified run number are displayed.

Table Name:
Specify the Table Name, for which the time taken must be displayed. This field can be used to display the time for a single table else all the tables for the specified company is displayed.

This table explains the fields in the session:

Fields	Description
Run number	The run number and the description of the run number for which the record count is displayed.
Company	This field displays the company number for which the record is displayed.
Conversion Phases	This field displays the conversion phase for which the records are displayed. The possible phases are: Pre-Conversion After Conversion After Post Conversion After Reorganize tables
Table Name	This field displays the table code for which the record count is stored.
Number of Records	This field displays the record count for the table.
Command	Description
Count	Starts the “Count number of records in Conversion Phases” (bcgct1251s000) session. The above session is used to start the record count of the migrated tables for a specific phase and a specified range of companies.

Count number of records in Conversion Phases (bcgct1251s000)

Use this session to start a record count of the migrated tables for the different migration phases at the target engine side per company or for a range of target companies.

The different phases:

- Pre-conversion:
  Before the data is loaded into target environment.
- After Conversion:
  After conversion run and before the post migration sessions are carried out.
- After Post-Conversion:
  After all the post migration session are carried out.
- After Reorganize Tables:
  After the session “Reorganize tables” (ttaad4225m000) is carried out.

The data is stored in the system linked to the run number and can be viewed in the “Number of records in Conversion Phases” (bcgct1551m000) session.

This table explains the fields in the session:

Fields	Description
Run number	The run number for which the record count is initiated.
Conversion Phases	Use this field to specify the conversion phase for which the records must be stored. The possible phases are: Pre-Conversion After Conversion After Post Conversion After Reorganize tables
Company	Use this field to specify for which company the record must be run. Using the “From” and “To” field the user can specify a single company or a range of companies.
Command	Description
Count	Starts the record count of the migrated tables for the specified range of companies.

Event Viewer (bcrun9599m000)

Use this session to view all the migration related generated processing messages. During the loading processes several processes log informational or error messages in the event view log.

The session shows which load process logged the message and gives the type of the message. Possible message types:

- Critical
  Used for blocking processing errors such as: “Cannot continue process because file is not present”.
- Warning
  Used for non-blocking processing errors like: “Process completed but not possible to send error report”.
- Information
  Used for informational messages such as: “Process started/stopped”.

To prevent overloading of the event logging table the user is able to clear all events from the table.

This table explains the fields in the session:

Fields	Description
Time	This field displays the date and time when the message was logged.
Source	This field displays the source process which logged the message.

Description	This field displays the description of the message.
User	This field displays the user who started a process which logged a message.
Type	This field displays the message Type of the logged message. This field can have these message types: • Critical • Warning • Information

Command	Description
Clear All Events	Starts clearing all migration events stored in the current company.

Loading Errors (bccnv5500m000)

Using this session, the user can analyze the loading errors for a conversion run. During the actual upload of the data files into the data base, a loading error can occur. These errors are generated by the native database loader used and are stored in the loading log file of the table.

All loading errors must be analyzed for each conversion run and if possible resolved. Not resolving a loading error can lead to a processing issue while running LN application processes and therefore must be reported to the migration customer.

Using the “Get Loading Errors” command from the specific menu the user can start retrieving the loading errors from the log files. Using the “Print” command the user can print the errors found.

This table explains the fields in the session:

Fields	Description
Company	This field displays the company number.
Table	This field displays the table for which a loading error is present.
Remark Series	This field displays the sequence of the loading error per table. If more than one loading error is present per table this series field is used to make the loading error unique for the “Error Report” (bccnv500) table.
Records Loaded	This field displays the amount of records loaded into the table.
Records Rejected	This field displays the amount of records rejected for the table.
Remark	This field displays the Loading Error message from the log file.

Command	Description

Fields	Description
Get Loading Errors	Starts “Get Loading Errors” (bccnv5200m000) session. In this session the user can specify a range for which the loading errors are retrieved.
Print	Starts the print session: “Loading Errors” (bccnv5400s000).

Get Loading Errors (bccnv5200m000)

Use this session to collect the loading error information from the loading log files. The loading log files are created by the native data base loader. Depending on the Data base the log files are analyzed for specific error information. After the loading errors are retrieved the user can view the errors in “Loading Errors” (bccnv5500m000) session.

Before retrieving the loading errors per company the current records present records are cleared. If no loading errors are detected the message “No data within selection” is shown. After retrieval, the loading errors are stored in the “Error Report” (bccnv500) table.

