QAD Enterprise Applications
Enterprise Edition

Conversion Guide
Progress Database

78-0886A
QAD Enterprise Applications 2010.1
Enterprise Edition
September 2010

ProgressConversion_CG_v2010_1EE.pdf/biw/biw

QAD Inc.
100 Innovation Place
Santa Barbara, California 93108
Phone (805) 566-6000
http://www.qad.com

Contents

Chapter 1 Conversion Overview 1

Introduction 2
QAD Enterprise Edition Conversions 2
Conversion Path Diagram 3
Prerequisite Skills 3

Chapter 2 Pre-conversion 5

Introduction 6
Pre-conversion Overview 6
Install Pre-conversion Code 6
Upgrades from eB2.1 or Later 7
Prepare Source Data 7
Table Extension Domain Utilities (eB2.1 and later only) 7
Archive/Delete Transaction Data (optional) 8
Global Tax Management Conversion (pre-eB) 9
Run Pre-conversion Utilities 9
Employee Entity Utility (uxempent.p) 10
End User Contact Utility (uxendupd.p) 10
Control Account Utility (uxctrl.p) 11
GL Account/Project Range Utility (uxglproj.p) if pre-eB 11
Orphaned Address Utility (uxaddrfix.p) 12
Credit Terms Utility (uxcrterms.p) 12
Data Preparation Report (gpdatarp.p) 13
Country Code Utility (uxctryup.p) 16
Tax ID Utility (uxtaxid.p) 17
Code Search and Replace Reports (eB2.1 and later only) 17
GL Account Type Utility (uxglacup.p) – Report Mode 19
Pre-conversion Integrity Report (gpinckrp.p) 23
Set Conversion Parameters 24
Conversion Parameters Utility (utfinpar.p) 24
Pre-conversion Completion 28
Process Pending Transactions 28
Close the Production Database to Users 28
Rerun Pre-conversion Integrity Report (gpinckrp.p) 29

Rerun GL Account Type Utility in Update Mode (uxglacup.p) 29
Run the Converted GL Account Definition Report 29
Standard Period Closing Reports 29
Rerun (Finalize) Data Preparation Report 30

Chapter 3 Conversion Execution 31

Introduction 32
Conversion Execution Overview 32
Prepare Environment 32
Prepare Source Databases 32
Install and Configure Software 32
Install QDT 32
Install QAD Enterprise Edition 33
Conversion Setup 33
Enable Large Files 33
Enable Conversions 34
Convert QAD Enterprise Edition from Previous Release 35
Create Live Main Database 35
Character Client Code: Generate R-Code 37
Update UI Configuration 38
Convert QAD Enterprise Edition Database from Previous Version . . . 39
Conversion Program Selection 40
Domain Conversion 41
OID Generator Value 43
EDI eCommerce Conversions 44
Execute Conversion 44
Conversion Validation 45
Enter License Codes 45
Conversion Execution Troubleshooting 46

Chapter 4 Post-conversion 47

Introduction 48
Overview 48
Post-conversion Utilities 48
Fixed Assets Migration Utility (32.25.2 - facvmt.p) 48
Table Extension Domain Conversions - Part 2 48
Sales Order Balance Update (36.16.23.6 - utcsob.p) 49
Document Credit Terms Update (36.25.83- uxdoccrterms.p) 49
Data Validation 49
System Consistency Check (36.16.23.1 - utsyscon.p) 49
Financials Consistency Check (36.16.23.2 - utfincon.p) 50
Operational Account Structure Validation (36.9.20 - uxacval.p) 51

Post Conversion Integrity Check (36.16.23.3 - acinckrp.p) 52
Post-conversion Reconciliation Reports 54
Post-conversion Reports 55
Process Flow and Static Data Validation 56
Mandatory Post-conversion Setup 56
Structured Reports 56
Invoice Status Codes (36.1.11) 56
Daybooks 57
Security 57
Tax Periods 57
Reporting Periods 58
Profiles 58
Configure Daemons 58
Optional Post-conversion Setup 58
Accounting Layers 58
Cash Groups 58
Report Structures 58
Taxes 59
Supplementary Analysis Fields 59
Customer Credit Checking 59
Customer/Supplier Payment Statuses 59
Customer/Supplier Control Accounts 60
Additional Profiles 60
Chart of Account (COA) Mask 60

Chapter 5 Upgrading QAD Enterprise Edition 61

Overview 62
Prepare Environment 62
Prepare Source Databases 62
Back up Environment 62
Install and Configure Software 63
Migrating QDT to Progress 10.2A 63
Install QDT 64
Install QAD Enterprise Edition 64
Upgrade Setup 67
Upgrade QAD Enterprise Edition to a New Release 67
Upgrade Live Main Database 68
Character Client Code: Generate R-Code 69
Update UI Configuration 70
Execute Upgrade 71
Upgrade Validation 72

Appendix A GTM Conversions 73

GTM Conversions Summary 74
Pre-conversion Planning 74
Post-conversion Procedures 75
Converting VAT Taxes to GTM 75
Implementing GTM 76
Converting Master Records 80
Converting Transaction Records 83
Converting US Taxes to GTM 88
Implementing GTM 89
Converting Master Records 96
Converting Transaction Records 99
Converting to GTM From No Taxes 104
USA to GTM Setup 104
USA to GTM Masters 105
Converting Canadian Taxes to GTM 105
Implementing GTM 106
Converting Master Records 114
Converting Transaction Records 117

Appendix B Running the Fixed Assets Migration Utility 123
Running the Fixed Assets Migration Utility 124
Buttons 124
Setting Migration Defaults 125
Mapping Legacy Data 126
Conversion Methods 126
Converting Books 128
Converting Locations 129
Converting Classes 130
Migration Reporting 131

Appendix C Converted Data 133
Overview 134
Layers 134
Business Relations 134
Shared Sets 134
Entities 135
Profiles 135
Accounts 135
Analysis Type and Analysis Limitations 135
COA Mask 136
Conversion Accounts 137
Project Status Codes 139
Project Groups 139

Security 140
Users 140
User Roles 140
Daybooks 140
Generalized Codes 140
Deleted Generalized Codes 140
New Generalized Codes 141
SAF Codes 141
Exchange Rates 141
Supplier Types 141
Purchase Types 142
Customer Types 142
Credit Terms 142
Credit Ratings 142
Invoice Status Codes 143
Payment Status Codes 143
Customer Payment Status Codes 143
Supplier Payment Status Codes 144
Supplier Bank Data 145
Payment Formats 146
Non-European Accounting 146
European Accounting 146
Consolidation 146
Voucher Detail Records 147
Pre-conversion 147
Conversion 147
Unconfirmed Supplier Vouchers 147

Appendix D Configuring Access to the Progress Editor 149
Introduction 150
Setup for Progress Editor Access 150

Appendix E Conversion Troubleshooting 153
Introduction 154
Conversions Not Enabled in QDT 154
Errors in Data Preparation Report 154
reindex.log Errors 154
qdtadmin.log Errors 155
Invalid Characters in Database Field 155
Number of Characters in Database Field Exceeds Limit 155
Role Name Contains Unsupported Characters 155
Pay Format Directory Not Found 155
Progress Errors During Conversion Execution 156

Default Daybook Codes Not Found 156
Could Not Start Financial Session 156

Appendix F Log Files. 157
Introduction 158
Log File Naming Conventions 159
Reviewing Log Files After Conversion 159

Appendix G Snapshots 161
Introduction 162
Pause Points - convprogpauselist.ini 162
Backup and Restoring a Snapshot 163
Creating a Snapshot 164
Restoring a Snapshot 164

Index. 165

Chapter 1

Conversion Overview

This chapter describes the requirements and process for upgrading a Progress database to QAD Enterprise Edition.

Introduction 2

QAD Enterprise Edition Conversions 2

Conversion Path Diagram 3

Prerequisite Skills 3

Introduction

This chapter provides an overview of the conversion and upgrade processes and describes the skills required to perform a database conversion.

The QAD Deployment Toolkit (QDT) Convert QAD EE from Previous Release option provides a means to convert a pre-QAD Enterprise Edition database to QAD Enterprise Edition. If you already have Enterprise Edition installed, you should not use this feature. See “Convert QAD Enterprise Edition from Previous Release” on page 35 for a detailed overview of this process.

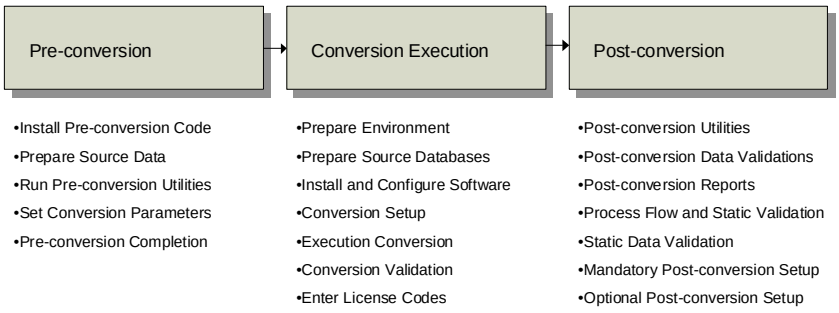
The QDT Upgrade QAD EE to New Release option allows you to upgrade an existing QAD Enterprise Edition installation to the new Enterprise Edition release. Do not use this feature if you do not have an existing Enterprise Edition installation. Refer to Chapter 5, “Upgrading QAD Enterprise Edition,” on page 61 for a detailed description of this process.

QAD Enterprise Edition Conversions

The QAD Enterprise Edition conversion process covers the steps for upgrading the database of an earlier QAD ERP system version to QAD Enterprise Edition. Figure 1.1 summarizes this process.

Note This overview is for the convert option only. It does not apply to the upgrade option.

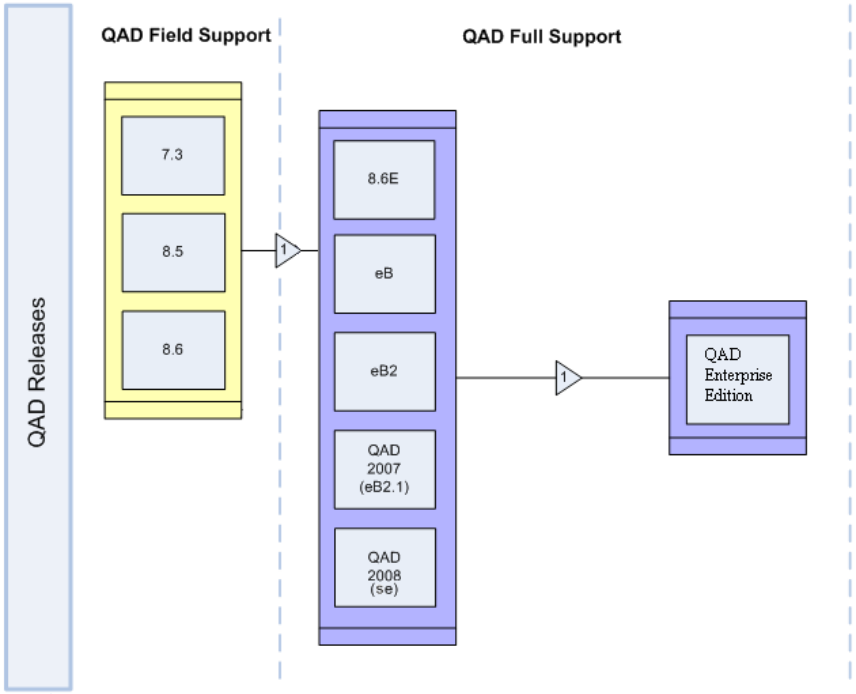
Fig. 1.1
Conversion Summary



Conversion Path Diagram

The following diagram illustrates the QAD Enterprise Edition conversion path.

Fig. 1.2
QAD Enterprise Edition Conversion Path



As depicted in Figure 1.2, direct conversions to QAD Enterprise Edition are supported for MFG/PRO versions 8.6E and above only.

If the database to convert is on a version before 8.6E, it must be updated to a supported version (8.6E and above) before conversion to QAD Enterprise Edition.

Prerequisite Skills

The procedures presented in this manual should only be attempted by system administrators and other qualified personnel with the following training and experience:

- QAD Enterprise Edition installation and administration training
- A working knowledge of QAD Enterprise Edition products
- System maintenance expertise (for example, adding menu items and custom programs)
- Starting and stopping AdminServer, AppServers, and NameServers
- Basic Progress database administration skills (`probkup`, `prorest`, dump and load)
- Familiarity with Linux/HP-UX and command line navigation

QAD strongly recommends that all persons involved in the conversion have the following background:

- Progress database tuning parameters
- Basic Progress coding skills for debugging - simple Progress queries

- Basic Tomcat administration skills (starting, stopping, installing, and changing ports and permissions)
- Familiarity with X Window and VNC server

Chapter 2

Pre-conversion

This chapter describes the preparation of a database for upgrade to QAD Enterprise Edition.

Introduction 6

Pre-conversion Overview 6

Install Pre-conversion Code 6

Prepare Source Data 7

Run Pre-conversion Utilities 9

Set Conversion Parameters 24

Pre-conversion Completion 28

Introduction

This chapter describes the pre-conversion steps required to prepare a database for upgrade to QAD Enterprise Edition (QAD EE). It covers pre-conversion code installation, data preparation, conversion preparation, and pre-conversion finalization.

Note This chapter only applies when converting from versions before Enterprise Edition. It is not required when upgrading an existing Enterprise Edition installation.

Pre-conversion Overview

The pre-conversion process prepares the source database for conversion by correcting the following types of issues:

- Data that is missing from the old Financials, but required in the QAD Enterprise Edition Financials
- Data that exists in the old Financials, but is not allowed in the same state in QAD Enterprise Edition Financials
- Data that is invalid in both the old and new Financials

The Pre-conversion process consists of the following major tasks:

- Install Pre-conversion Code
- Prepare Source Data
- Run Pre-conversion Utilities
- Set Conversion Parameters
- Pre-conversion Completion

The database is prepared using pre-conversion utilities and reports to highlight items that require changing or updating for the conversion. These run in a character environment only. All text is in English and no translations are available for these reports and utilities.

The pre-conversion utilities and reports do not appear on any menus. They are executed by typing the program name at any menu command line. The programs do not support CIM and cannot run in batch mode.

Install Pre-conversion Code

The QDT media contains all of the pre-conversion reports and utilities needed to prepare a database for conversion. The pre-conversion source code is version independent and supports MFG/PRO versions 8.6E through QAD Standard Edition and Progress versions 8.3e to the latest Progress version. As a result, attempting to compile the pre-conversion code will cause Progress errors due to field and table references that do not exist in all versions.

Therefore, QAD recommends running these programs as uncompiled source code.

To install the pre-conversion code, use the following steps:

1. The pre-conversion code is located in the `preconvrep/preconvrep.zip` file. Locate this file in the QDT media.
2. Unzip the pre-conversion code into its own directory (QAD suggests `preconvrep`).
3. Add the following the directories to the beginning of the propath:
   - *<preconvrep install dir>*
   - *<preconvrep install dir>*`/us`

Each program can then be run by typing the name at the character menu command line.

Upgrades from eB2.1 or Later

If upgrading from eB2.1 or later, the QDT media contains data files (`*.d`) that also must be loaded into the qadadm and main databases for both the Domain Table Extension utilities and Search and Replace reports. The table below indicates which files should be loaded into which database.

Table 2.1
Database File Inventory

Database	Files in <preconvrep dir>
qaddb	`cd_det.d`, `lpmd_det.d`, `mnd_det.d`, `mnt_det.d`, `msg_mstr.d`
qadadm	`lbld_det.d`, `lbl_mstr.d`, `pgmi_mst.d`

If you are running eB2.1 SP6 (QAD 2007.1) or later, you will receive errors when loading some of these files because the system already contains this data. In such cases, ignore the errors.

In addition, the following programs must be compiled and put into the `us/ut` subdirectory under the directory where the pre-conversion code was installed:

- `utqtabs.p`
- `utqtabsa.p`
- `utqtplsd.p`
- `utqtpsda`
- `utsarrp1.p`
- `utsarrp2.p`
- `utsarrp3.p`

Note If you are running eB2.1 SP6 (QAD2007.1) or later, you already have `utqtabs.p`, `utqtabsa.p`, `utqtplsd.p`, and `utqtpsda` installed and do not need to compile these files.

Prepare Source Data

Table Extension Domain Utilities (eB2.1 and later only)

For customers upgrading from eB2.1 and later, selected table extension (qtbl_ext) records must be updated with domain information before running the conversion. This is done using the following utilities:

- `utqtabs.p` – Covers supplier lots on shippers

- `utqtplsd.p` – Covers Expense Due account in Sales Account Maintenance (1.2.17, `ppplsmt.p`)

These utilities must be run in update mode once from any domain even if these features are not being used (all domains are updated in a single execution of each utility). They provide a simulation mode for previewing updates.

The effect of these utilities on future transaction processing is limited to the extent these features are used. They ensure data from the correct domain is used.

Archive/Delete Transaction Data (optional)

You can reduce the time required to complete the process by archiving and deleting as many records as possible before conversion. This is optional, but recommended.

You can use the following functions to archive and delete data:

- GL Transaction Consolidation (25.13.11, `glcons.p`)

  Transaction consolidation should include all accounts and be done for single accounting periods per consolidation.

  Warning Do this by period, not account, and preferably by month as opposed to by year. If done by year, any given account will have an opening balance but zero period amount for any period other than the period for the consolidation effective date.
- Invoice History Delete/Archive (7.13.23, `soivup.p`)
- Uninvoiced Receipt Delete/Archive * (28.22, `aprcup.p`)
- Uninvoiced Logistics Charge Delete/Archive (28.21, `aplaup.p`)
- Closed PO Receipt Delete/Archive * (5.22, `porcup.p`)
- Closed PO Delete/Archive * (5.23, `popoup.p`)
- Supplier Schedule Delete/Archive * (5.5.3.23, `rsdel.p`)
- PO Shipper Delete/Archive * (5.13.23, `rsscdel.p`)
- Customer Schedule Delete/Archive * (7.5.23, `rcdel.p`)
- PRO/PLUS Sequence Schedule Detail Delete (7.5.4.22, `rcsqscdl.p`)
- Shipper Delete/Archive * (7.9.23, `rcsdel.p`)
- Container Delete/Archive * (7.7.23, `rcctdel.p`)
- Expired Quote Delete/Archive (7.12.23, `sqqoup.p`)
- Inventory Transaction Delete/Archive * (3.21.23, `ictrup.p`)
- Zero Balance Delete/Archive (3.23, `icldup.p`)
- Cost Set Delete (30.23, `cscsdel.p`)
- Intrastat Delete/Archive (2.22.23, `iehup.p`)
- Retired Asset Delete/Archive (32.23, `fartup.p`)
- Tag Delete/Archive (3.16.23, `pitdup.p`)
- WIP Lot Trace Delete/Archive (3.22.13.23, `wldel.p`) – PRO/PLUS Module
- Intersite Request Delete/Archive (12.15.23, `dsdmup.p`)
- Closed Intersite Demand Delete/Archive (12.17.23, `dsdoup.p`)

- Product Structure Delete/Archive (13.23, `bmpsdel.p`)
- Routing Delete/Archive (14.13.23, `rwrodel.p`)
- Work Order Delete/Archive (16.23, `wowoup.p`)
- Operation History Delete/Archive (17.23, `sfopup.p`)
- Cumulative Work Order Delete/Archive (18.23.2, `recwoup.p`)
- PCR/PCO Delete/Archive (1.9.15, `ecarcdel.p`) – PCC Module
- Lot Master Delete/Archive (1.22.23, `clltup.p`) – Compliance Module
- Service/Repair Order Delete/Archive (7.23.23, `srsroup.p`)
- Closed Project Delete/Archive (10.23.23, `pjpjup.p`) – PRM Module
- Call/Quote History Delete/Archive (11.1.1.23, `fscaarc.p`)
- Service Request Delete/Archive (11.1.15.23, `fssrarc.p`)
- Field Notification Delete/Archive (11.3.12.23, `fsfnarc.p`)
- Contract Delete/Archive (11.5.13.23, `fssaarc.p`)
- Revenue Delete/Archive (11.5.18.23, `fsdefarc.p`)
- Flow Delete/Archive (17.21.23, `flschup.p`)
- Test Results Delete/Archive (19.22, `mpcaup.p`)
- Quality Order Delete/Archive (19.23, `qcqcup.p`)
- Family Hierarchy Delete/Archive (33.3.23, `spfhup.p`)
- Operations Plan Delete/Archive (33.15.23, `spfpup.p`)
- Simulation Delete/Archive (33.17.23, `spfspup.p`)

* Indicates activities providing the greatest benefit

To archive or delete data, start the desired utility by entering the utility name (or menu number) on the command line. After you have completed any deletion or archiving, skip to “Global Tax Management Conversion (pre-eB)”.

Global Tax Management Conversion (pre-eB)

If you are converting from MFG/PRO eB or later, skip to the next section. If you are converting from a release before MFG/PRO eB, you must convert the tax environment to use Global Tax Management.

Refer to Appendix A, “GTM Conversions,” on page 73 for more information.

Run Pre-conversion Utilities

Many changes to existing static data are required before upgrading to QAD Enterprise Edition. Some of the utilities described in this section provide the means to perform mass updates, while other utilities are required to create underlying data used in the conversion.

All of the utilities and reports listed in this section prompt you to specify a report output directory. Once supplied, this location is saved and used as the user default for each subsequent execution of a utility or report. This value can be changed and the new value is saved as the new output directory for that user (for example, each user can have a separate output directory location).

Note The Data Preparation Report is described in more detail in a later section and is the pre-conversion master report. It highlights all data errors and the recommended utilities to correct them. It also highlights which utilities must be run before beginning a conversion. The first time this report is executed, it may produce a large number of errors. To minimize the number of errors in the Data Preparation Report, QAD recommends running the following utilities before the Data Preparation Report:

- Employee Entity
- End User Contact
- Control Account
- GL Account/Project
- Orphaned Address
- Credit Terms

Employee Entity Utility (uxempent.p)

QAD Enterprise Edition requires every employee (emp_mstr) to have an entity code. This utility reports and updates employees with a user-specified entity code.

- It shows employees missing an entity code and employees already assigned an entity code.
- To see Employees missing entity codes, leave Entity field blank and Update = No.

The utility can be run for a range of employees when multiple entity codes must be assigned.

The utility provides a simulation mode that previews the effect of the update.

1 Run this utility. For eB2.1 and later, this utility must be run for each active domain.

The report name is `uxempent-<dbname>-<domain>-<date>_<time>.prn`.

2 Correct any discrepancies identified in the report.

This utility does not affect future transactions; it updates an unused QAD reserved field in the Employee Master (emp_mstr) table.

End User Contact Utility (uxendupd.p)

QAD Enterprise Edition requires every end user (eu_addr) to have one primary contact with a unique contact name. This utility reports and updates end users who are missing a contact name, have duplicate contact names, or have multiple primary contacts.

The utility provides a simulation mode that previews the effect of the update.

The report name is `uxendupd-<dbname>-<date>_<time>.prn`.

For eB2.1 and later, one execution processes all active domains.

Correct any discrepancies listed in the report. Some of these errors must be manually corrected through End User Maintenance (11.9.1, `adeumt.p`).

This utility may affect future processing by making contact names and primary contact designations different from previous transactions.

Control Account Utility (uxctrl.p)

The conversion routines must know which accounts are used as Accounts Receivable and Accounts Payable control accounts in the AR and AP modules of the pre-conversion installation.

The conversion cannot determine this information programmatically because the routines overlook accounts not referenced to customers, suppliers, or transactions.

The utility prompts you for lists of account codes currently used as control accounts. It is permissible to specify the same account for AR and AP if that is the current practice. However, the GL Account Type Utility will ultimately indicate that one of these accounts must be different after conversion. See the *GL Account Type Utility (uxglacup.p) – Report Mode* on page 19 section for more information.

The report output filename is `uxctrl-<dbname>-<domain>-<date>_<time>.prn`

Run this utility as many times as necessary. Previous answers are displayed and can be removed if desired. For eB2.1 and later, it must be run in each active domain.

The output from the utility is used later by the GL Account Type Utility, Data Preparation Report, and during the conversion.

This utility determines the GL account codes that will be defined as Customer and Supplier Control Accounts within Enterprise Edition. It does not affect the current system or transactions. It only creates qad_wkfl records that store the user's input values.

GL Account/Project Range Utility (uxglproj.p) if pre-eB

The following information only applies if you are an MFG/PRO 8.6E or 9.0 user. If you are not, go to the next section.

Prior to version eB, project codes were not a validated part of the GL account structure. When project codes were in use they could be used with all valid GL account, sub-account and cost center combinations.

The conversion must know which GL accounts are valid for use with project codes in the QAD Enterprise Edition Financials so that it can correctly set the analysis settings of the GL Accounts (that is, Analysis Type and Analysis Limitation).

Note Where a GL account is determined to allow project codes, Chart of Account (COA) Mask records are created allowing projects for use with all valid combinations of sub-account and cost center for that account. Review the COA Mask records after the conversion finishes.

This utility prompts you for GL account code ranges to use with project codes after conversion. Note, however, that not all GL accounts are allowed to use project codes after conversion.

The report output filename is:

`uxglproj-<dbname>-<date>_<time>.prn`.

Run this utility as many times as necessary. Previous results are displayed and can be removed if desired.

This utility does not affect future transactions; it only creates qad_wkfl records storing the user's input values.

Orphaned Address Utility (uxaddrfix.p)

To be converted, all customer, supplier, end user, and remit-to addresses must have a related List Type Master (ls_mstr). Execution of this utility may or may not be necessary. If required, it is noted on the Data Preparation Report.

This utility reports and creates missing List Type Masters (ls_mstr) in the following scenarios:

- A customer exists (with cm_mstr and ad_mstr) but no List Type Master record for “customer.”
- A supplier exists (with vd_mstr and ad_mstr) but no List Type Master record for “supplier.”
- An end user exists (with eu_mstr and ad_mstr) but no List Type Master record for “enduser.”
- A Supplier exists and has a reference to a remit-to address (vd_remit <> ““) which has valid address but no List Type Master record for “remit-to.”
- A site exists with an address but no List Type Master of “company” is found.

The utility provides a simulation mode that previews the effect of the update.

For eB2.1 and later, one execution processes all active domains.

The report output filename is `uxaddrfix-<dbname>-<date>_<time>.prn`.

This utility may affect future transactions by making previously unusable addresses usable.

Credit Terms Utility (uxcrterms.p)

All customers and suppliers must have a valid non-blank credit terms code assigned to them before upgrading to QAD Enterprise Edition. Execution of this utility may or may not be necessary. If required, it is noted on the Data Preparation Report.

A blank credit terms code (that is, ct_code = <blank>) is no longer permissible and is deleted by the conversion. To simulate the scenario of a customer or supplier with no credit terms, a new credit terms code must be created in Credit Terms Maintenance (2.19.1, `adcrtmt.p`) with 0 Disc Days and 0% Disc Pct.

Any credit terms code assigned to customers or suppliers must exist in Credit Terms Master (ct_mstr).

The utility must be run separately for customers and suppliers. The utility can be run for a range of customers or suppliers when multiple credit terms codes must be assigned. It reports any customers and suppliers having a blank or invalid credit terms code and updates missing terms with the user-specified code.

To see Customers or Suppliers with invalid Credit Terms assigned, leave the Credit Terms field blank and set Update to No.

The utility provides a simulation mode that previews the effect of the update.

The report output filename is:

`uxcrterms-cust-<dbname>-<domain>-<date>_<time>.prn`

or

`uxcrterms-supp-<dbname>-<domain>-<date>_<time>.prn`.

For eB2.1 and later, the utility must be run for each active domain.

This utility alters future transactions involving the Customers or Suppliers updated with new Credit Terms.

Data Preparation Report (gpdatarp.p)

This report highlights any data issues that must be resolved before conversion such as:

- Data that is missing from the old Financials, but required in the new Financials.
- Data that exists in the old financials, but is not allowed in the same state in the new Financials.
- Data that is invalid in the old and new Financials.

Warning Although this report may not indicate an error, county and state data in eB2.1 and later must be reviewed before conversion to eliminate similar or inconsistent entries. Pre- and post-conversion state and county data must be compared to verify data integrity. Otherwise, the conversion may make state and county data unusable.

This report is the gatekeeper for the conversion. The conversion cannot be run until this report has zero errors.

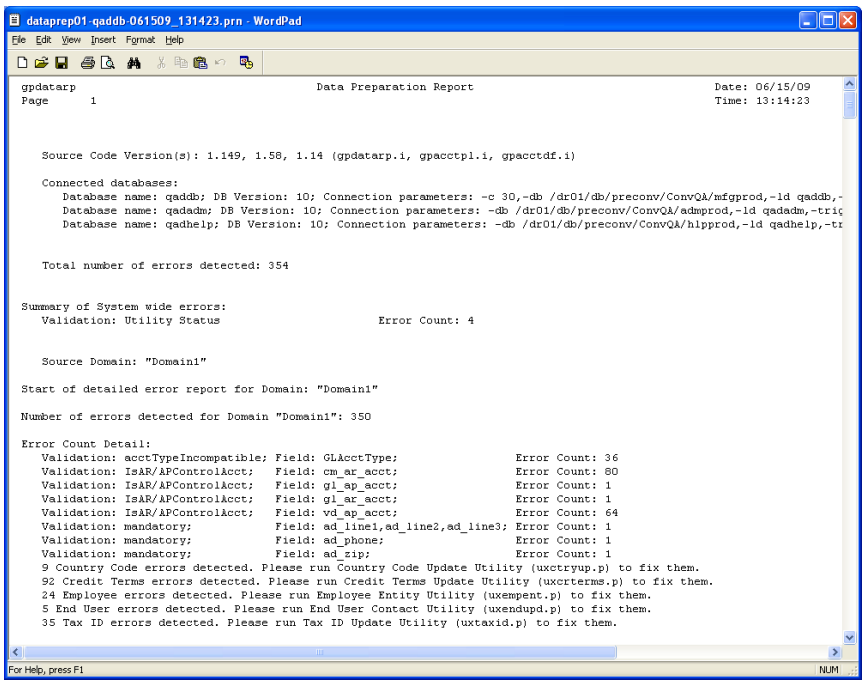
Although the name indicates it is a report, it does create a qad_wkfl record in the database once no errors are found. The conversion looks for this record before it begins.

The report output filename is:

`dataprep#-<dbname>-<date>_<time>.prn`

where # represents a segment number. The report may have multiple segments.

Fig. 2.1
Data Preparation Report



For eB2.1 and later, one execution processes all active domains.

The following new utilities are provided to assist with mass correction of some data errors.

- Country Code, `uxctryup.p`
- Tax ID, `uxtaxid.p`

You must run these utilities if the Data Preparation Report indicates there are errors in these areas. Their use is optional because the data can alternatively be manually corrected through the menu functions listed on the respective utility's reports.

It is very likely the Data Preparation Report will need to be run multiple times before all the errors are resolved.

Running this report does not affect future processing.

Refer to Table 2.2 on page 16 for the expected category for each Enterprise Edition account type.

Account Type Incompatibility Error

Most errors shown on the Data Preparation Report are self-explanatory. One exception is the Account Type Incompatibility error. In the old Financials it is possible to define an account in Account Code Maintenance as an Expense type account, but use that account code in static data for non-expense type purposes (for example, in Product Line Maintenance as an Inventory account).

The QAD Enterprise Edition Financials require that accounts used for specific purposes must have the appropriate account type definition based on the financial statement where they appear. Account codes used in database fields for static data that are associated with the Balance Sheet must be defined as type (ac_type) A(sset) or L(iability); not I(ncome) or E(xpense). Similarly, Account codes used in database fields for static data that are associated with the Income Statement must be defined as type I(ncome) or E(xpense); not A(sset) or L(iability).

For example, the Inventory account code (pl_inv_acct), or the WIP Control account (pl_wip_acct) used in Product Line Maintenance must be defined as an Asset or Liability type account; preferably Asset. (The important thing is that the account code be represented on the correct financial statement.) It is not acceptable to use an account code normally used for Purchases Expensed as Inventory or WIP Control accounts.

There are two ways to resolve Account Type Incompatibility errors.

- If the account code also appears as a conflict on the GL Account Type Utility report, it is possible to correct the account code by running the utility in update mode and specifying an alternate account code to resolve the conflict. The replacement account is applied during conversion and therefore has no affect on transactions occurring before conversion.
- If the account code does not appear as a conflict on the GL Account Type Utility report, the only course of action is to update the field in error (for example, pl_inv_acct) through its maintenance program with an account code having the appropriate account type.

Warning

This affects future transactions once such a change is made. Therefore, it should only be done after closing the database to transaction processing in preparation for conversion.

Do not correct these errors by writing a Progress utility to update the account type on the offending accounts. Doing so causes the Balance Sheet to be out of balance and Retained Earnings to no longer match the amount initially recorded.

Please be aware that changes to correct these errors will result in information being reported differently on financial reports after conversion. Continuing the example from before, if a Purchases Expensed account were used as the Inventory account in Product Line Maintenance, any transactions involving this account appear on the Balance Sheet post-conversion, whereas before they were on the Income Statement. For instance, a PO receipt involving such an account is reflected as Inventory on post-conversion Balance Sheet.

The Account Type Incompatibility errors are a result of the tighter controls used by the QAD Enterprise Edition Financials.

Beginning with QAD 2010.1 Enterprise Edition, Closing accounts are no longer restricted to Asset or Liability type accounts. Any account type (A, L, I, E) is allowed.

Another change introduced in QAD 2010.1 Enterprise Edition is the Rounding Differences account is no longer restricted to an Income or Expense type account. Any account type (A, L, I, E) is permitted. The Purchase Order Receipts account now must be a Liability type account.

The impact of these changes may be apparent (to the extent exceptions involving these account types exist in the pre-conversion environment) in the Account Type Incompatibility section of the Data Preparation Report and when choosing replacement accounts in the GL Account Type Utility for these types of accounts.

Table 2.2 lists the expected category for each Enterprise Edition account type.

Table 2.2 Expected Category by Account Type

Account Type	Category
Bank Account	Asset or Liability
Closing Account	any
Cross-Company Control Account	Asset or Liability
Customer Control Account	Asset or Liability
Customer Payment Account	Asset or Liability
Fixed Assets Account	Asset or Liability
Inventory Control Account	Asset or Liability
Supplier Control Account	Asset or Liability
Supplier Payment Account	Asset or Liability
Tax Account	any
WIP Control Account	Asset or Liability
Purchase Order Receipts	Liability
Realized Exchange Gain	any
Realized Exchange Loss	any
Result of Previous Years	Asset or Liability
Result of the Current Year	Asset or Liability
Rounding Differences	any
Unmatched Invoices	Asset or Liability
Unrealized Exchange Gain	any
Unrealized Exchange Loss	any

Unposted GL Transactions Error

The Data Preparation Report checks for any unposted GL transactions by accumulating glt_det records. It is possible that some of these records may be for a zero transaction amount and therefore will not show up on the Unposted Transactions Register (25.13.14, `glutrrp.p`). These zero-amount records can be deleted using GL Transaction Delete/Archive (36.23.2, `mgmgrp01.p`).

Country Code Utility (uxctryup.p)

QAD Enterprise Edition requires that all addresses and employees have a non-blank, valid country code.

This utility reports addresses and employees having a blank or invalid country code and updates them (in same execution) with a user-specified country code. Execution of this utility may or may not be necessary. If required, it is noted on the Data Preparation Report.

You can specify a range of addresses (which also covers employees) when multiple country codes must be assigned.

The utility provides a simulation mode that previews the effect of the update.

To see Addresses and Employees with invalid Country Codes, leave Country Code blank and set Update to No.

The report output filename is `uxctryup-<dbname>-<domain>-<date>_<time>.prn`.

Correct any discrepancies identified in the report.

For eB2.1 and later, it must be run for each active domain.

This utility will potentially alter future transactions involving tax calculations due to the changed country code on customer and/or supplier addresses updated with new country codes. For this reason, it may be preferable to delay running this utility until the database is closed to transaction processing.

Tax ID Utility (uxtaxid.p)

QAD Enterprise Edition requires all US suppliers for 1099 reporting (vd_1099 = true) have unique non-blank Federal tax IDs (ad_gst_id).

This utility reports duplicate or missing Federal tax IDs and lists VAT registration IDs that do not comply with the ISO format, when applicable. Execution of this utility may or may not be necessary. If required, it is noted on the Data Preparation Report.

If 1099 reporting is required, these errors must be manually corrected through Supplier Maintenance (2.3.1, `advnmt.p`).

If 1099 reporting is not used, set this field to No in the utility and it will assign vd_1099 = false for all U.S. suppliers, reducing the number of Federal tax ID errors.

Additionally, if Validate VAT Registration in Global Tax Management Control (2.13.24, `txtxcmt.p`) is true, VAT registration IDs for customers and suppliers in EU countries must conform to the ISO format for their country.

For eB2.1 and later, the utility must be run in each active domain. If the Validate VAT Registration is true for any single domain, the validation is considered true for all domains. This is necessary because Business Relations are created from the address records and Business Relations are system-wide within QAD Enterprise Edition.

The utility provides a simulation mode that previews the effect of the update.

The report output filename is `uxtaxid-<dbname>-<domain>-<date>_<time>.prn`.

Correct any discrepancies identified in the report.

The utility only affects 1099 reporting for US-based users and only when producing year-end 1099 forms.

Code Search and Replace Reports (eB2.1 and later only)

Run this suite of reports/utilities only if you are upgrading from eB2.1 and later. If you are not upgrading from eB2.1 and later, go to “Set Conversion Parameters” on page 24.

Use of these utilities and reports is optional (unless specifically indicated by the Data Preparation Report), but strongly recommended. They require that Table Extension Domain Update Utilities be run first.

The Code Search and Replace Reports are used to validate a number of areas that changed between eB2.1 and QAD Enterprise Edition.

Due to schema and architecture changes introduced in QAD Enterprise edition, the scope of some data objects has changed from domain level to system level.

During the conversion, if two records exist with the same code in more than one domain, they are assumed to be the same and only one record is converted. As a result, some domain-level data could potentially be lost when converting a database with more than one domain.

The purpose of the Code Search and Replace utilities is to provide a way to identify and correct instances of duplicate domain-level data.

Examples of Data Moved from Domain Level to System Level

- In QAD Enterprise Edition, addresses remain at the domain level, but map to business relations, which are at the system level in the new Financials. Each business relation must have a unique code.
- In QAD Enterprise Edition, entities remain at the domain level, but map to companies (now called entities) in the new Financials. The entity code must be unique in the database going forward.
- Before QAD Enterprise Edition, credit ratings were stored in the code master (code_mstr) table against field cm_cr_rating. Code master is a domain-level table in eB2.1 and later. In QAD Enterprise Edition, credit ratings are stored in a new Financials table and become system-level data.
- Before QAD Enterprise Edition, Customer and Supplier Types were stored in the code master (code_mstr) table against fields cm_type and vd_type. In QAD Enterprise Edition, customer and supplier types are stored in a new Financials table and become system-level data.
- Before QAD Enterprise Edition, voucher types were stored in the code master (code_mstr) table against field vo_type. In QAD Enterprise Edition, voucher types are called purchase types and stored in a new Financials table as system-level data.
- State and county codes are stored in the code master (code_mstr) table against fields ad_state and ad_county. In QAD Enterprise Edition, states and counties become system-level data.
- Before QAD Enterprise Edition, tax classes were stored in the code master (code_mstr) table against field taxc_taxc. In QAD Enterprise Edition, tax classes are stored in a new table (txcl_mstr) and become system-level data.
- Before QAD Enterprise Edition, tax usages were stored in the code master (code_mstr) table against field tx2_tax_usage. In QAD Enterprise Edition, tax usages are stored in a new table (txu_mstr) and become system-level data.
- Before QAD Enterprise Edition, tax types were stored in the code master (code_mstr) table against field txt_tax_type. In QAD Enterprise Edition, tax types are stored in a new table and become system-level data.
- Tax zones are currently stored in the tax zone master (txz_mstr) table. This is a domain-level table in eB2.1 and later. In QAD Enterprise Edition, the domain field was removed and tax zones are system-level data.
- Tax environments are stored in the tax environment master (txe_mstr) and tax environment detail (txed_det) tables. These are domain-level tables in eB2.1 and later. In QAD Enterprise Edition, the domain field was removed and tax environments are system-level data.

- Credit terms are stored in the credit terms master (ct_mstr) and credit terms detail (ctd_det) tables. These are domain-level tables in eB2.1 and later. In QAD Enterprise Edition, credit terms are stored in the payment condition table in the new Financials. Payment conditions are system-level data. Payment conditions are maintained in Financials and automatically replicated to all domains in the operational area, which means each domain will contain the same credit terms records.
- Rounding methods are stored in the rounding methods master (rnd_mstr) table. This is a domain-level table in eB2.1 and later. In QAD Enterprise Edition, the domain field was removed and rounding methods are system-level data. Rounding methods are maintained in Financials and automatically replicated to the operational area.

Duplicate Code Report (36.28.1.1 – utsarrp1.p)

This report highlights codes (entities, credit ratings, credit terms, tax parameters, and so on) occurring in multiple domains. These moved to the system level in QAD Enterprise Edition, and only one instance of each code remains. This report allows you to identify any differences in the codes across domains.

The report output can be directed to a printer or a user-specified filename. This report does not prompt for an output location. If you specify a filename, the output location can vary by installation. Possible locations include the directory from which MFG/PRO was launched and the user's home directory.

Correct any discrepancies listed in the report.

Code Usage Report (36.28.1.2 – utsarrp2.p)

This report highlights the number of instances where a particular code is used in the current domain, providing more insight into the use and analysis of the code.

The report output can be directed to a printer or a user-specified filename. This report does not prompt for an output location. If you specify a filename, the output location can vary by installation. Possible locations include the directory from which MFG/PRO was launched and the user's home directory.

Correct any problems identified in the report.

Replace Code Utility (36.28.1.3 – utsarrp3.p)

Use this utility to replace specified code values in the current domain with a new user-specified value.

Warning This utility can have a broad effect on the system and future transactions, depending on the codes replaced. Use it judiciously and only after backing up the database.

GL Account Type Utility (uxglacup.p) – Report Mode

In QAD Enterprise Edition Financials, each General Ledger (GL) account has an account type, a classification of how the account is used. These types include control accounts, Intercompany accounts, bank and cash accounts, and special accounts (classified as System accounts) dedicated

to period closing and exchange rate fluctuations. Any other accounts not falling into these specialized classifications are classified as Standard accounts. These are predominantly income/expense accounts, but also include some Balance Sheet accounts as well.

Control, Banking, Cash, Intercompany and System type accounts have restrictions on how and where they can be used in transactions. Standard accounts have no restrictions other than they cannot be used for purposes associated with the other account types.

