



Training Guide

BI v3 Technical Level 1

This document contains proprietary information that is protected by copyright and other intellectual property laws. No part of this document may be reproduced, translated, or modified without the prior written consent of QAD Inc. The information contained in this document is subject to change without notice.

QAD Inc. provides this material as is and makes no warranty of any kind, expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. QAD Inc. shall not be liable for errors contained herein or for incidental or consequential damages (including lost profits) in connection with the furnishing, performance, or use of this material whether based on warranty, contract, or other legal theory.

QAD and MFG/PRO are registered trademarks of QAD Inc. The QAD logo is a trademark of QAD Inc.

Designations used by other companies to distinguish their products are often claimed as trademarks. In this document, the product names appear in initial capital or all capital letters. Contact the appropriate companies for more information regarding trademarks and registration.

Copyright ©2013 by QAD Inc.

BI_TechLevel1_TG_v039.pdf/r9m/r9m

QAD Inc.
100 Innovation Place
Santa Barbara, California 93108
Phone (805) 684-6614
Fax (805) 684-1890
<http://www.qad.com>

Contents

| | |
|---|-----------|
| ABOUT THIS COURSE | 9 |
| QAD BI TECHNICAL LEVEL 1 CERTIFICATION – PART 1 | 12 |
| QAD BI v3 -Technical Level 1 Certification - Part I | 13 |
| Course Objectives | 17 |
| Course Benefits | 18 |
| Agenda - Day 1 | 19 |
| Agenda - Day 2 | 20 |
| Agenda - Day 3 | 21 |
| 1 – 1 Overview – What is BI? | 22 |
| An Example of BI | 23 |
| BI Overview | 24 |
| BI Positioning | 25 |
| QAD BI 3 | 26 |
| QAD BI Components - Review | 27 |
| BI Glossary – From a Functional Standpoint | 28 |
| BI Glossary –From a Technical Standpoint | 29 |
| On-line Transaction Processing - OLTP | 31 |
| On-line Analytical Processing - OLAP | 32 |
| Facts & Dimensions | 33 |
| Dimensions | 37 |
| BI Architecture | 38 |
| BI Architecture – Technical Perspective | 39 |
| BI Modules – BI v3.9 | 44 |
| Review Question #1 | 46 |
| Review Question #2 | 47 |
| Review Question #3 | 48 |
| In Review – BI Overview | 49 |
| Installing the QAD BI Solution | 50 |
| Overview - Installing the QAD BI Solution | 51 |
| Installation - Getting Started | 52 |
| Pre-Install Customer Checks | 53 |
| Login to Training Environment | 54 |
| Confirm Database Settings | 56 |
| Confirming the Database Points to the Correct Directory | 57 |
| Define a Portal User | 60 |
| Install BI via Installer Program | 63 |
| Progress database(s) ODBC Pre-Setup | 70 |
| Install BI via Installer Program | 71 |
| Install BI Modules via Installer Program | 85 |
| BI Designer – Set Parameters | 93 |
| Running INITIAL_JOB_SETUP | 99 |
| Double-Check the Parameters | 103 |