This table explains the fields in the session:

Fields	Description
Path of Data Files	Use this field to specify the location of the Data files. The loading log files are stored in the same directory. The loading errors are retrieved from the log files.
Company	Use this field to specify the range of companies for which the loading errors must be retrieved.
Command	Description
Continue	Starts retrieving the loading errors for the companies in the range.

Loading Errors (bccnv5400s000)

Use this session to print the loading error information per company and table. After specify the print selection the user can specify an output device for the report.

This table explains the fields in the session:

Fields	Description
Company	Specify a range of companies for which the loading errors must be printed.

Table	Specify the range of tables for which the loading errors must be printed.

Command	Description
Print	Starts retrieving the loading errors for the specified selection range and sends the information to a specified output device.

Delete Data (bccnv5101m000)

Use this session to flush data from tables using the flush table functionality of the native data base loaders.

In case the current company must be reused for a conversion data load process you might speed up the loading process by deleting the current content of the tables. For a few tables the Delete and Create table option from the LN tooling is an option but for a large range of tables this might take some time.

This session provides an alternative method and uses the native data base loaders to flush the table of the specified tables.

Before you can delete data, a list must be generated. After the list is generated the user can set “Delete?” on or off. During the data deletion process only tables with the “Delete?” check box marked are deleted.

This table shows the fields and commands in the session:

Fields	Description
Company	Displays the company number for which data in the tables must be deleted.
Table Name	This field shows the table name.
Delete?	Select this field to delete the data from the table. Clear the field to keep the current data in the table.

Command	Description
Generate	Starts the “Generate” (bccnv5202m000) session to generate the list of tables applicable for the current company to be deleted.
Delete	Starts the delete data process of the tables marked for deletion.

Generate (bccnv5202m000)

This session generates a list of tables for a company which is applicable for the delete process. The tables applicable for the process are based upon the package combination for the company (RDD).

For all tables in the list the field “Delete?” is set to “Yes”. Only all tables related to the BC package are by default set to “Delete?” is “No”. As you are in a conversion process you do not want to clear the data of the current conversion run of previous runs.

This table explains the fields in the session:

Fields	Description
Company	Specify the company range for which the delete list must be generated.
Command	Description
Generate	Generates the list of tables applicable for the company in range.

Analyze Current Environment (bccnv5495s000)

Use this session to retrieve information on the current environment based upon the selected options:

- User Details
- Database details
- Tables and existing data Details

The details are printed to a report.

This table explains the fields in the session:

Fields	Description
User Details?	Select this option if you want to print the current logged on user details.
Database Details?	Select this option if you want to print details about the currently used database schema.
Tables and existing data Details?	Select this option if you want to print the tables applicable for the current company and if the table has existing data.
Command	Description
Print	This command prints the data according to the specified options to an output device selected by the user.

Details of Users connected (bccnv5406s000)

Use this session to print a report of all connected users to the current system.

Print Scheduled Jobs (bccnv5402s000)

Use this session to print a report of all currently scheduled Jobs. The user can specify which jobs must be checked for a given date/time range.

While running the migration processes to have the best performance all running and scheduled jobs must be stopped.

This table explains the fields in the session:

Fields	Description
Operating System Jobs?	Select this option to check the jobs for the Operating system.
LN Jobs?	Select this option if the LN Jobs must be checked.
From all Companies?	Select this option to also check all jobs for all companies specified for the current environment. When this option is not selected only the jobs specified for the current company are checked.
Selection Range	Use the selection range to specify a date range to check for scheduled jobs.
Command	Description
Print	This command prints the scheduled job data according to the specified options to an output device selected by the user.

Print Loading Status (bccnv9442m000)

Use this session to view the status of the loading process. With this session you can print all loaded tables for the current batch or all tables to load for the current batch.

The result is send to a report and the user can decide on which device to use for printing the report.

Note: You can generate and view the status report even when a conversion run is in the process.

This table explains the fields in the session:

Fields	Description
Batch	This field displays the current batch.

Fields	Description
Options	Use this field to specify input criteria for kind of tables to consider while generating the status report. The possible values are: • Tables Loaded: All the tables that are loaded are shown in the report. • Tables to Load: All the tables loaded and are waiting to be loaded are shown in the report.
Command	Description
Print	This command prints the “Tables to load” or the “Tables loaded” for the current batch.

Select Table Range for Load only without Dumping (bccnv0205m000)

Use this session to load the data on the target server. This session can be used in these two situations:

- If the source and the target environments are present on two separate servers without a ERP client-server connection and the data files after copying them manually to the target environment server, must be loaded into the target database tables.
- If you must reload all or some of the data from the data files into the target database tables.

You have the flexibility to load tables in a specific range, or a specific table. Make the selection by package, by module, or by a table using Select Table Range for “Load Only without Dumping” (bccnv0205m000) session.