System type accounts can have one only one GL account per domain defined for each individual purpose. For example, only one account can be used for Unrealized Exchange Rate Gains in a domain, regardless of the currency involved. Similarly, another single GL account must be dedicated to Exchange Rate Rounding Differences. The one exception to this rule for System type accounts is PO Receipts. Multiple accounts can be used for PO Receipts.

Intercompany accounts are similar to System type accounts, but they are not classified as a System type account. In the old Financials, it is possible to have different Intercompany accounts for the debit side and credit side for each functional area (AR, AP, Inventory, and Fixed Assets) and each entity (and in eB2.1 and later for each domain). QAD Enterprise Edition handles this granularity differently.

Separate accounts for Intercompany debits and credits are no longer supported. Further, the Intercompany account used for a functional area must be the same for that functional area in every entity within the same domain. If this level of detail is not desired, the same Intercompany account can be used for any or all functional areas in a domain.

The utility distinguishes between Tax accounts used for AR and Tax accounts used for AP. However, separate account codes are not required for each of these areas. You can use the same account code for AR and AP taxes if desired.

All banks belonging to the same entity (within the same domain, if applicable) and having the same Cash account must also share the same Payment in Process (PIP) account if the Use Payment In Process Acct field is enabled in Accounts Payable Control. Similarly, all banks belonging to the same entity (within the same domain, if applicable) and having the same Cash account must share a common Drafts Payable account if the Use Draft Management field is enabled in Accounts Payable Control. It is acceptable to use the same account for both PIP and Drafts Payable when there is an overlap of bank code, entity and Cash accounts for PIP and Drafts Payable, as both account types are Supplier Payment accounts in QAD Enterprise Edition. These requirements are only applicable when European Accounting is not in use for the associated domain (if applicable) or database.

For example, consider the following bank definitions, which assume both fields in Accounts Payable Control are enabled and European Accounting is not used in the domain.

Table 2.3
Bank Definitions

Domain	Entity	Bank	Cash Acct	PIP Acct	Drafts Pay Acct
demo1	1000	AA	1040	2110	2300
demo1	1000	A2	1040	2111	2300
demo1	1000	BB	1041	2110	2300
demo1	1000	B2	1041	2110	2301
demo1	1000	XX	1040	1040	2300
demo1	1000	X2	1040	1041	2301

Domain	Entity	Bank	Cash Acct	PIP Acct	Drafts Pay Acct
demo1	2000	CC	1040	1040	2300
demo1	2000	C2	1041	1041	2301

- Banks AA and A2 are in the same domain and entity and use the same Cash and Drafts Payable accounts, but have different PIP accounts. The GL Account Type Utility requires that a single PIP account be designated for these two banks.
- Banks BB and B2 are in the same domain and entity and use the same Cash and PIP accounts, but have different Drafts Payable accounts. The GL Account Type Utility requires that a single Drafts Payable account be designated for these two banks.
- Banks XX and X2 are in the same domain and entity and use the same Cash account, but have different PIP and Drafts Payable accounts. The GL Account Type Utility requires that a single PIP account be designated for these two banks. It also requires that a single Drafts Payable account be designated for these two banks.

If desired, you can use the same account for any or all of the above PIP and Drafts Payable conflicts.

- No changes are required for the PIP and Drafts Payable accounts in Banks CC and C2. They do not share the same Cash account.

GL Allocation codes are no longer permissible in any account type field other than Standard Account types. Even within Standard Accounts, their use is limited. The Data Preparation Report will highlight any database fields using a GL allocation code where it is not allowed.

The GL Account Type Utility identifies GL accounts in static data (not transactional data) that will not pass the requirements described above when moved to QAD Enterprise Edition Financials. For every exception encountered in the database, you are prompted for a GL account to use during conversion to correct the exception.

When checking for exceptions involving the Accounts Receivable and Accounts Payable control accounts, the utility uses the account information entered through the Control Account utility as its basis for comparison.

The utility provides a report-only option listing the GL accounts in conflict. The report can be run in Detail or Summary mode.

- Detail mode lists every instance of an account that is in conflict, the menu function where the conflict occurs, and some key values to aid in identifying the specific offender.
- Summary mode lists only the first instance of an account in conflict and the menu function where it is defined. There can be additional instances other than the one shown.

The report output filename is `uxglacup-dtl-<dbname>-<domain>-<date>_<time>.prn`.

or

`uxglacup-sum-<dbname>-<domain>-<date>_<time>.prn`.

The report-only mode should be used first as a planning tool to determine how to resolve the exceptions.

Once this is done, the corrections can be done in several ways.

- Correct the offending account directly in the program shown on the report.

- Use the GL Account Type Utility to make the correction.
- Use a combination of the above two methods.

QAD recommends using the utility to make corrections. Otherwise, any changes can affect how future transactions are booked to the General Ledger. When using the utility and assigning new accounts to Standard Account types, pay careful attention to the detail report to understand all of the places where the New Acct value will be applied.

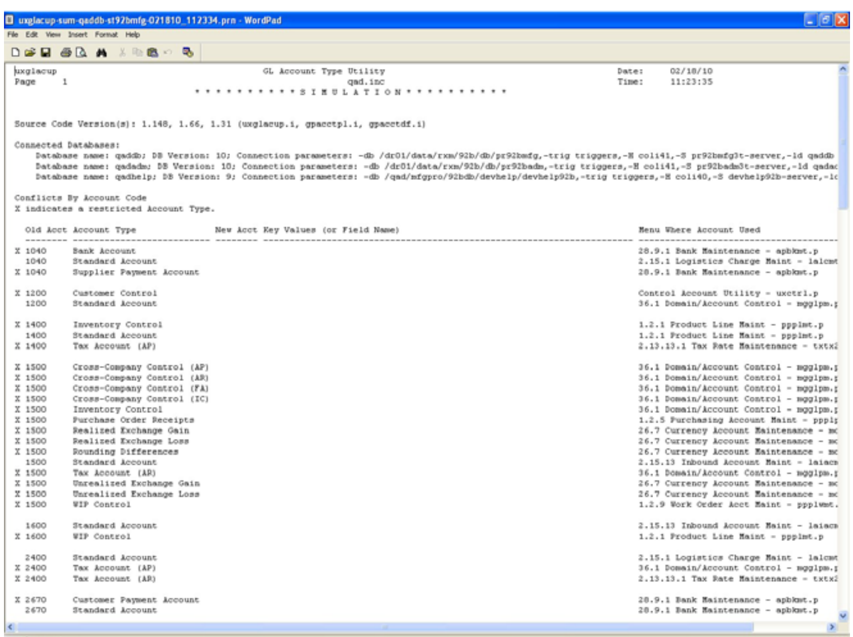
Run the utility as many times as necessary. If it was previously run in Update mode, the previous answers are displayed on the screen. Each time the utility runs, it checks for additional exceptions since the last execution. Therefore, you must run the utility a final time in Update mode just before starting the conversion.

For eB2.1 and later, the utility must be run in each active domain. The Data Preparation Report checks for this as well as verifying that all exceptions were resolved. If not, errors are reported.

This utility, when run in Update mode, does not affect the system or future transactions. It only creates qad_wkfl records. However, any account conflicts resolved by modifying the accounts directly in their menu functions (for example, Product Line Maintenance, System Control, and so on) will affect subsequent transactions involving the account fields updated. Take this into consideration when developing a plan to resolve the account conflicts noted on the report produced by this utility.

The following report uses account 1040 as an example.

Fig. 2.2
GL Account Type Utility Output



```
uxglacup-sum-qaddb-st92bmfg-021810_112334.prn - WordPad
File  Edit  View  Insert  Format  Help

uxglacup                              GL Account Type Utility                              Date:     02/18/10
Page      1                                          qad.inc                               Time:     11:23:35
                          * * * * * * * * * * * S I M U L A T I O N * * * * * * * * * * *


Source Code Version(s): 1.148, 1.66, 1.31 (uxglacup.i, gpacctpl.i, gpacctdf.i)

Connected Databases:
    Database name: qaddb; DB Version: 10; Connection parameters: -db /dr01/data/rxm/92b/db/pr92bmfg,-trig triggers,-H coli41,-S pr92bmfg3t-server,-ld qaddb
    Database name: qadadm; DB Version: 10; Connection parameters: -db /dr01/data/rxm/92b/db/pr92badm,-trig triggers,-H coli41,-S pr92badm3t-server,-ld qadac
    Database name: qadhelp; DB Version: 9; Connection parameters: -db /qad/mfgpro/92bdb/devhelp/devhelp92b,-trig triggers,-H coli40,-S devhelp92b-server,-lc

Conflicts By Account Code
X indicates a restricted Account Type.

  Old Acct Account Type               New Acct Key Values (or Field Name)                                       Menu Where Account Used
  -------- -------------------------- -------- ------------------------------------------------------------------ ------------------------------------
X 1040     Bank Account                                                                                         28.9.1 Bank Maintenance - apbkmt.p
  1040     Standard Account                                                                                     2.15.1 Logistics Charge Maint - lalcmt
X 1040     Supplier Payment Account                                                                             28.9.1 Bank Maintenance - apbkmt.p

X 1200     Customer Control                                                                                     Control Account Utility - uxctrl.p
  1200     Standard Account                                                                                     36.1 Domain/Account Control - mgglpm.p

X 1400     Inventory Control                                                                                    1.2.1 Product Line Maint - ppplmt.p
  1400     Standard Account                                                                                     1.2.1 Product Line Maint - ppplmt.p
X 1400     Tax Account (AP)                                                                                     2.13.13.1 Tax Rate Maintenance - txtx2

X 1500     Cross-Company Control (AP)                                                                           36.1 Domain/Account Control - mgglpm.p
X 1500     Cross-Company Control (AR)                                                                           36.1 Domain/Account Control - mgglpm.p
X 1500     Cross-Company Control (FA)                                                                           36.1 Domain/Account Control - mgglpm.p
X 1500     Cross-Company Control (IC)                                                                           36.1 Domain/Account Control - mgglpm.p
X 1500     Inventory Control                                                                                    36.1 Domain/Account Control - mgglpm.p
X 1500     Purchase Order Receipts                                                                              1.2.5 Purchasing Account Maint - pppip
X 1500     Realized Exchange Gain                                                                               26.7 Currency Account Maintenance - mc
X 1500     Realized Exchange Loss                                                                               26.7 Currency Account Maintenance - mc
X 1500     Rounding Differences                                                                                 26.7 Currency Account Maintenance - mc
  1500     Standard Account                                                                                     2.15.13 Inbound Account Maint - laiacm
X 1500     Tax Account (AR)                                                                                     36.1 Domain/Account Control - mgglpm.p
X 1500     Unrealized Exchange Gain                                                                             26.7 Currency Account Maintenance - mc
X 1500     Unrealized Exchange Loss                                                                             26.7 Currency Account Maintenance - mc
X 1500     WIP Control                                                                                          1.2.9 Work Order Acct Maint - ppplwmt.

  1600     Standard Account                                                                                     2.15.13 Inbound Account Maint - laiacm
X 1600     WIP Control                                                                                          1.2.1 Product Line Maint - ppplmt.p

  2400     Standard Account                                                                                     2.15.1 Logistics Charge Maint - lalcmt
X 2400     Tax Account (AP)                                                                                     36.1 Domain/Account Control - mgglpm.p
X 2400     Tax Account (AR)                                                                                     2.13.13.1 Tax Rate Maintenance - txtx2

X 2670     Customer Payment Account                                                                             28.9.1 Bank Maintenance - apbkmt.p
  2670     Standard Account                                                                                     28.9.1 Bank Maintenance - apbkmt.p

For Help, press F1                                                                                                                      NUM
```

The instances with an X must have a unique account code separate from the others.

It is not necessary to change each instance of account 1040 to a new account. Nor is it necessary to change every instance of account 1040 with an X to a new account.

The objective is for all three of these instances to have different accounts after the conflicts for 1040 are resolved. Any one of them can retain the original account 1040 as long as neither of the other two is assigned account 1040.

Refer to Table 2.2 on page 16 for the expected category for each Enterprise Edition account type.

Impact of GL Account Type Utility During a Conversion

The data entered in the GL Account Type Utility is used for two purposes during a conversion:

- Where the Type of an account was conflicting pre-conversion, the data entered in the utility is used to set the Type of the GL Account.
- Where you have elected to replace an account with a new Account Code for a particular usage, a new account is created and assigned to the relevant fields.

Pre-conversion Integrity Report (gpinckrp.p)

This report assesses the status of financial transaction data before conversion. It reports AR, AP and GL transaction integrity information in a single report. Inconsistencies are highlighted that can optionally be corrected before conversion.

The report output filename is:

`gpinckrp-dtl-<dbname>-<date>_<time>.prn`

or

`gpinckrp-sum-<dbname>-<date>_<time>.prn`.

Setting Build Integrity Check Records to Yes stores data in the database for the current report execution, enabling later reproduction of the report showing the same information. Anytime the current status of the database is required, set this field to Yes. Set this field to No to run the report using the data collected from a previous execution.

Report Integrity Check Records displays the information collected during the first step. This field is usually set to Yes.

For eB2.1 or later, the reports span all active domains regardless of the domain where they are launched.

If data is corrected, run this report again to capture an accurate pre-conversion snapshot. Rerun this report as many times as necessary.

AR Transaction Integrity

This portion of the report compares the sum of open AR invoices and payments by account to the sum of the corresponding amounts for each AR GL account. It reports any differences in the local and/or transaction currency amounts. The report also lists any non-AR GL transactions posted against an AR control account. Finally, the sum of open AR invoices and unapplied payments by customer is compared to each customer's Open Balance in Customer Maintenance, reporting any differences. You can correct differences by running the Adjust Customer Balance utility (36.25.5, `utcsbal.p`).

It also performs various database integrity checks:

- The customer referenced on each invoice still exists.
- Every invoice has detail line information.

AP Transaction Integrity

This portion of the report compares the sum of open AP vouchers by account to the sum of the corresponding amounts for each AP GL account.

It reports any differences in the local and/or transaction currency amounts. The report also lists any non-AP GL transactions posted against an AP control account. Finally, the sum of open AP vouchers by supplier is compared to each supplier's Open Balance in Supplier Activity Inquiry, reporting any differences. You can correct differences by running the Adjust Supplier Balance utility (36.25.4, `utvdbal.p`).

It also performs various database integrity checks:

- The supplier referenced on each voucher still exists.
- Every voucher has detail line information.

GL Transaction Integrity

The totals of posted GL transactions by account in the gltr_hist table are compared to the amounts stored in the Account Balance table (acd_det). Differences are reported and can be corrected by running the Recalculate acd_det Totals utility (36.25.39, `utacdfix.p`).

Warning Running the Recalculate acd_det Totals utility in a database containing consolidated GL transactions will zero out the period totals for all periods other than the consolidation period.

Any out-of-balance transactions are reported because the conversion has to balance them by creating offsetting entries. If the database contains GL transactions consolidated by selected accounts, the report should be run in Summary mode, which verifies the transactions are in balance for the year, not by individual transaction.

It also performs these database integrity checks:

- The entity in each GL transaction still exists.
- The account for each GL transaction still exists and is active.
- All effective dates for GL transactions have corresponding periods in the GL calendar.

If any of these types of errors are reported, manually fix them before beginning conversion.

Set Conversion Parameters

Conversion Parameters Utility (utfinpar.p)

This utility prompts you for values for the parameters used by the QAD Enterprise Edition conversion when creating new Financial objects. Customers upgrading from eB2.1 and later with multiple domains may want to provide a different set of values for each domain. For eB2.1 and later, the utility must be run in each active domain, regardless of whether values differ by domain.

The majority of these parameters are for Daybook codes used in various areas of the applications. Other parameters are items such as default Credit Terms, Sub-account, Cost Center, and so on. Some parameters can be specified by domain whereas others are system-level and apply to all domains.

Since there are so many parameters, QAD suggests running this utility well in advance of the conversion to allow ample time for planning and decision-making. The table below lists the parameters along with a brief explanation of each. eB2.1 and later users may find it helpful to extract this table to a spreadsheet with one column per domain for planning purposes.

If new validations were added to the utility since it was last run, the Data Exists flag for parameters with existing values is set to No. This causes a warning to appear when the utility starts because the existing values are treated as invalid. Attempting to reaccept an invalid parameter displays an error message that explains why the parameter is no longer valid.

Rerun the utility as often as needed, deleting or changing values.

Table 2.4
New Financial Object Parameters

Level	Parameter	Use
Domain	Default AR Finance Charges Daybook	Daybook code to use for this default daybook type. It is not required if Finance Charges are not used.
Domain	Default AR Credit Note Daybook for Operational Invoices	Daybook code to use for this default daybook type.
Domain	Default AR Credit Note Daybook for Financial Invoices	Daybook code to use for this default daybook type.
Domain	Default AR Invoice Daybook for Operational Invoices	Daybook code to use for this default daybook type.
Domain	Default AR Invoice Daybook for Financial Invoices	Daybook code to use for this default daybook type.
Domain	Default AR Payment Daybook	Daybook code to use for this default daybook type.
Domain	Default Customer Adjustment Daybook	Daybook code to use for this default daybook type.
Domain	Prefix for AR Tax Register Daybook Sets	The prefix to use when creating AR Tax Register Daybooks
Domain	Prefix for AP Tax Register Daybook Sets	The prefix to use when creating AP Tax Register Daybooks
Domain	Default AP Credit Note Daybook for Financial Invoices	Daybook code to use for this default daybook type.
Domain	Default AP Invoice Daybook for Financial Invoices	Daybook code to use for this default daybook type.
Domain	Default AP Payment Daybook	Daybook code to use for this default daybook type.
Domain	Default ERS Supplier Invoice Daybook	Daybook code to use for this default daybook type. This prompt only appears if ERS is used.
Domain	Default ERS Supplier Credit Note Daybook	Daybook code to use for this default daybook type. This prompt only appears if ERS is used.

Table 2.4 — *New Financial Object Parameters* (Page 1 of 4)

Level	Parameter	Use
Domain	Default Intercompany Daybook	Daybook code to use for this default daybook type. A Daybook code must be specified for each functional area (IC, FA, Inventory, SO, and WO) but they can share the same code if desired.
Domain	Default Matching Entry Daybook	Daybook code to use for this default daybook type.
Domain	Default Banking Entry Daybook	Daybook code to use for this default daybook type.
Domain	Default Inventory Daybook	Daybook code to use for this default daybook type.
Domain	Default Work Order Daybook	Daybook code to use for this default daybook type.
Domain	Default Sales Order Daybook	Daybook code to use for this default daybook type.
Domain	Default Fixed Assets Daybook	Daybook code to use for this default daybook type.
Domain	Default Daybook for Journal Entries	Used for any domain that has not previously used daybooks when converting GL transactions. Note: This should be a new daybook code that the conversion creates and not one of the default daybooks specified above.
Domain	Default Daybook Sequence Effective Date (MM/DD/YYYY)	The earliest date when the new Financial daybooks can be used.
Domain	Account for Posting Balances	This account is used by the conversion when it balances any unbalanced double-sided GL transactions. This is the account where such offsets are posted. Note: This should be a new account that does not exist in the database. The conversion creates it.
Domain	Account for Year-End Balances	New account to be created by the conversion for posting the offset to year-end closing entries.
Domain	Account for Results of Current Year	This prompt only appears if the GL Report Writer is not in use or no account was assigned for this purpose in that module. If prompted for this account, it must be an account with no transactions posted against it. It can be a new account. If so, it is created by the conversion. It is used to accumulate the YTD profit/loss total for printing on the Balance Sheet. It is not created if the conversion finds GL transactions posted to the co_ctrl.co_pl account in the General Ledger Control; co_ctrl.co_pl is used instead in that case.

Table 2.4 — *New Financial Object Parameters* (Page 2 of 4)

Level	Parameter	Use
Domain	Account for Current Year Income Offset	This prompt only appears if the GL Report Writer is not used or no account was assigned for this purpose in that module. If prompted for this account, it must be new and not exist. It is created by the conversion.
Domain	Account for Results of Previous Years	The conversion creates this account. It is used to accumulate the profit/loss total for previous unclosed years when printing the Balance Sheet.
Domain	Default Account for Unmatched Invoices	Default GL account to use for Unmatched Invoices. If using European Accounting, the use Waiting Expenses account in European Accounting Control (25.23, `eueupm.p`) will default to this field and cannot be modified by the user. If not using European Accounting, specify a new account to create during conversion.
Domain	Default Sub-Account	Default sub-account to use when an account requires a sub-account. Note: This should be a new value that does not exist in the database. See also “COA Mask” on page 136.
Domain	Default Cost Center	Default cost center to use when an account or sub-account requires a cost center. Note: This should be a new value that does not exist in the database. See also “COA Mask” on page 136.
Domain	Default Project	Default project code to use when an account, sub-account or cost center requires a project code. Note: This should be a new value that does not exist in the database. See also “COA Mask” on page 136.
Domain	Default Credit Term	Used when a credit term cannot be converted because it has a due date (or if the term has stages and one of the stages has a due date).
Domain	Convert Unused Cashbooks?	Should the conversion convert “not used” cashbooks (for example, ba_mstr records where ba_status = “NU”)?
Domain	Create Entity Tax Periods?	Should the conversion create Tax Periods for each entity? If this option is selected, the conversion creates the Tax Periods that are the same as the existing GL Periods.
System	Default Tax class for Suspended and Delayed Tax	The default tax class to use for Suspended and Delayed Taxes. This prompt only appears if the European Accounting module is in use. (SP3 and above only)

Table 2.4 — *New Financial Object Parameters* (Page 3 of 4)

Level	Parameter	Use
System	Management Currency	This is used to calculate the Management Currency amounts during the transactional conversions. It is also used to set the Management Currency in QAD Enterprise Edition Financials after the conversion. Note: Setting the correct value here is critical, as it cannot be changed after any transactions are posted. Note: This value is also used as the default value for the Statutory Currency.
System	Country for default Tax Box	Country code for the default Tax Box used in Tax Reporting
System	Source directory for Payment Format XML files	The directory from which the Payment format XML files will be loaded. This option will only appear if the source database uses European Accounting.
System	Convert existing security setup?	Should the conversion convert the existing security setup or just convert Users, Groups and allow the user to re-implement the security setup after the conversion?

Table 2.4 — *New Financial Object Parameters* (Page 4 of 4)

You can exit the utility without providing all of the answers, but it must be completed before running the conversion.

The report output filename is `utfinpar-<dbname>-<domain>-<date>_<time>.prn`.

This utility does not affect the system or future transactions. It only creates qad_wkfl records.

Pre-conversion Completion

Process Pending Transactions

Process pending transactions as follows:

1. Print any pending invoices (7.13.3 Pending Invoice Print, `sosorp10.p`).
2. Post any remaining invoices (7.13.4 Invoice Post, `soivpst.p`).
3. Confirm all unconfirmed vouchers (28.6 Voucher Confirmation Automatic, `apvoco01.p`; 28.7 Voucher Confirmation Manual, `apvoco.p`).

   Note This step is optional. Refer to “Unconfirmed Supplier Vouchers” on page 147.
4. Post all GL transactions (25.13.7 Transaction Post, `gltrpst.p`).

Close the Production Database to Users

Use Menu Security (36.10, `mgpwmt.p`) to prevent the addition of new data to the database and further transactions.

Rerun Pre-conversion Integrity Report (gpinckrp.p)

Rerun the Pre-conversion Integrity Report. Refer to “Pre-conversion Integrity Report (gpinckrp.p)” on page 23

Rerun GL Account Type Utility in Update Mode (uxglacup.p)

1 Rerun the GL Account Type Utility in Update mode (in all active domains for eB2.1 and later). Refer to “GL Account Type Utility (uxglacup.p) – Report Mode” on page 19.

An error is displayed if the GTM Conversion was not done and conflicts were found in the accounts from the obsolete tax tables (tax_mstr, vt_mstr). The utility is not flagged as complete until the GTM Conversion is done. This utility must be run again to check for conflicts involving accounts in the new tax table (tx2_mstr).

2 Resolve any new conflicts.

Run the Converted GL Account Definition Report

This report provides a preview of how the GL accounts and their key attribute settings will look after the conversion. Use the output from this report to more accurately plan and prepare a company’s GL account data before conversion execution.

This report can only be executed after the following utilities are run:

- Control Account Utility
- GL Account Type Utility (in update mode)
- Conversions Parameters Utility
- GL Account Project Code Utility (if pre-eB)

The report is provided for informational purposes and is not mandatory. For versions eB2.1 and later, the report content is specific to the domain in which the report was initiated.

This report has no effect on the system or future transactions and can be run as often as desired.

Standard Period Closing Reports

Although their use is optional and does not directly affect the conversion, QAD recommends running the standard period closing reports for later comparison with the equivalent post-conversion reports. This helps ensure that data was unchanged by the conversion. The reports are as follows:

- Trial Balance (25.15.4, `gltbrp.p` or 25.15.5, `gldtbrp.p`)
- Balance Sheet (25.15.8, `glbsrp.p`)
- Income Statement (25.15.13, `glinrp.p`)
- AR Aging as of Effective Date (27.18, `arcsrp05.p`)
- AP Aging as of Effective Date (28.17.6, `apvorp04.p`)
- AR Tax by Tax Rate (2.13.15.14, `txarrp01.p`)
- AP Tax by Tax Rate (2.13.15.17, `txaprp01.p`)
- AR Balance Report (27.20, `arcsrp.p`)

- Inventory Valuation (3.6.13, `ppptrp03.p`)
- Unvouchered Receipts Report (5.13.10, `poporp11.p`)
- Purchase Receipt Report (5.13.5, `poporp06.p`)
- Asset Owned Report (32.5.11, `faaorp.p`)
- Sales Order Report (7.15.1, 2 or 3, `sosorp.p`, `sosorp01.p`, `sosorp02.p`)

Rerun (Finalize) Data Preparation Report

Rerun the Data Preparation Report to ensure no errors were introduced. Correct the data until no errors are present.

Chapter 3

Conversion Execution

This chapter describes how to execute a Progress database conversion.

Introduction 32

Conversion Execution Overview 32

Prepare Environment 32

Prepare Source Databases 32

Install and Configure Software 32

Conversion Setup 33

Execute Conversion 44

Conversion Validation 45

Enter License Codes 45

Conversion Execution Troubleshooting 46

Introduction

This section provides detailed instructions on the steps to execute the conversion.

Conversion Execution Overview

The conversion process consists of the following major tasks:

- Prepare Environment
- Prepare Source Databases
- Install and Configure Software
- Conversion Setup
- Execute Conversion
- Conversion Validation
- Enter License Codes

Prepare Environment

The following environment variables must be defined before starting QDT:

- `export DLC=<Progress install directory>`
- `export CATALINA_HOME=<tomcat install directory>`
- `export JAVA_HOME=<Java install directory>`

Note QAD recommends defining these variables in a script and running this script when the user logs into the system

Prepare Source Databases

If the main and admin database versions differ from the Progress version to be used in the new QAD product environment, the databases must be converted to the new Progress version using the Progress `proutil` utility.

Note For additional information about the `proutil` utility, please refer to the *Progress OpenEdge Data Management: Database Administration* manual.

Conversion execution will not start if there are errors in the Data Preparation Report. All errors must be resolved before proceeding with the conversion.

Install and Configure Software

Install QDT

Install the latest version of QDT that includes conversions.

Refer to *QAD Enterprise Applications Enterprise Edition Installation Guide: Progress Database* for more information on QDT installation.

Note The conversion software is not deployed on QAD Enterprise Edition media. A separate QDT download that includes conversions is required.

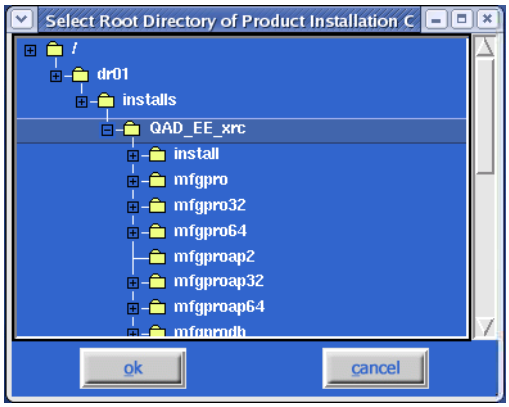
Install QAD Enterprise Edition

Using the latest version of QDT (that includes conversions), install Enterprise Edition.

Refer to *QAD Enterprise Applications Enterprise Edition Installation Guide: Progress Database* for detailed QAD Enterprise Edition installation information.

Note During QAD Enterprise Edition installation, the following screen prompts you to enter the location of the QAD Enterprise Edition installation media. This is necessary because the QDT release used for conversions is not part of the standard QAD Enterprise Edition.

Fig. 3.1
Installation Location Selection



Note

QDT and QDTAdmin must run in an X Window/VNC Server.

The QAD Enterprise Edition patch level may require updating. Refer to *QAD Enterprise Applications Enterprise Edition Installation Guide: Progress Database* for more information.

Do not run the Configure QAD EE option during a conversion. It is only required when doing a clean install with no conversion planned.

Conversion Setup

Enable Large Files

If applicable, enable large files.

Note If the source database uses large files, the target database may have to be configured to use large files.

Enable Conversions

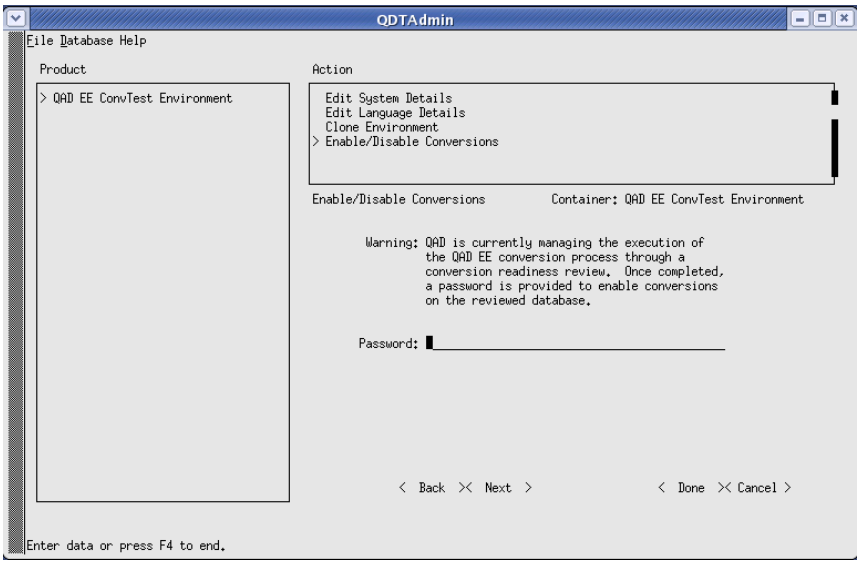
1 Start QDT. Navigate to <*QDT installation directory*> and run `./qadinst` (Linux) or `./qadinst.ksh` (UNIX).

2 The QDT Main Menu appears again. Press the Admin button.

Fig. 3.2
QDT Main Menu

3 You must enable conversions before proceeding.

Fig. 3.3
Enable Conversions

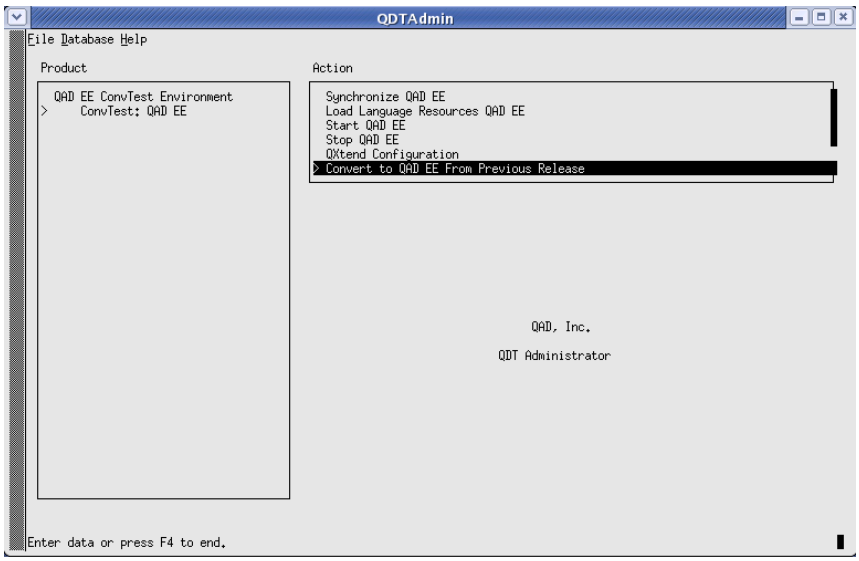


The password required to enable conversions is supplied with the separate release of QDT.

Convert QAD Enterprise Edition from Previous Release

1 Once the conversion is enabled, a screen similar to Figure 3.4 appears. Press Enter on the left-hand menu item that corresponds to the environment name.

Fig. 3.4
Convert From Previous Release



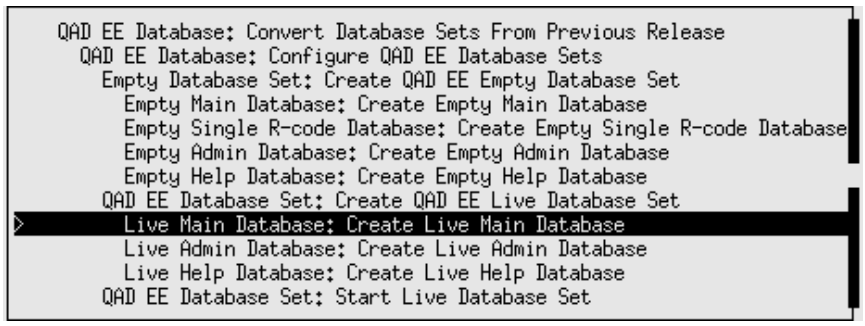
2 The QAD EE sub-menu option appears. Press Tab on this menu item. This will bring the cursor over to the right-hand menu.

Note Do not select the Configure QAD EE option. Instead, select Convert QAD EE from Previous Release as shown above. Note the QDT UI can only display six options at a time. If you scroll down to the Convert QAD EE from Previous Release option after enabling conversions, you cannot see the first item in the list (Configure QAD EE).

Create Live Main Database

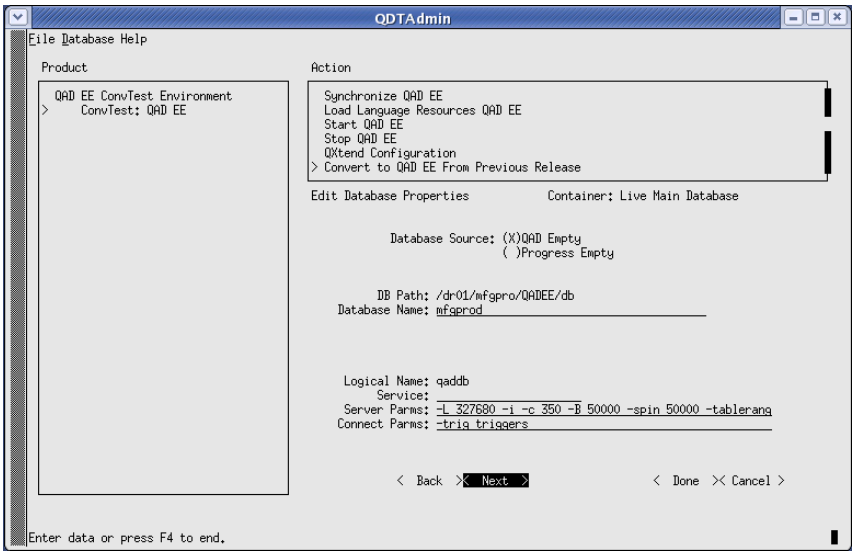
1 Select Live Main Database: Create Live Main Database as shown in Figure 3.5.

Fig. 3.5
Conversion Option Menu



2 Tab to the Next button and press Enter to bring you to the second option menu (Figure 3.6).

Fig. 3.6
Live Main Database Option Screen 2



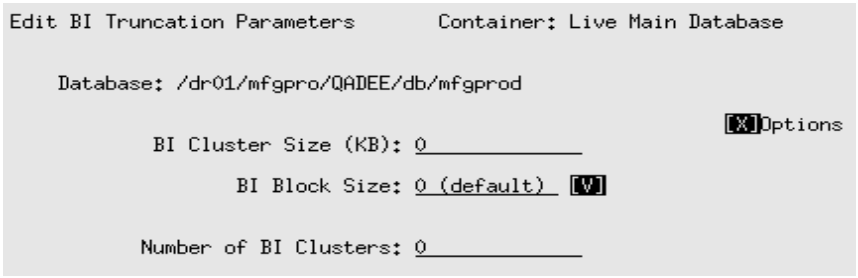
3 Change the values in the Server Parms field to the following:

- set -L: `327680`
- Include: `-i`
- Set -B: `50000`

4 Select Next twice, pressing Enter each time. You should now be on the fourth menu screen for Live Main Database: Create Live Main Database.

5 Tab to the Options field and press Enter to bring up more settings as shown below in Figure 3.7.

Fig. 3.7
BI Block Options



6 Change the BI parameters as follows:

- BI Cluster Size (KB): `32768`
- BI Block Size: `16k`
- Number of BI Clusters: `6`

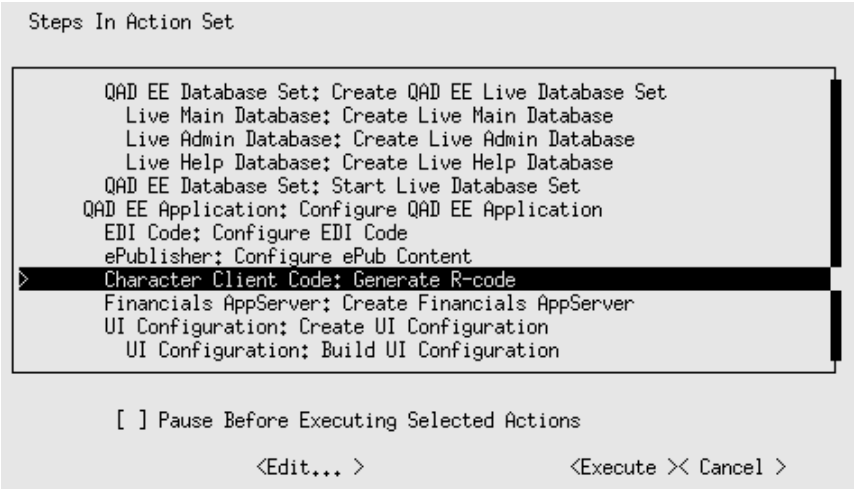
These changes to the BI cluster size can greatly improve performance, but may lead to issues at the compile step. If you encounter problems, change these settings after the compile step by running the following command:

```
proutil mfgprod –C truncate bi –bi 32768 –biblocksize 16
```

Character Client Code: Generate R-Code

1 Select Character Client Code: Generate R-Code and press Enter.

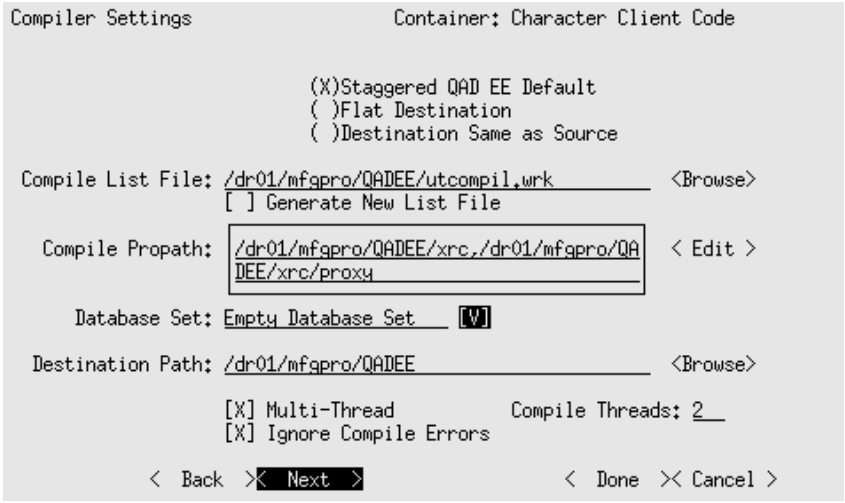
Fig. 3.8
Generate R-Code Menu Option



2 Change the Compiler Settings (Figure 3.9). Use the spacebar to select and configure the following settings:

- Multi-thread: Enabled
- Ignore Compile Errors: Enabled
- Compile Threads: Enter the number of CPU cores

Fig. 3.9
Compiler Settings



3 Tab to Next and press Enter.

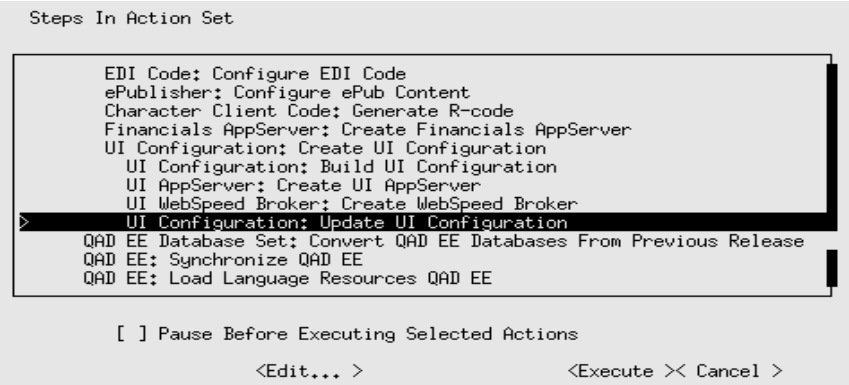
4 Create Dir: Yes.

5 Tab to Next and press Enter. This returns to the main conversion option screen.

Update UI Configuration

1 Select UI Configuration: Update UI Configuration.

Fig. 3.10
UI Configuration Menu Item



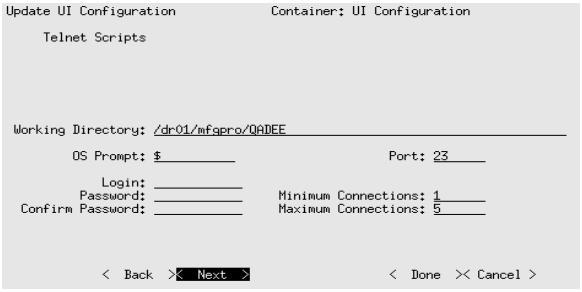
2 In the UI configuration screen (Figure 3.11), make the following changes:

- Working Directory: <defaulted in for your specific environment>
- Login: `mfg`
- Password: *<mfg user's password>*
- Confirm Password: *<mfg user's password>*

Notes:

- The MFG login is the O/S login ID and password.
- The password fields do not show key entries.