| | |
|--|------------|
| Language Installations | 104 |
| Language Installation – Pre DWD 6.2.2.0 | 105 |
| Language Installation – DWD 6.2.2.0 and Up | 106 |
| Language Installation | 107 |
| Database Size Validation Prior to Hist Load | 111 |
| Start the Historical Load | 114 |
| BI Portal Web Application – Getting Started | 115 |
| Install Standard Dashboards and Reports | 117 |
| BI Portal – Create New Users | 121 |
| Day 1: In Review | 125 |
| Questions? | 126 |
| QAD BI TECHNICAL LEVEL 1 CERTIFICATION – PART 2 | 128 |
| QAD BI v3 - Technical Level 1 Certification - Part 2 | 129 |
| Agenda - Day 2 | 130 |
| Day 1 Review | 131 |
| Confirm Historical Loads Have Run | 132 |
| Review of Day 1 - Problems Encountered | 133 |
| Review of Day 1 – Setting Connections and Scheduler | 134 |
| DWD Scheduler | 137 |
| Scheduling the DAILY Job Run | 138 |
| Scheduler – Jobs Inside Jobs | 139 |
| Scheduler – /F Switch Failure | 140 |
| Scheduler – “Over-packed” Columns | 145 |
| The Pieces So Far- How This All Ties Together | 146 |
| Cubes | 148 |
| Cubes – What are they? | 149 |
| Cube Structure | 150 |
| DWD Cube Types | 151 |
| Cubes Setup – Set Connections | 152 |
| Cubes Setup – Build the Olap Cubes | 154 |
| Cubes Setup – Build the Cubes (v5) | 155 |
| Cubes Processing – Run the Job | 156 |
| Setup Microsoft Excel for Cubes | 157 |
| A Complete Review of the DWD Tool | 161 |
| QAD Administrator | 162 |
| DWD Interface | 163 |
| Logging in to QAD DWD | 164 |
| DWD Builder/Browser Window Panes | 165 |
| DWD Builder/Browser Panes | 166 |
| DWD Builder/Browser Toolbar | 167 |
| WhereScape RED© | 168 |

| | |
|---|-----|
| DWD Concepts..... | 169 |
| Data Flow – Normalized Models..... | 170 |
| DWD Objects..... | 171 |
| Data Objects Types..... | 172 |
| DWD Object Builder..... | 174 |
| DWD Object Builder – Groups / Projects..... | 175 |
| Connections..... | 177 |
| Load Tables..... | 178 |
| Load Tables - Properties..... | 179 |
| Staging Tables..... | 180 |
| Table Update Procedures..... | 182 |
| Dimension Tables and Dimension Views..... | 183 |
| Fact Tables..... | 184 |
| Data Stores..... | 185 |
| Cubes..... | 186 |
| Procedures..... | 187 |
| Index..... | 188 |
| Diagrams..... | 189 |
| Diagram..... | 190 |
| Source Tracking Diagram..... | 193 |
| Reports..... | 194 |
| DWD Reports..... | 195 |
| Fact to Dimension Matrix..... | 196 |
| Export Report To Excel..... | 197 |
| Scheduler Components..... | 198 |
| Data Flow..... | 199 |
| Scheduler Hierarchy..... | 200 |
| Common Task Actions..... | 201 |
| Objects & Procedures..... | 202 |
| Jobs and Tasks..... | 203 |
| Scheduler Navigation..... | 204 |
| Scheduler Access..... | 205 |
| Scheduler Toolbars and Menus..... | 206 |
| Scheduler Panes..... | 207 |
| Scheduler States..... | 208 |
| Scheduler States - JOBS..... | 209 |
| Scheduler States - TASKS..... | 211 |
| Scheduler Status..... | 212 |
| Refresh Display..... | 213 |
| Troubleshooting Scheduler Issues..... | 214 |
| Audit Logs..... | 215 |
| Audit Trail at Job Level..... | 216 |
| Manually Starting a Job..... | 217 |
| DAILY_START..... | 218 |

| | |
|--|------------|
| Scheduler Documentation | 219 |
| Job Diagram | 220 |
| Job Documentation | 221 |
| DWD Reports | 222 |
| Course Review | 223 |
| Course Review – DWD Scheduler | 224 |
| DWD – Features – Data Tools | 225 |
| DWD – Features – Diagram Views | 226 |
| DWD – Features – Diagram Views – Options | 227 |
| DWD – Features – Diagram Views – Star Schema View | 228 |
| DWD – Features – Diagram Views – Source Tracking | 229 |
| DWD – Features – Diagram Views – Data Join Track Back | 230 |
| DWD – Features – Diagram Views – Track Forward | 231 |
| DWD – Features – Diagram Views – Linked Tables | 232 |
| DWD – Features – Diagram Views – Jobs | 233 |
| DWD – Features – Diagram Views – Exercises | 234 |
| DWD – How Data Flows Into the Warehouse | 235 |
| QAD Data Warehouse Designer | 236 |
| DWD – Features – Column Track Back | 237 |
| DWD – Features – Column Track Back - Exercise | 242 |
| DWD – Features – Search For String | 243 |
| DWD – Features – Column Transformations | 245 |
| HISTORY TABLES | 247 |
| SNAPSHOT FROM HISTORY | 250 |
| DWD – Jobs | 251 |
| DWD – Jobs and Parameters | 252 |
| Important Notes – DWD JOBS | 253 |
| Important Notes – Other Key DWD Components | 255 |
| BI Portal – Administration | 258 |
| BI Portal – User | 259 |
| Day 2: In Review | 260 |
| QAD BI TECHNICAL LEVEL 1 CERTIFICATION – PART 3 | 262 |
| QAD BI v3 - Technical Level 1 Certification - Part 3 | 263 |
| Agenda - Day 3 | 264 |
| Day 2 Summary | 265 |
| Building the CFO Dashboard | 266 |
| DWD – Creating Documentation | 284 |
| Day 3 - In Review | 290 |