After the selection is complete you can hit the continue button to start the “load only” process. The system uses the Data load parameter settings and scans for the dump files in the “Data dump file” directory. Only the data files falling in the specified range are selected and passed onto the “Loading into LN” (obcrun90505) processing screen.

This table explains the fields in the session:

Fields	Description
Package	Use this field to specify a range of packages for the data load process.
Module	Use this field to specify a range of modules for the data load process.
Table	Use this field to specify a range of tables for the data load process.

Command	Description
Continue	This command starts the “Loading into LN” (obcrun90505) processing screen and initiates the Loading process.

Loading into LN (obcrun90505)

This screen is started from the “Select Table Range for Load only without Dumping” (bccnv0205m000) session.

This session is used to monitor the actual data load of the dumped files into the LN database. The user can see the number of data files present for loading and falling in the specified range and also how many tables are loaded. A progress bar displays the overall progress of the data load session.

The screen also shows the amount of Load engines initiated and the currently loaded table per load process. Initially the program starts a number of loaders as specified in the “Data Load Parameters”. The user is able to add or remove additional loaders. Monitor the system performance to determine if additional loaders can be initiated or removed.

Note:The progress bar does not represent the progress of the loading process in terms of volume of data loaded but indicates the progress based on the number of tables loaded so far.

After the loading process is finished, Loading Complete message box appears.

This table explains the fields in the Screen:

Fields	Description
Tables to load	This display field shows the number of tables available for loading and within the specified loading range.
Tables Loaded	This display field shows the number of tables currently loaded.
Progress bar	The progress bar show the progress graphically based upon the number of tables to load and loaded. It does not depend on the amount of records to load per table.
Loading Shells	In the loading shell field, the user can see which table is currently processes/loaded by a specific loading shell. If more loading shells are invoked additional loading shell fields are initiated.
Command	**Description**
Plus/minus button	Using these commands, the user can revoke loading shells or invoke additional loading shells. Click the [+] button to start additional loading process. In case your system’s resources are underutilized, you can start an additional load process for faster loading. Click [-] button to remove a loading process, when your system is overloaded.

Loading Daemon (bcldr9100m000)

Use this session when loading data onto LN using LNUI. The previous used data load sessions are still available but does not work in an LNUI environment.

When initiating the loading from the dashboard the dashboard detects LNUI is used or not. If session “Loading Daemon” (bcldr9100m000) is used for the loading process else session “Select Table Range for Load only without Dumping” (bccnv0205m000) and “Loading into LN” (obcrun90505) are used.

This session is used to monitor the actual data load of the dumped files into the LN database. The screen shows the amount of Load engines initiated and the currently loaded table per load process. Initially the program starts a number of loaders as specified in the “Data Load Parameters”. The user is able to add or remove additional loader processes. Monitor the system performance to determine if additional loaders can be initiated or removed.

The session can be used in a “Only Load” or “Dump and Load” situation.
“Only Load”:
Use this option when the dumping process is finished and the data files are available for the LN Load engine to be loaded. You can choose to define ranges to only load a specific range or even a specific table.

“Dump and Load”:
Use this option when the dumping process is started at the dump engine side and the user wants to start loading data for tables finished with the dumping process. This saves total process time as the Dumping and loading process are running in parallel. This process is only possible when the dump and loading engine can both access a shared location where the data files and communication files can be stored.
In the dump and load setting the user cannot set ranges as all dumped tables are loaded.

This table explains the fields in the session:

Fields	Description
Group	**Selection Range**
Package	Use this field to set a range of packages to load.
Module	Use this field to set a range of modules to load.
Table	Use this field to set a range of tables to load.
Group	**Progress**
Tables to Load	Displays the number of tables to load.
Loaded	Displays the number of tables loaded.
Percentage Complete	Displays the percentage of the tables completed for loading.
# Processes	Displays the number of processes used for dumping.
Grid	
Process Number	Displays the Engine process number.

Table	Displays the table and description of the table which are currently loading in the engine process.
Company No.	Displays the company number.
Status	Displays the loading status. The possible loading statuses are: • **Started** – Engine process is started. **Loading** – Engine process is loading data into a table. • **Cancelled** – Engine process is cancelled. • **Completed** – Engine process is completed. There are no more tables to load.

Command	Description
Start Loading	Starts the loading process for the specified loading processes.
Stop Loading	Stops the loading processes.
Add Process	Adds a process for loading.
Remove Process	Removes a process for loading.