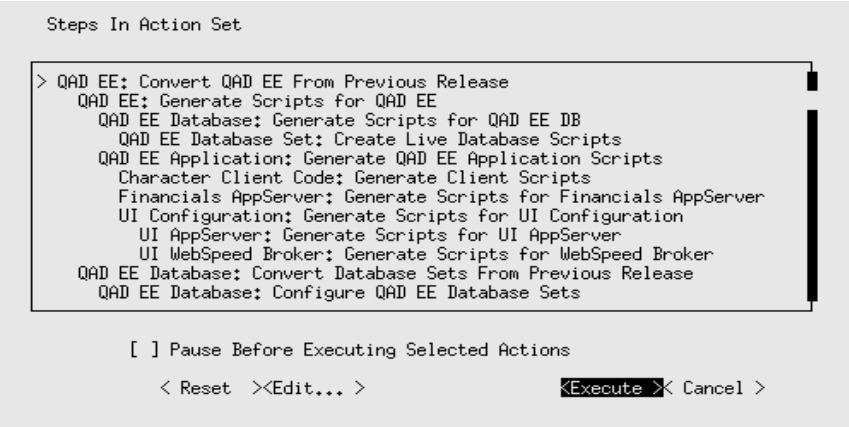
Fig. 3.11
UI Configuration Screen



Convert QAD Enterprise Edition Database from Previous Version

1 Select QAD EE Database Set: Convert QAD EE Databases From Previous Release as shown in Figure 3.12 and press Enter.

Fig. 3.12
Convert Enterprise Edition

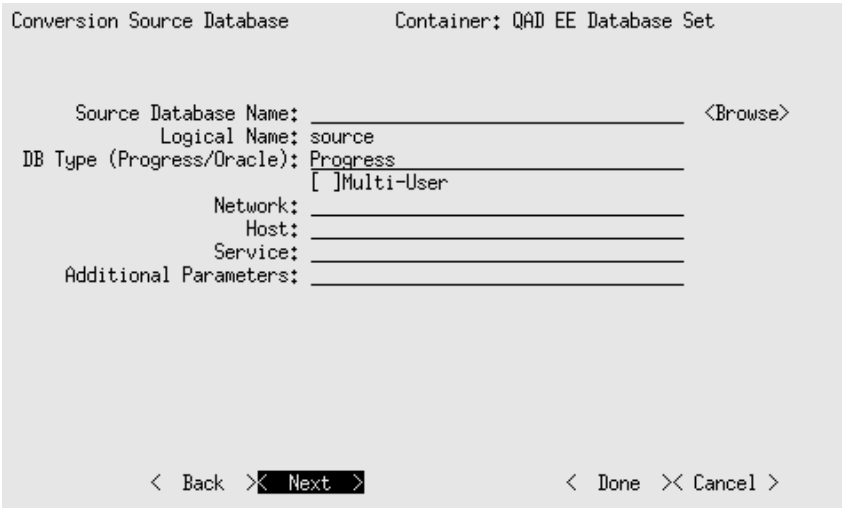


2 Tab to Next on the first screen and press Enter.

3 Provide the following information:

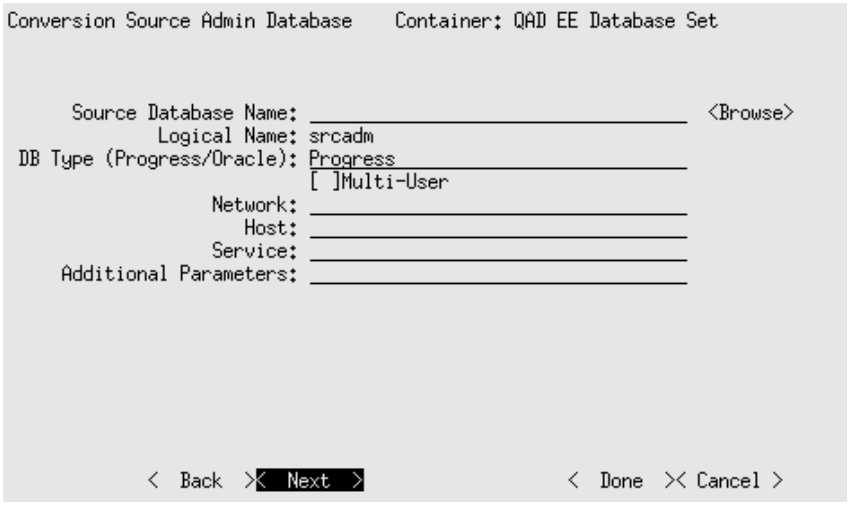
- Select Yes to all Create prompts.
- Source `qaddb` database name is the full path of the source production database.

Fig. 3.13
Conversion Source Database Screen



- Source `adm` database name is the full path of the source admin database.

Fig. 3.14
Conversion Source Admin Database Screen

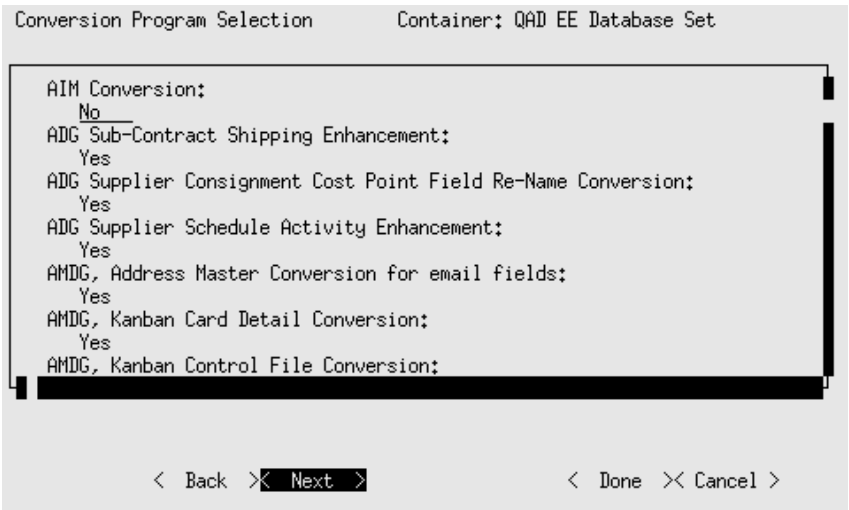


Conversion Program Selection

1 Select the needed conversion functions.

Note The available list of conversions depends on the source database version.

Fig. 3.15
Conversion Program Selection



2 Select Next.

3 If the database being converted is before eB2.1, some domain information must be set up. Tab to Next, press Enter, and proceed to Domain Conversion.

Domain Conversion

All QAD Enterprise Edition databases require a minimum of two domains. One - the system domain - is created when you install the QAD Enterprise Edition target version. For converted databases, a production or primary domain is created during the conversion process.

In this process, the conversion routines make assumptions based on information in the source database. If there are no connection records (dc_mstr) entries for other databases in the source database, a new connection master is created and the production domain is assigned to this database.

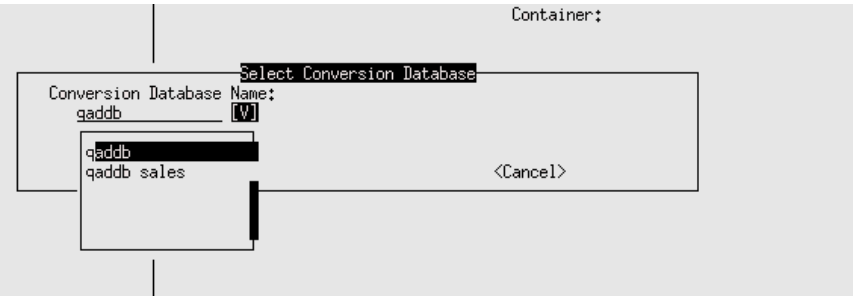
If dc_mstr entries for other databases exist, you are first asked to choose the current database from the list. You must then assign a primary domain for this database. After the primary domain is assigned, you must assign a domain to each of the other databases with dc_mstr records. These domains are then required to support future connections to these other databases.

Important When you convert the other databases, all the domain names must be identical. For example, if you assign domain names of sales, production, and distribution to databases named db1, db2, db3 respectively in the first database you convert, the same domains for the same databases must be used when converting subsequent databases.

Multiple Databases – More than One dc_mstr

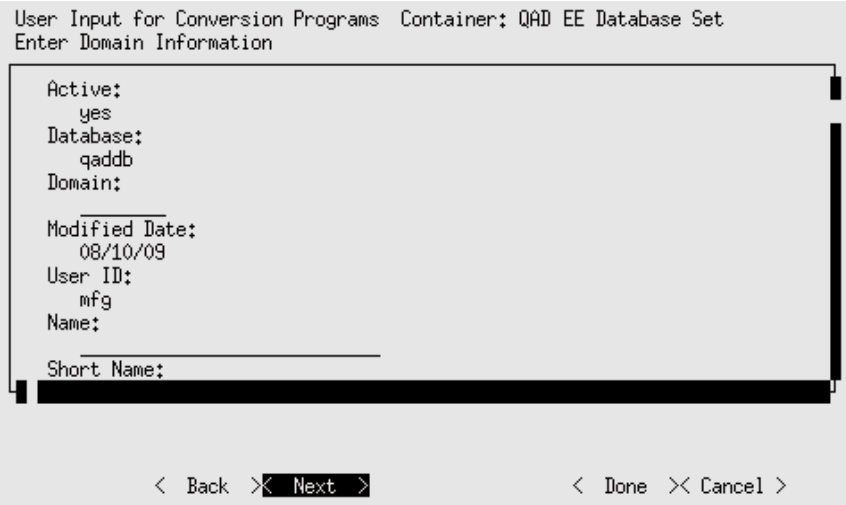
1 If the source database contains more than one database connection master, the first screen to display is the Select Conversion Database screen. Select the database you are converting and press OK.

Fig. 3.16
Select Conversion Database



2 You are prompted to create a domain for each database with a connection record in alphanumeric order.

Fig. 3.17
Creating a Database Domain - Multiple Databases



3 The master domain screen displays for each database identified with a dc_mstr in the conversion database. Enter values based on the following field descriptions:

- Active: Indicate whether this domain is currently active.

  Yes (the default): This domain can be associated with users in User Maintenance (36.3.1) and specified at log-in.

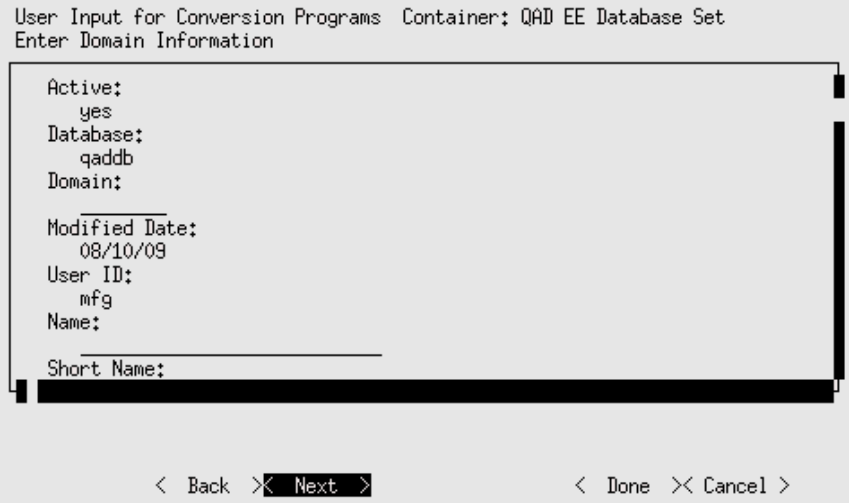
  No: This domain is not active in the current database.
- Database: The logical name of the database displays but cannot be modified.
- Domain (Code): Enter a unique code for each database identifying a specific domain.
- (Domain) Name: Enter a descriptive name to associate with this domain (up to 28 characters). This name must be unique within a database and across connected databases. This name displays in the lookup associated with domain fields and on various reports and inquiries, as space permits.
- (Domain) Short Name: Enter a brief name (up to 14 characters) to associate with this domain. This name must be unique within a database and across connected databases. The domain short name displays in the program title bar in the character interface based on the setting of Header Display Mode in Security Control (36.3.24).
- Domain Type: This displays the default value.

4 Click OK when you finish.

Single Database – No dc_mstr

1 If this database has no dc_mstr records, the Select Conversion Database screen displays with the logical database name displayed. Use the domain entry screen to create the required dc_mstr for the conversion (target) database.

Fig. 3.18
Creating a Database Domain - Single Database



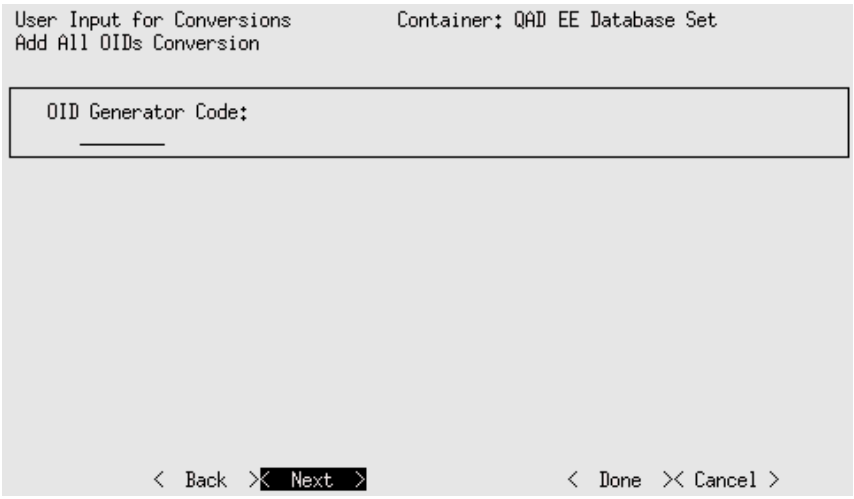
2 Enter the required information. The previous section describes the required fields.

3 Choose Next when you finish.

OID Generator Value

The next screen displayed is the OID Generator code (Figure 3.19).

Fig. 3.19
OID Generator Code



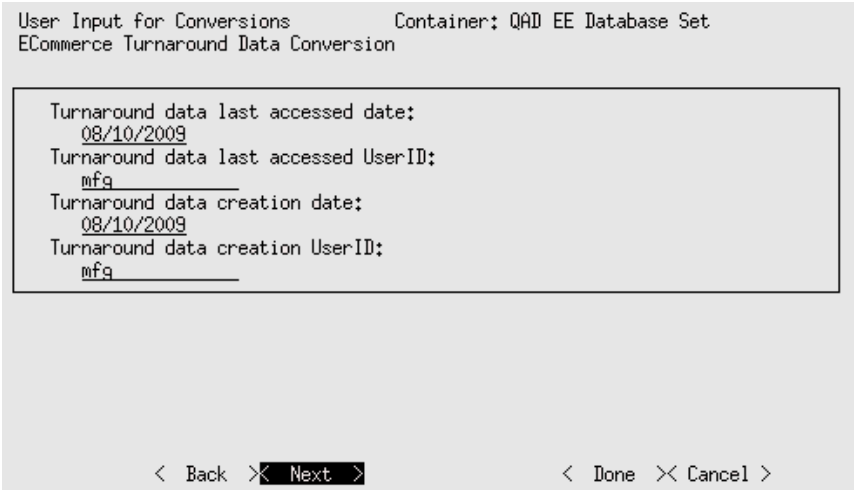
Enter a value for the OID Generator Code.

EDI eCommerce Conversions

The conversion prompts for eCommerce turnaround data if appropriate. Use the screen and field descriptions to complete this dialog.

Note This screen only appears if you have selected ECommerce Data Conversion in the Conversion Functions screen.

Fig. 3.20
eCommerce Turnaround Data Conversion

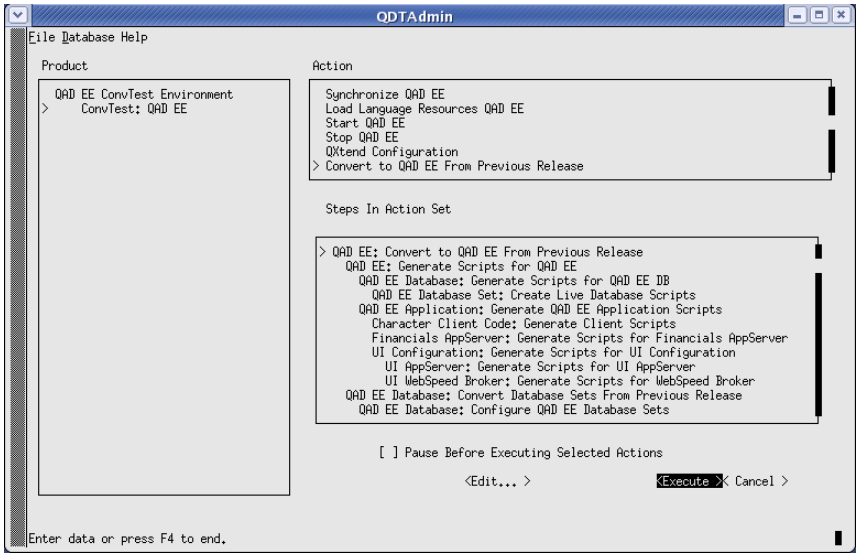


eCommerce turnaround data is any data in an ECommerce document that the system does not use. The data is, instead, stored in a side table and placed in the related export document when it is sent.

Execute Conversion

1 Tab to Execute and press Enter to start the conversion as shown in Figure 3.21.

Fig. 3.21
Starting the Conversion



The conversion will proceed until finished or an error occurs. No further input is required until the conversion ends.

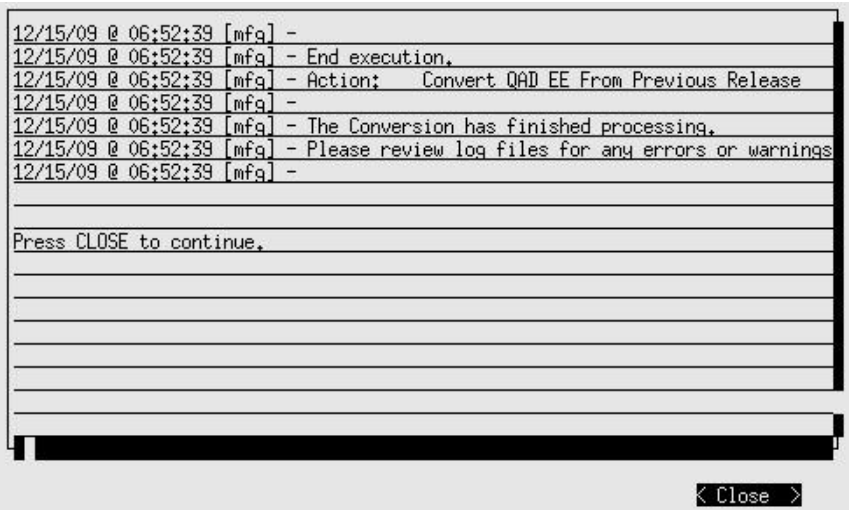
2 When the conversion finishes, quit QDT.

Note Use the Pause Before Executing Selected Actions check box to prompt you before executing each step. This allows you to make backups (snapshots) during the conversion. Refer to Appendix G, “Snapshots,” for further information. QAD does not recommend using this option for a live production conversion.

Conversion Validation

The following screen should appear in QDT Admin once the conversion has completed.

Fig. 3.22
QDT Admin Screen After Conversion



Review and check the following log files for errors:

- `<QDT install dir>/logs/qdtadmin.log`
- `<QDT install dir>/logs/qdtadmin001.log`
- `...`
- `<QDT install dir>/logs/qdtadminxxx.log`

Refer to Appendix F, “Log Files,” for information about log files.

Enter License Codes

1 Navigate to the scripts directory and run the client script for the desired language.

```
cd <QDT install directory>/scripts/<envname>/scripts
./client.<envname>
```

2 Log into the system using the following entries:

- User: `mfg`
- Password: [blank]

- Domain: QAD

3 When prompted, select Add and enter the applicable license codes.

Conversion Execution Troubleshooting

For information on troubleshooting conversion problems, see Appendix E, “Conversion Troubleshooting,” on page 153.

Chapter 4

Post-conversion

This chapter describes the validation and setup activities following database conversion.

Introduction 48

Overview 48

Post-conversion Utilities 48

Data Validation 49

Static Data Validation 56

Mandatory Post-conversion Setup 56

Optional Post-conversion Setup 58

Introduction

This chapter describes the validation and setup activities following database conversion.

Note This chapter only applies when converting from a pre-Enterprise Edition version to Enterprise Edition. It is not required when upgrading an existing Enterprise Edition installation.

Overview

Post-conversion has the following objectives:

- Test and validate a converted database.
- Prepare a converted database for a go-live implementation.

Post-conversion activities consist of the following major tasks:

- Post-conversion Utilities
- Post-conversion Data Validations
- Post-conversion Reports
- Process Flow and Static Data Validation
- Static Data Validation
- Mandatory Post-conversion Setup
- Optional Post-conversion Setup

Post-conversion Utilities

Fixed Assets Migration Utility (32.25.2 - facvmt.p)

You should only run this utility if you are upgrading from MFG/PRO 8.6E or 9.0 and using the Fixed Assets module. See Appendix B, “Running the Fixed Assets Migration Utility,” on page 123 for more information.

Table Extension Domain Conversions - Part 2

This utility completes the conversion of supplier lot data previously held in table extension records. You must must run the appropriate menu option for each active domain in the system.

If you are upgrading to QAD Enterprise Edition from eB2 or eB2.1 with any Service Pack before SP 6, run this utility from menu 5.25.7 (`utvdtr92.p`).

If you are upgrading to QAD Enterprise Edition from eB2.1 SP 6 or higher, run this utility from menu 5.25.8 (`utvdt92b.p`).

Warning Execute one menu item only.

Sales Order Balance Update (36.16.23.6 - utcsob.p)

This utility recalculates the open sales order balance by customer and updates the outstanding balance on the customer master. The balance is checked when a credit check is performed and the Include Sales Order flag is checked.

Document Credit Terms Update (36.25.83- uxdoccrterms.p)

This utility updates various documents (calls, invoice history, sales orders, purchase orders, service contracts, price list detail, sales quotes) of your choosing with a user-specified credit terms code. This credit terms code is applied to documents meeting the selection criteria that have blank or invalid credit terms assigned to them. Blank or invalid credit terms are not converted.

Run the Document Credit Terms Update utility for each domain and separately for each document type.

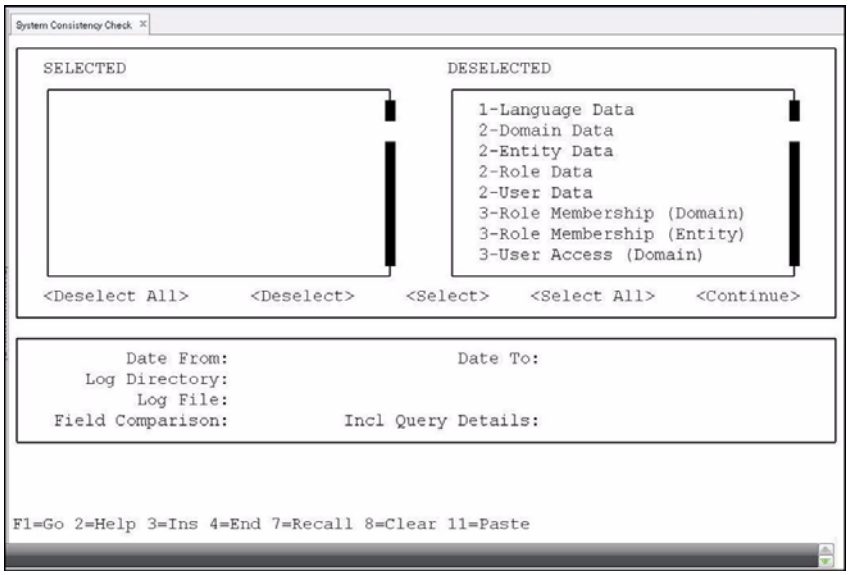
Data Validation

System Consistency Check (36.16.23.1 - utsyscon.p)

This utility ensures that master data, which is maintained in the Financials and replicated to the operational modules, remains consistent. For example, if there is a record in the operational part of the system, there must be a record in the Financials and vice versa. This check ensures that the conversion created new records in the Financials for every master record in the original version.

1 Start the System Consistency Check utility.

Fig. 4.1
System Consistency Check



The resulting report contains a summary and details of any inconsistencies.

Fig. 4.2
System Consistency Check Report

```
------------------------------------------------------------------------------------------
2009-05-12>02:49:21: REPORT STATISTICS
                              White Missing * MFGPRO Missing * Inconsistent
        ok 1-Language Data                0 *              0 *            0
        ok 2-Domain Data                  0 *              0 *            0
        ok 2-Entity Data                  0 *              0 *            0
        ok 2-Role Data                    0 *              0 *            0
        ok 2-User Data                    0 *              0 *            0
        ok 3-Role Membership (D           0 *              0 *            0
        ok 3-Role Membership (E           0 *              0 *            0
        ok 3-User Access (Domai           0 *              0 *            0
        ok 3-User Access (Entit           0 *              0 *            0
       ERR 4-Menu Data                    0 *              1 *            0
       ERR 5-Address Data                 0 *              8 *            0
        ok 5-Country Data                 0 *              0 *            0
        ok 5-Credit Terms Data            0 *              0 *            0
        ok 5-Credit Terms Stage           0 *              0 *            0
        ok 5-Currency Data                0 *              0 *            0
       ERR 5-Exchange Rate Data           9 *              9 *            0
        ok 5-GL Calendar Data             0 *              0 *            0
        ok 5-GL Period Data               0 *              0 *            0
        ok 5-Rounding Method Da           0 *              0 *            0
        ok 6-Cost Center Data             0 *              0 *            0
        ok 6-Daybook Data                 0 *              0 *            0
        ok 6-GL Account Data              0 *              0 *            0
        ok 6-GL Mask Data                 0 *              0 *            0
        ok 6-Project Data                 0 *              0 *            0
        ok 6-Sub-Account Data             0 *              0 *            0
        ok 6-Tax Data                     0 *              0 *            0
        ok 7-Customer Data                0 *              0 *            0
       ERR 7-Employee Data               80 *             12 *            0
       ERR 7-End User Contact D           9 *              0 *            2
       ERR 7-End User Data                1 *              0 *            0
        ok 7-Supplier Data                0 *              0 *            0
       ERR 8-AP Matching Data            82 *              0 *            0
       ERR 8-AR Invoice Data              0 *             30 *            0
       ERR 8-Posting Data                 0 *             83 *            0
------------------------------------------------------------------------------------------

2009-05-12>02:49:21: >>>> Finished System Consistency Check <<<<
```

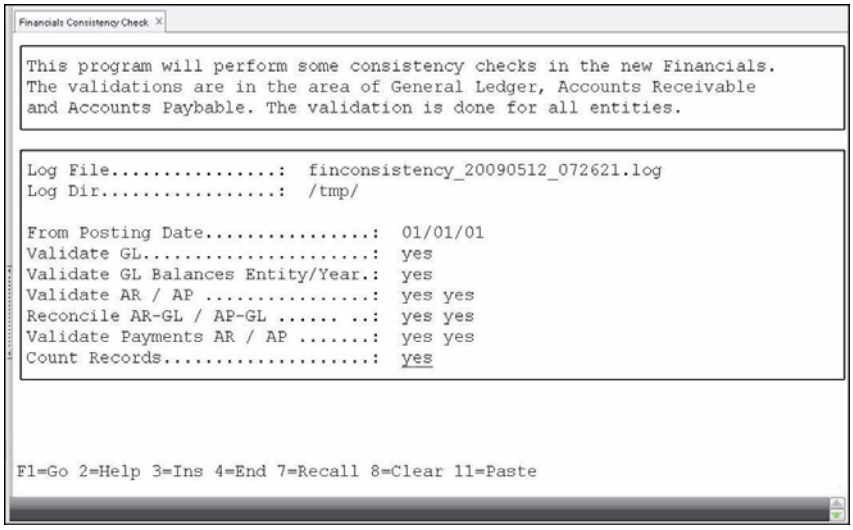
2 Analyze any inconsistencies.

Financials Consistency Check (36.16.23.2 - utfincon.p)

This utility validates the consistency, completeness and integrity of converted financial data. It also checks that all table structures in the new Financials are in order and their links are correct. This utility validates some balances (AR, AP) to ensure transactions balance across entities with regards to debits and credits and that the sub-ledgers balance to GL.

1 Start the Financial Consistency Check utility.

Fig. 4.3
Financials Consistency Check



The resulting report contains the details of any inconsistencies.

Fig. 4.4
Financials Consistency Check Report

```
2009-05-11>04:03:34: >>>> START COUNTING <<<<
        Posting#= 1410
        PostingLine#= 3099
        PostingHist#= 477
        QPostingLine#= 0

        DInvoice#= 42
        DInvoiceMovement#= 48
        QDInvoiceMovement#= 0
        DHist#= 28

        CInvoice#= 48
        CInvoiceMovement#= 85
        QCInvoiceMovement#= 0
        CHist#= 29

        CDocument#= 36
        CCollection#= 8
        CDocumentInvoiceXref#= 37
        CDocInvoiceXrefStage#= 0

        DDocument#= 10
        DCollection#= 6
        DDocumentInvoiceXref#= 10
        DDocInvoiceXrefStage#= 0
2009-05-11>04:03:34: >>>> END COUNTING <<<<

2009-05-11>04:03:34: >>>> Start Validation of GL (01/01/00) <<<<<
        POSTING;590842;Posting not in balance for CC amounts;04/26/09;mfg
        PostingLine;591858;Transactions without sub-accounts found on account which is enabled for sub-accounts; 04/26/09;mfg
        PostingLine;591872;Transactions without sub-accounts found on account which is enabled for sub-accounts; 04/26/09;mfg
        PostingLine;590887;Transactions on costcentres found on account which is disabled for costcentres; 04/26/09;mfg
        PostingLine;590987;Transactions on costcentres found on account which is disabled for costcentres; 04/26/09;mfg
        PostingLine;591885;Transactions without sub-accounts found on account which is enabled for sub-accounts; 04/26/09;mfg
        PostingLine;591958;Transactions without sub-accounts found on account which is enabled for sub-accounts; 04/26/09;mfg
        PostingLine;591946;Transactions without sub-accounts found on account which is enabled for sub-accounts; 04/26/09;mfg
        PostingLine;590900;Transactions on costcentres found on account which is disabled for costcentres; 04/26/09;mfg
        PostingLine;591227;Transactions on costcentres found on account which is disabled for costcentres; 04/26/09;mfg
        PostingLine;591203;Transactions on costcentres found on account which is disabled for costcentres; 04/26/09;mfg
```

2 Analyze any inconsistencies.

Operational Account Structure Validation (36.9.20 - uxacval.p)

Use this utility to generate a report of all of the places in the system where default GL accounts, sub-accounts, and cost centers can be specified and are used to generate GL transactions in the operational modules (for example, Product Line Maintenance, Department Maintenance, and so on). The report indicates any invalid combinations.

Invalid combinations can exist for a number of reasons, but more typically occur when accounts, sub-accounts, cost centers are made inactive. Such invalid settings can result in GL transactions that cannot be posted to the Financials.

1 Run the Operational Account Structure Validation utility for each active domain.

Fig. 4.5
Operational Account Structure Validation Menu

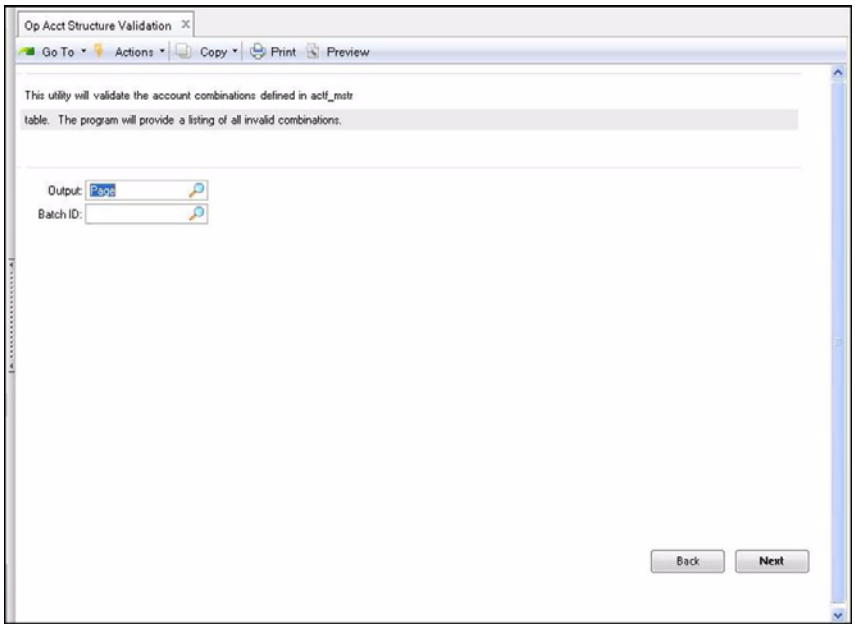


Fig. 4.6
Operational Account Structure Validation Report

2 Analyze the output from running this utility.

Correct invalid combinations to prevent the generation of invalid GL transactions.

Post Conversion Integrity Check (36.16.23.3 - acinckrp.p)

Use this utility to produce a post-conversion report to assess the status of the financial transaction data after conversion.

Note You must run the corresponding pre-conversion report (`gpinckrp.p`) before conversion. Otherwise, the contents of the report will be inaccurate.

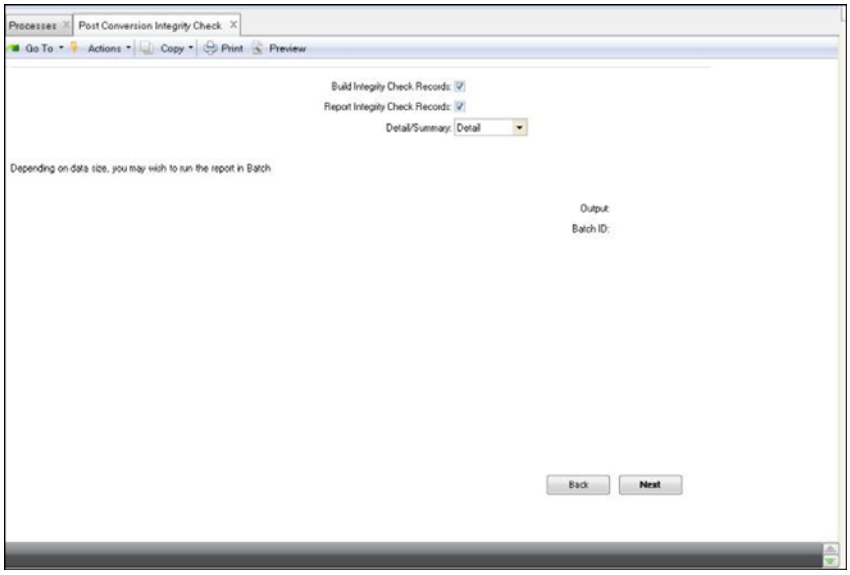
The report is split across domains with one section per domain. Each of the domain sections is broken down into the sub-sections listed below:

- GL Transaction Integrity
- AR Transaction Integrity
- AP Transaction Integrity

GL Integrity has two modes. Summary mode only displays accounts containing differences. Detail mode shows all accounts, even when there are no differences between pre-conversion and post-conversion balances.

1 Run the Post Conversion Integrity Check utility.

Fig. 4.7
Post-conversion Integrity Check Menu



The report should show that the data is in the same condition as before conversion with the exception of any unbalanced GL transactions from the current period, which the conversion will balance.

Fig. 4.8
Post-conversion GL Integrity (Summary)

Domain: Domain1

GL Balance Validation

Entity	Account	Balance Preconv	Balance Postconv	Difference	Conversion Account
6000	440000	407.72	407.74	0.02	QAD-0086
6000	410000	100.00	100.05	0.05	QAD-0085
6000	430000	404.87	404.84	0.03	QAD-0086
6000	460000	301.99	309.00	8.99	QAD-0086
6000	480000	1024.00	1022.55	2.45	QAD-0086

Fig. 4.9
Post-conversion GL Integrity (Detail)

Domain: Domain1

GL Balance Validation

Entity	Account	Balance Preconv	Balance Postconv	Difference	Conversion Account
6000	000000	0.00	0.00	0.00	
6000	003500	0.00	0.00	0.00	
6000	022000	0.00	0.00	0.00	
6000	023000	0.00	0.00	0.00	
6000	050100	108,267.00	108,267.00	0.00	
6000	051100	721,164.70	721,164.70	0.00	
6000	053100	3,865,197.00	3,865,197.00	0.00	
6000	053900	0.00	0.00	0.00	
6000	054000	92,021.00	92,021.00	0.00	
6000	055000	123,326.00	123,326.00	0.00	
6000	056000	0.00	0.00	0.00	
6000	056100	114,251.00	114,251.00	0.00	
6000	056200	431,145.00	431,145.00	0.00	
6000	059000	0.00	0.00	0.00	
6000	070000	4,383,915.00	4,383,915.00	0.00	
6000	073000	0.00	0.00	0.00	
6000	075100	207,514.00	207,514.00	0.00	
6000	075200	0.00	0.00	0.00	

Fig. 4.10
Post-conversion AP Integrity

AR Account Validation

Entity	Account	Balance Preconv	Conv Acct Balance	Conversion Account	Balance Postconv	Difference
6000	240000	0.00	0.00	QAD-0062	0.00	0.00
6000	240001	0.00	0.00	QAD-0063	0.00	0.00
6000	240002	0.00	0.00	QAD-0064	0.00	0.00
6000	250003	0.00	0.00	QAD-0065	0.00	0.00
6000	250005	0.00	0.00	QAD-0066	0.00	0.00
6000	250007	0.00	0.00	QAD-0067	0.00	0.00
6000	250010	21,669.62	0.00	QAD-0068	21,669.62	0.00
6000	250012	0.00	0.00	QAD-0069	0.00	0.00
6000	250014	0.00	0.00	QAD-0070	0.00	0.00
6000	250017	0.00	0.00	QAD-0071	0.00	0.00
6000	250030	0.00	0.00	QAD-0072	0.00	0.00
6000	250032	0.00	0.00	QAD-0073	0.00	0.00
6000	250041	0.00	0.00	QAD-0074	0.00	0.00
6000	250042	0.00	0.00	QAD-0075	0.00	0.00
6000	250043	0.00	0.00	QAD-0076	0.00	0.00
6000	250044	12,936.00	0.00	QAD-0077	12,936.00	0.00
6000	250045	0.00	0.00	QAD-0078	0.00	0.00
Total		34,605.62	0.00		34,605.62	0.00

Fig. 4.11
Post-conversion AR Integrity

AP Account Validation

Entity	Account	Balance Preconv	Conv Acct Balance	Conversion Account	Balance Postconv	Difference
6000	440000	-36,407.68	0.00	QAD-0025	-36,407.68	0.00
6000	440001	0.00	0.00	QAD-0026	0.00	0.00
6000	440002	-2,315,203.57	0.00	QAD-0027	-2,315,203.57	0.00
6000	465012	-830.00	0.00	QAD-0034	-830.00	0.00
6000	465013	0.00	0.00	QAD-0035	0.00	0.00
6000	465014	-231,902.19	0.00	QAD-0036	-231,902.19	0.00
6000	465015	0.00	0.00	QAD-0037	0.00	0.00
6000	465016	0.00	0.00	QAD-0038	0.00	0.00
6000	465017	0.00	0.00	QAD-0039	0.00	0.00
6000	465018	-30,844.61	0.00	QAD-0040	-30,844.61	0.00
6000	465020	0.00	0.00	QAD-0041	0.00	0.00
6000	465030	0.00	0.00	QAD-0042	0.00	0.00
6000	465031	0.00	0.00	QAD-0043	0.00	0.00
6000	465032	-604.48	0.00	QAD-0044	-604.48	0.00
6000	465039	0.00	0.00	QAD-0051	0.00	0.00
6000	465040	-72,888.38	0.00	QAD-0052	-72,888.38	0.00
6000	465046	0.00	0.00	QAD-0058	0.00	0.00
6000	465047	0.00	0.00	QAD-0059	0.00	0.00
6000	465048	0.00	0.00	QAD-0060	0.00	0.00
6000	465049	0.00	0.00	QAD-0061	0.00	0.00
Total		-2,688,680.91	0.00		-2,688,680.91	0.00

2 Analyze the output produced by this utility.

Post-conversion Reconciliation Reports

AP Reconciliation Report (36.16.23.10 - appcrnrp.p)

Use this optional utility to the report pre- and post-conversion balances of open AP transactions.

The report displays base and transaction currency balances and identifies any discrepancies between the pre- and post-conversion values. The output options are Detail, which shows all open transactions, or Summary, which shows only unbalanced transactions. The report can be sorted by Supplier, Effective Date, or Transaction Type, and can be executed for a specific domain or all domains.

Note You must initially run the utility before beginning transaction processing within the AP module of the converted database. Otherwise, the contents of the report output will be inaccurate.

During the initial run of the utility, each transaction is flagged as reconciled or unreconciled. In subsequent runs of the utility, the balances for unreconciled transactions are updated to account for corrections made. The transactions are flagged as reconciled when appropriate.

AR Reconciliation Report (36.16.23.9 - arpcrnrp.p)

Use this optional utility to report the pre- and post-conversion balances of open AR transactions.

The report displays base and transaction currency balances and identifies any discrepancies between the pre- and post-conversion values. The output options are Detail, which shows all open transactions, or Summary, which shows only unbalanced transactions. The report can be sorted by Customer, Effective Date, or Transaction Type and can be executed for a specific domain or all domains.

Note You must initially run the utility before starting transaction processing within the AR module of the converted database. Otherwise, the contents of the report output will be inaccurate.

When the utility is first run, each transaction is flagged as reconciled or unreconciled. During subsequent runs of the utility, the balances for unreconciled transactions are updated to account for corrections made. The transactions are flagged as reconciled when appropriate.

Post-conversion Reports

Rerun the following reports and compare the pre- and post-conversion data:

- Trial Balance
- Balance Sheet (25.15.5.4)
- Income Statement (25.15.5.5)
- Customer Aging Analysis
- Supplier Aging Analysis
- Inventory Valuation as of Date
- Unmatched PO Receipts as of Date (5.13.10)
- Asset Owned Report (Fixed Assets Valuation)
- Open Sales Order Balances

The *QAD Enterprise Edition Financials User Guide* describes many of the reports in detail.

Process Flow and Static Data Validation

Following conversion, QAD recommends that you process the entire inbound, outbound, and manufacturing transaction life cycles. Review the results at each step to ensure that they are as expected.

Static Data Validation

Following conversion, take a sampling of static data (for example, customer and supplier addresses) and verify that everything is as expected.