| | |
|--|------------|
| QAD BI TECHNICAL LEVEL 1 CERTIFICATION – EXTRAS | 293 |
| QAD BI v3 - Technical Level 1 Certification - Extras..... | 294 |
| QAD BI 3.x Customization Methodology..... | 295 |
| Dim Table Customizations | 296 |
| Non-History Fact Table Customization | 297 |
| History Fact Table Customization | 298 |
| Customizations – Why We Chose Our Methodology..... | 299 |
| Adding a Column to a Table..... | 309 |
| Adding a Column to a Table – Drag and Drop Methodology..... | 313 |
| Adding a Column to a Table..... | 344 |
| Changing Procedures and Finding Differences Between Procedure Versions | 346 |
| Columns that Exist in Some Versions of the ERP in a Table and Not in Others..... | 357 |
| Adding a Table from QAD’s ERP Progress Database..... | 358 |
| Creating a Patch/Installing a Patch | 374 |
| Environment Comparisons..... | 387 |
| Importing a Flat File with Foreign Characters..... | 393 |
| Running Jobs Stand-Alone. | 403 |
| HISTORIC RELOAD..... | 404 |
| Pre-run dbtool Progress Database Check..... | 407 |
| Eb2 – Pre-domain Environments Integration Into our BI Solution. | 416 |
| The Pieces so far- How This all Ties Together | 417 |
| Summarization..... | 418 |

Change Summary

The following table summarizes significant differences between this document and the previous version.

| Date/Version | Description | Reference |
|---------------------|-------------------------|------------------|
| November 2013 | 1 st Release | -- |
| | | -- |
| | | -- |
| | | -- |
| | | |
| | | |

About This Course

Course Description

This course covers the basics to install and support the Business Intelligence solution and includes activities and exercises throughout the course. The topics covered include the following:

- Installing BI3, including Data Warehouse Designer, Modules, BI Portal and Tomcat
- Connecting to Source systems
- Using the BI Scheduler
- Configuring and Executing BI Data Loads
- Data Validation
- Activities and exercises throughout the class.

Course Objectives

This class is designed to provide you with a fairly intensive overview of the BI installation process. You will use the BI Installer to do the following:

- Setup a database
- Setup ODBC connections
- Install QAD BI DWD (Data Warehouse Designer)
- Install QAD BI Portal
- Support QAD's BI solution using some advanced features including troubleshooting
- Build cubes
- Integrate outside flat file data sources into DWD
- Run simple queries in SQL Server.

Audience

QAD and QAD Partner services consultants who plan to implement for QAD clients and those preparing for the BI Technical Level 1 certification exam. This course is not open to QAD customers at this time.

Prerequisites

Information about prerequisites is sent to participants by the instructor in an email prior to class start date. Before the class, participants are required to watch the following self-study videos:

- Running Queries and Manipulating Data in SQL Server 2008
- SQL Server 2008 Installation
- Completed Business Intelligence (BI): 1-2 Overview
- Completed Business Intelligence (BI): 1-3 Portal User Introduction

- Completed Business Intelligence (BI): 1-4 Modules
- Completed Business Intelligence (BI): 1-10 Data Warehouse Designer Introduction
- Completed Business Intelligence (BI): 1-11 Data Warehouse Scheduler Introduction

In addition, an intermediate knowledge of SQL Query is required. Also, a basic knowledge of SQL Server Management Studio, Tomcat and Excel Pivot tables is required.