Loading Engines/processes:

The initiated loading engines are invoked in the background. The loading engines create control files to provide the input to the native Database Loader. For Oracle these control files have the *.ctl extension and for Mssql the control files have the *.fmt extension. The control files hold information how the native loaders must be started and also contain the field format of the LN Table. The Field format in the control files must match with the dumped data file. The control files are created in the “Data Files Directory” as specified in the “Data Load Parameters”.

The loading engines also create files to invoke the native database loaders. The Database loaders use the control file and data files as input. Depending on the database any error occurred in the native database loader is logged in the *.log file and if possible the rejected record is stored in a *.bad file.

The *.log files contains the result of the loading like, loading time, loaded records, and rejected records. These files are also stored in the “Data Files Directory” as specified in the “Data Load Parameters”.

After all the tables in range are loaded the created *.log files and *bad files must be analyzed. To analyze the log files, the “Log Files” (bcrun8588m000) session can be used. The reported errors should be resolved before continuing with the post migration activities.

Log Files (bcrun8588m000)

Use this session to analyze the loading files generated by the native data base loaders. The session scans the data dump directory as specified in the Data Load Parameters for data dump files. If a data dump file is present and the data load for this table is initiated there is a loading log file for the table present. This file contains information about the load status of the table like:

- Loaded data file
- Load option used
- Expected fields for the table and the field type
- Number records loaded and or rejected
- Start time and end Time of loading the table
- Elapsed time taken for the load of the table.
- Detected loading errors (depending on the database)

In case a load error for the table is present this must be analyzed and resolved before continuing with the post migration activities.

From this session the log files and rejected (*.bad) files, can be opened and viewed. The user can specify which local editor program must be used to view the files.

This table explains the fields in the session:

Fields	Description
Company	Displays the current company number.
Client Program	Use this field to specify a local client program to view the log files. The specified setting applies to all files.
Table	Displays the table code for which a data file is present in the data dump directory as specified in the Data Load Parameters.
Log File	Displays the location and name of the log file for the table.
Error?	This field shows if there is an error detected in the log file or not. If an error is present, the field is marked else the field is not marked. All log files with errors must be evaluated and the issue must be resolved.
Missing Log?	This field indicates if for a log file for the table was present (checked) or not (not checked). It can indicate an error while processing the load of the table or the data dump file is empty.
Command	Description
Open	Use this command to open and view the log file in the specified local editor.

View Bad file	Use this command to open the *.bad file with rejected records for the table.
Show error	Use this command to open “Loading Errors” (bccnv5500m001) session for the specified table and company. While starting the option scans the log file for an available loading error.
Scan for Log files	This command starts the retrieval of the log files information from the Data dump directory as specified in the Data Load Parameters. The retrieved log file information can be viewed in the current session.

Loading Errors (bccnv5500m001)

This is a similar session as “Loading Errors” (bccnv5500m000) for checking the loading per company. The “Loading Errors” (bccnv5500m001) session shows the loading errors per company and table.

For details on the session and the fields see “Loading Errors” (bccnv5500m000).

Load Utilities

The session listed under the menu Load Utilities are all kinds of additional session related to loading the data in the target LN environment.

Session	Session Code
Generate License Key	bcrun0000m0000
Conversion Runs (*)	bcgct1550m000
Data Validation Analyzer	Submenu
Query verification	bccnv9190m000
Customer Critical Data Verification Sessions	bccnv9191m000
Export/Import of Sessions/Query Data	bccnv9290m000

(*) This LN Engine session is described elsewhere in this chapter.

Generate License Key (bcrun0000m0000)

Use this session to generate a license key to use the BaanIVc4 dump engine in a two server scenario without client/server connection. In all other cases it is not required to use this session.

To generate the license key just start session and click the generate button.

Two server scenario without client/server connection:
When no valid validation key is present at the BaanIVc4 Engine during the data dump the “Enter Validation Key” (bcrun0000m000) session is displayed.

Copy the generated license key number at the LN Engine and paste this key in the “Enter Validation key” (bcrun0000m000) session at the BaanIVc4 Engine and click “OK”.

You must only provide the license key once as long as you do not change the dump parameters settings the key stays active.

This table explains the fields in the session:

Fields	Description
License Key	This field displays the generated license key. The user can copy the key to the BaanIVc4 engine.
Command	Description
Generate	Use this command to generate the license key.

Query Verification (bccnv9190m000)

Use this session to create validation queries for the data at the dump side (source queries) and the data at the target side (target queries). After the queries are created the session can run the queries and the output is presented on a report. For running the source queries a connection to the source engine is required.

With the query verification the user is able to create special verification checks to assure the migration process ran correctly.