Mandatory Post-conversion Setup

Warning For eB2.1 and later, county and state data must be reviewed before conversion to eliminate similar or inconsistent entries. Pre- and post-conversion state and county data must be compared to verify data integrity. Otherwise, the conversion may make state and county data unusable.

Structured Reports

Structured reports are run following conversion and reviewed to verify that they are in the desired form.

Refer to the *QAD Enterprise Edition Financials User Guide* for more information regarding structured reports.

Balance Sheet (25.15.5.4)

The Balance Sheet Report runs based on report structures implemented using the Budget function. The system constructs the balance sheet based on the accounts specified in the report structure. All other accounts are excluded. Refer to the *QAD Enterprise Edition Financials User Guide* for more information.

Run the Balance Sheet report and check the resulting report for the correct structure.

Income Statement (25.15.5.5)

The Income Statement Report runs based on report structures implemented using the Budget function. The system constructs the income statement based on the accounts you specify in the report structure. All other accounts are excluded. Refer to the *QAD Enterprise Edition Financials User Guide* for more information.

Run the Income Statement Report and check the resulting report for the correct structure.

Invoice Status Codes (36.1.11)

The conversion only creates three invoice status codes. Consider the number of codes needed for receiver and financial matching; decide at what point in the process invoices are deemed approved, released for payment, and so on; and set up new status codes and assign them to suppliers (and customers) as appropriate.

Run the Invoice Status Code report and check the resulting report for the correct structure. Rename or replace the codes provided by the conversion.

Refer to the *QAD Enterprise Edition Financials User Guide* for more information.

Daybooks

Daybooks are mandatory with the new Financials.

Daybook

1. Decide if any further reporting granularity is required.
2. Add new daybooks as necessary and assign to transaction types using Default Daybook Maintenance.
3. Optionally, delete or deactivate any unused daybooks.

Daybook Sets

The conversion creates two Daybook Sets, an AP Daybook Set and an AR Daybook set. You can optionally use additional daybook sets to facilitate multiple invoice/credit note number ranges. Daybook sets can be defined for an entire domain or for individual sites. In the latter case, the daybooks sets must be assigned to the desired sites.

Refer to the *QAD Enterprise Edition Financials User Guide* for more information.

Security

Roles and Permissions

Set up notification roles, including e-mail notification, and associated users for customers, suppliers, end users, and engineers. Roles and permissions are similar to User Group Maintenance and Menu Security Maintenance in older versions for granting access by role.

For more information on roles and permissions, refer to the *QAD Enterprise Edition Security and Controls User Guide*.

Update Domain/Entity/User

Grant access by user to entities and domains.

For more information on updating domains, entities, and users, refer to the *QAD Enterprise Edition Security and Controls User Guide*.

Tax Periods

If the Create Entity Tax Periods parameter was set in the Conversion Parameters utility, the Tax Periods were created from the GL Periods during the conversion. However, if this parameter was not set, Tax Periods must be created manually within the application

Create tax periods from the GL calendar. Otherwise, skip to “Reporting Periods”.

For more information, refer to the *QAD Enterprise Edition Financials User Guide*.

Reporting Periods

Reporting periods are required if the new Financials budgeting is used. If GLRW and budgets are used, this is not required. Reporting periods must be set-up to run structured reports.

If the new Financials budgeting is not used, skip to “Profiles”.

Report periods mark a specific time span for producing budget reports. They are independent of GL periods and tax periods, and can span multiple GL periods across multiple entities.

Set up reporting periods.

For more information, refer to the *QAD Enterprise Edition Financials User Guide*.

Profiles

4. The conversion creates the required profiles for each element in a GL transaction. Review these profiles to verify they are linked to the correct shared set.
5. Additionally, the account codes for the specialized account profiles must be reviewed to ensure that the correct default sub-account, cost center, and project codes (as well as their associated linked shared sets) are set for each specialized account.

Configure Daemons

The History and Balance daemons are mandatory for correct operation of the system. The budget and replication daemons should also be reviewed for correct configuration. For more information on system daemons, see *User Guide: QAD System Administration*.

Optional Post-conversion Setup

Accounting Layers

The conversion creates transient and official layers. Add desired additional layers such as adjustments, internal reporting, and so on.

Cash Groups

Cash groups are used to group GL accounts for cash flow reporting purposes. Define new cash groups for petty cash (as opposed to the bank accounts for AR, AP, and Payroll).

Assign the cash groups to the required GL account codes.

Report Structures

6. Define any desired new reports.

Taxes

QAD Enterprise Edition has a new concept called Tax Groups. With tax groups (and boxes), you can set up tax reporting. During normal transaction processing, the application populates the tax transaction with the tax group defined against the tax rates.

Tax Codes

Define any new tax codes needed to take advantage of new functionality such as Suspended/Delayed Taxes.

Tax Boxes

Tax boxes contain the individual elements of a transaction (for example, Tax Amount or Tax Base Amount, as they are reported in official returns to the authorities).

Tax Groups

A tax group contains one or more tax boxes.

Update Tax Rates

Add any new tax groups defined to the applicable tax rate.

Posting Tax Group Update

The conversion creates one default tax group. This default is assigned to transactions until the required setup is defined.

This utility updates existing tax transactions with new tax groups added to a tax rate. This allows transaction processing to start without the definition of tax groups/boxes.

Supplementary Analysis Fields

Define any SAF codes and assign to accounts, sub-accounts, and cost centers if this level of detail analysis is desired. Refer to the *QAD Enterprise Edition Financials User Guide* for more information.

Customer Credit Checking

QAD Enterprise Edition significantly enhances customer credit checking. To take full advantage of all the new capabilities in this area, review the credit checking parameters to ensure they are set according to the desired behavior for each customer.

Customer/Supplier Payment Statuses

Review and modify or create additional payment statuses for customers and suppliers.

Customer/Supplier Control Accounts

Review and modify or create additional the control accounts for customers and suppliers.

Additional Profiles

Review and modify any new profiles set up for customer and supplier credit notes and prepayments.

Chart of Account (COA) Mask

Review and modify or create additional COA Mask records. Additional COA Mask records may be required for GL account codes created by the conversion when those accounts fall outside of current GL validation ranges (for example, replacement accounts created within the GL Account Type Utility, or accounts defined within the Conversion Parameters Utility).

Chapter 5

Upgrading QAD Enterprise Edition

This section describes how to upgrade the components of an existing QAD Enterprise Edition installation.

Overview 62

Prepare Environment 62

Prepare Source Databases 62

Back up Environment 62

Install and Configure Software 63

Upgrade Setup 67

Execute Upgrade 71

Upgrade Validation 72

Overview

This section provides detailed instructions on how to perform a QAD Enterprise Edition Upgrade. This process consists of the following tasks:

- Prepare Environment
- Prepare Source Databases
- Backup Environment
- Install and Configure Software
- Upgrade Setup
- Execute Upgrade
- Upgrade Validation

Note Only use the Upgrade option to upgrade an existing Enterprise Edition environment to the latest release. Do not use this option if your current environment is not Enterprise Edition.

Prepare Environment

The following environment variables must be defined before starting QDT:

- `export DLC=<Progress install directory>`
- `export CATALINA_HOME=<Tomcat install directory>`
- `export JAVA_HOME=<Java install directory>`

Note QAD recommends defining these variables in a script and running it when the user logs into the system.

Prepare Source Databases

If the Progress version of the databases (qaddb and qadadm) differ from the Progress version to be used in the new QAD production environment, the databases must be converted to the new Progress version.

Note

- Refer to the *Progress OpenEdge Data Management: Database Administration* manual for more information.
- Unlike a pre-Enterprise Edition to Enterprise Edition conversion, pre-conversion work is not required to upgrade an existing Enterprise Edition database.

Back up Environment

You must back up the existing environment before attempting an upgrade. To back up the existing QDT installation directory, use the following steps:

1. Exit any QDT sessions.
2. Back up the QDT installation directory.

Back up the QAD Enterprise Edition environment to be upgraded.

1. Stop the QAD Enterprise Edition environment by running the following script:
   `<QDT install directory>/envs/<env name>/scripts/stopenv.<env name>`
2. Back up the QAD EE installation directory (including the database directory if it is not included within the QAD Enterprise Edition installation directory structure).

Back up the following folders under the Tomcat WEB-APPS directory.

1. Stop the Tomcat server by running the following script:
   `$CATALINA_HOME/bin/shutdown.sh`
2. Back up the following folders under `$CATALINA_HOME/webapps`:
   - `epub`
   - `search`
   - `qadhome`
   - `qadui`

Back up the `ubroker.properties` file:

`$DLC/properties/ubroker.properties`

Install and Configure Software

Migrating QDT to Progress 10.2A

QAD 2010 requires that Progress version 10.2A is installed. To update QDT to use Progress 10.2A during the upgrade process, use the following steps:

1. Shut down the existing environment using the following script:
   `<QDT install dir>/envs/<env name>/scripts/stopenv.<env name>`
2. Shut down the Progress 10.1C admin server:
   `proadsv –stop`
3. Install Progress 10.2A (refer to the Progress documentation).
4. After Progress 10.2A is installed, start the admin server:
   `proadsv –start`
5. Update the `$DLC` environment variable to use Progress 10.2A.
6. Start QDT.
7. Select Edit|System Default Settings from the QDT Menu.
8. Change the Progress Home (DLC) field to the Progress 10.2A home directory.
9. If the name server port for your Progress 10.2A install is different than in the previously used version, update the port number here.
10. Click OK. At this point the internal references in QDT are adjusted so that clicking the Admin button will start QDTAdmin using the new Progress version.

11 Click Admin to start QDTAdmin.

12 In the Products window, select the environment to update.

13 Select Edit System Defaults in the Action window.

14 Select adminserver in the browse to edit the values.

15 Change the Version field to 10.2A.

16 Click the Browse button or type the Progress home directory in the fill-in.

17 Click the Done button. There is a long pause while the system is updated.

18 Select NameServer from the system components.

19 Make sure the port number is correct.

20 Quit QDT and check the following files to ensure the name server port was updated correctly.

   a In `<QDT install dir>/envs/ <env name>/configs` check:
   - `server.xml`
   - `cbserver.xml`

   b In `<QDT install dir>/envs/ <env name>/scripts` check:
   - `startqadfin*.ksh`
   - `stopqadfin*.ksh`
   - `checkqadfin*.ksh`

Install QDT

Install the latest QDT version that includes the upgrade and conversion software. Refer to *Installation Guide: Progress Database - Enterprise Edition* for more information on QDT installation.

Note For upgrades, you must install QDT over the existing QDT installation (it must be installed to the same directory location).

Install QAD Enterprise Edition

Use the latest version of QDT (which includes the upgrade and conversion software) to install QAD Enterprise Edition.

Refer to *Installation Guide: Progress Database - Enterprise Edition* for detailed QAD Enterprise Edition installation information.

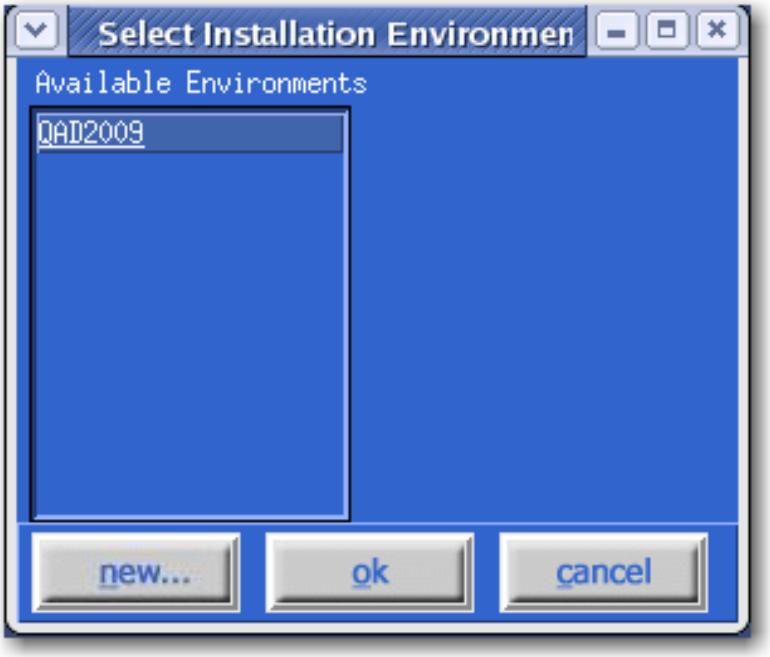
Note

- QDT and QDTAdmin must run in an X Window/VNC Server.
- The QAD Enterprise Edition patch level may require updating. Refer to *Installation Guide: Progress Database - Enterprise Edition* for more information.
- Do not run the Configure QAD EE option during an upgrade. It is only used when doing a clean install with no upgrade planned.

- For upgrades, QAD Enterprise Edition must be installed over the existing QAD EE environment. In other words, QAD Enterprise Edition must be installed to the same directory location as the version being upgraded.

During installation you must select the environment you want to update from the list of available environments.

Fig. 5.1
Available Environment Selection Panel

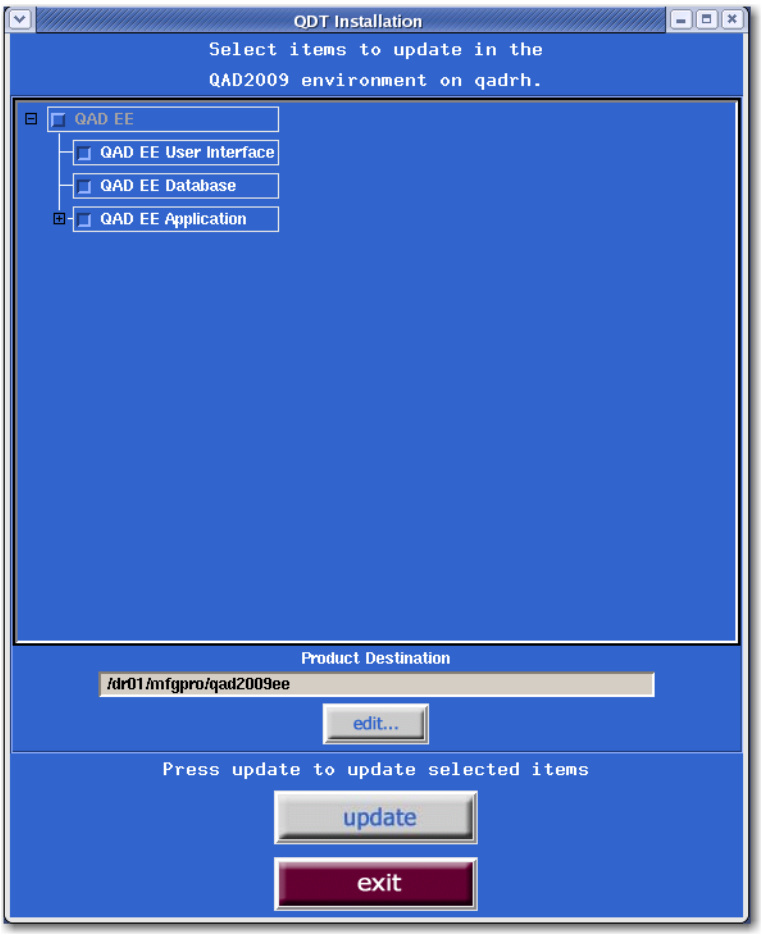


By default, the following options are not selected:

- QAD EE User Interface
- QAD EE Database
- QAD EE Application

You must ensure they are selected (see the figure below).

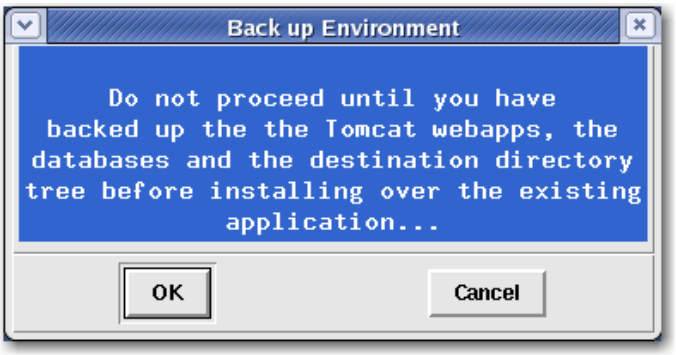
Fig. 5.2
Installation Option Window



You must also ensure that the Product Destination path is set to the directory location of the version of QAD Enterprise Edition being upgraded.

A warning appears to back up the current install. Select OK to continue.

Fig. 5.3
Backup Warning Pop-Up



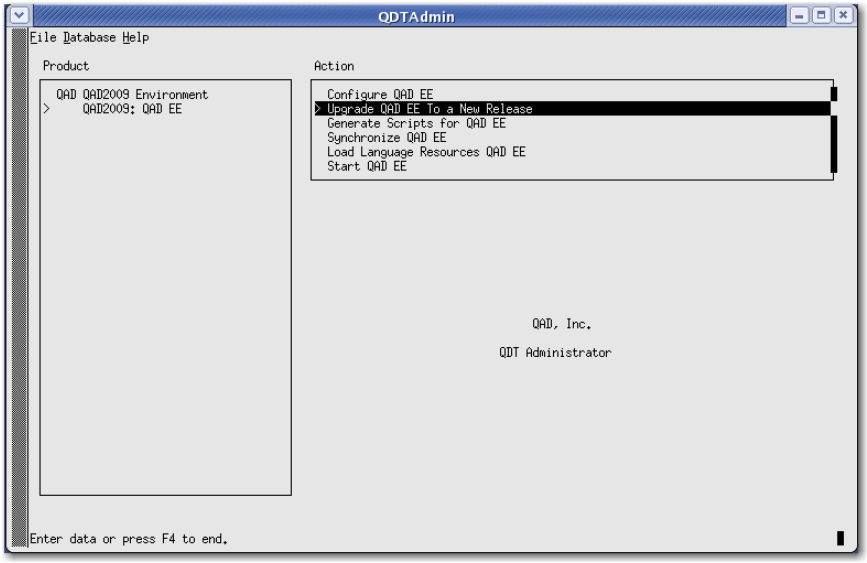
Upgrade Setup

The following screens have default settings. QAD recommends reviewing these to verify they are correct before beginning the upgrade.

Upgrade QAD Enterprise Edition to a New Release

1 Navigate the following screen within QDTAdmin. Press enter on the left-hand menu item that corresponds to the environment name.

Fig. 5.4
Select Environment for Upgrade



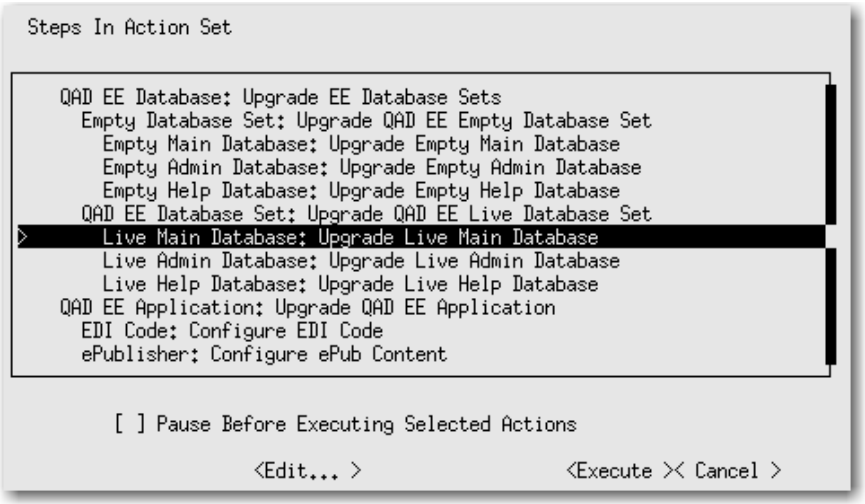
2 A sub-menu option named QAD EE displays. Press Tab on this menu item. This brings your cursor over to the right-hand menu.

Note Do not select the Configure QAD EE option. Instead, select Upgrade QAD EE to a New Release as shown above.

Upgrade Live Main Database

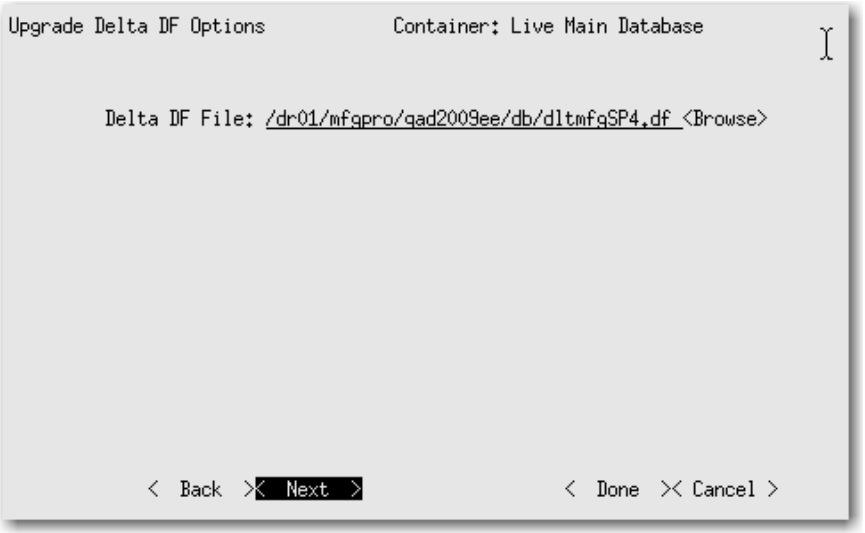
1 Select Live Main Database: Upgrade Live Main Database as shown below.

Fig. 5.5
Live Main Database: Upgrade Live Main Database



2 The following screen allows you to specify the path to the Delta DF file to use during the upgrade. This is automatically set by default.

Fig. 5.6
Specify the Delta DF File Path



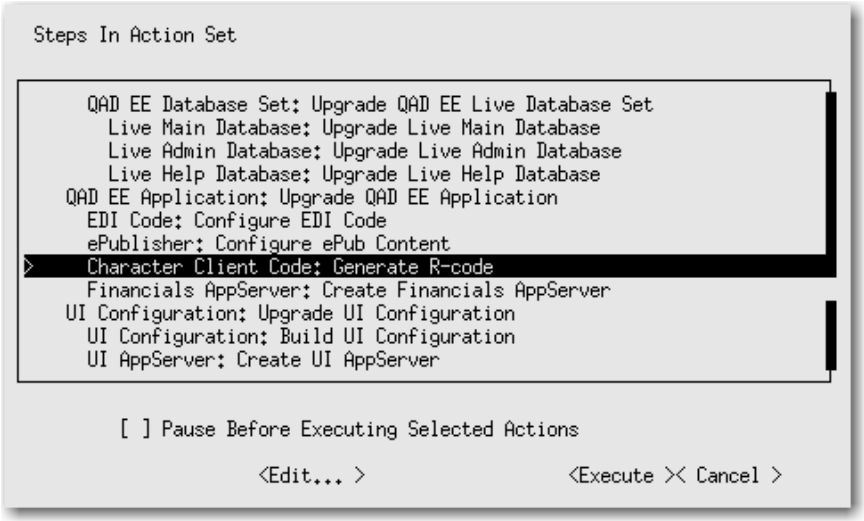
Note Do not change this path unless you are using a customized Delta DF file.

3 Tab to Next and press Enter. This returns to the main conversion option screen.

Character Client Code: Generate R-Code

1 Select Character Client Code: Generate R-Code and press Enter.

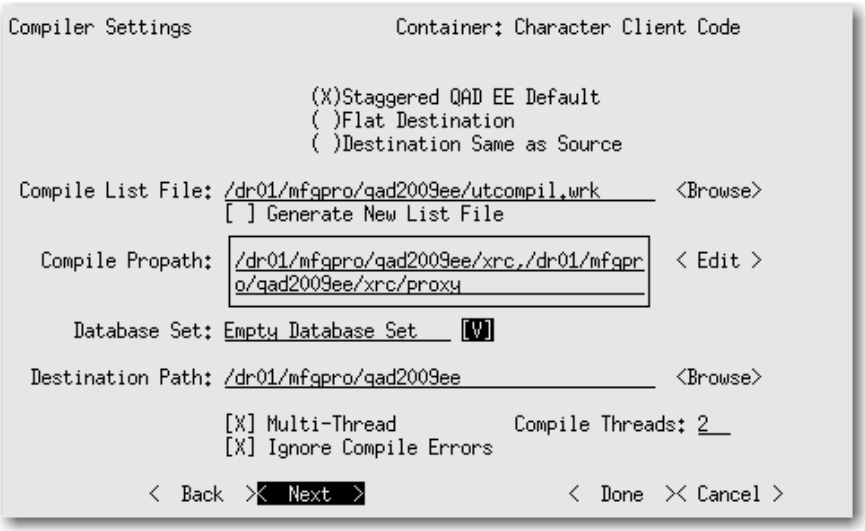
Fig. 5.7
Character Client Code: Generate R-Code



2 Change the Compiler Settings. Use the spacebar to select and configure the following settings:

- Multi-thread: Enabled
- Ignore Compile Errors: Enabled
- Compile Threads: Enter the number of CPU cores.

Fig. 5.8
Compiler Settings

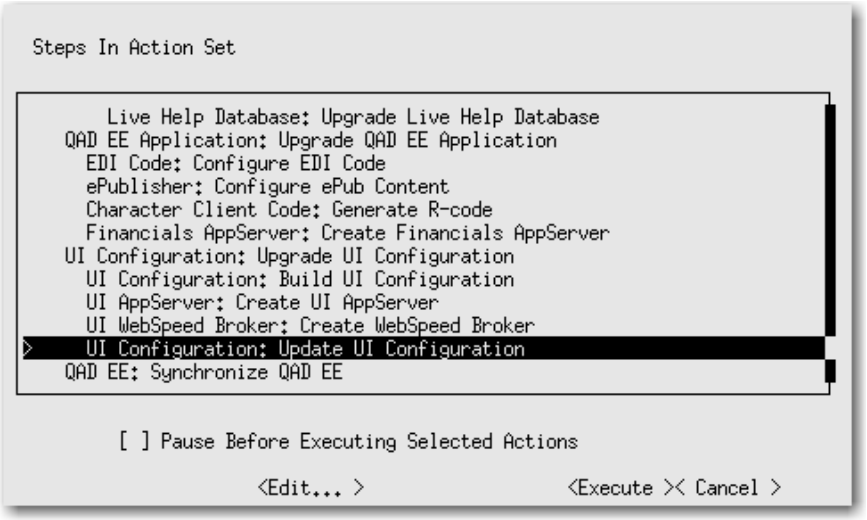


3 Tab to Next and press Enter. This returns to the main conversion option screen.

Update UI Configuration

1 Select UI Configuration: Update UI Configuration.

Fig. 5.9
UI Configuration: Update UI Configuration



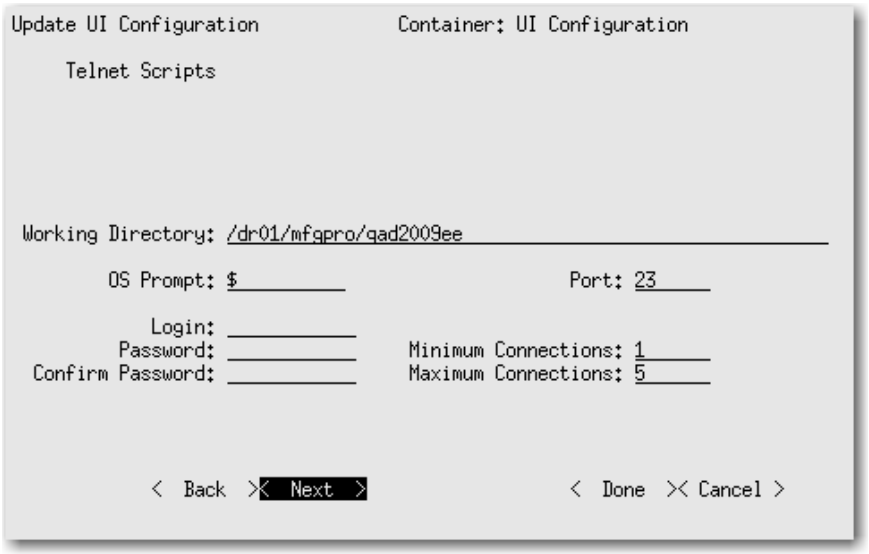
2 In the UI configuration screen, make the following changes:

- Working Directory: *<defaulted-in for your specific environment>*
- Login: `mfg`
- Password: *<mfg user's password>*
- Confirm Password: *<mfg user's password>*

Note

- The MFG login is the O/S login ID and password.
- The password fields do not show key entries.

Fig. 5.10
Update UI Configuration Telnet Scripts

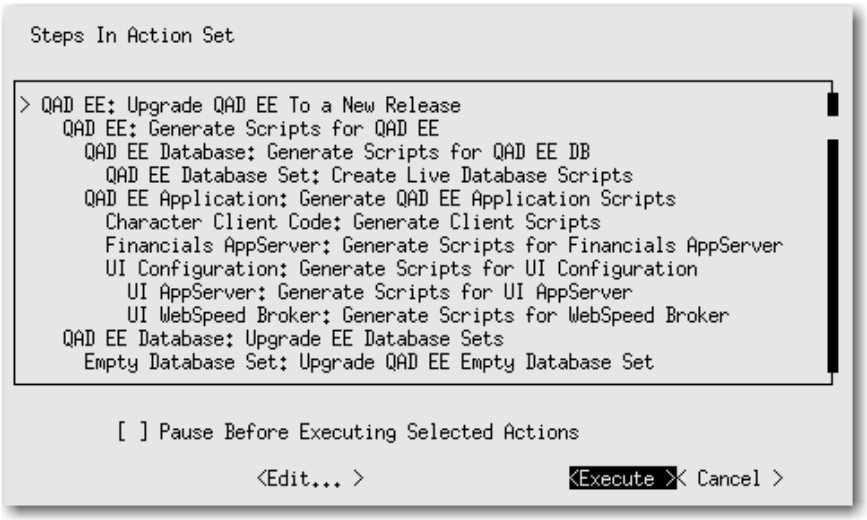


3 Tab to Next and press Enter. This returns to the main conversion option screen.

Execute Upgrade

Tab to Execute and press Enter to start the upgrade as shown below.

Fig. 5.11
Execute Upgrade

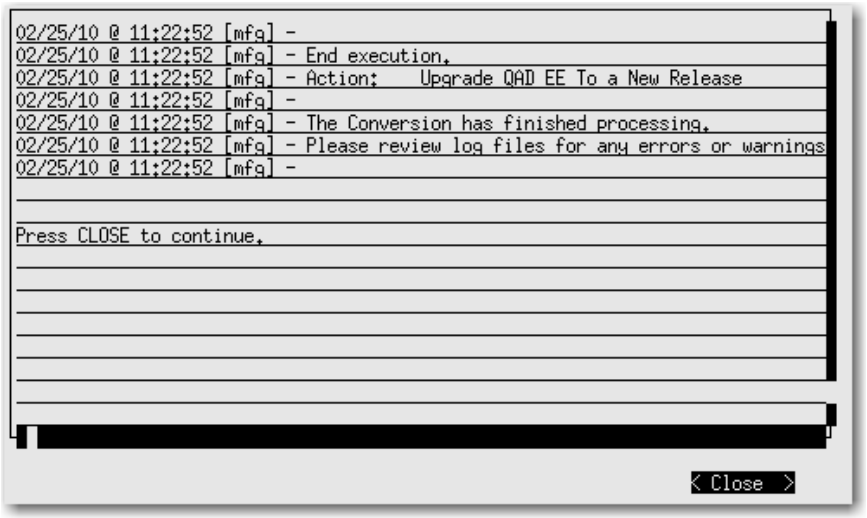


The upgrade proceeds until it finishes or an error occurs. No further input is required until the upgrade ends.

Upgrade Validation

The following QDT Admin screen should appear when the upgrade finishes.

Fig. 5.12
Upgrade Validation



1 Review and check the following log files for errors:
   - `<QDT install dir>/logs/qdtadmin.log`
   - `<QDT install dir>/logs/qdtadmin001.log`
   - `...`
   - `<QDT install dir>/logs/qdtadminxxx.log`

Note Refer to Appendix F, “Log Files,” on page 157 for further details on log files.

2 If the upgrade was successful, quit QDT.

Appendix A

GTM Conversions

This appendix describes GTM conversion considerations.

GTM Conversions Summary 74

Converting VAT Taxes to GTM 75

Converting US Taxes to GTM 88

Converting to GTM From No Taxes 104

Converting Canadian Taxes to GTM 105

GTM Conversions Summary

Before eB, MFG/PRO supported four tax processing systems:

- Value-added tax (VAT)
- United States (US) taxes
- Canadian taxes
- Global Tax Management (GTM)

See “Converting to GTM From No Taxes” on page 104.

Note Some companies may not have implemented any tax system, because they were using an external tax package or had some other alternate method for tracking taxes. In this case, some preparation is still required to convert to GTM and continue without a tax system.

Of the four systems, GTM offers the most precise calculations and the greatest flexibility for calculating taxes for multiple countries. Therefore, the system no longer supports regional tax system.

QAD supplies programs with GTM to automate most conversion tasks (Table A.1). These programs generate GTM codes and update existing eB2.1 and later records. They also create records required for tax reporting.

Table A.1
GTM Conversion Programs

Menu Number	Program
2.13.22.1	VAT to GTM–Setup
2.13.22.2	VAT to GTM–Masters
2.13.22.3	VAT to GTM–Transactions
2.13.22.5	USA to GTM–Setup
2.13.22.6	USA to GTM–Masters
2.13.22.7	USA to GTM–Transactions
2.13.22.9	CAN to GTM–Setup
2.13.22.10	CAN to GTM–Masters
2.13.22.11	CAN to GTM–Transactions

Pre-conversion Planning

To save time and reduce the likelihood of errors, address the following issues before starting the GTM conversion process:

- Timing

  Perform the conversion any time in transaction processing. You do not have to close open transactions or post transactions to the general ledger beforehand. However, for a clearer division of reporting, consider converting at the beginning of a new financial period.

  To prevent record contention conflicts, only run the conversion programs when no one else is using the system.

- Records to Convert

  Determine the range of records to convert. Master records such as customers and items are converted first, followed by transaction records. Records must be converted in the order in which their selection options display on the conversion screen. Finally, transactions that are prerequisites for other transactions must be converted first. For example, purchase order receipts must be converted before their respective vouchers.

  The conversion programs select records by number, not creation date or effective date. To convert records for a specific date range, specify the first record number for the starting date and/or the last record number for the ending date.

  The conversion programs do not perform cross-checks on your selection of records to convert. For example, for accounts payable, they do not verify that selected payment records are the ones associated with the selected voucher records.

- Code Naming Conventions

  Each conversion has default naming conventions for GTM tax classes, tax zones, and tax environments. Review these and decide if they are what you want.

- Integration of GTM Enhancements

  The objective of the conversion is to move your existing tax processing configuration into GTM. You must complete the conversion before you can incorporate new GTM features. There are two reasons for this. First, your current configuration does not have the data to support these features. Second, some of the conversion subprograms expect to encounter specific data values. They will not run correctly if you change these values prematurely.

- Custom Programming

  For some situations, custom programming is required. An example of such a situation is the need to merge two VAT class codes to one GTM tax class code without using GTM tax usage codes.

Practice running the conversion on a copy of your live database. You can identify problems in existing records, as well as familiarize yourself with the conversion process.

Post-conversion Procedures

Once you complete the conversion, you still have to exercise some controls to ensure a clean division for pre- and post-GTM reporting:

- Handling of Closed Transactions

  Closed transactions that were not included in the conversion should never be reversed or deleted once you start using GTM.

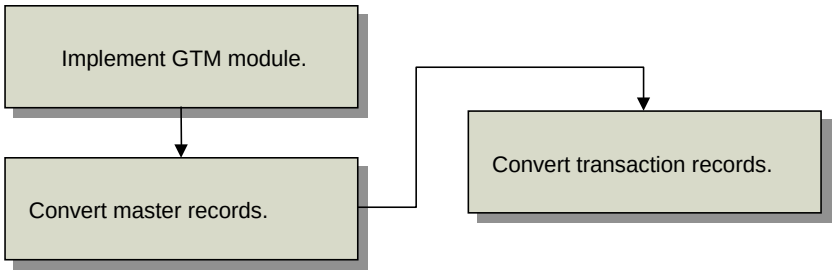
- Effective Date for Tax Reports

  Tax reports should be printed with a post-conversion tax date, to exclude transactions you did not bring into GTM.

Converting VAT Taxes to GTM

This section describes converting to GTM from the VAT system available in earlier versions.

Fig. A.1
VAT to GTM Conversion Process



The VAT to GTM conversion translates VAT data to GTM equivalents and updates existing eB2.1 and later records. Figure A.1 summarizes the conversion workflow, which revolves around three sets of activities.

Implementing GTM. Implementing GTM for VAT, you make planning decisions and then run a setup program. How this program works depends on whether a country is a member of the European Community.

Converting Master Records. Run a program that populates database tables for customer and supplier addresses, items, product lines, and other master records with GTM data values.

Converting Transaction Records. Run a second program that populates transaction records. All transactions subject to tax are affected, including sales, purchasing, accounts payable, accounts receivable, and service/support management.

Important After each of these activities, review the corresponding reports and audit trails. Mistakes can be pervasive and costly.

Table A.2 lists the eB2.1 and later programs used during the conversion.

Table A.2
eB2.1 and later Programs Used to Convert VAT to GTM

Activity	eB2.1 and Later Programs
Implementing GTM	Country Code Maintenance (2.14.1)
	VAT to GTM–Setup (2.13.22.1)
Converting master records	VAT to GTM–Masters (2.13.22.2)
Converting transaction records	VAT to GTM–Transactions (2.13.22.3)

Implementing GTM

An automated setup program can create most of the codes needed to implement GTM, based on how your VAT taxes are defined. Before executing this program, you should understand the options it provides and the default logic it uses.

Country Codes and Tax Environments

The first step in implementing VAT taxes in GTM is to ensure that all countries are defined in Country Code Maintenance (2.14.1). Countries that are part of the European Community must have the EC field set to Yes. This is important because two options are available during the automated setup:

1. You can create tax zones and tax environments with countries summing into one of two predefined tax zones: union and non-union. To do this, you must specify a code representing the union. By default, this is EU. The system then creates two tax zones—EU and NON-EU. These zones correspond to special country code values of ~1 (EU) and ~0 (NON-EU).

   This approach results in a minimum number of tax environments:

   - One environment for each country defined in the country master.
   - IN-EU for transactions within the union but outside the borders of one country
   - FROM-EU for transactions between a member of the union and a nonmember
   - TO-EU for transactions from a nonmember to a member of the union
   - NON-EU for transactions between two countries that are both nonmembers of the union

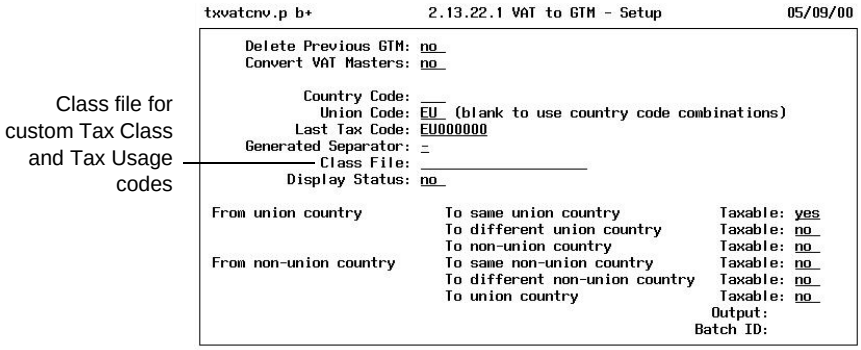
2. You can create tax zones and environments based on each defined country without summing into a union. This option results in the maximum number of tax environments—one for each combination of ship-from and ship-to countries. If, for example, you do business with 10 countries, 100 tax environments are created.

   If you choose this option, you do not need to specify a union code, and the setup program does not create the EU and NON-EU zones or the ~1 and ~0 countries.

Defining Custom Tax Class and Usage Codes

By default, the conversion setup generates tax classes that correspond to your VAT classes, without associated GTM tax usage codes. You can override this by creating your own map for the setup program and specifying it in the Class File field. The programs that convert master data and transaction records also reference the same class file.

Fig. A.2
Class File in VAT to GTM–Setup (2.13.22.1)



You should create a class file if:

- Your company plans to change tax class codes during the conversion.

- Within a tax class, a company can be taxed based on its nature of operation or the way it intends to use an item. Tax usage codes identify these conditions in GTM.

The class file is an ASCII file with text strings in the following format:

```
"Current VAT Class" "GTM Tax Class" "GTM AP Tax Usage" "GTM AR Tax Usage"
```

GTM tax classes are a maximum of three characters, and tax usage codes are eight characters. A null string (" " or "") represents an unused optional value.

The class file can have any name or extension. However, code values in .csv files must be separated by commas instead of blank characters. The file must be located in the home directory for the Progress session. A `.csv` file is a Windows comma-separated value file format for saving values recorded in a spreadsheet.

The class file accommodates companies that use different tax class and/or tax usage codes for AP and AR processing. If you use one set of codes for both kinds of tax processing, simply specify the same usage code for both.

Example Your current VAT classes are 1 and 2. You want to map VAT class 1 to GTM tax class A, tax usage code FOOD, and VAT class 2 to GTM tax class B, tax usage DRUG.

```
"1" "A" "FOOD" "FOOD"
"2" "B" "DRUG" "DRUG"
```

AP and AR usage codes are applied differently during the conversion to master records and transactions.

Table A.3
AP and AR Usage Codes

Usage Code	Applied to...
AP Usage Codes	The master conversion applies AP usage codes to supplier records. The transaction conversion uses them to update purchasing and accounts payable records.
AR Usage Codes	The master conversion applies AR usage codes to customers, warranty types, and contract types. The transaction conversion uses them to update sales, accounts receivable, and service/support management transactions.

Processing Logic

VAT to GTM–Setup creates records as described in Table A.4.

Table A.4
New GTM Records

Type of Record	Explanation
Tax zones	Based on the value of Union Code, setup builds the tax zone hierarchy for EU and non-EU countries in your current system or tax zones for all countries.
Tax types	Setup creates a tax type of VAT and NON-TAX.
Tax environments	Based on the value of Union Code, setup generates sums-into tax environments or environments for all ship-to/ship-from country combinations.
Tax rates	Based on VAT rates, setup generates tax rates for the tax jurisdictions and percentages used in your current system, as well as a nontaxable tax rate.
Tax classes	By default, setup generates tax classes based on your current VAT classes and a NOT-TAX class for nontaxable transactions. To create different classes, define a class file (see "Defining Custom Tax Class and Usage Codes" on page 77).

Type of Record	Explanation
Tax usages	By default, setup does not generate tax usages. However, you can create these with a class file.
Country code	The setup generates a record for the default country code you specify when you run the setup. If you enter a union code, setup creates `~1` and `~0` country codes.
Company addresses and address list types	In GTM, company sites require a corresponding company address record because taxes are calculated by address, not site. The setup verifies that each company site has an address record and creates any missing ones, along with any needed address list type records. The setup also creates a `~taxes` address record to provide a default tax address whenever a transaction is missing a company site code.

GTM Control Settings

Setup defines Global Tax Management Control (2.13.24) as described in Table A.5.

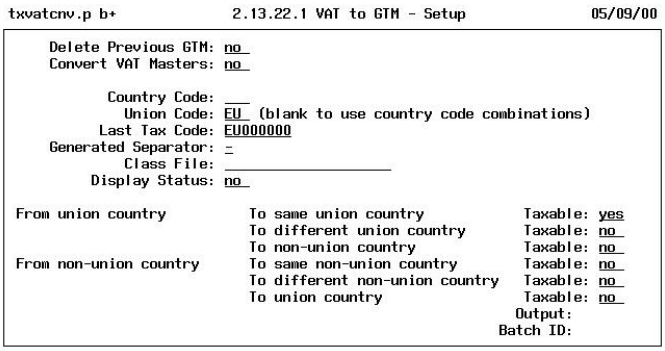
Table A.5
Updates to GTM Control Settings

Field	Explanation
Country Code	Value specified in VAT to GTM–Setup
Tax Method	01
Tax-By-Line	No
Accrue Tax at Receipt	No
Discount Tax at Invoice	Same setting as in VAT Control (2.15.2.24)
Discount Tax at Payment	Same setting as in VAT Control (2.15.2.24)
Last Tax Code	Value specified in VAT to GTM–Setup

VAT to GTM–Setup

Based on your implementation decisions, use VAT to GTM–Setup (2.13.22.1) to set up GTM for VAT tax processing.

Fig. A.3
VAT to GTM–Setup (2.13.22.1)



Delete Previous GTM. This option determines whether the setup deletes previously created GTM records from the database. If you select this option, the setup deletes tax zones, tax types, tax environments, transaction tax details, and other GTM records.

- Enter No if you have not yet converted your database to GTM.
- Enter Yes to clean up the database if it contains GTM records from unsuccessful conversion or installation attempts.

Convert VAT Masters. This option determines whether setup generates GTM records based on country codes and VAT classes.

- Enter Yes to create GTM records corresponding to VAT classes.
- Enter No if you only want to delete previous GTM records and do not want the setup program to generate new GTM records.

Country Code. Enter the default country code for the GTM control program.

Union Code. Enter a three-character code (default is EU) representing the European Union if you want countries to sum into a union and non-union zone. Leave this blank to create tax environments for each combination of ship-to and ship-from countries.

Last Tax Code. Enter a value to update the corresponding field in the Global Tax Management Control. In GTM, tax codes identify individual tax rates. Codes are generated sequentially based on the value of Last Tax Code in the GTM control program.

The default Last Tax Code is the union code followed by zeros. For example, for union code EU, the default Last Tax Code is EU000000. This value is recommended if you specified a value in Union Code.

If you are not summing into a union code, enter the default country code followed by zeros.

Generated Separator. Enter a character to use as a separator in system-generated tax zones and environments. Using a separator can improve the readability of the component elements of these codes.

The default separator is the dash (–), but you can enter any character. A sample GTM code that uses the dash separator is PAR–FR for Paris, France. If you do not want to use separators, enter blank. However, you cannot use blank as a separator character.

Note The system- generated nontaxable tax type is NON-TAX, regardless of the separator you specify.

Class File. To provide custom mapping of VAT classes to GTM classes and usage codes, specify an ASCII file with conversion information.

See “Defining Custom Tax Class and Usage Codes” on page 77.

Display Status. This setting determines whether the system displays status messages online during the conversion. These messages list database tables and their indexes as they are converted. If you select this option, messages display on the screen and the printed report.

Taxable. Enter the appropriate values for the six possible combinations of transactions between EC countries and non-EC countries. Yes indicates the transaction is taxable.

Converting Master Records

Once you finish implementing VAT processing in GTM, the next activity is to update tax settings in the following master records:

- Suppliers
- Customers
- Product lines
- Items
- Trailer codes

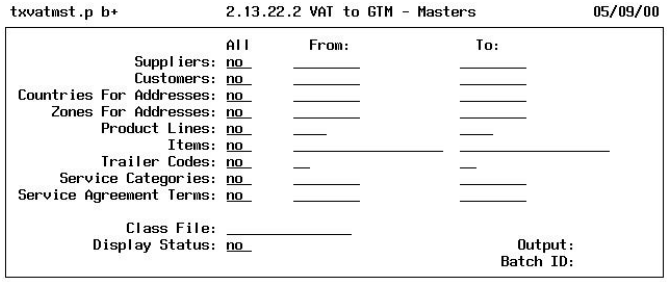
- Service categories
- Service agreement terms

GTM has additional fields and may require new values for existing fields.

VAT to GTM–Masters

To convert master records, run VAT to GTM–Masters (2.13.22.2).

Fig. A.4
VAT to GTM– Masters (2.13.22.2)



In addition to updating the tax settings in the master records, this program assigns tax zone codes to supplier, customer, and company address records. For verification of changes, the program generates an audit trail.

For each type of record, you can convert all records, a range of records, or individual records. The program converts records in the same order as the options on the screen. For separate audit trails, run the report separately for each type of record.

If you created a class file during the setup step, specify its name in the Class File field. Display status, output, and Batch ID are the same as in the setup program.

See “Defining Custom Tax Class and Usage Codes” on page 77.

Master Conversion Audit Trail

The master conversion prints a report of changed records. The format varies depending on the records included in the conversion. For each group of converted records, the report shows the record number and name followed by the before and after tax information, such as country code, tax zone code, taxable status, whether tax is included in item amounts, tax class, and tax usage.

Groups of converted records print in the same order as the screen selection criteria, and a page break separates each group. Warning and error messages identify potential conversion problems.

If you specified a class file, the report prints the VAT class and the corresponding GTM tax class and tax usage if any.

Figure A.5 and Figure A.6 show representative audit trail formats.

Fig. A.5
Customer Audit Trail

```
Processing: Customers

Address Name                                  Taxable Tax In TxC TaxUsage
-------- ----------------------------        ------- ------ --- --------
10000001 Consolidated Industries Ltd. Before     No      No     E   1-P-MFG
                                      After      No      No     E   1-P-MFG
10000002 Office Automation B.V.       Before     Yes     No     H   1-P-MFG
                                      After      Yes     No     H   1-P-MFG
10000003 MMB Verkehrssysteme GmbH     Before     Yes     No     G   1-P-MFG
                                      After      Yes     No     G   1-P-MFG
```

Fig. A.6
Countries for Addresses Audit Trail

```
Processing: Countries For Addresses

Address Name                                 Ctry Country
-------- ----------------------------        ---- --------------------------
10000001 Consolidated Industries Ltd. Before      United Kingdom
                                      After  UK   United Kingdom
10000002 Office Automation B.V.       Before      Netherlands
                                      After  NL   Netherlands
10000003 MMB Verkehrssysteme GmbH     Before      Germany
                                      After  D    Germany
```

Troubleshooting the Master Conversion

The error messages in the audit trail identify conditions you should analyze and address before you convert transactions. Table A.6 lists some common problems along with explanations. Before you make corrections, restore the database from backup.

Warning Do not proceed to the transaction conversion until the master conversion audit trail is free of errors.

Table A.6
Troubleshooting the Master Conversion

Error	Explanation
Tax class cannot be converted.	Class file does not contain VAT class that matches the one in the master record.
Blank tax class not allowed.	VAT class is blank in the class file.
Tax class cannot exceed 3 characters (*xxx*).	VAT class in the class file is longer than three characters. Message shows the first three characters.
Tax class does not exist (*x*).	VAT class in the class file not in the VAT master.
Tax class is not unique (*x*).	VAT class occurs in multiple places in the class file.
Tax class does not exist (*xxx*).	GTM tax class in the class file does not exist in the GTM tax class master.
Tax usage cannot exceed 8 characters (*xxxxxxxx*).	GTM tax usage in the class file is longer than eight characters. Message shows the first eight characters.
Tax usage does not exist (*xxxxxxxx*).	GTM tax usage in the class file does not exist in the GTM tax usage master.
Tax class/tax usage combination is not unique (*xxx xxxxxxxx*).	GTM tax class and tax usage combination occurs in multiple places in the class file.

Note *x*, *xxx*, and *xxxxxxxx* are placeholders for the actual codes displayed in the error message.

How the Conversion Changes Master Records

The following is a technical description of how the master conversion updates the database.

The menu-level program for VAT to GTM–Masters is `txvatmst.p`. This program calls subprograms that set the GTM tax values in the individual database tables. These programs can set the existing VAT class value or retrieve an alternate value from a class file.

Table A.7 lists the affected database tables and summarizes the changes.

Table A.7
Changes to Master Records

Table	Summary of Changes
ad_mstr	In supplier records, `txvatvd.p` sets ad_taxable from vd_taxable. It also sets ad_taxc and ad_tax_usage from ad_taxc or from the class file with AP usage if any. In customer records, `txvatcm.p` sets ad_taxable from cm_taxable and ad_tax_in from cm_tax_in. It also sets cm_taxc, ad_taxc, and ad_tax_usage from cm_taxc and from the class file with AR tax usage if any. In all address records, `txvatct.p` sets ad_ctry from ad_country and vice versa. `txvatzn.p` calls `txtxzget.p` to set ad_tax_zone.
fsc_mstr	`txvatfsc.p` sets fsc_taxc from fsc_taxc or from the AR tax usage if any.
pl_mstr	`txvatpl.p` sets pl_taxc from pl_taxc or from the class file if any.
pt_mstr	`txvatpt.p` sets pt_taxc from pt_taxc or from the class file if any.
sv_mstr	`txvatsv.p` sets sv_taxc from sv_taxc or from the class file with AR tax usage if any.
trl_mstr	`txvattrl.p` sets trl_taxc from trl_taxc or from the class file if any.

Converting Transaction Records

Once you finish converting master records, you can convert the following transaction records:

- Purchase orders and receipts
- Accounts payable vouchers and payments
- Service contracts, calls, and return material authorizations
- Sales quotes and orders
- Accounts receivable memos, invoices, and payments

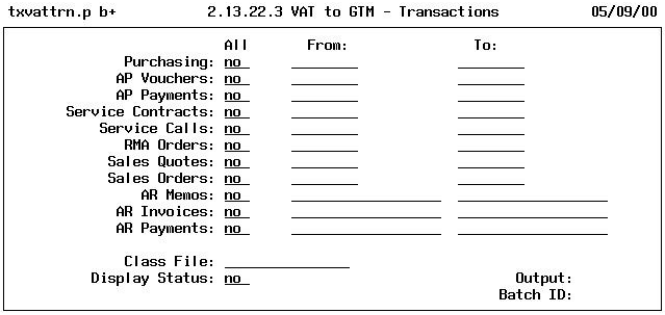
In GTM, every transaction subject to tax has a transaction tax detail record. This record stores the information used to calculate tax. It also separates the tax into component elements such as recoverable and non-recoverable amounts.

Note If you already have GTM transaction records in the database when you perform transaction conversion, the conversion process updates them using the current default tax values. Values replaced in these records include tax environment, class, usage, and so on.

VAT to GTM–Transactions

To convert existing transaction records so they are accessible in GTM, run VAT to GTM–Transactions (2.13.22.3).

Fig. A.7
VAT to GTM– Transactions (2.13.22.3)



In addition to updating transactions, this program generates an audit trail for verification of changes.

You can convert all records, a range of records, or individual records. The program converts records in the same order they display on the screen.

Note In some cases, the record sequence is important. Purchasing transactions must be converted before accounts payable vouchers and vouchers before payments. Accounts receivable memos and invoices must be converted before payments.

If you created a class file during the setup step, specify its name in the Class File field. Display status, output, and Batch ID are the same as in the setup program.

Transaction Audit Trail

The transaction conversion prints a report of changed records. The format varies depending on the records included in the conversion. For each group of converted records, the report shows the transaction number and name followed by the before and after tax information for each line item, such as taxable status, tax environment, tax class, and tax usage. Groups of converted records print in the same order as the screen selection criteria with a page break separating each group.

Warning and error messages identify potential conversion problems. Messages that appear at the end of a transaction apply to the entire transaction; those that appear between the Before and After line apply only to that line. If you specified a class file, the report prints the VAT class and the corresponding GTM tax class and tax usage if any.

Figure A.8 and Figure A.9 show representative audit trail formats.

Fig. A.8
Purchasing Audit Trail

```
Processing: Purchasing

Order    Receiver  Ln         Tax TxC TaxUsage Tax Env  prh_tax_at
-------- -------- ---         --- --- -------- -------- ----------
01104533             Before  No
                     After   No  E            BE-NE
                   1 Before  Yes e
                     After   Yes E            BE-NE
         RC1290    1 Before      e                    E
                     After       E            BE-NE    Yes
```

Fig. A.9
Service Calls Audit Trail

```
Processing: Service Calls

Call ID  Call/SR  Line Record               Tax TxC TaxUsage Tax Env
-------- -------- ---- ----------           --- --- -------- ----------------
CA127                  Call       Before    No  0
                                  After     No  0   1-P-MFG  GER-NE
         CA127       1 Item       Before    No  0
                                  After     No  0   1-P-MFG  GER-NE
         CA127       1 Billing    Before    No  0
                                  After     No  0   1-P-MFG  GER-NE
```

Troubleshooting Transaction Conversion

The warning and error messages in the audit trail identify conditions you should analyze and address before you resume live GTM processing. Table A.8 lists some common problems along with explanations. Before you make corrections, restore the database from backup.

Warning Do not resume live processing until the transaction conversion audit trail is free of errors.

In addition to examining the audit trail, you should review the Tax Detail by Transaction Report (2.13.15.3). This report shows the tax environments, tax types, and tax amounts for the converted records. Verify that the tax calculations are as expected.

Note Converted transactions may have minor differences in before/after tax amounts. These can occur because GTM uses a different calculation algorithm or rounding method than your current system. To synchronize the general ledger with the converted transactions, record adjusting entries.

Table A.8
Troubleshooting the Transaction Conversion

Error	Explanation
Detail tax environment must match header.	In accounts payable vouchers and accounts receivable debit/credit memos, the tax environment must be the same in both the header and detail lines.

Warning Do not correct transaction records programmatically. This approach often causes additional problems.

To eliminate ambiguity, the audit trail shows before and after values for purchasing, accounts payable, and accounts receivable records by their Progress database field name, not their screen label. For example, the audit trail displays voucher line types in the vod_type column.

Use Table A.9 to Table A.13 to interpret audit trails for the transaction conversion. These tables summarize before and after tax values.

Note In these tables, quotation marks indicate a value that cannot be translated.

Table A.9
VAT to GTM, Purchasing Transactions

Status	Tax System	pod_taxable	prh_tax_at
Taxable	VAT	Yes	VAT class
	GTM	Yes	“Yes”
Nontaxable	VAT	No	“0”[1]
	GTM	No	Blank

1. First VAT class with a zero percentage.

Table A.10
VAT to GTM, AP Voucher Receiver Lines

Status	Amt	Tax System	Tax	TxC	vod_type	vod_tax	vod_tax_at
Taxable	Item	VAT	No	VAT class	“R”	Blank	VAT class
		GTM	Yes	Tax class	“R”	Blank	“Yes”
	Tax	VAT	No	Blank	Blank	VAT class	Blank
		GTM	No	Blank	“T”	“t”	“No”
Nontaxable	Item	VAT	No	VAT class	“R”	Blank	VAT class
		GTM	No	Tax class	“R”	Blank	Blank
	Tax[1]	VAT	No	Blank	Blank	VAT class	Blank
		GTM	–	–	–	–	–

1. The conversion deletes VAT tax lines resulting from nontaxable amounts.

Table A.11
VAT to GTM, AP Voucher Memo Lines

Status	Amt	Tax System	Tax	TxC	vod_type	vod_tax	vod_tax_at
Taxable	Item	VAT	No	Blank	Blank	Blank	VAT class
		GTM	Yes	Tax class	Blank	Blank	“Yes”
	Tax	VAT	No	Blank	Blank	VAT class	Blank
		GTM	No	Blank	“T”	“t”	“No”
Nontaxable	Item	VAT	No	Blank	Blank	Blank	“0”[1]
		GTM	No	Tax class	Blank	Blank	“No”
	Tax	VAT	No	Blank	Blank	VAT class	Blank
		GTM	–	–	–	–	–

1. The conversion deletes VAT tax lines resulting from nontaxable amounts.

Table A.12
VAT to GTM, AR Invoices

Status	Amt	Tax System	TxC	ard_tax	ard_tax_at
Taxable	Item	VAT	Blank	VAT class	Blank
		GTM	Tax class	Blank	Tax class
	Tax	VAT	Blank	Blank	VAT class
		GTM	Blank	“t”	“No”
Nontaxable	Item	VAT	Blank	Blank	“0”[1]
		GTM	Tax class	Blank	Tax class
	Tax[2]	VAT	Blank	“0”	Blank
		GTM	–	–	–

1. First VAT class with a zero percentage.
2. The conversion deletes VAT tax lines resulting from nontaxable amounts.

Table A.13
VAT to GTM, AR DR/CR Memos

Status	Amt	Tax System	TxC	ard_tax	ard_tax_at
Taxable	Item	VAT	Blank	Blank	VAT class
		GTM	Tax class	Blank	“Yes”
	Tax	VAT	Blank	VAT class	Blank
		GTM	Blank	“t”	“No”
Nontaxable	Item	VAT	Blank	Blank	“0”
		GTM	Tax class	Blank	“No”
	Tax[1]	VAT	Blank	“0”[2]	Blank
		GTM	–	–	–

1. The conversion deletes VAT tax lines resulting from nontaxable amounts.
2. First VAT class with a zero percentage.

How the Conversion Changes Transaction Records

The following is a technical description of how the transaction conversion updates the database.

The menu-level program for VAT to GTM–Transactions is `txvattrn.p`. This program calls subprograms that set the GTM tax values in the individual database records. For all transactions, the conversion also generates corresponding tax detail records in the tx2d_det database table.

When setting the GTM tax class value, these programs can set the existing VAT class value or retrieve an alternate value from a class file.

Table A.14 lists the affected database tables and summarizes the changes.

Table A.14
Changes to Transaction Records

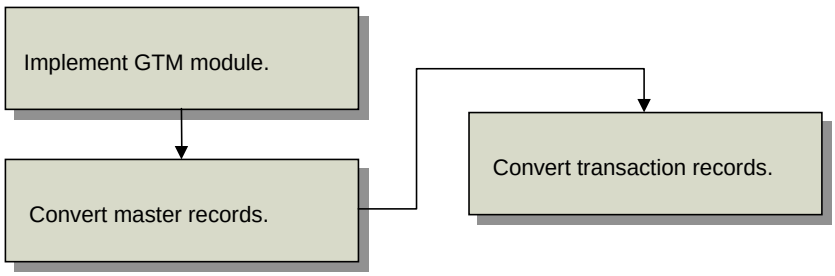
Tables	Summary of Changes
ard_det	For debit/credit memos, `txvatarm.p` sets ard_tax, ard_tax_at, ard_taxc, and ard_tax_usage from the class file with AR usage if any. For invoices, `txvatari.p` sets ard_taxc and ard_tax_usage from the class file with AR tax usage if any, and ard_tax and ard_tax_at. Duplicate records for unique keys are merged into one record.
ar_mstr	For debit/credit memos, `txvatarm.p` sets ar_tax_env.

Tables	Summary of Changes
ca_mstr	`txvatca.p` sets ca_taxc and ca_tax_usage from ca_taxc or from the class file with AR tax usage if any. It also sets ca_tax_env using `txtxeget.p`.
idh_hist	`txvatari.p` sets idh_taxc and idh_tax_usage from idh_taxc or from the class file with AR tax usage if any. It also sets idh_tax_env using `txtxeget.p`.
ih_hist	`txvatari.p` sets ih_taxc and ih_tax_usage from ih_taxc or from the class file, with AR tax usage if any. It also sets ih_tax_env using `txtxeget.p`.
itm_det	If itm_prefix is CA and itm_type is any value except INV, `txvatca.p` sets itm_taxc and itm_tax_usage from itm_taxc or from the class file with AR usage if any. It also sets itm_tax_env using `txtxeget.p`.
pod_det	For inventory items, `txvatpo.p` sets pod_taxc and pod_tax_usage from pod_taxc or from the class file, with AP tax usage if any. For memo items, `txvatpo.p` sets pod_taxc from po_taxc and pod_tax_usage from po_tax_usage.
po_mstr	`txvatpo.p` sets po_tax_pct[1], po_tax_pct[2], and po_tax_pct[3] to 0. It sets po_tax_usage from ad_tax_usage and po_taxc from ad_taxc. It also sets po_tax_env using `txtxeget.p`.
prh_hist	For inventory items, `txvatpo.p` sets prh_taxc and prh_tax_usage from prh_taxc or from the class file, with AP tax usage if any. For memo items, `txvatpo.p` sets prh_taxc from pod_taxc and prh_tax_usage from pod_tax_usage. For all items, `txvatpo.p` also sets prh_tax_at from pod_taxable.
qod_det	`txvatqo.p` sets qod_taxc and qod_tax_usage from qod_taxc or from the class file with AR usage if any. If qo_taxable and qod_taxable are No and the quote is for an inventory item, `txvatqo.p` sets qod_taxc from pt_taxc. It also sets qod_tax_env.
qo_mstr	`txvatqo.p` sets qo_tax_pct[1], qo_tax_pct[2], and qo_tax_pct[3] to 0. `txvatqo.p` also sets qo_taxc and qo_tax_usage from qo_taxc or from the class file with AR usage if any. It also sets qo_tax_env.
rma_mstr	`txvatrma.p` sets rma_taxc from rma_taxc or from the class file with AR usage if any.
sad_det	`txvatsc.p` sets sad_taxc and sad_tax_usage from sad_taxc or from the class file with AR usage if any. `txvatsc.p` sets sad_tax_env from sa_site and sa_taxc using `txtxeget.p`.

Converting US Taxes to GTM

This section describes converting to GTM from MFG/PRO US taxes.

Fig. A.10
USA to GTM Conversion Process



The USA to GTM conversion process translates United States tax data to GTM equivalents and updates existing eB2.1 and later records. Figure A.10 summarizes the conversion workflow:

Implementing GTM. Run a setup program to implement GTM for US tax processing.

Converting Master Records. Run a second program to populate database tables for customer and supplier addresses, items, product lines, and other master records with GTM data values.

Converting Transaction Records. Run a third program to populate transaction records. All transactions subject to tax are affected, including sales, purchasing, accounts payable, accounts receivable, and service/support management.

Note After each of these activities, review the corresponding reports and audit trails. Mistakes can be pervasive and costly.

Table A.15 lists the eB2.1 and later programs used during the conversion.

Table A.15
eB2.1 and Later Programs Used to Convert US Taxes to GTM

Activity	eB2.1 and Later and Programs
Implementing GTM	USA to GTM–Setup (2.13.22.5)
Converting master records	USA to GTM–Masters (2.13.22.6)
Converting transaction records	USA to GTM–Transactions (2.13.22.7)

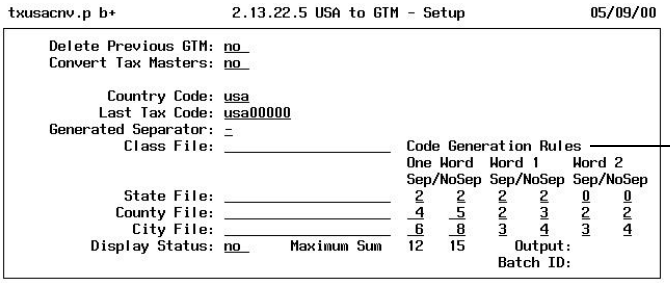
Implementing GTM

An automated setup program creates the codes needed to implement GTM, based on how your US taxes are currently defined. Before executing this program, you should understand the options it provides and the default logic it uses.

Code Generation Rules

GTM codes for tax types, tax zones, and tax environments consist of text strings that uniquely identify the state, county, and city of a tax jurisdiction. Manual setup of these codes would be a tedious process, since there are thousands of them.

Fig. A.11
Code Generation Rules in USA to GTM–Setup (2.13.22.5)



Used to generate codes for Tax Types, Tax Zones, and Tax Environments

Therefore, by default, USA to GTM–Setup creates codes based on a set of rules. These rules systematically select characters from the state code, county name, and city name in the tax master.

To determine if the generated codes are appropriate for your company, run the setup and review the audit trail. If you need a different coding scheme, read the rest of this section and settings for the code generation rules as necessary. Then, rerun the setup with Delete Previous GTM set to Yes and Convert Tax Masters set to Yes.

Table A.16 lists the default generated code formats for US taxes.

Table A.16
Default Generated Code Formats in USA to GTM– Setup (2.13.22.5)

Code	Format	Explanation
Tax zones and tax environments	*SS-CCCC-cccccc*	*SS* is the 2-character state code, *CCCC* is the 4-character county name, and *cccccc* is the 6-character city name. A dash (–) separates each text string.
Tax Types	*SS-CCCC-cccccc-#*	This format is the same as the previous one, except that tax types have an extra digit (#) to identify whether the tax type applies to the state (1), county (2), or city (3).

The rules that determine the characters to select depend on two factors:

- Whether the US code or name used to generate the text string consists of one word or multiple words (text separated by blank spaces).
- Whether separator characters are used

Table A.17 lists the default number of characters for each text string under the different conditions. However, you can change the number of characters and use a different separator or no separator, as long as the total number of generated characters—including the separator and any ending integers—is 16 or less.

Table A.17
Code Generation Rules in USA to GTM–Setup (2.13.22.5)

	One Word		Multiple Words			
			Word 1		Word 2	
Code	Sep	No Sep	Sep	No Sep	Sep	No Sep
State	2	2	2	2	0	0
County	4	5	2	3	2	2
City	6	8	3	4	3	4
Maximum Total Characters, With Separator						12
Maximum Total Characters, Without Separator						15

Example If the original state code for Arkansas is ARKA, the generated text string is AR regardless of whether separators are used. For the county of Orange, the generated text string is Oran if separators are used and Orang if they are not. For the city of North Hollywood, the generated text string is NorHol if separators are used and NortHoll if they are not.

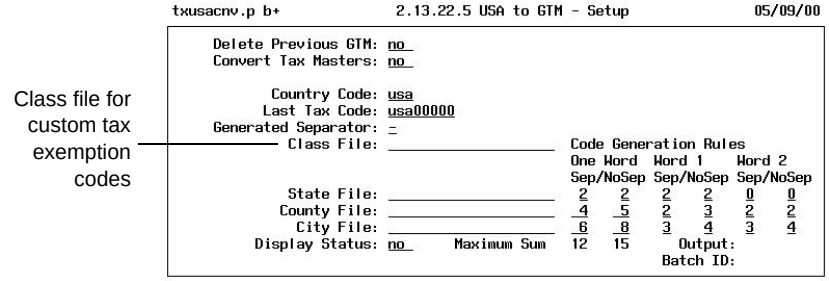
To override duplicate strings such as AR for the states Arkansas and Arizona, see “Defining Custom Codes for States, Counties, and Cities” on page 91.

The setup retains the capitalization from the original US code or name. If the original code or name contains punctuation such as a period, the code generation rules treat it the same as any other non-blank character.

Defining Custom Tax Exemption Codes

By default, the conversion generates corresponding GTM tax classes for your current tax exemption codes. For example, for tax exemption 1, the conversion generates GTM tax class 1. You can override this by creating your own map for the setup program and specifying it in the Class File field. The same class file is also referenced in the programs that convert master data and transaction records.

Fig. A.12
Class File in USA to GTM–Setup (2.13.22.5)



You should create a class file if:

- Your company plans to change its tax exemption codes during the GTM conversion.
- You want to convert exemption codes to tax usages instead of tax classes.

The class file is an ASCII file with text strings in the following format:

```
"Current Tax Exemption Code" "GTM Tax Class" "GTM Tax Usage"
```

GTM tax classes can have a maximum of three characters and tax usage codes eight characters. A null string (" " or "") represents an unused optional value.

The file name can have any name or extension. However, code values in .csv files must be separated by commas instead of blank characters. The file must be located in the home directory for the Progress session. A .csv file is a Windows comma-delimited file format that saves values recorded in a spreadsheet.

Example Your current exemption codes are 1 and 2. You want to map these to GTM tax classes 01 and 02.

```
"1" "01" " "
"2" "02" " "
```

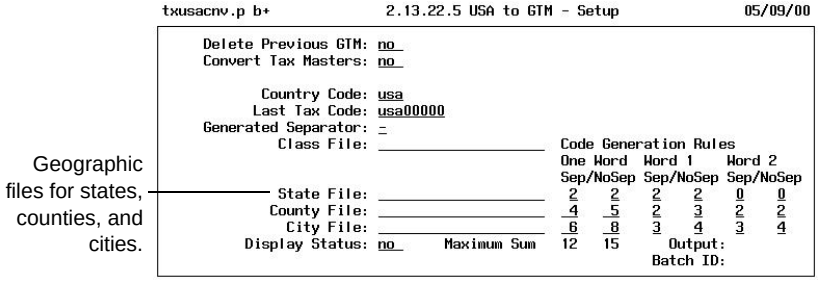
Defining Custom Codes for States, Counties, and Cities

The generated codes for tax zones, tax environments, and tax types consist of text strings that identify the state, county, and city. By default, the code generation rules define the structure of these text strings. However, if you need a different naming convention, you can create geographic files for state codes, county names, and/or city names. See "Updates to Company Addresses" on page 95.

In addition to supporting alternate naming conventions, such files can resolve code generation conflicts. For example, for state codes ARIZ and ARKA, the generated GTM code is AR. A state file is necessary to provide unique values.

Note You must only define codes for conditions not supported by the code generation rules.

Fig. A.13
State, County, and City Files in USA to GTM–Setup (2.13.22.5)



Create a separate file for each kind of text string and reference the file in USA to GTM–Setup as shown in Figure A.13. A geographic file is the same as a class file, except that it uses the following format:

```
"Current Code or Name" "GTM Text String"
```

Example To create unique codes for Arizona and Arkansas, create a state file with these lines:

```
"ARIZ" "AZ"
"ARKA" "AR"
```

Processing Logic

USA to GTM–Setup creates records as described in Table A.18.

Table A.18
New GTM Records

Type of Record	Explanation
Tax zones	Based on code generation rules or a class file, setup builds the tax zone hierarchy for the country and all state/county/city combinations in your current system.
Tax types	Based on code generation rules or a class file, setup generates tax types for all state/county/city combinations in your current system.
Tax environments	Based on code generation rules or a class file, setup generates tax environments for all ship-to tax zones. In the US, the tax environment's ship-from tax zone is the default country code, and the customer or supplier tax class is blank.
Tax rates	Based on code generation rules or a class file, setup generates tax rates for the tax jurisdictions and percentages in the tax master.
Tax classes	By default, setup generates corresponding tax classes for existing US tax exemption codes. However, if you reference a class file, the mappings in the class file determine the actual exemption codes.
Tax usages	By default, the setup does not generate tax usages. However, you have the option to do so in the class file.
Rounding method	The setup enters the rounding method specified in System/Account Control (36.1) as the GTM rounding method in Global Tax Management Control (2.13.24).
Country code	The setup generates a record for the default country code you specify when you run the setup. This country code is the top-level zone in the tax zone hierarchy.
Company addresses and address list types	In GTM, company sites require a corresponding company address record because taxes are calculated by address, not site. The setup verifies that each company site has an address record and creates any missing ones, along with any needed address list type records. The setup also creates a ~taxes address record to provide a default tax address whenever a transaction is missing a company site code.

GTM Control Settings

The setup resets Global Tax Management Control (2.13.24) as described in Table A.19.

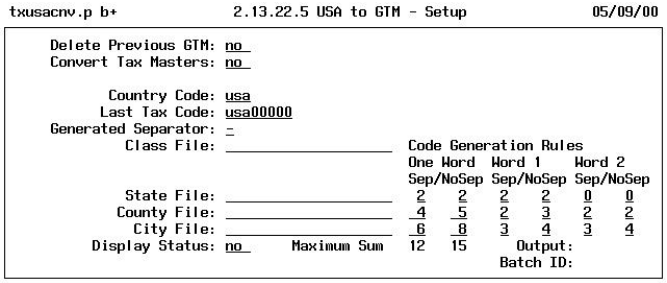
Table A.19
Updates to GTM Control Settings

Field	Explanation
Country Code	As specified during the setup
Tax Method	01
Tax-By-Line	No
Accrue Tax at Receipt	Yes
Discount Tax at Invoice	No
Discount Tax at Payment	No
Last Tax Code	As specified during the setup
Rounding Method	Value specified in System/Account Control (36.1)

USA to GTM–Setup

Based on your implementation decisions, use USA to GTM–Setup (2.13.22.5) to set up GTM for US tax processing.

Fig. A.14
USA to GTM–Setup (2.13.22.5)



Warning As noted previously, the purpose of the conversion is to replicate your existing tax processing setup in GTM. Do not attempt to implement new GTM functionality until after the entire conversion is complete. Do not change settings in the new GTM records or in Global Tax Management Control (2.13.24). If you do, conversions may fail.

Delete Previous GTM. This option determines whether the setup deletes previously created GTM records from the database. If you select this option, the setup deletes tax zones, tax types, tax environments, transaction tax details, and other GTM records from the database.

- Enter No if you have not yet converted your database to GTM.
- Enter Yes to clean up the database if it contains GTM records from unsuccessful conversion or installation attempts.

Convert Tax Masters. This option determines whether the setup generates the GTM master records from USA tax master records.

- Enter Yes to create corresponding GTM records for the USA tax master: tax classes, tax types, tax zones, tax environments, tax rates, and so on. The audit trail shows the USA tax master records and the new tax zone codes.

- Enter No if you only want to delete previous GTM records and do not want the setup program to generate new GTM records.

Country Code. This country code is the top-level tax zone in the tax zone hierarchy. All other tax zones sum into this one.

If Global Tax Management Control (2.13.24) already has a country code, it displays here. Otherwise, the setup sets the default country code to USA. If you override the value here, the setup assigns it to the control program.

Last Tax Code. Enter a value to update the corresponding field in Global Tax Management Control. In GTM, tax codes identify individual tax rates. Codes are generated sequentially based on the value of Last Tax Code in the GTM control program.

The default Last Tax Code is an 8-character value that consists of the GTM country code and a right-justified integer with placeholder zeros. For example, for country code USA, the default Last Tax Code is USA00000. The system assigns the number USA00001 to the first tax rate record created in GTM and increments this number for subsequent rates.

If you want tax codes to have a different format, enter a different prefix. Codes display alphanumerically in screens and reports. Tax codes that are totally numeric are left justified and have no placeholder zeros. For example, codes 1 through 30 display in a report column as follows:

```
1
...
19
2
20
...
30
```

Generated Separator. Enter a character to use as a separator in system-generated tax zone, type, and environment codes. Using a separator improves readability of the elements of these codes.

The default separator is the dash (–), but you can enter any character. A sample GTM code that uses the dash separator is CA–SBa–SBa for Santa Barbara, California. If you do not want to use separators, enter blank. However, you cannot use blank as a separator character.

Class File. To provide custom mapping of US tax exemption codes to GTM classes and usage codes, specify an ASCII file with conversion information.

State, County, City File. To override default code generation rules, specify specific values for geographic locations in an ASCII file.

Code Generation Rules. Enter appropriate values for your organization.

Display Status. This setting determines whether the system displays status messages online during the conversion. These messages list database tables and their indexes as they are converted. If you select this option, messages display on the screen and the printed report.

Updates to Company Addresses

In GTM, company sites require a corresponding company address record because taxes are calculated by address, not site. The setup creates any missing company address records for company sites. However, the setup does not populate these new address records with the city, county, state, and country. You must supply this information manually in Company Address Maintenance (2.12).

Also, set up tax zone codes to support these new addresses if the setup did not already generate codes for these tax jurisdictions. Do this in Tax Zone Maintenance (2.13.3.13). Then, assign the tax zone to the address.

Setup Audit Trail

USA to GTM–Setup prints an audit trail of updated tax master records. For each record, the report shows the state/county/city combination, tax effective date, the tax rates for the effective date, and the taxable status of trailer charges. It also shows the corresponding generated tax zone and its sums-into tax zone.

Figure A.15 shows the audit trail format.

Fig. A.15
Setup Audit Trail

```
Processing: Create GTM from tax masters
                                                    Tax                Sums-Into
State County City      Effective  Tax    Tax    Tax  Trl Tax Zone       Tax Zone
----- ------ --------- ---------- ------ ------ ------ --- -------------- --------
FL    ORANGE KISSIMMEE 08/07/97  8.00% 2.00% 6.00% NO  FL-ORAN-KISSIM USA
FL    ORANGE ORLANDO   10/17/92  7.00% 0.00% 9.00% NO  FL-ORAN-ORLAND USA
```

Troubleshooting GTM Setup

After you run USA to GTM–Setup, verify that the GTM setup is correct before you continue with the conversion. The problems listed in Table A.20 can cause errors or unexpected values. Before you proceed to the master conversion, review the audit trail, the GTM reports for the new records, and Global Tax Management Control settings. Correct any problems before continuing.

Subsequent setups do not automatically overwrite records created by previous ones. To set up new records, you must first delete the old ones. If you rerun the setup, you must remove the records created by the earlier setup attempt by setting Delete Previous GTM to Yes and Convert Tax Masters to Yes.

Note If you must rerun the setup after you have run any of the other conversion programs, restore the database first. Then, rerun the setup and any other conversion programs you ran previously. This is necessary to propagate changes to master data, transactions, and tax details.

Table A.20
Troubleshooting the GTM Setup

Error	Explanation
Tax system must be USA.	The USA to GTM setup can be run only on a US tax system.

Error	Explanation
Must delete previous GTM when converting.	When you set Convert Tax Masters to Yes, you must also set Delete Previous GTM to Yes.
Tax-trailers has changed, cannot convert prior to this date.	The setup can convert only the current tax environment, not previous variations. If the taxable status of trailer charge codes changed in the time span included in the conversion, the setup creates tax environments only for current conditions.

Warning Do not correct records programmatically. This approach often causes additional problems.

How the Setup Changes GTM Records

The following is a technical description of how the setup updates the database.

The menu-level program for USA to GTM–Setup is `txusacnv.p`. This program calls subprograms (primarily `txusatax.p`) that set the GTM tax values in the individual database records. Table A.21 lists the affected database tables and summarizes the changes.

Table A.21
Changes to GTM Records

Tables	Summary of Changes
ad_mstr	`txusatax.p` creates one ~taxes record for the database. It also scans si_mstr and creates an address record for any company site that does not already have one.
code_mstr	For the nine US exemption codes in tax_mstr, `txusatax.p` creates corresponding tax classes. If a class file is referenced, it creates the specified tax classes. For each of the three rates that tax_mstr stores for US tax jurisdictions, `txusatax.p` creates a tax type. For non-taxable transactions, it also creates a default NON-TAX tax type. Finally, for each ship-to tax zone, `txusatax.p` creates a tax environment and assigns it the tax types associated with the tax zone.
ctry_mstr	`txusatax.p` creates a record for the default country code specified in the selection data.
ls_mstr	`txusatax.p` creates ls_mstr record for each new ad_mstr record, if any.
tx2_mstr	For each of the three rates that tax_mstr stores for US tax jurisdictions, `txusatax.p` creates a tax rate. It also runs txtx2_nt.i to create a non-taxable tax rate and txtxmeth.i to create tax method 01.
txc_ctrl	`txusatax.p` sets txc_ctry_code and txc_tax_code from the selection data. It sets txc_method to 01, txc_by_line, txc_inv_disc, and txc_pmt_disc to No, and txc_rcpt_tax_point to Yes.
txe_mstr	`txusatax.p` creates tax environment zone detail records for every tax environment code it generates for the code_mstr.
txed_det	`txusatax.p` creates tax environment tax type detail records for every tax environment code it generates for the code_mstr.
txz_mstr	`txusatax.p` creates a top-level sums-into tax zone for the new ctry_mstr record. For each state/county/city combination in tax_mstr, it creates a ship-to tax zone.

Converting Master Records

Once you finish the GTM setup, the next activity is to update tax settings in the following master records:

- Suppliers
- Customers
- Product lines

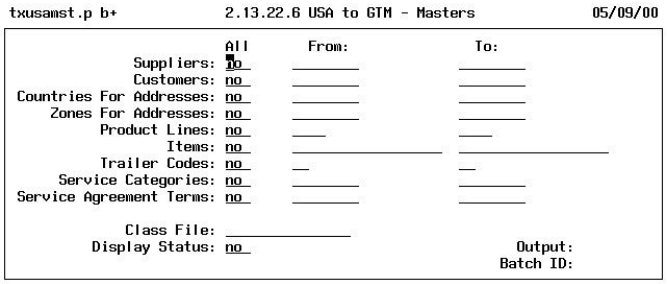
- Items
- Trailer codes
- Service categories
- Service agreement terms

GTM has additional fields and may require new values for existing fields.

USA to GTM–Masters

To convert master records, run USA to GTM–Masters (2.13.22.6).

Fig. A.16
USA to GTM–Masters (2.13.22.6)



In addition to updating the tax settings in the master records, this program assigns tax zone codes to supplier, customer, and company address records. For verification of changes, the program generates an audit trail.

Important Before you run USA to GTM–Masters, do the following.

- Run USA to GTM–Setup (2.13.22.5).
- To avoid record contention conflicts with other users, shut down the database. Restart it with no other users on the system.

For each type of record, you can convert all records, a range of records, or individual records. This program converts records in the same order as the options on the screen. For separate audit trail reports, run the report separately for each type of record.

If you created a class file during the setup step, specify its name in the Class File field. Display Status, Output, and Batch ID are the same as in the setup program.

See “Defining Custom Tax Exemption Codes” on page 90.

Master Conversion Audit Trail

The master conversion prints a report of changed records. The format varies depending on the records included in the conversion. For each group of converted records, the report shows the record number and name followed by the before and after tax information, such as country code, tax zone code, taxable status, whether tax is included in item amounts, and tax class.

Groups of converted records print in the same order as the screen selection criteria with a page break separating each group. Warning and error messages identify potential conversion problems.

Figure A.17 and Figure A.18 show representative audit trail formats.