In case of multiple target companies to load you can copy created queries from another target company into the current company by using the “Copy Query Data from Another Company” option from the Specific menu.

When a query identification is created and saved the user can double click on the records to enter the detail session of the query verification. In the detail session the source and target queries can be maintained.

Note: In a “Two server without client/server connection” migration scenario it is not possible to run source queries but only target queries.

This table explains the fields of the overview session:

Fields	Description
Query ID	The identification of the query. The query Id is automatically generated and follows a sequential order. But it can be adjusted manually.
Description	The description of the query identification.
Source Query	Display if the query is a source query or not. If the query ID contains a source query the check box for this field is marked else it is not marked.
Target Query	Display if the query is a target query or not. If the query ID contains a target query the check box for this field is marked else it is not marked.
Command	Description
Copy Query Data from another Company	Use this command to copy the available queries from another company to the current company. The command starts session “Copy Query Data from Another Company” (bccnv9290s002)

This table explains the fields of the details session:

Fields	Description
Query ID	The identification of the query. The query Id is automatically generated and follows a sequential order. But it can be adjusted manually.
Description	The description of the query identification.
Source Query	In this text box the user can define the source query. To give an example: select count(*) from tcmcs002 This query returns the number of records in the currency (tcmcs002) table at the source side.
Target Query	In this text box the user can define the target query. For example: select count(*) from tcmcs002 This query returns the number of records in the currency (tcmcs002) table at the target side.
Command	Description
Execute (source)	Use this command to run the source query in the source environment. The output of the result is shown as a report on the

	target side. The user can define a device to print the source query report. Note:This option cannot be used in a “Two server without client/server connection” scenario.
Execute (target)	Use this command to run the target query in the target environment. The output of the result is shown as a report on the target side. The user can define a device to print the source query report.

Copy Query Data from Another Company (bccnv9290s002)

Use this session to copy the verification queries from another company to the current company.

This table explains the fields of the session:

Fields	Description
Copy Data From Company	Use this field to specify another company from which the verification queries must be copied.
Command	**Description**
Continue	Use this command to copy all available Verification queries from another target company to the current company.

Customer Critical Data Verification Sessions (bccnv9191m000)

Use this session to run critical data validation sessions in the source environment and the corresponding sessions in the target environment. You can view the source and target sessions simultaneously.

Within the session you can records with an unique sequence number and link target to source sessions or only register a target data verification session.

Via the “execute” command in the specific menu you can start the listed session per record.

Via the “Copy Session Data from another Company” in the specific menu you can import Customer Critical Data Verification Sessions data from another target company to the current company.

Note:
Viewing source environment sessions is not possible in a “Two server without client/server connection” scenario.

This table explains the fields of the session:

Fields	Description
Sequence Number	This field unique identifies the record. The user can enter a sequence number upon record creation or it is generated.
Description	Use this field to add a description to the Sequence Number.
Source Session	Use this field to add a source session code. There is no check if the source code is valid.
Target Session	Use this field to add a Target session code. The session checks if the entered target session code is an existing target session. If the session code is not valid a message is displayed.
Command	Description
Execute	Use this command to invoke the source and target sessions for comparison.
Copy Session Data from another Company	Use this command to copy the available critical data validation sessions from another company to the current company. The command starts session “Copy Session Data from Another Company” (bccnv9291s002)

Copy Session Data from Another Company (bccnv9291s002)

Use the session to copy, the listed verification sessions, from another target company to the current company.

This table explains the fields of the session:

Fields	Description
Copy Data From Company	Use this field to specify another company from which the verification sessions must be copied.
Command	Description
Continue	Use this command to copy all available Verification sessions from another target company to the current company.

Export/Import of Sessions/Query Data (bccnv9290m000)

Use this session to export or import the “Customer Critical Data Verification Sessions” (bccnv9191m000) data and/or the Query Verification (bccnv9190m000) data to/from an ASCII file.

The exported file called “VerificationData.xml” is placed/loaded from a location specified by the user.

This table shows the fields and commands of the session:

Fields	Description
Action	Use this field to select the action for the session. The options are: • Export: Select this option to export data to an ASCII file. • Import: Select this option to import the data from the ASCII file into the current company.
Queries	If this check box is selected the available verification queries are imported or exported.
Sessions	If this check box is selected the available verification sessions are imported or exported
Path of Data Files	Use this field to specify the location of the “VerificationData.xml” file
Command	**Description**
Continue	Use this command to start the export to the file or to start the import of the verification data from the file into the current company

Chapter 7 Special Migration Features

This chapter describes some special migration features and guides you through the special requirements that can occur during a data conversion project.