Fig. A.17
Customer Audit Trail

```
Processing: Customers

Address  Name                                    Taxable TxC TaxUsage
-------- ----------------------------            ------- --- --------
32174893 Consolidated Industries Inc. Before     No       1
                                      After      No       1
32174895 Asheville Manufacturing      Before     Yes
                                      After      Yes
32174897 Hartford Electronics         Before     Yes
                                      After      Yes
```

Fig. A.18
Countries for Addresses Audit Trail

```
Processing: Countries For Addresses

Address  Name                                    Ctry Country
-------- ----------------------------            ---- -------------
32174893 Consolidated Industries Inc. Before          United States
                                      After      USA  United States
32174895 Asheville Manufacturing      Before          United States
                                      After      USA  United States
32174897 Hartford Electronics         Before          United States
                                      After      USA  United States
```

Troubleshooting the Master Conversion

The error messages in the audit trail identify conditions you should analyze and address before you convert transactions. Table A.22 lists some common problems along with explanations. Before you make corrections, restore the database from backup.

Warning Do not proceed to the transaction conversion until the master conversion audit trail is free of errors.

Table A.22
Troubleshooting the Master Conversion

Error	Explanation
Tax class cannot be converted.	Class file does not contain a tax exemption that matches the one in the master record.
Blank tax class not allowed.	Tax exemption is blank in the class file.
Tax class cannot exceed 3 characters (*xxx*).	Tax exemption in the class file is longer than three characters. The message shows the first three characters.
Tax class does not exist (*x*).	Tax exemption in the class file not in the US tax master.
Tax class is not unique (*x*).	Tax exemption occurs in multiple places in the class file.
Tax class does not exist (*xxx*).	GTM tax class in the class file does not exist in the GTM tax class master.
x, *xxx*, and *xxxxxxxx* are placeholders for the actual codes displayed in the error message.	

How the Conversion Changes Master Records

The following is a detailed description of how the master conversion updates the database.

The menu-level program for USA to GTM–Masters, `txusamst.p`, sets the GTM tax values in the individual tables.

When setting the GTM tax class value, the programs can set the existing tax exemption code or retrieve an alternate from a class file.

See “Defining Custom Tax Exemption Codes” on page 90.

Table A.23 lists the affected database tables and summarizes the changes.

Table A.23
Changes to Master Records

Tables	Summary of Changes
ad_mstr	In supplier records, `txusavd.p` sets ad_taxable from vd_taxable. It also sets ad_taxc and ad_tax_usage to blank. In customer records, `txusacm.p` sets ad_taxable from cm_taxable and ad_tax_in from cm_tax_in. It also sets cm_taxc, ad_taxc, cm_taxc, and ad_tax_usage from cm_taxc or from the AR class file, if any. In all address records, `txusact.p` sets ad_ctry from ad_country and visa versa. `txusazn.p` calls `txtxzget.p` to set ad_tax_zone.
fsc_mstr	`txusafsc.p` sets fsc_taxc from fsc_taxc or from the AR class file if any.
pl_mstr	`txusapl.p` sets pl_taxc from pl_taxc or from the class file, if any.
pt_mstr	`txusapt.p` sets pt_taxc from pt_taxc or from the class file, if any.
sv_mstr	`txusasv.p` sets sv_taxc and sv_tax_usage from sv_taxc or from the AR class file, if any.
trl_mstr	`txusatrl.p` sets trl_taxc from trl_taxc or from the class file, if any.

Converting Transaction Records

Once you finish converting master records, you can convert the following transaction records:

- Purchase orders and receipts
- Accounts payable vouchers and payments
- Service contracts, calls, and return material authorizations
- Sales quotes and orders
- Accounts receivable memos, invoices, and payments

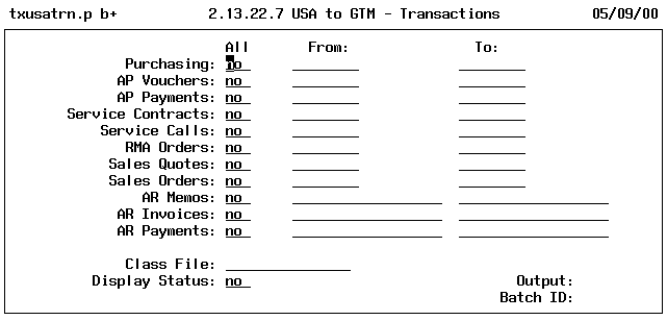
In GTM, every transaction subject to tax has a transaction tax detail record. This record stores the information used to calculate tax.

Note If you already have GTM transaction records in the database when you perform transaction conversion, the conversion process updates them using the current default tax values. Values replaced in these records include tax environment, class, usage, and so on.

USA to GTM–Transactions

To convert existing transaction records, run USA to GTM–Transactions (2.13.22.7).

Fig. A.19
USA to GTM–Transactions (2.13.22.7)



In addition to updating transactions, this program generates an audit trail for verification of changes.

Important Before you run USA to GTM–Transactions, do the following:

- Run USA to GTM–Setup (2.13.22.5) and USA to GTM–Masters (2.13.22.6).
- Avoid record-contention conflicts with other users. Shut down the database and restart it when no other users are on the system.

You can convert all records, a range of records, or individual records. The program converts records in the same order they display on the screen. For separate audit trail reports, run the report separately for each type of record.

Note In some cases, the record sequence is important. Purchasing transactions must be converted before accounts payable vouchers and vouchers before payments. Accounts receivable memos and invoices must be converted before payments.

If you created a class file during the setup step, specify its name in the Class File field. Display Status, Output, and Batch ID are the same as in the setup program.

See “Defining Custom Tax Exemption Codes” on page 90.

Transaction Audit Trail

The transaction conversion prints a report of changed records. The format varies depending on the records included in the conversion. For each group of converted records, the report shows the transaction number and name followed by the before and after tax information for each line item, such as taxable status, tax environment, and tax class. Groups of converted records print in the same order as the screen selection criteria with a page break separating each group.

Warning and error messages identify potential conversion problems. Messages that appear at the end of a transaction apply to the entire transaction; those that appear between the Before and After line apply only to that line.

Figure A.20 and Figure A.21 show representative audit trail formats.

Fig. A.20
Purchasing Audit Trail

```
Processing: Purchasing

Order    Receiver  Ln         Tax TxC TaxUsage Tax Env  prh_tax_at
-------- -------- ---        --- --- -------- -------- ----------
01104533             Before  No  1
                     After   No  1            NJ-TRENT
                   1 Before  Yes B
                     After   Yes B            NJ-TRENT
         RC1290      1 Before     B                    B
                     After        B            NJ-TRENT B
```

Fig. A.21
Service Calls Audit Trail

```
Processing: Service Calls

Call ID  Call/SR  Line Record              Tax TxC TaxUsage Tax Env
-------- -------- ---- ----------          --- --- -------- ----------------
CA127                  Call       Before    No  1
                                  After     No  1   1-P-MFG  NJ-TRENT
         CA127       1 Item       Before    No  1
                                  After     No  1   1-P-MFG  NJ-TRENT
         CA127       1 Billing    Before    No  1
                                  After     No  1   1-P-MFG  NJ-TRENT
```

Troubleshooting Transaction Conversion

The error messages in the audit trail identify conditions you should analyze and address before you resume live GTM processing. Table A.24 lists some common problems along with explanations. Before you make corrections, restore the database from backup.

Table A.24
Troubleshooting the Transaction Conversion

Error	Explanation
Detail tax environment must match header.	In accounts payable vouchers and accounts receivable debit/credit memos, the tax environment must be the same in both the header and detail lines.

Warning Do not resume live processing until the transaction conversion audit trail is free of errors. Do not correct transaction records programmatically. This approach often causes additional problems.

In addition to examining the audit trail, it is advisable to review the Tax Detail by Transaction Report (2.13.15.3). This report shows the tax environments, tax types, and tax amounts for the converted records. Verify that the tax calculations are as expected.

Converted transactions may have minor differences in before/after tax amounts. These occur because GTM uses a different calculation algorithm or rounding method than your source version. To synchronize the general ledger with converted transactions, record adjusting entries.

To eliminate ambiguity, the audit trail shows before and after values for purchasing, accounts payable, and accounts receivable records by their Progress database field name, not their screen label. For example, the audit trail displays voucher line types in the vod_type column.

As you interpret the audit trail for the transaction conversion, it can be helpful to refer to Table A.25 through Table A.29. These tables summarize the nature of before/after tax values.

Note In these tables, quotation marks indicate a value that cannot be translated.

Table A.25
USA to GTM, Purchasing Transactions

Status	Tax System	pod_taxable	prh_tax_at
Taxable	USA	Yes	“y”
	GTM	Yes	“Yes”
Non-taxable	USA	No	“n” or blank[1]
	GTM	No	Blank

1. An item is non-taxable if pod_taxable is n (No). If the transaction is non-taxable, the tax exemption code is optional.

Table A.26
USA to GTM, AP Voucher Receiver Lines

Status	Amt	Tax System	Tax	TxC	vod_type	vod_tax	vod_tax_at
Taxable	Item	USA	No	Blank	“R”	Blank	“y”
		GTM	Yes	Tax class	“R”	Blank	“Yes”
	Tax	USA	No	Blank	Blank	“y”	Blank
		GTM	No	Blank	“T”	“t”	“No”
Non-taxable	Item	USA	No	Exemption code or blank[1]	“R”	Blank	Blank
		GTM	No	Tax class	“R”	Blank	Blank
	Tax[2]	USA	No	Blank	Blank	“y”	Blank
		GTM	–	–	–	–	–

1. An item is non-taxable if its taxable status is No. If the transaction is non-taxable, the tax exemption code is optional.
2. The conversion deletes US tax lines resulting from non-taxable amounts.

Table A.27
USA to GTM, AP Voucher Memo Lines

Status	Amt	Tax System	Tax	TxC	vod_type	vod_tax	vod_tax_at
Taxable	Item	USA	No	Blank	Blank	Blank	Y
		GTM	Yes	Tax class	Blank	Blank	“Yes”
	Tax	USA	No	Blank	Blank	y	Blank
		GTM	No	Blank	“T”	“t”	“No”
Non-taxable	Item	USA	No	Blank	Blank	Blank	n or blank[1]
		GTM	No	Tax class	Blank	Blank	“No”
	Tax[2]	USA	No	Blank	Blank	y	Blank
		GTM	–	–	–	–	–

1. An item is non-taxable if its taxable status is n (No). If the transaction is non-taxable, the tax exemption code is optional.
2. The conversion deletes USA tax lines resulting from non-taxable amounts.

Table A.28
USA to GTM, AR Invoices

Status	Amt	Tax System	ard_tax	ard_tax_at	ard_taxc
Taxable	Item	USA	Blank	Blank	Blank
		GTM	Blank	Tax class	Tax class
	Tax	USA	Blank	Blank	Blank
		GTM	“t”	“No”	Blank
Non-taxable	Item	USA	Blank	Blank	Blank
		GTM	Blank	Tax class	Tax class
	Tax[1]	USA	–	–	–
		GTM	–	–	–

1. In the US tax system, no tax records are created for non-taxable amounts, so the conversion creates no new records for GTM.

Table A.29
USA to GTM, AR DR/CR Memos

Status	Amt	Tax System	ard_tax	ard_tax_at	ard_taxc
Taxable	Item	USA	Blank	“y”	Blank
		GTM	Blank	“Yes”	Tax class
	Tax	USA	“y”	Blank	Blank
		GTM	“t”	“No”	Blank
Non-taxable	Item	USA	Blank	“n”	Blank
		GTM	Blank	“No”	Tax class
	Tax[1]	USA	–	–	–
		GTM	–	–	–

1. In the US tax system, no tax records are created for non-taxable amounts, so the conversion creates no new records for GTM.

How the Conversion Changes Transaction Records

The following is a detailed description of how the transaction conversion updates the database.

The menu-level program for USA to GTM–Transactions is `txusatrn.p`. This program calls subprograms that set the GTM tax values in the individual database records. For all transactions, the conversion also generates corresponding tax detail records in the tx2d_det table.

When setting the GTM tax class value, these programs can set the existing tax exemption code value or retrieve an alternate value from a class file.

See “Defining Custom Tax Exemption Codes” on page 90.

Table A.30 lists the affected database tables and summarizes the changes.

Table A.30
Changes to Transaction Records

Tables	Summary of Changes
ard_det	For debit/credit memos, `txusaarm.p` sets ard_tax and ard_tax_at.
ar_mstr	For debit/credit memos, `txusaarm.p` sets ar_tax_env using `txtxeget.p`.
ca_mstr	`txusaca.p` sets ca_taxc and ca_tax_usage from ca_taxc or from the class file, if any. It also sets ca_tax_env using `txtxeget.p`.
idh_hist	`txusaari.p` sets idh_taxc and idh_tax_usage from idh_taxc or from the class file, if any. It also sets idh_tax_env using `txtxeget.p`.
ih_hist	`txusaari.p` sets ih_taxc and ih_tax_usage from ih_taxc or from the class file, if any. It also sets ih_tax_env using `txtxeget.p`.
itm_det	If itm_prefix is CA and itm_type is any value except INV, `txusaca.p` sets itm_taxc and itm_tax_usage from itm_taxc or from the class file, if any. It also sets itm_tax_env using `txtxeget.p`.
pod_det	If pod_taxable is Yes, `txusapo.p` sets pod_taxc to blank and sets pod_tax_env using `txtxeget.p`. If pod_taxable is No, `txusapo.p` sets pod_tax_env from po_tax_env.
po_mstr	`txusapo.p` sets po_tax_pct[1], po_tax_pct[2], and po_tax_pct[3] to 0. It sets po_tax_usage from ad_tax_usage and po_taxc from ad_taxc. It also sets po_tax_env using `txtxeget.p`.
prh_hist	`txusapo.p` sets prh_taxc to blank if prh_tax_at is “y.” It also sets prh_tax_env from pod_tax_env. If pod_taxable is Yes, `txusapo.p` sets prh_tax_at to “Yes”; otherwise, to blank.
qod_det	`txusaqo.p` sets qod_taxc and qod_tax_usage from qod_taxc or from the class file, if any. It also sets qod_tax_env using `txtxeget.p`.

Tables	Summary of Changes
qo_mstr	`txusaqo.p` sets qo_tax_pct[1], qo_tax_pct[2], and qo_tax_pct[3] to 0. `txusaqo.p` also sets qo_taxc and qo_tax_usage from qo_taxc or from the class file, if any. It also sets qo_tax_env using `txtxeget.p`.
rma_mstr	`txusarma.p` sets rma_taxc from rma_taxc or from the class file, if any.
sad_det	`txusasc.p` sets sad_taxc and sad_tax_usage from sad_taxc or from the class file, if any. `txusasc.p` sets sad_tax_env from sa_site and sa_taxc using `txtxeget.p`.
sa_mstr	`txusasc.p` sets sa_tax_pct[1], sa_tax_pct[2], and sa_tax_pct[3] to 0. It sets sa_taxc and sa_tax_usage from sa_taxc or from the class file, if any. It also sets sa_tax_env using `txtxeget.p`.
sfb_det	`txusaca.p` sets sfb_taxc and sfb_tax_usage from sfb_taxc or from the class file, if any. `txusaca.p` sets sfb_tax_env using `txtxeget.p`.
sod_det	`txusaso.p` sets sod_taxc and sod_tax_usage from sod_taxc or from the class file if any. `txusaso.p` sets sod_tax_env using `txtxeget.p`.
so_mstr	`txusaso.p` sets so_tax_pct[1], so_tax_pct[2], and so_tax_pct[3] to 0. `txusaso.p` sets so_taxc and so_tax_usage from so_taxc or from the class file, if any. `txusaso.p` sets so_tax_env using `txtxeget.p`.
tx2d_det	`txusapo.p` creates tax details for purchase orders (GTM transaction type 20), receivers (21), reconciliations (23), and returns (25). `txusaapv.p` creates tax details for vouchers (22) and recurring vouchers (32). `txusaapp.p` creates tax details for accounts payable tax on discount at payment (29). `txusaqo.p` creates tax details for sales quotes (10). `txusaso.p` creates tax details for invoiced service calls (38), return material authorizations (36), sales orders (11), and pending invoices (13). `txusaarm.p` creates tax details for debit/credit memos (18). `txusaari.p` creates tax details for invoices (16). `txusaarp.p` creates tax details for accounts receivable tax on discount at payment (19). `txusasc.p` creates tax details for service quotes (33) and service contracts (34).
vod_det	`txusaapv.p` sets vod_taxable, vod_taxc, vod_tax_usage, vod_tax, vod_tax_at, vod_type, and vod_tax_env. If vod_type is “r” and vod_tax_at is “Y”, vod_taxc is set to blank.
vo_mstr	For vouchers, `txusaapv.p` sets vo_tax_pct[1], vo_tax_pct[2], and vo_tax_pct[3] to 0. It also sets vo_taxable, vo_taxc, vo_tax_usage, and vo_tax_env.

Converting to GTM From No Taxes

If you are not using any tax method in your source database, follow the steps in “Converting US Taxes to GTM” on page 88 to convert master records and transactions details. The modifications to the process documented in that section are as follows:

USA to GTM Setup

Under “Implementing GTM” on page 89, when you run USA-to-GTM– Setup (2.13.22.5), use the following settings:

Table A.31
USA-to-GTM Setup

Field	Value
Delete Previous GTM	Yes
Convert Tax Masters	Yes

Field	Value
Country Code	Non-blank
Last Tax Code	USA00000
Display Status	Yes

Accept default values for all other fields.

Enter any new country codes entered in US-to-GTM Setup in Country Code Maintenance (2.14.1). In addition, create a country code called `All` with a name of `All`. The Name field in Country Code Maintenance must match the Country field in the address master table (`ad_mstr`).

USA to GTM Masters

Under “Converting Master Records” on page 96, run the following programs:

- Tax Zone Maintenance (2.13.3.13). Create a Tax Zone `All` with a country code of `All`. In addition, create tax zones for each country code. Set the option Sums-Into Tax Zone to `All`.
- Tax Environment (2.13.5.1). Create a non-taxable environment with the following setup:

Table A.32
Non-taxable Environment Setup

	Field	Value
	Tax Environment	non-tax
1st record:	Ship-From Zone	All
	Ship-To Zone	blank
2nd record:	Ship-From Zone	blank
	Ship-To Zone	All

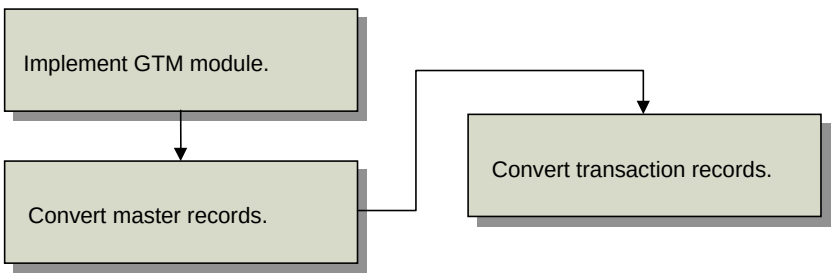
- Global Tax Management Control (2.13.24). Enter the default tax zone and tax environment. Set Tax Zone to the zone created for your country and set Tax Environment to `non-tax`.

Continue on with “USA to GTM–Masters” on page 97 and “USA to GTM–Transactions” on page 99.

Converting Canadian Taxes to GTM

This section describes how to convert to GTM from MFG/PRO’s Canadian tax system.

Fig. A.22
Canadian to GTM Conversion Process



The Canadian to GTM conversion process translates Canadian tax data to GTM equivalents and updates existing eB2.1 and later records. Figure A.22 summarizes the conversion workflow, which revolves around three sets of activities.

Implementing GTM. You run a setup program to implement GTM for Canadian tax processing.

Converting Master Records. You run a second program to populate database tables for customer and supplier addresses, items, product lines, and other master records with GTM data values.

Converting Transaction Records. You run a third program to populate transaction records. All transactions subject to tax are affected, including sales, purchasing, accounts payable, accounts receivable, and service/support management.

Note After each of these activities, it is crucial to review the corresponding reports and audit trails. Mistakes can be pervasive and costly.

Table A.33 lists the eB2.1 and later programs used during the conversion.

Table A.33
eB2.1 and Later Programs Used to Convert Canadian Taxes to GTM

Activity	eB2.1 and Later Programs
Implementing GTM	CAN to GTM–Setup (2.13.22.9)
Converting master records	CAN to GTM–Masters (2.13.22.10)
Converting transaction records	CAN to GTM–Transactions (2.13.22.11)

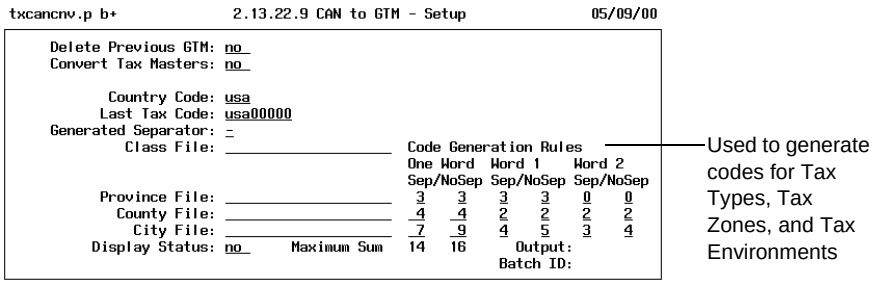
Implementing GTM

An automated setup program creates most of the codes needed to implement GTM, based on how your Canadian taxes are defined. Before executing this program, you should understand the options it provides and the default logic it uses.

Code Generation Rules

GTM codes for tax types, tax zones, and tax environments consist of text strings that uniquely identify the province, county, and city of a tax jurisdiction. In Canada, manual setup of these codes would be a tedious process, since there are thousands of them.

Fig. A.23
Code Generation Rules in CAN to GTM–Setup (2.13.22.9)



Therefore, by default, CAN to GTM–Setup generates codes based on its code generation rules (Figure A.23). These rules systematically select characters from the province code, county name, and city name in the tax master.

To determine if generated codes are suitable for your company, run the setup and review the audit trail. If you need a different coding scheme, read the rest of this section and change the settings for the code generation rules as necessary. Then, rerun the setup with Delete Previous GTM set to Yes and Convert Tax Masters set to Yes.

For Canadian taxes, the default generated code format for tax zones, tax types, and tax environments is:

```
PPP-CCCC-ccccccc
```

PPP is the 3-character province code, *CCCC* the 4-character county or district name, and *ccccccc* the 7-character city name. Each text string is separated by a dash (–).

The rules used to determine the characters to select depend on:

- Whether the Canadian code or name used to generate the text string consists of one word or multiple words (text strings separated by blank spaces).
- Whether separator characters are used.

Table A.34 lists the default number of characters for each text string under the different conditions. However, you can change the number of characters and use a different separator or no separator, as long as the total number of generated characters—including the separator and any ending integers—is 16 or less.

Table A.34
Code Generation Rules in CAN to GTM–Setup (2.13.22.9)

			Multiple Words			
	One Word		Word 1		Word 2	
Code	**Sep**	**No Sep**	**Sep**	**No Sep**	**Sep**	**No Sep**
Province	3	3	3	3	0	0
County	4	4	2	2	2	2
City	7	9	4	5	3	4
Maximum Total Characters, With Separator						14
Maximum Total Characters, Without Separator						16

Example If the original province code for Alberta is ALBA, the generated text string is ALB regardless of whether separators are used. For the city of Southampton, the generated text string is Southam if separators are used and Southampt if they are not. For the city of Thunder Bay, the generated text string is ThunBay if separators are used and ThundBay if they are not.

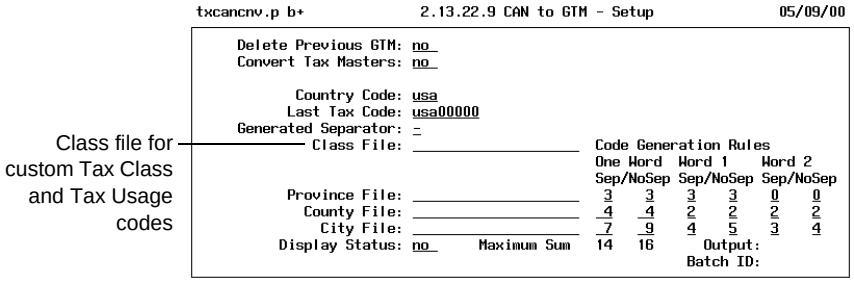
Setup retains the capitalization from the original Canadian code or name. If the original code or name contains punctuation such as a period, the code generation rules treat it the same as any other non-blank character.

Defining Custom Tax Class and Tax Usage Codes

By default, the setup generates tax classes that directly correspond to your Canadian GST master records. To distinguish the two sets of codes, it appends the letter P to the GST + PST code. For example, if your current GST classes are 0, 1, and 2, the generated GTM classes are 0, 1, and 2 (for GST only) and 0P, 1P, and 2P (for GST + PST). The system does not automatically generate tax usages.

You can override this by creating your own map for the setup program and specifying it in the Class File field. The same class file is also referenced in the programs that convert master data and transaction records.

Fig. A.24
Class File in CAN to GTM–Setup (2.13.22.9)



You should create a class file if:

- Your company plans to change its tax class codes during the GTM conversion. In this situation, you must define custom codes for GST only and GST + PST.
- Within a tax class, a company can be taxed based on its nature of operation or the way it intends to use an item. Tax usage codes identify these conditions in GTM.

The class file is an ASCII file with text strings in the following format:

```
"Current GST Class" "GST Class" "GST Usage" "Class for GST + PST" "Usage for GST + PST"
```

GTM tax classes can have a maximum of three characters, and tax usage codes eight characters. A null string (" " or "") represents an unused value.

The class file can have any name or extension. However, commas instead of blank characters must separate code values in .csv files. The file must be located in the home directory for the Progress session. A .csv file is a Windows comma-separated values file format that saves values recorded in a spreadsheet.

Example Your current GST codes are 0, 1, and 2, and you want to change them to A, B, and C, plus add a G to indicate GST only.

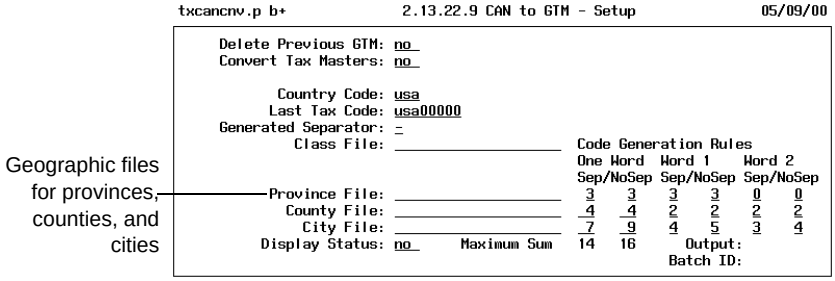
```
"0" "AG" "" "A" ""
"1" "BG" "" "B" ""
"2" "CG" "" "C" ""
```

Defining Custom Codes for Provinces, Counties, and Cities

The generated codes for tax zones, tax environments, and tax types consist of text strings that identify the province, county, and city. By default, the code generation rules define the structure of these text strings. However, if you need a different naming convention, you can create geographic files for province codes, county names, and/or city names.

See "Code Generation Rules" on page 106.

Fig. A.25
Province, County, and City Files in CAN to GTM–Setup (2.13.22.9)



Create a separate file for each kind of text string and reference the file in CAN to GTM–Setup as shown in Figure A.25.

Note You only need to define codes for conditions not already supported by code generation rules.

A geographic file is the same as a class file, except that it uses the following format:

```
"Current Code or Name" "GTM Text String"
```

Example To map province codes for the provinces of Manitoba and Quebec, create a province file with these lines:

```
"MANI" "MB"
"QUE" "QB"
```

Processing Logic

This program creates eB2.1 and later records as described in Table A.35.

Table A.35
New GTM Records

Type of Record	Explanation
Tax zones	Based on code generation rules or a class file, setup builds the tax zone hierarchy for the country and all province/county/city combinations used in your current system.
Tax types	Based on code generation rules or a class file, setup generates tax types for the province/county/city combinations used in your current system.
Tax environments	Based on code generation rules or a class file, setup generates tax environments for all ship-to tax zones.
Tax rates	Based on code generation rules or a class file, setup generates tax rates for the tax jurisdictions and percentages used in your current system.
Tax classes	By default, the setup generates tax classes based on your Canadian GST master records, in which GST and PST are represented as separate codes. However, the GTM setup generates tax classes for GST only and GST + PST. It sets up PST only as GST + PST with a GST tax rate of 0%. It sets up non-taxable as GST only with GST and PST tax rates of 0%. To distinguish the generated codes, the setup appends the letter P to the GST + PST code. For example, if your current GST classes are 0, 1, and 2, the generated GTM classes are 0, 1, and 2 (for GST only) and 0P, 1P, and 2P (for GST + PST). The system does not automatically generate tax usages. You can choose to bypass the default setup behavior by defining a class file. For more information, see “Defining Custom Codes for Provinces, Counties, and Cities” on page 108.
Tax usages	By default, setup does not generate tax usages. However, you can do this with a class file.

Type of Record	Explanation
Country code	The setup generates a record for the default country code you specify when you run the setup. This country code is the top-level zone in the tax zone hierarchy.
Company addresses and address list types	In GTM, company sites require a corresponding company address record because taxes are calculated by address, not site. The setup verifies that each company site has an address record and creates any missing ones, along with any needed address list type records. The setup also creates a ~taxes address record to provide a default tax address whenever a transaction is missing a company site code.

GTM Control Settings

Setup defines Global Tax Management Control (2.13.24) as described in Table A.36.

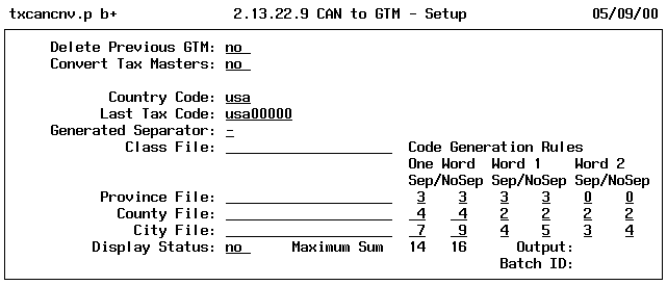
Table A.36
Updates to GTM Control Settings

Field	Explanation
Country Code	As specified during the setup
Tax Method	01
Tax-By-Line	No
Accrue Tax at Receipt	Yes
Discount Tax at Invoice	No
Discount Tax at Payment	No
Last Tax Code	As specified during the setup
Rounding Method	Value specified in System/Account Control (36.1)

CAN to GTM–Setup

Based on your implementation decisions, use CAN to GTM–Setup (2.13.22.9) to set up GTM for Canadian tax processing.

Fig. A.26
CAN to GTM–Setup (2.13.22.9)



Warning As noted previously, the purpose of the conversion is to replicate your existing tax processing setup in GTM. Do not attempt to implement new GTM functionality until after the entire conversion is complete. Do not change settings in the new GTM records or in Global Tax Management Control (2.13.24). If you do, the master and/or transaction conversions may fail.

Delete Previous GTM. This option determines whether the setup deletes previously created GTM records from the database. If you select this option, the setup deletes tax zones, tax types, tax environments, transaction tax details, and other GTM records from the database.

- Enter No if you have not yet converted your database to GTM.
- Enter Yes to clean up the database if it contains GTM records from unsuccessful conversion or installation attempts.

Convert Tax Masters. This option determines whether the setup generates the GTM master records from the master tables for Canadian GST and PST.

- Enter Yes to create GTM records corresponding to Canadian tax masters: tax classes, tax types, tax zones, tax environments, tax rates, and so on. The audit trail shows the new GTM records.
- Enter No if you only want to delete previous GTM records and do not want the setup program to generate new GTM records.

Country Code. This country code is the top-level tax zone in the tax zone hierarchy. All other tax zones sum into this one.

If Global Tax Management Control already has a country code, it displays here. Otherwise, the setup sets the default country code to CAN. If you override the value here, the setup assigns it to the control program.

Last Tax Code. Enter a value to update the corresponding field in Global Tax Management Control. In GTM, tax codes identify individual tax rates. Codes are generated sequentially based on the value of Last Tax Code in the GTM control program.

The default Last Tax Code is an 8-character value that consists of the GTM country code and a right-justified integer with placeholder zeros. For example, for country code CAN, the default Last Tax Code is CAN00000. The system assigns the number CAN00001 to the first tax rate record created in GTM and increments this number for subsequent rates.

If you want tax codes to have a different format, enter a different prefix. Codes display alphanumerically in screens and reports. Tax codes that are totally numeric are left-justified and have no placeholder zeros. For example, codes 1 through 30 display in a report column as follows:

```
1
...
19
2
20
...
30
```

Generated Separator. Enter a character to use as a separator in system-generated tax zones, types, and environments. Using a separator can improve the readability of the component elements of these codes.

The default separator is the dash (–), but you can enter any character. A sample GTM code that uses the dash separator is BC–Van for Vancouver, British Columbia. If you do not want to use separators in codes, enter blank. However, you cannot use blank as a separator character.

Class File. To provide custom mapping of Canadian tax classes to GTM classes and usage codes, specify an ASCII file with conversion information.

Province, County, City File. To override default code generation rules, specify specific values for geographic locations in an ASCII file.

Code Generation Rules. Enter appropriate values for your organization.

Display Status. This setting determines whether the system displays status messages online during the conversion. These messages list database tables and their indexes as they are converted. If you select this option, messages display on the screen and the printed report.

Updates to Company Addresses

In GTM, company sites require a corresponding company address record because taxes are calculated by address, not site. The setup creates any missing company address records for company sites. However, the setup does not populate these new address records with the city, county, province, and country. You must supply this information manually in Company Address Maintenance (2.12).

Also set up tax zone codes to support these new addresses if the setup did not already generate codes for these tax jurisdictions. Do this in Tax Zone Maintenance (2.13.3.13). Then, assign the tax zone to the address.

Set Up Audit Trail

CAN to GTM–Setup prints a report of updated tax master records for GST and PST rates. For GST rates, the report shows the GST class, description, starting and ending effective dates, the GST percent, and the general ledger tax accounts for accounts payable and accounts receivable. For PST, the report shows the province/county/city, tax effective date, tax rate for the effective date, whether PST is based on GST, and the generated GTM tax zone and sums-into zone.

Figure A.27 and Figure A.28 show the audit trail formats.

Fig. A.27
GST Audit Trail

```
Processing: Create GTM GST from GST masters

GST Class Description Start Eff End Eff   GST Pct AP GST Acct AR GST Acct
--------- ----------- --------- --------- -------- ----------- -----------
0         GST 0       01/01/97              0.00% 1400        2400
1         GST 1       01/01/97              5.00% 1400        2400
2         GST 2       05/29/97  12/31/99    7.00% 1400        2400
3         GST 3       07/02/97  12/31/99   10.00% 1400        2400
```

Fig. A.28
PST Audit Trail

```
Processing: Create GTM PST from PST masters
                                                                  Sums-Into
Province County City        Effective  Tax   On GST Tax Zone     Tax Zone
-------- ------ ----------- --------- ------ ------ ------------ ---------
ALB             Calgary     01/01/96  0.00%   No    ALB-Calgary  ALB
BC              Vancouver   01/01/96  7.00%   Yes   BC-Vancouv   BC
NFL             Bay Roberts 01/01/96  8.00%   No    NFL-BayRob   NFL
ONT             Bala        01/01/96  8.00%   Yes   ONT-Bala     ONT
PEI             Souris      01/01/96  10.00%  Yes   PEI-Souris   PEI
```

Troubleshooting GTM Setup

After you run CAN to GTM–Setup, verify that the GTM setup is correct before you continue with the conversion. The problems listed in Table A.37 can cause errors or unexpected values. Before you proceed to the master conversion, review the audit trail, the GTM reports for the new records, and Global Tax Management Control settings. Correct any problems before going on.

Subsequent setups do not automatically overwrite records created by previous ones. To set up new records, you must first delete the old ones. If you rerun the setup, you must remove the records created by the earlier setup attempt. To do this, rerun the setup with Delete Previous GTM set to Yes and Convert Tax Masters set to Yes.

Note If you must rerun the setup after you have run any of the other conversion programs, restore the database first. Then, rerun the setup and any other conversion programs you ran previously. This is necessary to perpetuate changes to master data, transactions, and tax details.

Table A.37
Troubleshooting the GTM Setup

Error	Explanation
Tax system must be Canadian.	The CAN to GTM setup can only be run on a Canadian tax system.
Must delete previous GTM when converting.	When you set Convert Tax Masters to Yes, you must also set Delete Previous GTM to Yes.
On GST has changed, cannot convert prior to this date.	The setup can convert only the current tax environment, not previous variations. If a city's GST Only status changed in the span of time included in the conversion, the setup creates tax environments only for current conditions.

Warning Do not correct records programmatically. This approach often causes additional problems.

How the Setup Changes GTM Records

The following is a technical description of how the setup updates the database.

The menu-level program for CAN to GTM–Setup is `txcancnv.p`. This program calls subprograms (primarily `txcantax.p`) that set the GTM tax values in the individual database tables.

Table A.38 lists the affected database tables and summarizes the changes.

Table A.38
Changes to GTM Records

Tables	Summary of Changes
ad_mstr	`txcantax.p` creates one ~taxes record for the database. It also scans si_mstr and creates an address record for any company site that does not already have one.
code_mstr	For each vt_mstr record, `txcantax.p` creates tax classes for GST only and GST + PST. If a class file is referenced, it creates the specified tax classes and tax usages, if any. For the GST rate in vt_mstr and each PST rate in tax_mstr, `txcantax.p` creates a tax type code. For non-taxable transactions, it also creates a default NON-TAX tax type. Finally, for each ship-to tax zone, `txcantax.p` creates a tax environment and assigns it the tax types associated with the tax zone.
ctry_mstr	`txcantax.p` creates a record for the default country code specified in the selection data.
ls_mstr	`txcantax.p` creates ls_mstr record for each new ad_mstr record, if any.
tx2_mstr	For each vt_mstr record, `txcantax.p` creates a tax rate for GST only and GST + PST. For each tax_mstr record, `txcantax.p` creates multiple tx2_mstr records, one for each combination of (1) GST only and GST + PST and (2) each GST class. (For example, if there are three GST classes, the setup generates six tx2_mstr records.) It also runs `txtx2_nt.i` to create a non-taxable tax rate and `txtxmeth.i` to create tax method 01.
txc_ctrl	`txcantax.p` sets txc_ctry_code and txc_tax_code from the selection data. It sets txc_method to 01, txc_by_line to Yes, txc_inv_disc and txc_pmt_disc to No, and txc_rcpt_tax_point to Yes.

Tables	Summary of Changes
txbd_det	`txcantax.p` creates a tax base record for PST + GST.
txe_mstr	`txcantax.p` creates tax environment zone detail records for every tax environment code it generates for code_mstr.
txed_det	`txcantax.p` creates tax environment tax type detail records for every tax environment code it generates for code_mstr.
txz_mstr	`txcantax.p` creates a top-level sums-into tax zone for the new ctry_mstr record. For each province/county/city combination in tax_mstr, it creates a ship-to tax zone.

Converting Master Records

Once you finish the GTM setup, the next activity is to update tax settings in the following master records:

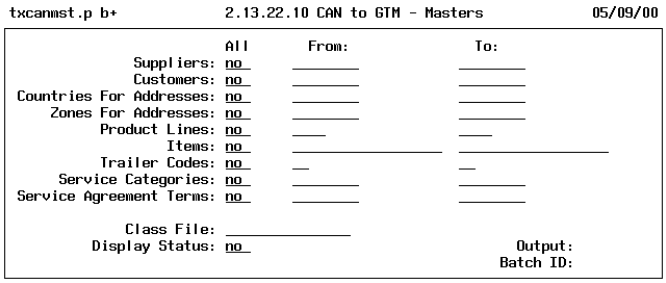
- Suppliers
- Customers
- Product lines
- Items
- Trailer codes
- Service categories
- Service agreement terms

GTM has additional fields and may require new values for existing fields.

CAN to GTM–Masters

Use CAN to GTM–Masters (2.13.22.10) to convert master records.

Fig. A.29
CAN to GTM–Masters (2.13.22.10)



In addition to updating the tax settings in the master records, this program assigns tax zone codes to supplier, customer, and company address records. For verification of changes, the program generates an audit trail.

Important Before you run CAN to GTM–Masters, do the following:

- Run CAN to GTM–Setup (2.13.22.9).
- To avoid record-contention conflicts with other users, shut down the database and restart it when no other users are on the system.

You can convert all records, a range of records, or individual records. This program converts records in the order in which their selection options display on the screen. For separate audit trail reports, run the report separately for each type of record.

If you created a class file during the setup step, specify its name in the Class File field. Display Status, Output, and Batch ID are the same as in the setup program.

See “Defining Custom Tax Class and Tax Usage Codes” on page 107.

Master Audit Trail

CAN to GTM–Master prints a report of changed records. The format varies depending on the records included in the conversion. For each group of converted records, the report shows the record number and name followed by the before and after tax information, such as country code, tax zone code, GST and PST taxable status, whether tax is included in item amounts, tax class, and tax usage.

Groups of converted records print in the same order as the screen selection criteria with a page break separating each group. Warning and error messages identify potential conversion problems.

Figure A.30 and Figure A.31 show representative audit trail formats.

Fig. A.30
Customers Audit Trail

```
Processing: Customers

Address  Name                                    GST PST Tax In TxC TaxUsage
-------- ----------------------------            --- --- ------ --- --------
10000000 Harris Steel                Before      No  No  No     0
                                     After       No      No     0
10000001 Computer Services           Before      Yes Yes No     1
                                     After       Yes     No     1P
10000002 Niagara Byteworks           Before      Yes Yes No     2
                                     After       Yes     No     2P
```

Fig. A.31
Countries for Addresses Audit Trail

```
Processing: Countries For Addresses

Address  Name                                    Ctry Country
-------- ----------------------------            ---- -------
10000000 Harris Steel                Before           Canada
                                     After       CAN  Canada
10000001 Computer Services           Before           Canada
                                     After       CAN  Canada
10000002 Niagara Byteworks           Before           Canada
                                     After       CAN  Canada
```

Troubleshooting Master Conversions

The error messages in the audit trail identify conditions you should analyze and address before you convert transactions. Table A.39 lists some common problems along with explanations. Before you make corrections, restore the database from backup.

Warning Do not proceed to the transaction conversion until the master conversion audit trail is free of errors.

Table A.39
Troubleshooting the Master Conversion

Error	Explanation
Tax class cannot be converted.	Class file does not contain a GST class that matches the one in the master record.
GST class must exist 0%.	GST class master must have at least one GST class with a zero percentage.
Blank tax class not allowed.	GST class is blank in the class file.
Tax class cannot exceed 3 characters (*xxx*).	GST class in the class file is longer than three characters. The message shows the first three characters.
Tax class does not exist (*x*).	GST class in the class file does not exist in the GST master.
Tax class is not unique (*x*).	GST class occurs in multiple places in the class file.
Tax class does not exist (*xxx*).	GTM tax class in the class file does not exist in the GTM tax class master.
Tax usage cannot exceed 8 characters (*xxxxxxxx*).	GTM tax usage in the class file is longer than eight characters. Message shows the first eight characters.
Tax usage does not exist (*xxxxxxxx*).	GTM tax usage in the class file does not exist in the GTM tax usage master.
Tax class/tax usage combination is not unique (*xxx xxxxxxxx*).	GTM tax class and tax usage combination occurs in multiple places in the class file.
Conversion will ignore tax class (*x*).	Class file does not have an entry for a GST class that is in the class master.
x, *xxx*, and *xxxxxxxx* are placeholders for the actual codes displayed in the error message.	

How the Conversion Changes Master Records

The following is a detailed description of how the master conversion updates the database.

The menu-level program for CAN to GTM–Masters, `txcanmst.p`, calls subprograms that set the GTM tax values in the individual tables.

When setting the GTM tax class value, these programs can set the existing tax class code or retrieve an alternate value from a class file. See “Implementing GTM” on page 106.

Table A.40 lists the affected database tables and summarizes the changes.

Table A.40
Changes to Master Records

Tables	Summary of Changes
ad_mstr	In supplier records, `txcanvd.p` sets ad_taxable from vd_taxable. It also sets ad_taxc and ad_tax_usage from ad_taxc or from the class file, if any. If vd_taxable is No, ad_taxc is GST only; otherwise, GST + PST. In customer records, `txcancm.p` sets ad_taxable from cm_taxable and ad_tax_in from cm_tax_in. It also sets cm_taxc and ad_tax_usage from cm_taxc or from the class file, if any. If cm_pst is No, the tax class is GST only; otherwise, GST + PST. In all address records, `txcanct.p` sets ad_ctry from ad_country and visa versa. `txcanzn.p` calls `txtxzget.p` to set ad_tax_zone.
fsc_mstr	`txcanfsc.p` sets fsc_taxc from fsc_taxc or from the class file if any. Tax class is set to GST only.
pl_mstr	`txcanpl.p` sets pl_taxc from pl_taxc or from the class file, if any. If pl_pst is No, the tax class is GST only; if it is Yes, to GST + PST.
pt_mstr	`txcanpt.p` sets pt_taxc from pt_taxc or from the class file, if any. Tax class is set to GST only.

Tables	Summary of Changes
sv_mstr	`txcansv.p` sets sv_taxc and sv_tax_usage from sv_taxc or from the class file, if any. Tax class is set to GST only.
trl_mstr	`txcantrl.p` sets trl_taxc from trl_taxc or from the class file, if any. It also sets trl_taxable to Yes. If trl_pst is No, the tax class is GST only; if it is Yes, to GST + PST.

Converting Transaction Records

Once you finish converting master records, you can convert the following transaction records:

- Purchase orders and receipts
- Accounts payable vouchers and payments
- Service contracts, calls, and return material authorizations
- Sales quotes and orders
- Accounts receivable memos, invoices, and payments

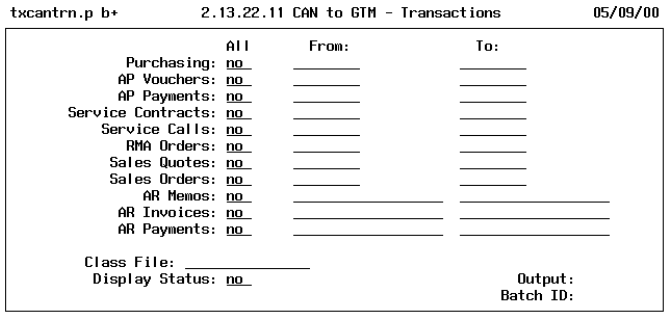
In GTM, every transaction subject to tax has a transaction tax detail record. This record stores the information used to calculate tax.

Note If you already have GTM transaction records in the database when you perform transaction conversion, the conversion process updates them using the current default tax values. Values replaced in these records include tax environment, class, usage, and so on.

CAN to GTM–Transactions

Use CAN to GTM–Transactions (2.13.22.11) to convert existing transaction records.

Fig. A.32
CAN to GTM– Transactions (2.13.22.11)



In addition to updating transactions, this program generates an audit trail for verification of changes.

Important Before you run CAN to GTM–Transactions, do the following:

- Run CAN to GTM–Setup (2.13.22.9) and CAN to GTM–Masters (2.13.22.10).
- To avoid record-contention conflicts with other users, shut down the database and restart when no other users are on the system.

You can convert all records, a range of records, or individual records. This program converts records in the same order they display on the screen.

Note For separate audit trail reports, run the report separately for each type of record.

In some cases, the record sequence is important. Purchasing transactions must be converted before accounts payable vouchers and vouchers before payments. Accounts receivable memos and invoices must be converted before payments.

If you created a class file during the setup step, specify its name in the Class File field. Display status, output, and Batch ID are the same as in the setup program.

See “Defining Custom Tax Class and Tax Usage Codes” on page 107.

Transaction Audit Trail

The transaction conversion prints a report of changed records. The format varies depending on the records included in the conversion. For each group of converted records, the report shows the transaction number and name followed by the before and after tax information for each line item, such as taxable status, tax environment, tax class, and tax usage. Groups of converted records print in the same order as the screen selection criteria with a page break separating each group.

Warning and error messages identify potential conversion problems. Messages that appear at the end of a transaction apply to the entire transaction; those that appear between the Before and After line apply only to that line.

Figure A.33 and Figure A.34 show representative audit trail formats.

Fig. A.33
Accounts Payable Audit Trail

```
Processing: AP Vouchers

Ref Type Supplier Ln       GST TxC TaxUsage Tax Env vod_type vod_tax vod_tax_at
--- ---- -------- --       --- --- -------- ------- -------- ------- ----------
105 VO   32487432    Before No
                     After  Yes 0P           PEI-GG
                   1 Before No  1                    R               1
                     After  Yes 1            PEI-GG  R               Yes
                   2 Before No                              1
                     After  No                        T       t       No
```

Fig. A.34
Service Calls Audit Trail

```
Processing: Service Calls

                         Cust
Call ID Call/SR Cust     PST   Ln Record            GST PST TxC TaxUsage Tax Env
------- ------- -------- ---- --- ---------        --- --- --- -------- -------
CA124           10000000 Yes    1 Call      Before   Yes     1
                                            After    Yes     1P           NB
CA125           10000001 Yes    1 Item      Before   Yes Yes 1
                                            After    Yes     1P           NB
CA126           10000002 Yes    1 Billing   Before   Yes Yes 1
                                            After    Yes     1P           NB
                                2 Billing   Before   No  Yes 0
                                            After    Yes     1P           NB
```

Troubleshooting Transaction Conversions

The error messages in the audit trail identify conditions you should analyze and address before you resume live GTM processing. Table A.41 lists some common problems along with explanations. Before you make corrections, restore the database from backup.

Warning Do not resume live processing until the transaction conversion audit trail is free of errors. Do not correct transaction records programmatically. This approach often causes additional problems.

In addition to examining the audit trail, it is advisable to review the Tax Detail by Transaction Report (2.13.15.3). This report shows the tax environments, tax types, and tax amounts for the converted records. Verify that the tax calculations are as expected.

Converted transactions may have minor differences in before/after tax amounts. These occur because GTM uses a different calculation algorithm or rounding method than your current system. To synchronize the general ledger with the converted transactions, record adjusting entries.

Table A.41
Troubleshooting the Transaction Conversion

Error	Explanation
GST class must exist 0%.	The GST class master must have at least one GST class with a zero percentage.
Freight, brokerage, or duty charges cannot be converted.	There is no equivalent ability in GTM.
Detail tax environment must match header.	In accounts payable vouchers and accounts receivable debit/credit memos, the tax environment must be the same in both the header and detail lines.

To eliminate ambiguity, the audit trail shows before and after values for purchasing, accounts payable, and accounts receivable records by their Progress database field name, not their screen label. For example, the audit trail displays voucher line types in the vod_type column.

As you interpret the audit trail for the transaction conversion, it can be helpful to refer to Table A.42 through Table A.46. These tables summarize the nature of before/after tax values.

Note In these tables, quotation marks indicate a value that cannot be translated.

Table A.42
CAN to GTM, Purchasing Transactions

Status	Tax System	pod_taxable	prh_tax_at
Taxable	CAN	Yes	Tax class
	GTM	Yes	“Yes”
Non-taxable	CAN	No	“0”
	GTM	Yes	“Yes”

Table A.43
CAN to GTM, AP Voucher Receiver Lines

Status	Amt	Tax System	GST	TxC	vod_type	vod_tax	vod_tax_at
Taxable	Item	CAN	Yes	GST class	“R”	Blank	GST class
		GTM	Yes	Tax class	“R”	Blank	“Yes”
	Tax	CAN	No	Blank	Blank	GST class	Blank
		GTM	No	Blank	“T”	“t”	“No”
Non-taxable	Item	CAN	No	Blank	“R”	Blank	0% GST class
		GTM	Yes	Tax class	“R”	Blank	Blank
	Tax[1]	CAN	No	Blank	Blank	GST class	Blank
		GTM	–	–	–	–	–

1. The conversion deletes Canadian tax lines resulting from non-taxable amounts.

Table A.44
CAN to GTM, AP Voucher Memo Lines

Status	Amt	Tax System	GST	TxC	vod_type	vod_tax	vod_tax_at
Taxable	Item	CAN	Yes	Blank	Blank	Blank	GST class
		GTM	Yes	Tax class	Blank	Blank	“Yes”
	Tax	CAN	No	Blank	Blank	GST class	Blank
		GTM	No	Blank	“T”	“t”	“No”
Non-taxable	Item	CAN	No	Blank	Blank	Blank	0% GST class
		GTM	Yes	Tax class	Blank	Blank	“No”
	Tax[1]	CAN	No	Blank	Blank	GST class	Blank
		GTM	–	–	–	–	–

1. The conversion deletes Canadian tax lines resulting from non-taxable amounts.

Table A.45
CAN to GTM, AR Invoices

Status	Amt	Tax System	ard_tax	ard_tax_at	ard_taxc
Taxable	Item	CAN	Blank	GST class	Blank
		GTM	Blank	Tax class	Tax class
	Tax	CAN	GST class or blank[1]	Blank	Blank
		GTM	“t”	“No”	Blank
Non-taxable	Item	CAN	Blank	0% GST class	Blank
		GTM	Blank	Tax class	Tax class
	Tax[2]	CAN	0% GST class	Blank	Blank
		GTM	–	–	–

1. Blank for PST.
2. The conversion deletes Canadian tax lines resulting from non-taxable amounts.

Table A.46
CAN to GTM, AR DR/CR Memos

Status	Amt	Tax System	ard_tax	ard_tax_at	ard_taxc
Taxable	Item	CAN	Blank	GST class	Blank
		GTM	Blank	“Yes”	Tax class
	Tax	CAN	GST class[1]	Blank	Blank
		GTM	“t”	“No”	Blank
Non-taxable	Item	CAN	Blank	0% GST class	Blank
		GTM	Blank	“No”	Tax class
	Tax[2]	CAN	–	–	–
		GTM	–	–	–

1. DR/CR memos do not calculate PST.

2. The conversion deletes Canadian tax lines resulting from non-taxable amounts.

How the Conversion Changes Transaction Records

The following is a technical description of how the transaction conversion updates the database.

The menu-level program for CAN to GTM–Transactions is `txcantrn.p`. This program calls subprograms that set the GTM tax values in the individual database tables. For all transactions, the conversion also generates corresponding tax detail records in tx2d_det.

When setting the GTM tax class value, these programs set the existing tax class code or retrieve an alternate value from a class file. See “Defining Custom Tax Class and Tax Usage Codes” on page 107.

Table A.47 lists the affected database tables and summarizes the changes.

Table A.47
Changes to Transaction Records

Tables	Summary of Changes
ard_det	For debit/credit memos, `txcanarm.p` sets ard_tax and ard_tax_at. It also sets ard_taxc (GST only), ard_tax_usage, and ard_tax_at.
ar_mstr	For debit/credit memos, `txcanarm.p` sets ar_tax_env.
ca_mstr	`txcanca.p` sets ca_taxc and ca_tax_usage from ca_taxc or from the class file, if any. If cm_pst is No, ca_taxc is GST only; otherwise, it is GST + PST. It also sets ca_tax_env using `txtxeget.p`.
idh_hist	`txcanari.p` sets idh_taxc and idh_tax_usage from idh_taxc or from the class file, if any. It sets idh_taxable to Yes. If idh_pst is No, idh_taxc is GST only; otherwise, it is GST + PST. It also sets idh_tax_env using `txtxeget.p`.
ih_hist	`txcanari.p` sets ih_taxable to Yes and ih_pst_pct to 0. It sets ih_taxc (GST only) and ih_tax_usage from ih_taxc or from the class file, if any. It also sets ih_tax_env using `txtxeget.p`.
itm_det	`txcanca.p` sets itm_taxc (GST + PST) and itm_tax_usage from itm_taxc or from the class file, if any. It sets itm_taxable to Yes. It also sets itm_tax_env using `txtxeget.p`.
pod_det	`txcanpo.p` sets pod_taxable to Yes and pod_tax_env using `txtxeget.p`. If pod_pst is No, then pod_taxc is 0; otherwise, pod_taxc is 0P.
po_mstr	`txcanpo.p` sets po_tax_pct[1], po_tax_pct[2], and po_tax_pct[3] to 0. It sets po_taxable to Yes. It sets po_taxc and po_tax_usage from the ad_taxc value for the corresponding supplier. It also sets po_tax_env using `txtxeget.p`.
prh_hist	`txcanpo.p` sets prh_taxc from pod_taxc, prh_tax_usage from pod_tax_usage, and prh_tax_env from pod_tax_env. Since po_taxable is Yes, it sets prh_tax_at to Yes.
qod_det	`txcanqo.p` sets qod_taxable to Yes. It sets qod_taxc and qod_tax_usage from qod_taxc or from the class file, if any. It sets qod_taxable to Yes. If qod_pst is No, tax class is GST only; otherwise, it is GST + PST. It also sets qod_tax_env using `txtxeget.p`.
qo_mstr	`txcanqo.p` sets qo_tax_pct[1], qo_tax_pct[2], and qo_tax_pct[3] to 0. It sets qo_taxable to Yes and qo_pst_pct to 0. It sets qo_taxc and qo_tax_usage from qo_taxc or from the class file, if any. If cm_pst is No, tax class is GST only; otherwise, it is GST + PST. it also sets qo_tax_env using `txtxeget.p`.
rma_mstr	`txcanrma.p` sets rma_taxc from rma_taxc or from the class file, if any. It sets rma_taxable to Yes. If cm_pst is No, tax class is GST only; otherwise, it is GST + PST.
sad_det	`txcansc.p` sets sad_taxc and sad_tax_usage from sad_taxc or from the class file, if any. It sets sad_taxable to Yes. If sad_pst is No or sa_prefix is QA, tax class is GST only; otherwise, it is GST + PST. It also sets sad_tax_env from sa_site and sa_taxc using `txtxeget.p`.

Table A.47 — *Changes to Transaction Records* (Page 1 of 2)

Tables	Summary of Changes
sa_mstr	`txcansc.p` sets sa_tax_pct[1], sa_tax_pct[2], and sa_tax_pct[3] to 0. It sets sa_taxable to Yes and sa_can_tax to 0. It sets sa_taxc and sa_tax_usage from sa_taxc or from the class file, if any. If cm_pst is No, tax class is GST only; otherwise, it is GST + PST. It also sets sa_tax_env using `txtxeget.p`.
sfb_det	`txcanca.p` sets sfb_taxc from itm_taxc and sfb_tax_usage from itm_tax_usage. It sets sfb_taxable to Yes. It also sets sfb_tax_env using `txtxeget.p`.
sod_det	`txcanso.p` sets sod_taxc and sod_tax_usage from sod_taxc or from the class file if any. It sets sod_taxable to Yes. If sod_pst is No, tax class is GST only; otherwise, it is GST + PST. It also sets sod_tax_env using `txtxeget.p`.
so_mstr	`txcanso.p` sets so_taxable to Yes and so_pst_pct to 0. It sets so_tax_pct[1], so_tax_pct[2], and so_tax_pct[3] to 0. It sets so_taxc and so_tax_usage from so_taxc or from the class file, if any. If cm_pst is No, tax class is GST only; otherwise, it is GST + PST. It also sets so_tax_env using `txtxeget.p`.
tx2d_det	`txcanpo.p` creates tax details for purchase orders (GTM transaction type 20), receivers (21), reconciliations (23), and returns (25). `txcanapv.p` creates tax details for vouchers (22) and recurring vouchers (32). `txcanapp.p` creates tax details for accounts payable tax on discount at payment (29). `txcanqo.p` creates tax details for sales quotes (10). `txcanso.p` creates tax details for invoiced service calls (38), return material authorizations (36), sales orders (11), and pending invoices (13). `txcanarm.p` creates tax details for debit/credit memos (18). `txcanari.p` creates tax details for invoices (16). `txcanarp.p` creates tax details for accounts receivable tax on discount at payment (19). `txcansc.p` creates tax details for service quotes (33) and service contracts (34).
vod_det	`txcanapv.p` sets vod_taxable to Yes. It sets vod_taxc (GST only) and vod_tax_usage from vod_tax_at. It sets vod_tax, vod_tax_at, and vod_type. It also sets vod_tax_env using `txtxeget.p`.
vo_mstr	For vouchers, `txcanapv.p` sets vo_tax_pct[1], vo_tax_pct[2], and vo_tax_pct[3] to 0. It sets vo_taxable, vo_taxc, and vo_tax_usage. It also sets vo_tax_env using `txtxeget.p`.

Table A.47 — *Changes to Transaction Records* (Page 2 of 2)

Appendix B

Running the Fixed Assets Migration Utility

This appendix describes how to run the Fixed Assets Migration utility.

Running the Fixed Assets Migration Utility ***124***

Buttons ***124***

Setting Migration Defaults ***125***

Mapping Legacy Data ***126***

Conversion Methods ***126***

Converting Books ***128***

Converting Locations ***129***

Converting Classes ***130***

Migration Reporting ***131***

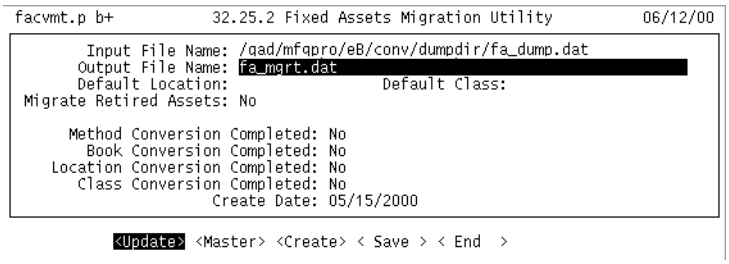
Running the Fixed Assets Migration Utility

The following sections describe the Fixed Assets Migration utility. Use this information to map the legacy data model to the enhanced model and to resolve discrepancies. When ready, use the Create button on the Fixed Assets Migration utility screen to load the data into the enhanced Fixed Assets module.

Fixed Assets Migration utility (32.25.2) tracks milestones of the migration process. It also lets you map the individual migration data types—methods, books, locations, and classes—in stages and then save these intermediate stages to a migration file. After you have mapped all of the legacy data to the enhanced data model, use the Create button to load the data from the migration file into the enhanced Fixed Assets module.

Important This update can only be done once.

Fig. B.1
Fixed Assets Migration Utility (32.25.2)



Most of the fields in the Fixed Asset Migration utility header cannot be updated. They display current mappings set in detail menus and screens. Use the buttons at the bottom to access the detail menus and screens. The following sections describe these features in detail.

Buttons

Update. Use this button to update the Output File Name, Default Location, Default Class, and the Migrate Retired Assets field.

Master. Use this button to access the Master Files Migration screen.

Create. Use this button to load legacy data into the enhanced Fixed Assets module after all the data is mapped to the enhanced model. This function can only be run once.

Save. Use this button to save the completed work to the file specified in the Output File Name field.

End. Use this button to end a migration utility session. All work is saved to the file specified in the Output File Name field.

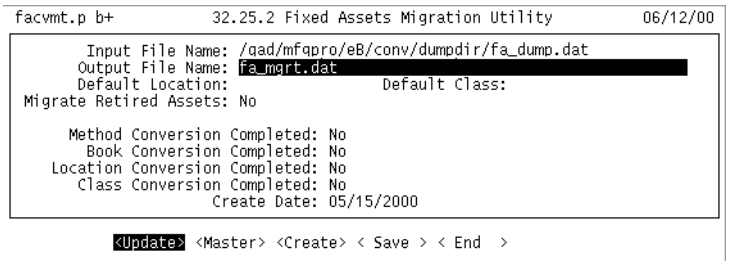
Setting Migration Defaults

Choose update to set up the default parameters for your migration.

Use the Default Location and Default Class fields to specify a legacy location and class to default for legacy assets that do not already have this information. The location and class must be part of legacy data. You can run the Fixed Assets Migration Report (32.25.3) to generate a list of your legacy classes and locations.

Set the Migrate Retired Assets field to Yes to migrate retired legacy asset data. Set the field to No to exclude retired assets from the migration.

Fig. B.2
Fixed Assets Migration Utility (32.25.2)



1. The Input File Name field defaults to `fa_dump.dat`. Enter only the file name. If you specified a different file name in the MFG/CONV environment prompts, specify the file name for your legacy data dump file. Press Enter to continue.
2. Press the Update button.
3. In the Output File Name field, specify the full path, including the file name, to the location where you want to store your migration file. Use this file to store your work as you map the legacy data to the new fixed assets model. The default file name is `fa_mgrt.dat`.
4. In the Default Location field, enter a default legacy location ID. Fixed asset location IDs refer to the accounting location of the fixed asset. This location does not have to be the physical location of the asset.
5. In the Default Class field, enter a default legacy class ID.

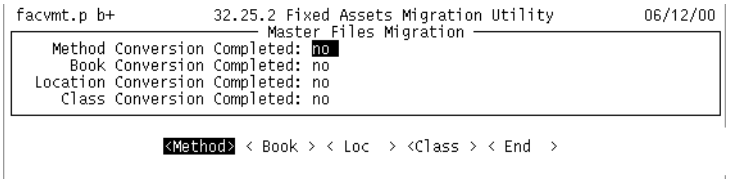
   Classes group similar fixed assets together and define:

   - GL accounts
   - Depreciation books
   - Service lives for calculating depreciation
   - Depreciation methods
6. In the Migrate Retired Assets field, enter Yes to migrate retired assets to the new system. Enter No if you do not want to migrate retired assets.

Mapping Legacy Data

Migrating legacy data to the enhanced module requires mapping the existing data models to the new models. Use the Master Files Migration screen to monitor the conversion of the legacy data models. Master Files Migration also provides access to the maintenance programs for each data model. These programs let you map legacy models to enhanced models.

Fig. B.3
Master Files Migration



Access the maintenance program for each data model using the buttons at the bottom of Master Files Migration.

Buttons

Method. Use this button to access Method Migration.

Book. Use this button to access Book Migration.

Loc. Use this button to access Location Migration.

Class. Use this button to access Class Migration.

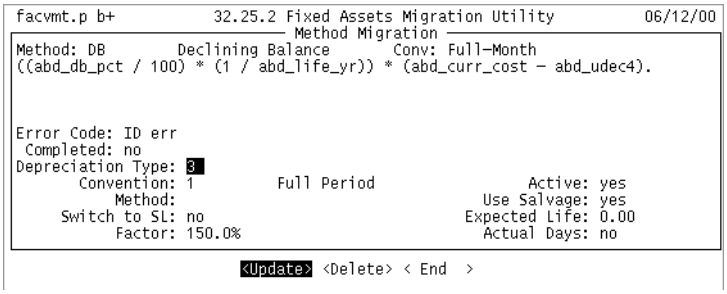
End. Use this button to end a Master Files Migration session and return to Fixed Asset Migration utility.

Conversion Methods

Use Method Migration to convert legacy depreciation methods to the depreciation methods supplied with the enhanced Fixed Assets module.

To convert legacy methods, use the following figure and instructions.

Fig. B.4
Method Migration



1 Use the arrow keys to scroll through the legacy methods. The legacy method, description, convention, and equation display.

2 If an asset does not use this method, press the Delete button to remove it. The system verifies the method is not used and prompts to confirm the deletion.

3 To convert the legacy method, press the Update button.

4 In the Depreciation Type field, select the enhanced depreciation method that most closely resembles the legacy method. The following depreciation methods are supplied with the enhanced module:

- 1: Straight Line
- 2: Units of Production
- 3: Declining Balance
- 4: Sum of the Years' Digits
- 5: Flat Rate
- 6: Custom Table

5 To modify the standard depreciation methods supplied with the enhanced Fixed Assets module, complete the following fields:

- Convention
- Method ID
- Switch to SL
- Factor
- Active
- Use Salvage
- Expected Life
- Actual Days

Note If you are using the custom-table depreciation method, you must define a custom table in Method Maintenance (32.1.1) before mapping it to a legacy method. Use the same method ID from Method Maintenance for the method in Method Migration.

6 If an error exists with the new method, the Error Code field displays one of the following tables:

- ID err: An error exists with the new method ID. For example, the method ID is missing or there is a duplicate method ID.

- type err: An error exists with the new method description. For example, the enhanced module does not support the method.
- conv err: An error exists with the new convention. For example, the convention is missing or the convention is not compatible with the method.
- table err: An error exist with the new table method. The following are examples of possible errors:

  The corresponding method defined in Method Maintenance is not a custom-table method.

  There is a discrepancy with the estimated life.

  Table detail is undefined in Method Maintenance.
- life err: An error exists with the new life. For example, the method is a custom table and the new life is zero.

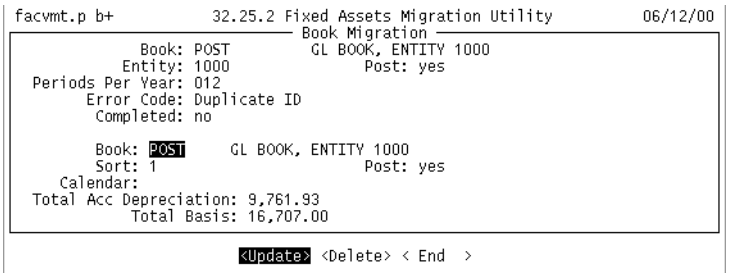
7 After you correct any errors, the Completed field is updated to Yes for the converted method.

8 Repeat steps 1 through 7 for each legacy method.

Converting Books

Use Book Migration to convert legacy books to books used in the enhanced Fixed Assets module. To convert legacy books, use the following figure and instructions. In the legacy system, asset cost is associated with depreciation books. Therefore, if an asset contains multiple books, the asset cost comes from the posting book.

Fig. B.5
Book Migration



1 Use the arrow keys to scroll through the legacy books. The legacy book ID, description, entity, and post field display.

2 If no assets use this book, press the Delete button to remove it. The system verifies the book is not used and prompts to confirm the deletion.

Note Any legacy asset books that use a depreciation type of none are not created in the new fixed asset system.

3 To convert the legacy book, press the Update button.

4 Complete the following fields:

- Book
- Description
- Sort Code

- Post
- Calendar

Note You can have only one posting book in the system. If the book was not a posting book in the legacy system, you cannot change it to a posting book for the new system.

5 Total Accumulated Depreciation and Total Basis are automatically calculated. Total Accumulated Depreciation displays the total accumulated depreciation for the legacy book. Total Basis displays the total basis for all of the fixed assets use the legacy book.

6 If duplicate book IDs exist, the Error Code field displays Duplicate ID.

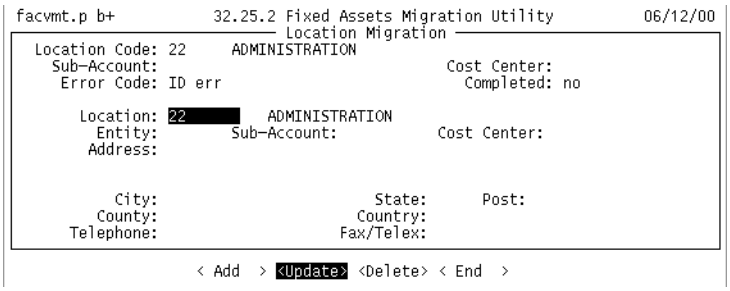
7 After you correct any errors, the completed field updates to Yes for the converted book.

8 Repeat steps 1 through 7 for each legacy book.

Converting Locations

Use Location Migration to convert legacy locations to locations used in the enhanced Fixed Assets module. To convert legacy locations, use the following figure and instructions.

Fig. B.6
Location Migration



1 Use the arrow keys to scroll through the legacy locations. The legacy location code, description, sub-account, and cost center display. For each legacy location, the location ID automatically fills in with the legacy location ID.

2 If an asset does not use this location, press the Delete button to remove it. The system verifies the location is not used and prompts to confirm to deletion.

3 To convert the legacy location, press the Update button.

4 Complete the following fields:

- Location ID
- Description
- Entity
- Sub-Account
- Cost Center
- Address and Telephone information

5 If an error exists with the new location, the Error Code field displays one of the following codes:

- ID err: An error exists with the new ID. For example, another module uses the location ID and you must set up a new ID.
- en err: An error exists with the new entity. For example, the entity code is undefined in Entity Code Maintenance (25.3.1).
- sub err: An error exists with the new sub-account. For example, the sub-account is undefined in Sub-Account Code Maintenance (25.3.17).
- cc err: An error exists with the new cost center. For example, the cost center is undefined in Cost Center Code Maintenance (25.3.20).

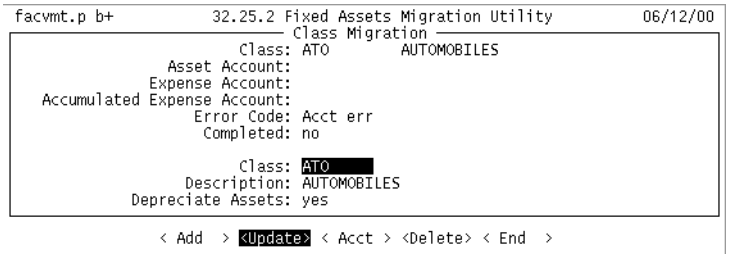
6 After you correct any errors, the Completed field updates to Yes for the converted location.

7 Repeat steps 1 through 6 for each legacy location.

Converting Classes

Use Class Migration to convert legacy classes to classes used in the enhanced Fixed Assets module and set up default GL accounts. To convert legacy classes, use the following figure and instructions.

Fig. B.7
Class Migration



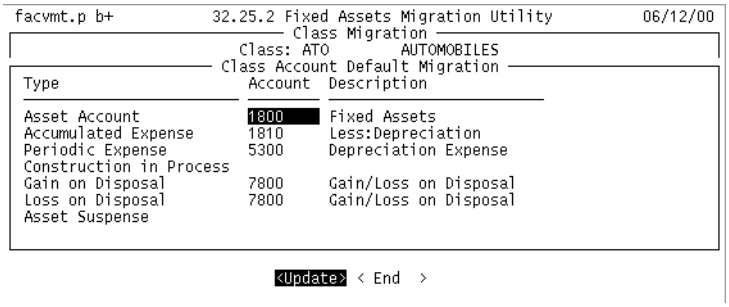
1 Use the arrow keys to scroll through the legacy classes. The legacy class ID, description, asset account, accumulated depreciation account, and depreciation expense account display. For each legacy class, the class ID and description automatically fill in with the legacy data.

2 If an asset does not use this class, press the Delete button to remove it. The system verifies the class is not used and prompts to confirm the deletion.

3 To convert the legacy class, choose Update and enter whether this fixed-asset class is depreciated.

4 Press the Acct button to modify the default GL accounts for the fixed-asset class.

Fig. B.8
Class Account Default Migration



5 For each class, you must specify a GL account for the following accounts:

- Asset Account
- Accumulated Expense
- Periodic Expense
- Construction In Process
- Gain on Disposal
- Loss on Disposal
- Asset Suspense

6 If any fixed-asset accounts are undefined, the Error Code field displays acct err.

7 After you correct any errors, the Completed field updates to Yes for the converted class.

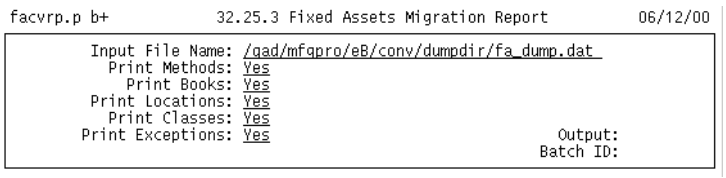
8 Repeat steps 1 through 7 for each legacy class.

Migration Reporting

Use Fixed Assets Migration Report (32.25.3) to generate a report that provides a summary of the migrated fixed-asset data. The report shows the migrated data from the legacy system and the newly mapped data for the new system.

You can choose to include books, methods, locations, classes, and exceptions in the report.

Fig. B.9
Fixed Assets Migration Report (32.25.3)



1 In the Print Books field, enter Yes to include migrated book data from the legacy system and the newly mapped book data in the report. Enter No to exclude book data.

2 In the Print Methods field, enter Yes to include migrated method data from the legacy system and the newly mapped method data in the report. Enter No to exclude method data.

3 In the Print Locations field, enter Yes to include migrated location data from the legacy system and the newly mapped location data in the report. Enter No to exclude location data.

4 In the Print Classes field, enter Yes to include migrated class data from the legacy system and the newly mapped class data in the report. Enter No to exclude class data.

5 In the Print Exceptions field, enter Yes to include exceptions for your data. Enter No to exclude the exceptions.

Appendix C

Converted Data

This appendix describes how selected data is converted.

Overview ***134***

Layers ***134***

Business Relations ***134***

Shared Sets ***134***

Entities ***135***

Profiles ***135***

Accounts ***135***

Project Status Codes ***139***

Project Groups ***139***

Security ***140***

Daybooks ***140***

Generalized Codes ***140***

Exchange Rates ***141***

Supplier Types ***141***

Purchase Types ***142***

Customer Types ***142***

Credit Terms ***142***

Credit Ratings ***142***

Invoice Status Codes ***143***

Payment Status Codes ***143***

Supplier Bank Data ***145***

Payment Formats ***146***

Consolidation ***146***

Unconfirmed Supplier Vouchers ***147***

Overview

The conversion creates many new objects, some of which are created based on user-specified values provided in the Conversion Parameters Utility for QAD Enterprise Financials. These new objects include items, such as business relations, shared sets, profiles, invoice status codes, project status codes, and project groups.

This section describes some of the objects created by the conversion, and the naming conventions used by the conversion when creating the objects. For additional information, refer to *User Guide: QAD Financials*.

Important This appendix only applies to conversions from a pre-Enterprise Edition version of the QAD ERP application to Enterprise Edition. It does not apply to upgrades from an Enterprise Edition installation.

Layers

The conversion creates the default accounting layers required by QAD Enterprise Financials.

Business Relations

The conversion creates a business relation and an address of type Head Office for employees and addresses of the following types:

- Customer
- Supplier
- Employee
- Carrier
- Company
- Slsprsn
- Engineer
- Our_bank (bank address for the business's operational cash)
- C/S_bank (customer and supplier bank addresses)
- Faloc (Fixed Asset location ID address)

Shared Sets

The conversion creates a number of shared sets for each of the domains in the source database.

The conversion only creates shared sets for active domains that belong to the database being converted. Each shared set is named using a combination of the source domain name and the shared set type.

The following table lists the shared sets created for each source domain.

Table C.1
Shared Sets

Shared Set Code	Shared Set Type
<domain>Cost Center	Cost Center
<domain>Customer	Customer
<domain>Daybook	Daybook
<domain>Ex Rate	Exchange Rate
<domain>Account	Account
<domain>Project	Project
<domain>SubAccount	SubAccount
<domain>Supplier	Supplier
<domain>COAMaskDiv	Sub-Account COA Mask
<domain>COAMaskCC	Cost Center COA Mask
<domain>COAMaskProj	Project COA Mask

Entities

The conversion links entities to the shared set of the domain to which the entity belongs.

Profiles

The conversion creates the default profiles listed in the following table:

Table C.2
Default Profiles

Profile Code	Profile Type
Default Bank-<domain>	Banking Entry Daybook Profile
<cost center>-<domain>-C	Cost Center Profile
<AR account>-<domain>-A	Customer Account Profile
<project>-<domain>-P	Project Profile
<Purchases account>-<domain>-PA	Purchase Account Profile
<Sales account>-<domain>-SA	Sales Account Profile
<sub-account>-<domain>-S	Sub Account Profile
<AP account>-<domain>-A	Supplier Account Profile

Accounts

The conversion does not convert memo and statistical type accounts or blank sub-accounts, cost centers, or projects. Instead, the conversion replaces the blank values with default values that the Conversion Parameters Utility prompts for.

Analysis Type and Analysis Limitations

The conversion uses Code Range Master (cr_mstr) to determine the analysis type setting for an account. For example, if an account code is within the range defined in Code Range Master for a cost center, the account is assigned an analysis type of Cost Center.

There are exceptions to the assignment of the analysis type setting for some account types. For example, system accounts (with the exception of PO Receipts system accounts) must have an analysis type of None. These limitations are defined by the validations in QAD Enterprise Financials.

The other exception when using the Code Range Master to assign the analysis type setting occurs when converting from 8.6E or 9.0. In this scenario, there are no ranges defined for projects. Therefore, the pre-conversion GL Account/Project Range Utility (`uxglproj.p`) is used to identify which accounts use project analysis. For the analysis limitations, the default value is None, but there are exceptions.

Enterprise Edition accounts that use automatic posting cannot have a combination of an analysis type of Both and an analysis limitation of None. Automatic accounts include customer and supplier control and payment accounts, and system accounts other than PO Receipt accounts. Therefore, for any accounts with a posting type of Automatic and an analysis type of Both, the conversion sets the analysis limitation to Both. All other accounts have an analysis limitation of None.

COA Mask

The conversion creates a COA mask range for every Code Range Master (cr_mstr) range in the source database. All pre-existing Account Sub-Account Cost Center Master (asc_mstr) records (combinations used in historical transactions) are deleted.

For MFG/PRO versions below eB2, the Code Range Master table is populated from sub-account and cost center combinations in the Sub-Account Validation Detail (sbd_det) table, the Cost Center Account Validation Detail (ccd1_det) table, and the Cost Center Sub-Account Validation Detail (ccd2_det) table as part of the conversion to QAD Enterprise Edition.

When converting a GL account, the conversion uses the ranges in Code Range Master to determine if the account uses cost center or projects analysis.

- If an account does not use cost center or project analysis, the analysis type is set to None.
- If an account only uses project analysis, the analysis type is set to Project.
- If an account only uses cost center analysis, the analysis type is set to Cost Center.
- If an account uses both cost center and project, the analysis type is set to Both.

Note For MFG/PRO versions below 9.0, project code ranges are set to All. This means that these project codes are valid for use with all accounts that allow project analysis (as defined in the GL Account/Project Range Utility).

The conversion only assigns a default cost center to an account if its analysis type is Cost Center. Similarly, the conversion only assigns a default project to an account if its analysis type is Project.

If an account has an analysis type of Both or None, the conversion sets the account's analysis limitation to None, and the default values are left blank. However, there is one exception to this rule. Currently, in QAD Enterprise Financials, if an account has a posting type of Automatic and an analysis type of Both, the analysis limitation must be set to Both, and the account must be assigned a default cost center and project.

Conversion Accounts

Some new GL accounts of type Conversion are created during conversion. These accounts are used only by the conversion when converting financial transactions for Accounts Receivable (AR) and Accounts Payable (AP). The naming convention for these special accounts uses the format QAD-xxxx, where xxxx is a sequential number assigned to the conversion accounts. The conversion creates the account description for the conversion accounts by concatenating the original account number and description. If needed, the conversion trims the account description to fit the Description field length in the GL Account record.

The conversion accounts remain active post-conversion so that you can post to them to remove any remaining balances. Once the conversion accounts are balanced, you must manually change the status of the conversion accounts to Inactive.

GL Transactions

The conversion does not convert the existing GL transactions against the original control account, but instead creates posting lines in QAD Enterprise Financials against the special conversion accounts. This process solves a potential balance discrepancy between the converted sub-ledger (AR or AP) and the converted control account balances for the corresponding accounts (AR control and AP control). Agreement between the sub-ledgers and control accounts is a key check in QAD Enterprise Financials, and you must resolve any discrepancies during the conversion. If a GL transaction is posted to a non-control account, the transaction is converted as it is.

Unbalanced GL Transactions

A database can contain unbalanced GL transactions. A potential cause of unbalanced transactions is that the database contains transactions from other financial software that were previously converted to the QAD application.

The conversion converts unbalanced transactions in a closed fiscal year as they are. If the entire closed year is out of balance for an individual entity, the conversion creates a balancing transaction. The reason for this is that if the historical years are unbalanced, the trial balance in QAD Enterprise Financials will never balance.

An accounting year is composed of all GL calendar periods for an individual GL calendar year. To determine if a balancing transaction is needed for an entity in a closed year, all converted GL transactions for an accounting year and entity are tracked. If the total does not balance, the conversion creates a new posting and posting line to balance the year and entity. The conversion creates the balancing transaction in the appropriate historical period, and uses the GL account entered in the Conversion Parameters Utility for the balancing transactions.

Note The calculations that determine if the accounting year is balanced do not include year-end postings (transactions of type YR). Year-end postings are discussed below.

The conversion modifies any unbalanced transactions in an open fiscal year so that the year-end closing process in QAD Enterprise Financials can run without error. The conversion automatically creates a single posting line for the out-of-balance amount against an account you specified for this purpose before starting the conversion. This process balances the unbalanced transaction. All balancing changes made by the conversion are recorded in a log file. The balancing transaction can also be easily identified by analyzing the postings made against this special account.

The conversion treats year-end transactions (type YR) differently because these transactions are always one-sided in earlier versions of the QAD ERP application. The conversion always creates a balancing posting line for year-end transactions. This balancing posting line is created in the historical period and uses the P&L (balance sheet) account from General Ledger Control (co_ctrl.co_pl). As noted previously, the year-end posting is not considered when determining if the overall year is balanced.

In European Accounting, the year-end transaction is not one-sided, but must be fully balanced. When European Accounting is used, the conversion does not create a balancing posting line for year-end transactions.

AR Transactions

The conversion posts each total invoice or payment amount to the AR account defined in the Customer Master, and then reverses this amount from the corresponding conversion account for the AR account of the original transaction.

If the AR account used for the original transaction is not an actual AR control account provided in the pre-conversion Control Account Utility or GL Account Type Utility, the posting line is changed to use the account from one of these utilities. The conversion records changes of this type in a separate log file.