To migrate Text Tables (tttxt001, ttxt002, and tttxt010)

During the migration of the text tables, pay attention to these points:

- You must not maintain any conversion rules for the tttxt010 table in the DCA Studio. By default, the object that dumps the data in this table to the ASCII file is shipped with the source Engine.
- Ensure that you maintain the rules for tttxt001 and tttxt002 tables in the profile of DCA Studio, for which you perform the conversion build.
- When you define a batch in LN Engine, ensure that the batch contains tttxt001, tttxt002, and tttxt010 text tables. During loading, the loader processes the data files in this specified order. The loader for table tttxt010 waits until table tttxt001 and table tttxt002 are loaded. If these tables are not specified, the loader for tttxt010 waits indefinitely.

Split mapping objects using History/multiple dumping process per table

Use this feature if the table contains a lot of (static) history data and processing this table would take too much processing time during a final Go-Live Migration. Using the “History/multiple dumping process per table” feature you can reduce the system downtime during the live migration. First migrate the static data of a migration object up to a specific date at a convenient moment and during the final Go-Live migration only run the migration object from that specific date up to current date. This is splitting the migration mapping object in 2 objects/ranges.

Or use this feature if the source table contains too much data and you want to split the mapping object over multiple dumping processes. To lower the total execution time of the mapping object it can be split into multiple objects where each object migrates a specified range of the main source table and can be carried out in another dumping process.

For example, if the migration of tcibd001 table would take four hours to run in one dump engine process. If the total available engines were four than one engine would process for four hours and three engines would wait (not utilized). By splitting the mapping object in four objects/ranges each object can run in another dump engine process. So now each dump engine is utilized and handles a specific range of the source table. Now the lead time is one hour the total dump time for table tcibd001 remains the same four times one hour is four hours.

If you can only use 1 dump engine than splitting a mapping object in multiple processes does not lower the total migration dumping time.

In both cases you require the DCA Studio to modify the standard delivered mapping object for a specified table. Complete the processing History data procedure before starting the actual conversion run.

For processing History data

1. Identify the field(s) in the source table which can help you recognize the history data. This can be a date field or a status field.
2. In DCA Studio, go to Master Data - History & Multi-process.
3. Using the new button add the conversion object required for the process. Define the main query table and select the date or status field required (For more information refers to the Data Conversion Adapter Studio 3.4 - User's Guide (U9093F US).
4. In Studio, go to Conversion Rules Modeler - Conversion Build
5. In the Conversion Build screen use the Settings button. Onto the “XML and Script Info” tab select the “Consider History and MultiProcess Setup” option. In the build details specify the name of the mapping object (defined in step 2).
6. Run the conversion build. The conversion build creates a specific migration mapping object and a splitinfo file is created in your source Engine environment, where the dll objects are located (By default $BSE/application/<Your VRC>/obcdll).

Continue in the LN dump Engine

1. Start the “History/Multiple dumping process per Target table” (bcdmp8571m000). From the Specific menu select the “Import from DCA generated file…” option. This option reads the information from the splitinfo file created by the DCA Studio.
2. Select the “Change number of process” option from the Specific menu and enter the value 1 (it is safe to use one process for dumping). By doing this you specified that this specific mapping object is started in 1 dumping engine.
3. Select the “Open Range values per processes” option from the Specific menu. On starting the session, you find two processes. Against the first process enter the “from” value from which you like to select the data for migration. Against the second process enter the “to” value up to which you are migrating. The values must be enclosed in {}. If you are using a date value to split the data, you must enter the corresponding long number of the date
4. During the dumping process this mapping object exports (dumps) the records falling into the specified range.

5 Alter the range for the final Go-Live migration to migrate the remainder of the data.

6 Load the data for this table in append mode. Another option is to first create a sequential dump of the LN table then load the table (truncate mode). Then reload the data from the sequential dump.

For splitting the mapping object in multiple ranges

1 Identify the field(s) in the source table which can help you specify ranges for the mapping object.

2 In Studio, go to Master Data - History & Multi-process.

3 Using the new button add the conversion object required for the process. Define the main query table and select fields required. Most of the time it is a part of the main index of the source table. For more information, see the Data Conversion Adapter Studio 3.4 - User's Guide (U9093F US.

4 In Studio, go to Conversion Rules Modeler - Conversion Build

5 In the Conversion Build screen use the Settings button. Onto the “XML and Script Info” tab select the “Consider History and MultiProcess Setup” option. In the build details specify the name of the mapping object (defined in step 2).

Note: By selecting the Multiple process option you cannot also select the 2GB option to set a maximum of data file size (Unix environment). If 2GB limitation is required and split process is required as well the user must specify enough ranges so each range does not exceed the 2GB dump file limit.

6 Run the conversion build. The conversion build creates a specific migration mapping object and a splitinfo file is created in your source Engine environment, where the dll objects are located (By default $BSE/application/<Your VRC>/obcdll).

Continue in the dump Engine

7 Start the “History/Multiple dumping process per target table” (bcdmp8571m000). From the Specific menu select the “Import from DCA generated file…” option. This option reads the information from the splitinfo file created by the DCA Studio.

8 Select the “Change number of process” option from the Specific menu and enter a value > 1. You can even choose a value higher than the maximum defined dump engines in the parameters (bcdmp000) table. By this you just define the amount of ranges and each range is presented to an available dump engine process.

Note: If you are dumping on a Unix LN environment and you are bound by the 2GB file size restriction than each specified range must result in a data file which does not exceeds this limit. If this is not the case extend the amount of processes or manually alter the ranges.

9 Select the “Open Range values per processes” option from the Specific menu. Within the session choose the “Generate Range Values per Process” option. This creates equal ranges based upon the amount of processes and the selection fields. If needed you can manually alter the specified ranges.

10 During the dumping process the mapping object exports (dumps) the records falling into the specified range and another defined range can be presented to another available engine.

11 During the load process the LN Engine evaluates the amount of available data files and loads the data in the de LN table.

Recoding

This feature enables you to recode master data, as a part of the data migration to Infor LN. During data migration, customers can choose to standardize their master data codes, for example, address codes, warehouse codes, and so on. During the migration the old source code is replaced by the new defined code.

Within the DCA Studio you must set up a conversion code, See also *Data Conversion Adapter 3.4 – Data Conversion Adapter Studio – User Guide (U9093F US)*

Using the DCA Recode Utility (part of the DCA studio installation) you can link domains to a conversion code. The recode utility searches through the specified logic pack and detects were the specified domain is used. On all places were the domain is used the mapping rule is modified and uses the predefined GetCode() function. During the dump process this function retrieves the new defined code based upon the old source code as specified in the Maintain Values by Conversion Codes (bcdmp9191m000) session.

As the recoding functionality can apply to all mapping objects of a Rules Pack, the new value is not only replaced in the master data table but also on all the places where the master data field is present. For instance, the recoding of a warehouse code is not only done in the generic warehouse table but on all places where a warehouse code is used like sales orders, purchase orders, warehouse orders, Production orders, invoicing, and so on.

To use the recoding functionality:

- Define a Conversion code in the DCA Studio
- Create a specific Recode profile in the DCA Studio
- Use the Recode Utility of the DCA Studio to apply the recoding code to the complete logic rules and create the changed mapping objects in the new created recode profile.
- Define the recode code in Conversion Codes (bcdmp9190m000) session in the source engine.
- Define the old / new values in the Maintain Values by Conversion Codes (bcdmp9191m000) in the source engine.
- Dump and load the data.

The conversion codes can be defined in the Conversion Codes (bcdmp9190m000) session. The defined conversion codes must be the same as defined in the DCA Studio.

Therse predefined conversion codes cannot be altered:

- "*COMP"* code is reserved for cross company migration. For example, to migrate data from Infor B50c company 100 to Infor LN company 200.
- *"UNIC"* code is reserved for the Unicode correction feature. This feature can replace illegal String characters in LN Strings. For example, when migrating to a Unicode environment it is not allowed to use Characters greater then character number 127. So the charter "Ü" is not allowed and can be replaces by "UE".

Note: As there is no standard LN to LN logic pack also the recoding company is not standard available.

The Maintain Values by Conversion Codes (bcdmp991 table) can be filled in these ways:

- Manually add the old value and the new value
- Retrieve the Old value list using the master data table at the source and fill the new values manually.
- Create a function to generate the new value based upon the old value.

It is not required to specify all the source codes and a new value for it, but only the codes you want to be replaced. If for a conversion code the old value is not specified, the function returns the unchanged master data value (old value). If for a conversion code the old value is specified but the new value is empty, the function returns also the unchanged master data value (old value).

Unicode Correction

The LN source environment can be a non-unicode environment where the standard installation of LN10.5 and higher are Unicode installations. This change can lead to migration issues when in the source characters are used which are not allowed in the target environment. During a full migration we do not want to scan all strings for illegal characters as this is bad for performance. First a migration to the target environment is done without a correction. At the target environment the migrated data is checked for which table fields illegal characters are used. For the list of fields, the migration logic pack must be adjusted so the migrated string is checked for not allowed characters. To make the modification the recoding utility can be used with the recoding code "UNIC". If this recoding code is used and the correct table field is entered the utility makes the required changes in the customized logic pack automatically. After rebuilding the Logic Pack the Unicode correction function is used to correct the string fields during the dumping process. The user can define in the "UNIC" recoding values which characters must be checked and how these characters must be replaced.