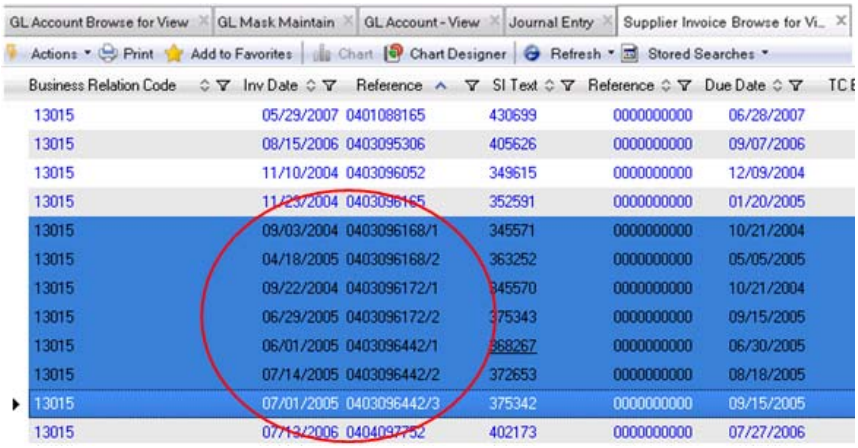
Note If GL transaction consolidation was performed in the pre-conversion database, and multiple AR control accounts are in use, it is likely that compensating balances will remain on the AR conversion accounts. You can correct these balances post-conversion using a journal entry.

AP Transactions

The conversion of AP transactions is similar to the conversion process for AR transactions. One difference, however, is the treatment of duplicate invoice numbers.

QAD Enterprise Edition prohibits duplicate invoice numbers for a single supplier. Standard Financials provides a warning in this situation. QAD Enterprise Financials displays an error if it finds duplicate invoice numbers. To work around this, when the conversion encounters a duplicate invoice number for a supplier, it appends a suffix of /1 or /2 to the invoice number, as illustrated in the following figure.

Fig. C.1
Duplicate Invoice Number Indication



When the conversion is complete, the remaining balance of the conversion accounts must be equal to the pre-conversion difference between the GL and sub-ledger. If additional balances remain, you must investigate and resolve them.

Residual balances in the AR or AP conversion accounts can result from transactions previously posted to the control accounts that did not truly relate to AR or AP. These differences are normally indicated in the Pre-conversion Integrity Report. Complete the following steps to identify specific errors:

- Balance aging by account to the GL for each month.
- Add the balances in all conversion accounts for the sub-ledger (AR or AP) to the converted control accounts for the sub-ledger. This total should be the same as the control account total before conversion.

You can post journal entries to conversion accounts, but not to control accounts, with the exception of Fixed Assets Control accounts.

If the Inventory Control or WIP Control accounts are out of balance, use Issues – Unplanned (3.7, `icunis.p`) or Receipts – Unplanned (3.9, `icunrc.p`) to correct the account balance.

Project Status Codes

The conversion creates two default project status codes:

- Open, with a system status of Open
- Closed, with a system status of Closed

Project Groups

The conversion creates a single default project group named All.

Security

Users

The conversion creates an active user for each user record in the database being converted, where the user ID is not blank. The conversion also creates user domains for each domain in the User Domain Detail (udd_det) table, and creates a user entity record linking the user to each entity in that domain. The conversion marks the user record as Enabled if the source record is Active.

User Roles

A user role is created for each User Group Detail (usrgd_det). The role name is derived from the linked user record. A User Role Domain (UsrRoleDomain) and User Role Entity (UsrRoleCompany) are also created based on the domain of the User Group Detail.

Daybooks

For each existing daybook with a non-blank code, the conversion creates a daybook record in the shared set of the domain to which the daybook belongs.

The converted daybook is assigned a daybook type of Journal Entry, a layer code of Official, and a control type of Operational. If the source daybook has not reached its expiration date, the daybook is marked as active in the converted database. Otherwise, the daybook is marked as inactive. The conversion ensures that the associated NRM sequence for active daybooks uses the new format of YYYY<*daybook code*>000000001.

Additional active default daybooks are created in each shared set with the type of Journal. These daybooks are the ones prompted for in the pre-conversion Conversion Parameters Utility.

Generalized Codes

Deleted Generalized Codes

A number of tables in QAD Enterprise Financials hold data previously stored in Generalized Codes (code_mstr). The conversion moves the following data from Generalized Codes to their new tables.

- Each generalized code defined for the tx2_tax_usage field is moved to the txu_tax_usage field in the Tax Usage Master (txu_mstr) table.
- Each generalized code defined for the taxc_taxc field is moved to the txcl_tax_cls field in the Tax Class Master (txcl_mstr) table.
- Each generalized code defined for the txt_tax_type field is moved to the txty_tax_type field in the Tax Type Master (txty_mstr) table.
- Each generalized code defined for the txe_tax_env field is moved to the txev_tax_env field in the Tax Environment Master (txev_mstr) table.

New Generalized Codes

The conversion validates that the data required in Generalized Codes exists. Generalized Codes (code_mstr) table entries are created for the following items:

- Each state code on employee records (emp_mstr.emp_state) and other address records (ad_mstr.ad_state) must exist in Generalized Codes with a code_fldname value of ad_state.
- Each customer type (cm_mstr.cm_type) must exist in Generalized Codes with a code_fldname value of cm_type.
- Each county code on addresses records (ad_mstr.ad_county) must exist in Generalized Codes with a code_fldname value of ad_county.
- Each tax class on address records (ad_mstr.ad_txc) must exist in Generalized Codes with a code_fldname value of ad_txc.
- Each voucher type (vo_mstr.vo_type) must exist in Generalized Codes with a code_fldname value of vo_type.
- Each supplier type (vd_mstr.vd_type) must exist in Generalized Codes with a code_fldname value of vd_type.

SAF Codes

The following Supplementary Analysis Field (SAF) codes are created using generalized codes:

- Item Group (pt_group)
- Item Type (pt_part_type)
- Customer Type (cm_type)
- Supplier Type (vd_type)
- Region (cm_region)

Additional SAF codes are created for sites and non-blank product lines. SAF codes are system-level data, and are not domain-specific.

Exchange Rates

The conversion sets the latest exchange rate for each currency as the active exchange rate for that currency, even if the exchange rate had an expiration date before conversion. Exchange rates are effectively open until a new exchange rate is created for that currency with a later start date.

In QAD Enterprise Financials, any exchange rates between two specific currencies must always be stated in terms of the From currency. For example, if an exchange rate is created from Currency 1 to Currency 2, you cannot then create an exchange rate from Currency 2 to Currency 1. If this situation exists pre-conversion, the conversion updates the exchange rates so that they are consistent.

Supplier Types

The conversion creates active supplier types for each value found in the Generalized Codes table, where the code_fldname value is vd_type and the generalized code is not blank. Supplier types are system-level data in QAD Enterprise Financials.

Purchase Types

The conversion creates active purchase types for each value found in the Generalized Codes table, where the code_fldname value is vo_type and the generalized code is not blank. Purchase types are system-level data in QAD Enterprise Financials.

Customer Types

The conversion creates active customer types for each value found in the Generalized Codes table, where the code_fldname value is cm_type and the generalized code is not blank. Customer types are system-level data in QAD Enterprise Financials.

Credit Terms

The conversion creates active credit terms with a type of Normal based on the credit terms code in the database being converted. If the pre-conversion credit term has multiple due dates, a staged credit term is created. In this instance, the associated credit term has a type of Staged rather than Normal.

Credit terms are system-level data in QAD Enterprise Financials. If the pre-converted database contains multiple credit terms with the same name, but in different domains and with different terms, the conversion creates only one credit term for the credit terms code.

QAD Enterprise Edition does not support fixed due dates for credit terms. As a workaround, the conversion:

- Does not convert credit terms with fixed due dates
- Identifies the credit term in the pre-conversion database with the most stages
- Creates a set of default credit term stages for each of the stages identified above

Note You provide the naming convention when running the Conversion Parameters Utility, and the system also appends a numeric extension to the end of the credit term code. However, because the credit term code can be a maximum of eight characters, if you supply an eight-character code, the conversion removes the last character in order to append the numeric extension (for example, 1, 2, 3, and so on). If only credit terms with fixed due dates exist (for example, no credit terms with multiple due dates), the conversion creates credit terms supplied within the Conversion Parameters Utility.

- When converting a credit term that uses a stage with a fixed due date, the conversion replaces the existing stages with new default stages.
- When a customer or supplier document that references a credit term whose stages were replaced with the default stages is converted, the new default stages are used.

Credit Ratings

The conversion creates active credit ratings for each value found in the Generalized Codes table, where the code_fldname value is cm_cr_rating and the generalized code is not blank. Credit ratings are system-level data in QAD Enterprise Financials.

Invoice Status Codes

The conversion creates three default invoice status codes:

- QAD-ALLOC for use with allocated transactions. The field values are as follows:
  - Allocation = Allocated
  - ReceiverMatching = True
  - ReceiverMatching = False (not allowed)
- QAD-UNALLOC for use in unallocated transactions. This code is linked to the QAD-ALLOC invoice code as the associated allocated invoice status code. The field values are as follows:
  - Allocation = Not allocated
  - ReceiverMatching = True (allowed)
- QAD-CONT for use with contested open items
  - Allocation = No Allocation
  - ReceiverMatching = False

You can easily change these codes and their descriptions post-conversion. Any change is reflected in AR and AP invoices where these codes are referenced.

Payment Status Codes

Customer Payment Status Codes

The conversion creates all, or a subset of, the customer payment status codes listed in the following table.

Table C.3
Customer Payment Status Codes

Payment Instrument	Status
Check	Initial
Check	Paid
Check	Bounced
Draft	Initial
Draft	For Collection
Draft	Paid
Draft	Paid Conditionally
Draft	Bounced
Draft	Allocated

Determining Factors:

- The conversion always creates customer payment status codes for check payment instruments. The status codes are created for each GL bank account.
- The conversion creates customer payment status codes for the payment instrument Draft for Accounts Receivable Master (ar_mstr) records with a type (ar_type) of D. The status codes are only created for the GL bank accounts against which the draft transactions were processed.

Note Standard Financials does not use AR payment in process (PIP) accounts. Therefore, the conversion does not create payments with a status of For Collection for customer payments of type Check. All payments are converted with the status Paid and an Open indicator of No.

Supplier Payment Status Codes

The conversion creates all, or a subset of, the supplier payment status codes listed in the following table.

Table C.4
Supplier Payment Status Codes

Payment Instrument	Status
Check	Initial
Check	For Collection
Check	Paid
Check	Void
Transfer	Initial
Transfer	For Collection
Transfer	Paid
Transfer	Void
Electronic Transfer	Initial
Electronic Transfer	For Collection
Electronic Transfer	Paid
Electronic Transfer	Void
Draft	Initial
Draft	For Collection
Draft	Paid
Draft	Paid Conditionally
Draft	Void

Determining Factors:

- The conversion always creates payment statuses for payment instruments of type Check. The statuses are created for each GL bank account.
- The conversion creates payment statuses for payment instruments of type Transfer if an Accounts Payable Master (ap_mstr) record exists with a type (ap_type) of CK and a check form (ap_ckfrm) of 3.
- The conversion creates payment statuses for payment instruments of type Electronic Transfer if an ap_mstr record exists with a type (ap_type) of CK and a check form (ap_ckfrm) of 4.
- The conversion creates payment statuses for payment instruments of type Draft if an Accounts Payable Master (ap_mstr) record exists with a type (ap_type) of CK and a check form (ap_ckfrm) of 5, 6 or 7.
- For payment instruments Check, Transfer, and Electronic Transfer, the conversion only creates the payment status For Collection if the Use Payment in Process Acct (apc_ctrl.apc_pip) field in AP Control (28.24, `appm.p`) is set to Yes. If this status is created, the Payment in Process (PIP) account defined in Bank Maintenance (bk_mstr.bk_pip_acct) is assigned to the GL account field for the Paid statuses.

Please note the following:

- If the Use Payment in Process Acct (apc_ctrl.apc_pip) field in AP Control is set to No, AP payments for the payment instruments Check, Transfer, and Electronic Transfer are converted with the status Paid and with an Open indicator of No.
- If the Use Payment in Process Acct (apc_ctrl.apc_pip) field in AP Control is set to Yes, AP payments for the payment instruments Check, Transfer, and Electronic Transfer are converted with the status For Collection and with an Open indicator of Yes.

When PIP accounts are not used in the pre-conversion environment, irrespective of the payment status, the conversion assigns all supplier payments a status of Paid.

QAD Enterprise Financials requires that PIP accounts are entered for payments with a status of Paid. However, unless the Conditional Collection status is used, the PIP account is retrieved from the For Collection status. The GL account is not required for this status and the use of PIP accounts is optional.

Regardless of whether PIP accounts are used or not in the pre-conversion environment, transactional processing can continue following the conversion to QAD Enterprise Edition. This is done without the need to change static and transactional data.

If you want to implement PIP accounts, you can do this pre- or post-conversion.

Pre-conversion

You can create the PIP account, populate the required bank accounts with the new PIP account, and change the Use PIP Accounts field to Yes in AP Control. For the post-conversion position to be correct, you must ensure that the GL balance for the PIP account matches the balance of the open checks. This adjustment cannot be made post-conversion because QAD Enterprise Edition does not allow journal entries to customer or supplier payment accounts.

If you follow this approach, the conversion creates an open payment with the For Collections status.

Post-conversion

You can create a payment status of For Collection and assign the required PIP account. This approach will be used going forward. You must maintain a manual list of open payments in the pre-conversion database until all of the payments are cleared or closed.

Supplier Bank Data

QAD Enterprise Financials requires that at least one active bank is defined for each supplier. If supplier bank data is maintained in Supplier Maintenance in the pre-conversion database, the data is used during the conversion. Optionally, you can create a default supplier bank account.

Supplier bank accounts are converted as active or inactive, depending upon the effective dates defined in Supplier Maintenance in the pre-conversion database. A single bank account for each supplier must be set as the default bank (this is determined by the source effective dates). If the

supplier has multiple bank accounts and more than one bank account is effective, the conversion sets all the supplier bank accounts to active and sets the first bank account in alphanumeric order as the default. You must review these settings after the conversion.

Payment Formats

Payment formats are used in customer and supplier payments to define the layout of the payment output. The term payment format is new to QAD Enterprise Financials, and replaces check forms in Standard Edition and payment methods in European Accounting.

The conversion determines the payment format using a combination of the bank and payment method on the supplier, customer, customer invoice, supplier invoice, or customer or supplier payment.

Non-European Accounting

The conversion creates a payment format for each of the AP check forms (1-8) in the format Check Form 1, Check Form 2, and so on.

The conversion also creates a default AR check payment format because earlier QAD versions did not have this concept. The AR check payment format is simply called AR Check.

If the pre-conversion system uses drafts (meaning that Accounts Receivable Master records exist where the type is D), the conversion also creates a default AR draft payment format called AR Draft.

European Accounting

QAD Enterprise Edition Financials includes a list of supported payment formats. These payment formats are defined in XML files that you can obtain by contacting QAD Support. The conversion loads the supported payment formats from the directory specified in the Conversion Parameters Utility.

The conversion attempts to convert non-standard (that is, not provided by QAD) payment formats to Enterprise Edition. However, the conversion may not always be completely successful in this process. The expected outcome is that any customers, suppliers, and banks that reference non-standard payment formats are configured correctly. However, the details of the actual payment format (that is, the output format of the file) may not be 100% correct due to differences that existed before the conversion to Enterprise Edition. You must review and modify the output from this process in Enterprise Edition, as required.

You can identify non-standard payment formats in Enterprise Edition by their name, which is the same as in the pre-conversion environment (for example, `euqptswp.p`). Standard, supported Enterprise Edition payment formats have names such as GENERIC-PAY-AP or DE-DTAUS-AR.

Consolidation

The conversion creates consolidation cycle records based on GL consolidation sets in the source database. Each of these records requires a unique daybook for each consolidation entity. The conversion creates these records using the following format:

<Four character daybook code provided in Conversion Parameters Utility><Entity Code>

After the conversion, review and update the following data, as necessary:

- Default sub-account, project, cost center, rounding GL account, and tax codes
- Default SAF values for GL accounts, cost centers, and projects
- Consolidation cycle status
- Daybooks associated with the management and transient layers
- COA cross reference codes

Due to considerable differences in the setup required by the From-Acct Cross-Reference Maintenance function in previous versions of QAD applications compared to the account and sub-account cross-references in QAD Enterprise Financials, consolidation cross-reference data is not converted. Instead, you must use Excel integration and the process of copying existing records to configure this data after conversion.

Voucher Detail Records

The conversion does not directly reference Voucher Detail (vod_det) records extensively.

Pre-conversion

The Pre-conversion Integrity Report reports any missing Voucher Detail records, but this output is for informational purposes only, and does not stop the conversion.

Conversion

The conversion only references Voucher Detail records when converting AP invoices (converting ap_mstr and vo_mstr records to CInvoice records). The conversion uses the Voucher Detail data in several ways:

- The conversion uses the vod_det.vod_dy_num field to populate the Posting.PostingOriginDaybookNumber field for the posting records created.
- If the record converted is a Waiting Expense Voucher transaction, the conversion uses the Voucher Detail data to determine the waiting expenses amounts.
- When converting unconfirmed vouchers, the conversion removes Voucher Detail records to allow receiver matching after conversion.

Unconfirmed Supplier Vouchers

When unconfirmed vouchers are converted to Enterprise Edition, GL distribution or receiver matching data is lost and must be re-entered.

In earlier QAD versions, the Accounts Payable module allows you to enter vouchers marked as unconfirmed. In Standard Financials, you can match unconfirmed vouchers to purchase order receipts and you can enter GL distribution lines without creating actual GL transactions.

In earlier QAD versions, you can modify or delete unconfirmed vouchers entirely. The vouchers are not widely visible within the system, and are not available for payment. Once the voucher and GL distribution or matching are approved, you can use a separate Confirmation function to set the voucher status to Confirmed. At that point, the system creates the required GL transactions, prohibits further modification to the voucher, and makes the voucher available for payment.

In Enterprise Edition, the creation and timing of general ledger postings within the supplier invoice process is controlled through the use of invoice status codes. However, Enterprise Edition does not contain an invoice status code that lets you create supplier invoices that contain detailed GL posting and matching data that can later be reversed or deleted. The Initial invoice status code provides a similar, but non-equivalent, functionality. Supplier invoice header data (for example, the supplier, PO number, and invoice total) is held, but GL or matching data is not. Therefore, converting unconfirmed vouchers erases the GL distribution or receiver matching data held against the supplier invoice.

Appendix D

Configuring Access to the Progress Editor

This appendix describes how to configure the system to access the Progress editor.

Introduction 150

Setup for Progress Editor Access 150

Introduction

In QAD Enterprise Edition, access to the Progress editor from the main menu (by entering P when attempting to exit) was removed. Instead, `mgeditor.p` is placed on menu 36.25.80. With the standard QAD Enterprise Edition .NET UI installation, use of mgeditor is disabled (it is still accessible through character telnet sessions). The following instructions explain how to set up User Option Telnet Maintenance for Progress editor access.

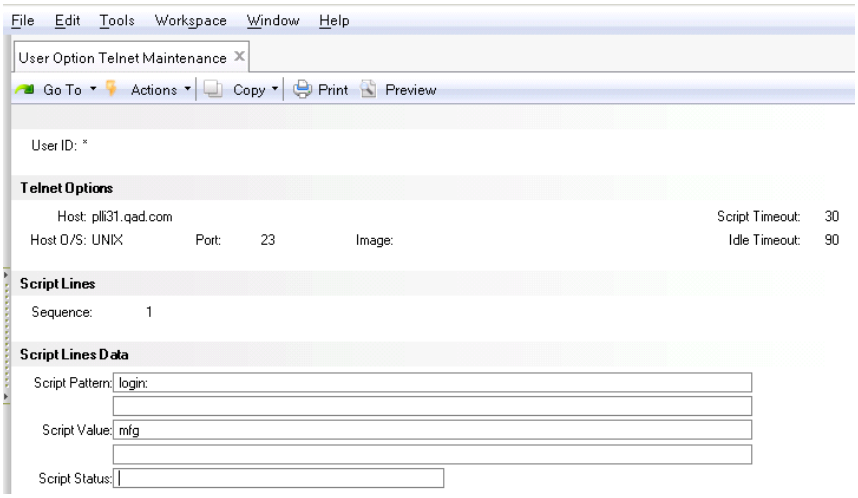
Setup for Progress Editor Access

The User Option Telnet Maintenance function creates the hidden script that logs in to the system via telnet. The script can be emulated by mirroring the steps using a normal telnet session; for example, match login and type user ID.

1 Launch User Option Telnet Maintenance (36.4.14) and configure the login script as follows:

- User ID: *
- Host: <*host-name for QAD Enterprise Edition Installation*>
- Port: 23
- Sequence: 1
- Script Pattern: login: <-- Validate the case of the login on the Telnet Screen (for example, is it L or l)
- Script Value: <*unix-login*> <-- Enter the UNIX user id

Fig. D.1
Sequence 1 Entries



2 Make the following entries for sequence 2:

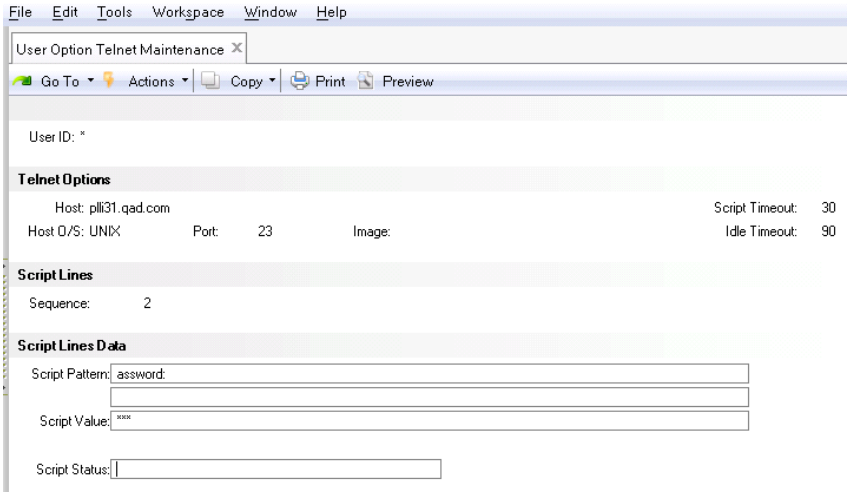
- Sequence: 2
- Script Pattern: assword:

Note Do not use the P because some operating systems have this as lower case and some have uppercase.

- Script Value: <*unix-password*> <-- Enter the UNIX password

Note This is the OS password. It is not necessarily the same as the QAD .NET UI password.

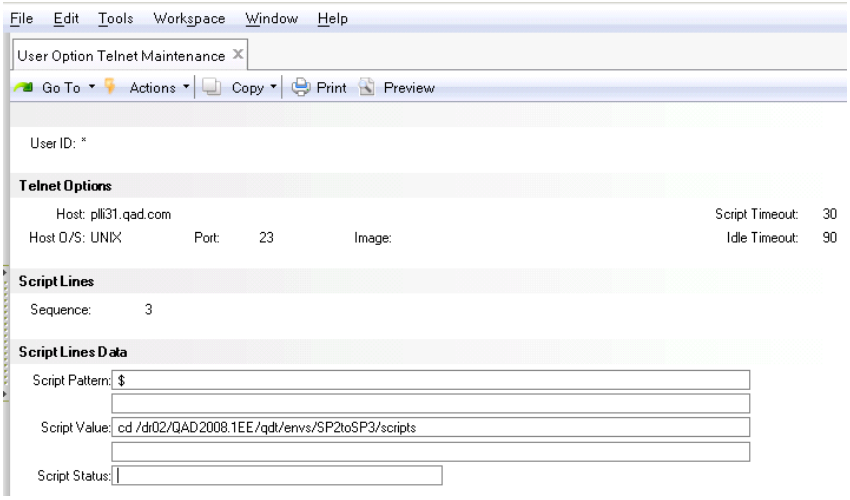
Fig. D.2
Sequence 3 Entries



3 Make the following entries for sequence 3:

- Sequence: 3
- Script Pattern: $ <-- Ensure the script pattern is correct after the user above logs in. Is it $ or # for example?
- Script Value: `cd <QDT-install-dir>/scripts/<env>/scripts`

Fig. D.3
Sequence 3 Entries



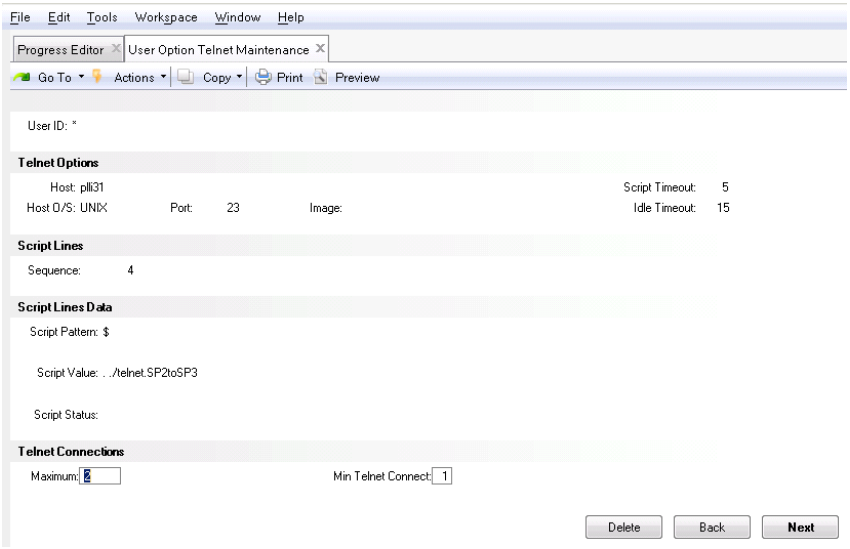
4 Change to the directory where the scripts are located for the environment under QDT.

5 Make the following entries for sequence 4:

- Sequence: 4
- Script Pattern: $
- Script Value: `. ./telnet.<env>` <-- where *<env>* represents the environment being accessed.

Note There is a space between the two periods.

Fig. D.4
Sequence 4 Entries



6 Click the Back button to advance to the Telnet Connections section of the frame.

7 Update the Maximum and Min Telnet Connect numbers.

8 Click Next when finished.

Appendix E

Conversion Troubleshooting

This appendix provides conversion troubleshooting information.

Introduction 154

Conversions Not Enabled in QDT 154

Errors in Data Preparation Report 154

reindex.log Errors 154

qdtadmin.log Errors 155

Progress Errors During Conversion Execution 156

Introduction

This appendix describes issues that may prevent a successful conversion.

The following are some of these issues:

- Conversions were not enabled in QDT.
- Errors in the Data Preparation Report
- `reindex.log` errors
- Invalid characters in a database field
- Number of characters in a database field exceed the character limit of that field
- Progress errors

Conversions Not Enabled in QDT

Symptom: QDT menu options do not contain Convert QAD EE from Previous Release.

Conversion routines are disabled and password protected by default. Therefore, this option must be enabled in advance.

Conversion routines are not deployed on QAD Enterprise Edition media. Therefore, a separate download of QDT with conversions included is required.

Errors in Data Preparation Report

Symptom: `Data Preparation Report is not clean` error message after you input the source database.

Before you can execute a conversion, you must run the Data Preparation Report on the source database and report zero errors.

To fix the problem, run the Data Preparation Report against the source database and correct any reported errors.

reindex.log Errors

`reindex.log` contains the output from rebuilding the indexes in the QAD Enterprise Edition database during the conversion execution stage.

The new unique index oid_<table name> can cause these errors. OID fields were available in some pre-QAD Enterprise Edition versions, but uniqueness was not enforced. This new index is used to enforce the uniqueness of OID fields.

Analyze and correct these errors before proceeding with the conversion. For OID fields, these can be reset to zero and the conversion will regenerate them. Note that some OID fields are used as foreign fields and should be manually corrected.

Restart the conversion execution from the beginning.

qdtadmin.log Errors

Invalid Characters in Database Field

Error message: `field cannot contain a comma, a pipe or any unprintable character`

QAD Enterprise Edition does not support commas, pipes, or unprintable characters in some fields. Remove these characters from the source database or replace them with supported characters such as a semicolon.

Restart the conversion execution from the beginning.

Number of Characters in Database Field Exceeds Limit

Error message: `value is too long`

QAD Enterprise Edition validates field length based on data definitions. If the value entered in the field exceeds the length defined in the database, QAD Enterprise Edition will raise an error during conversion.

These errors can be corrected by shortening or truncating some characters from the original value to make its length less than, or equal to, the length defined in database.

Restart the conversion execution from the beginning.

Role Name Contains Unsupported Characters

Error message: `The role name may not contain the following characters: '*', ',' or '!'`

QAD Enterprise Edition's role name does not support the characters “*, “,” or “!”.

They must be replaced in usrg_group_name of the source database with supported characters.

Restart the conversion execution from the beginning.

Pay Format Directory Not Found

Error message: `***** Error: Cannot find Pay Format Directory`

The `payformats` XML directory was entered incorrectly using the Conversion Parameters utility or the directory does not contain any of the required XML files.

Confirm that the directory entered is valid and contains the correct XML files. Restart the conversion execution from the beginning.

Progress Errors During Conversion Execution

Default Daybook Codes Not Found

Error message: `Record Already exists with Daybook Code = 0.`

The conversion could not find the default daybook codes because they were not entered or an earlier error caused the conversion not to write these values to the XML file.

Update the values using the Conversion Parameters utility.

The conversion execution stage must be restarted from the beginning.

Could Not Start Financial Session

Error message: `Could not start financial session. -5`

The possible causes of this error are:

- Financial application server is not running. Start the financial application server.
- License Keys are out of date. At some point, the conversion was stopped and restarted. The conversion execution must be restarted from the beginning.

Appendix F

Log Files

This appendix describes the conversion log files.

Introduction 158

Log File Naming Conventions 159

Reviewing Log Files After Conversion 159

Introduction

Conversion is executed within the QDTAdmin UI. Therefore, the `qdtadmin.log` is the primary source of conversion logging information.

Besides the `qdtadmin.log` file, several other log files may be created, especially if certain error conditions occur. The following table lists the log files that may be generated during a financial conversion.

Table F.1
Financial Conversion Log Files

Log File Name	Description
AddressMaster.log	When addresses and contacts are created during the conversion, End User detail (eud_det) records with a Name (eud_sort) field that exceeds 24 characters are truncated. This file contains a log of truncated records.
APInvoiceInvalidData.log	This file contains a log of AP Invoices which contained some invalid data, but were processed during the conversion. Examples of such invalid data are AP Invoices in which the Company, Currency, or Daybook information was invalid or referenced a nonexistent record. The AP Invoice record will still be fully converted, but post-conversion action may be required.
APPaymentInvalidData.log	This file contains a log of any AP Payments which contained some invalid data, but were processed during the conversion. Examples of such invalid data would be AP Payments in which the Company, Currency, or Supplier information was invalid or referenced a nonexistent record. The AP Payment record will still be fully converted, but post-conversion action may be required.
ARInvoiceInvalidData.log	This file contains a log of any AR Invoices which contained some invalid data, but were processed during the conversion. Examples of such invalid data would be AR Invoices in which the Company, Currency, or Daybook information was invalid or referenced a nonexistent record. The AR Invoice record will still be fully converted, but post-conversion action may be required.
ARPaymentInvalidData.log	This file contains a log of AR Payments which contained some invalid data, but were processed during the conversion. Examples of such invalid data are any AR Payments in which the Company, Customer, or Daybook information is invalid or references a nonexistent record. The AR Payment record will still be fully converted, but post-conversion action may be required.
BankFormatInvalidData.log	This file contains a log of errors encountered when determining the Bank Format details for a customer or supplier bank. An example of such invalid data are an invalid BankNumber and Payment Format combination. The record will still be fully converted, but post-conversion action may be required.
PaymentStatusInvalidData.log	This file contains a log of any PIP (Payment in Process) accounts that were defined with an incorrect GL Account Type.
ConversionAccountPostings.log	This file contains a log if there are any postings against the Control account of a Customer, or Supplier, but against the mirror of the transaction account.
ARBankNumberChange.log	This file contains a log of any records in which it was necessary to default the bank details for AR Transactions. One of two methods can be used to default the values: 1. The Bank details associated with ar_bank are used instead of ar_acct. This default is used if ar_acct is not defined as a valid bank. 2. A default bank number based on the Customer and Customer bank number. This default is used if both ar_acct and ar_bank fail to provide a valid bank.
BankInvalidData.log	This file contains a log of any errors encountered when determining the Supplier Bank details. An example of such invalid data would be blank pay format data for a supplier.

Log File Name	Description
GLInvalidData.log	This file contains a log of any invalid data encountered when converting GL History (gltr_hist) records. Examples of such invalid data are an invalid GL Entity (gltr_hist.gltr_entity) or GL Account (gltr_acct).
PostingBalance.log	This file contains details of any additional Posting which the conversion had to make to ensure a GL Transaction Posting balances. This log entry provides traceability for these newly created records.
BankConsolidation.log	During the pre-conversion process, you must specify a replacement Payment in Process (PIP) account or replacement Drafts Payable account in the GL Account Type Utility which the conversion will use for bank records having the same domain (where applicable) and entity and Cash account not the same PIP account or Drafts Payable account. Therefore, any transactions that were posted against the original PIP or Drafts Payable account are converted using the replacement account. This log file contains a record of such transactions where the PIP or Drafts Payable account were replaced with the new PIP or Drafts Payable account.

Note These log files and the `qdtadmin.log` file are created in the QDT logs directory.

Log File Naming Conventions

The `qdtadmin.log` file is renamed if you select to clear the log file before beginning the execution process or if it reaches a size greater than 1 MB.

It is renamed using the following logic:

`qdtadmin.log` is renamed `qdtadmin001.log`. If `qdtadmin001.log` already exists, it is renamed `qdtadmin002.log`, and so on.

Also, at the start of the financial conversion any existing financial log files are renamed using logic similar to `qdtadmin`. For example,

`AddressMaster.log` becomes `AddressMaster001.log`

Reviewing Log Files After Conversion

Once a conversion has completed (or a section of the conversion if using the conversion snapshots functionality, described in “Snapshots” on page 161), the `qdtadmin.log` file should be reviewed for any errors and warnings. Also, if any of the financial conversion log files were created, they too should be reviewed and analyzed.

All data related errors and warnings should be reviewed by a Financial Consultant to determine if there is a need for any corrective action.

Appendix G

Snapshots

This appendix describes conversion snapshots.

Introduction 162

Pause Points - convprogpauselist.ini 162

Backup and Restoring a Snapshot 163

Introduction

You can configure the conversion process to pause at predefined points to allow you to make backups (snapshots). You can use snapshots to easily restore and restart the conversion at the point where a snapshot was made.

Note The snapshot feature is only available for conversions. It is not available for upgrades.

Pause Points - convprogpauselist.ini

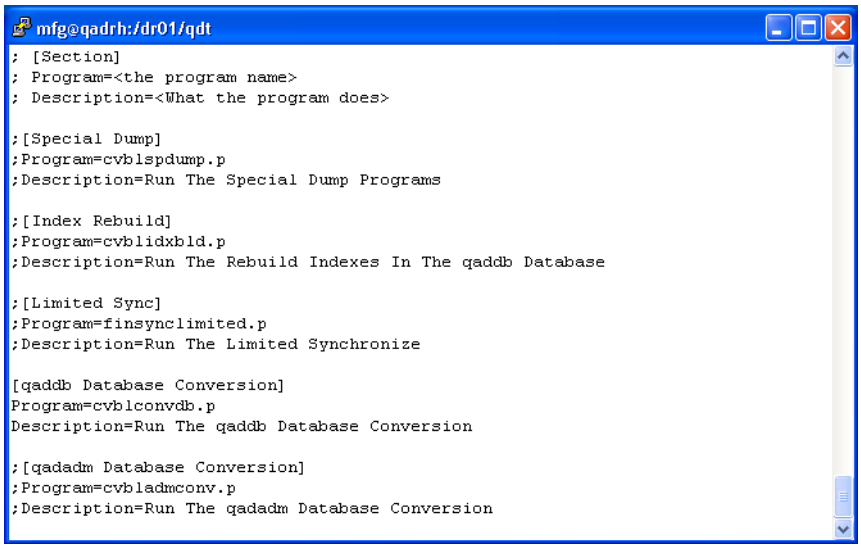
The pause points are defined in the `convprogpauselist.ini` configuration file. The conversion pauses at these points if the Pause Before Executing Selected Actions option is checked for the conversion. The following table lists the pause points in the `convprogpauselist.ini` file.

Table G.1
Pause Points

Pause Point Name	Prompt Message	Comments
[Special Dump]	Run The Special Dump Programs	The conversion finishes environment creation and compile. The next step is to run the special dump programs.
[Index Rebuild]	Run The Rebuild Indexes In The qaddb Database	The conversion finishes all the steps before and including BufferCopy. The next step is to rebuild indexes in the qaddb database.
[Limited Sync]	Run The Limited Synchronize	The conversion finishes all the steps before and including index rebuild. The next step is to run the Limited Synchronize.
[qaddb Database Conversion]	Run The qaddb Database Conversion	The conversion finishes all the steps before and including Limited Sync. The next step is to run the qaddb database conversion.
[qadadm Database Conversion]	Run The qadadm Database Conversion	The conversion finishes the steps before and including qaddb data base conversion. The next step is to run the qadadm database conversion.

In `convprogpauselist.ini`, only the qaddb Database Conversion pause point is enabled by default. Thus, the conversion will only pause before it tries to run the qaddb database conversion.

Fig. G.1
Pause Point Messages



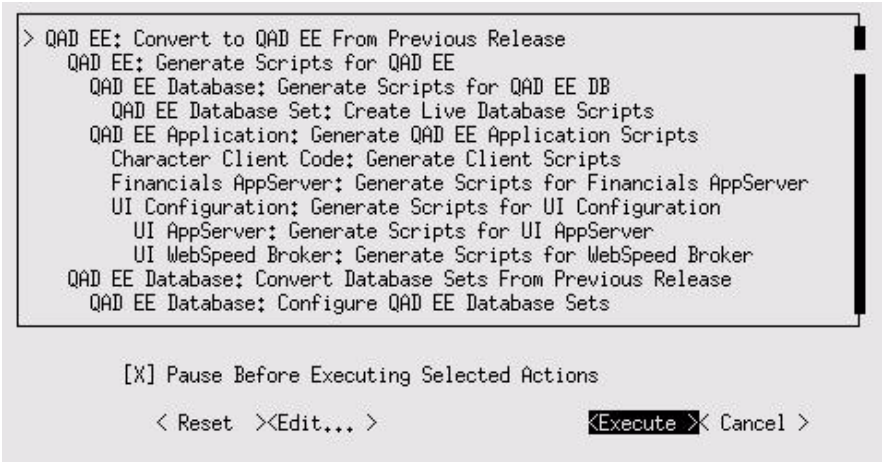
The remaining pause points are disabled by default. Removing the leading semicolons from these sections will enable the pause point while adding leading semicolons to the section disables the pause point.

Note If all the pause points are commented-out from the configuration file, it will pause before executing each action, not just the ones in the configuration file.

Backup and Restoring a Snapshot

Checking the Pause Before Executing Selected Actions is the prerequisite for taking a snapshot.

Fig. G.2
Pause Before Executing Selected Actions



Once selected, the conversion pauses at the steps defined in the configuration file. Before running each step, qdtadmin prompts you to continue. This gives you the opportunity to take a snapshot at this particular step by selecting No, exiting from QDT, and taking a snapshot. You can then restart QDT and restart the conversion from where you left off.

Creating a Snapshot

1. Exit QDT at the step from which the snapshot is to be taken.
2. Stop the environment by running the `stopenv.<environment name>` script under `<QDT installation directory>/envs/<environment name>/scripts`.
3. Back up the following files and databases:
   - `<QDT installation directory>/finconvdata.ini`
   - `<QDT installation directory>/xml/QDT_<hostname>_<environment name>.xml`
   - qaddb and qadadm databases under `<QAD EE installation directory>/db/` (use the `probkup` utility)
4. Restart the environment by running the `startenv.<environment name>` script under `<QDT installation directory>/envs/<environment name>/scripts`.
5. Start QDT and execute the conversion as outlined in Chapter 3, “Conversion Execution,” on page 31.

Note To maintain the integrity of the snapshot, QAD recommends placing all of the files backed up during a snapshot into one location.

Restoring a Snapshot

1. Close any QDT sessions.
2. Stop the environment by running the `stopenv.<environment name>` script under `<QDT installation directory>/envs/<environment name>/scripts`.
3. Restore the following files and databases:
   - `<QDT installation directory>/finconvdata.ini`
   - `<QDT installation directory>/xml/QDT_<hostname>_<environment name>.xml`
   - qaddb and qadadm databases under `<QAD EE installation directory>/db/` (use the `prorest` utility)
4. Restart the environment by running the `startenv.<environment name>` script under `<QDT installation directory>/envs/<environment name>/scripts`.
5. Start QDT and execute the conversion as outlined in Chapter 3, “Conversion Execution,” on page 31.

Index

B

books, converting 128
business relations 134

C

Canadian taxes to GTM, converting 105
classes, converting 130
consolidation 146
conversion
  accounts 137
  execute 44
  execution 31
  methods 126
  overview 1
  parameters, set 24
  setup 33
  validation 45
conversion execution
  overview 32
conversions 2
converted data 133
convprogpauselist.ini 162
credit
  ratings 142
  terms 142
customer types 142

D

daybooks 140

E

entities 135
environment, prepare 32
errors
  conversions not enabled 154, 159
  Data Preparation Report 154, 159
  Progress errors during conversion 156
  qdtadmin.log 155
  reindex.log 154
exchange rates 141

F

Fixed Assets Migration Utility 123
  running 124

G

generalized codes 140
GTM
  conversions 73
  conversions summary 74
  from no taxes, converting 104

I

invoice status codes 143

L

layers 134
legacy data, mapping 126
license codes, enter 45
locations, converting 129
log files 157

M

migration
  defaults, setting 125
  reporting 131

P

pause points 162
payment
  formats 146
  status codes 143
post-conversion 47
  data validations 49
  overview 48
  reports 55
  setup, mandatory 56
  setup, optional 58
  utilities 48
pre-conversion 5
  code, install 6
  completion 28
  overview 6
  utilities, run 9
prerequisite skills 3
process flow validation 56
profiles 135
Progress editor access 149
project
  groups 139
  status codes 139
purchase types 142

S

security 140
shared sets 134
snapshots 161
  backup 163
  creating 164
  pause points 162

restoring 163, 164
software
install and configure 32
source
data, prepare 7
databases, prepare 32
static
data validation 56
validation 56
supplier types 141

T
Troubleshooting 153, 157

U
upgrade path diagram 3
US taxes to GTM, converting 88

V
VAT taxes to GTM, converting 75