UTC conversion

Until LN10.5 the maximum allowed UTC date was 19-janurary-2038. From LN10.5 you can have UTC dates up to 30-december-9999. When the Target environment can handle UTC dates beyond

2038 the migration dump parameter “Is Target UTC40 format?” (bcgct000.ituf) must be set to “Yes” else it must be set to “No”.

During the migration the UTC values from the source are migrated as is to the target. In case the Migration logic must fill an UTC date, for instance when there is no equivalent source field, and the field must filled with the maximum allowed UTC value than based upon the “Is Target UTC40 format?” (bcgct000.ituf) parameter, the maximum UTC value is set.

Note: If in the source a maximum UTC value is set and in the target an equivalent field is present than the field is migrated as is and the maximum UTC value is not changed even if the “Is Target UTC40 format?” (bcgct000.ituf) parameter is set to “Yes”.

Source field is a Date type, target field is an UTC type:
The source field is converted to an UTC value. Date types can allow date beyond 2038.
If the “Is Target UTC40 format?” (bcgct000.ituf) parameter is set to “No” then date is converted to an UTC value. All dates after 19-janurary-2038 is set to: 19-janurary-2038.
If the “Is Target UTC40 format?” (bcgct000.ituf) parameter is set to “Yes” then date is converted to an UTC value as is.

Appendix A DCA3.x Differences

This section gives a short over view of the enhanced features and fixes compared with the previous DCA versions. If your previous version is before DCA31 see the changes in DCA3.3 and DCA3.4 section. If your previous version is DCA32 see the changes in the DCA3.4 section.

DCA3.2

Enhancements

Synchronized dump and Load (LNUI):
The LN engine is now fully LNUI enabled. As from the target environment it is no longer allowed to start another environment process the synchronized dump and load process is changed. Dumping data and loading data at the same time is still possible but the user must start the dump engine and then start the load engine.

New LNUI sessions:

Some of the existing sessions are not working in a LNUI environment and therefore the below new sessions are created to replace the old sessions.

- New Loading progress session for LNUI: Loading Daemon (bcldr9100m000)
- New Dumping progress session for LNUI: Dumping Daemon (bcdmp9100m000)

UTC40:
The dump engine can now also convert dates in UTC for dates beyond 19-january-2038.

Database support:
The DCA3.3 LN dump engine now also can handle the migration to an EDB-Postgress as a target database.

String Conversion functions:
New conversion function added in bcrunconvdll for handling string conversions. Each string is now checked if the string is filled. If the migrated string is empty, the string is migrated as a single space

string for Mssql and Oracle target databases. For EDB as a target database the migrated field is an empty string in this case. For the other databases the string field gets the full string length. For Mssql, Oracle and EDB target databases the size of the data dump files is therefore reduced.

Fixes

UTC conversion improvements:
The UTC() function in bcrunconvdll is redesigned for the implementation of the UTC40 change.

Error during simulation or debug a mapping Object:
Logging error during simulation (or debug) mode resulted in max.retry errors.

DCA3.3

Enhancements

Log Files:

When Multiple data files for the same table must be loaded the errors in the multiple logfiles for this table are now evaluated and the results are stored in bcrun888.

Export Table Information for DCA:

New session added in the LN tools layer called: Export Table Information for DCA (ttmig1200m000). This session is available from porting set 9.2b / 10.6.1
With the session is it now possible to get the standard Data Dictionary information plus additional information from the actual LN database via an xml file. The new xml file per table holds information about:

- The actual field sequence in the LN database.
- The runtime added fields for a table which are not present in the Data dictionary files. Like workflow fields and/or fields for Multi Language enabling.
- The table description and the table field descriptions

The Data dictionary information together with the Additional (xml) information can be used by the DCA34 Studio to create Target LN applications.

Fixes

Mandatory defines where not present.

Added Statics to UTC$ function to improve performance.

Now possible to start Verify in debug mode from DCA34 Studio.

Cyrillic MB strings are no longer cutoff to a length of 60 characters.

DCA3.4

Enhancements

DSN Information:

DSN Information is detected from XML Files and prompted in the Application.

Detection tool started as Admin:

Now the Tool detects if it has been started with an user with Administrator rights.

This to preserve (unreported) problems in a later stage.

Others:

Some cosmetic improvements.

Stability Improvement.

Fixes

‘Run Time Error 53’ problem solved.