Course Credit and Scheduling

This course provides 30 credit units. It is designed to be taught in 5 days.

Virtual Environment Information

The hands-on exercises in this book should be used with the latest BI learning environment.

Additional Resources

If you encounter questions on QAD software that are not addressed in this book, several resources are available. The QAD corporate Web site (<http://www.qad.com/>) provides product and company overviews. From the main site, you can access the QAD Learning or Support site and the QAD Document Library. Access to some portions of these sites depends on having a registered account.

QAD Learning Center

To view available training courses, locations, and materials, use the QAD Learning Center. Choose Education under the Services tab to access this resource. In the Learning Center, you can reserve a learning environment if you want to perform self-study and follow a training guide on your own.

QAD Document Library

To access release notes, user guides, training guides, and installation and conversion guides by product and release, visit the QAD Document Library. Choose Document Library under the Support tab. In the QAD Document Library, you can view HTML pages online, print specific pages, or download a PDF of an entire book. To find a resource, you can use the navigation tree on the left or use a powerful cross-document search, which finds all documents with your search terms and lets you refine the search by book type, product suite or module, and date published.

QAD Support

Support also offers an array of tools depending on your company's maintenance agreement with QAD. These include the Knowledgebase and QAD Forums, where you can post questions and search for topics of interest. To access these, choose Visit Online Support Center under the Support tab.

CHAPTER 1

QAD BI Technical Level 1 Certification – Part 1

QAD BI v3 -Technical Level 1 Certification - Part I

QAD BI v3 -Technical Level 1 Certification - Part I



Introductions

Introductions



Facilities



Rules of the Road

Rules of the Road

- First class 
- Interactive 
- Questions are welcome



Course Objectives

Course Objectives

- Using the BI Installer to do the following

- Setup a database
- Setup ODBC connections
- Install QAD BI DWD (Data Warehouse Designer)
- Install QAD BI Portal

Also

- Support QAD's BI solution using some advanced features including troubleshooting
- Build cubes
- Integrate outside flat file data sources into DWD
- Run simple queries in SQL Server



By the end of this class, you will know how to

Course Benefits

Course Benefits

- Additional skill sets
- Customer satisfaction
- BI is a cornerstone of QAD's future plans

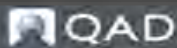


By learning these skills we benefit by

Agenda - Day 1

Agenda - Day 1

- Introductions, Facilities and Rules of the Road
- Course Objectives & Benefits
- Agenda Review
- BI and Module Overview
- Install the QAD BI Data Warehouse Designer (DWD) and BI Portal
- Install Languages
- Start Historical Load
- Setup BI Portal – Dashboards, Reports, Users



Agenda - Day 2

Agenda - Day 2

- Day 1 Review
- Confirm historical data loads were successful
- Discuss various failures & possibilities
- Build and populate cubes
- Connect ad hoc Excel to cubes
- Review of the DWD and the Admin tool
- Review and build the CFO dashboard

Agenda - Day 3

Agenda - Day 3

- Day 2 Review
- Basic and some advanced features of DWD
- DWD reporting options
 - Track Back / Track Forward
 - Diagram views
- Column Transformations
- Modules
- Jobs and Parameters
- Certification Exam

1 – 1 Overview – What is BI?

1 – 1 Overview – What is BI?

Business intelligence (BI) refers to computer based techniques used in identifying, extracting, and analyzing business data. BI technologies provide historical, current and predictive views of business operations. Common functions of business intelligence technologies are reporting, online analytical processing, analytics, data mining, business performance management, benchmarking, text mining and predictive analytics.

An Example of BI

An Example of BI

Sales revenue by products and/or departments, or by associated costs and incomes.



BI Overview



Use this video for the BI overview.

<https://share.qad.com/gm/document-1.9.697589/BI1-2%20BI%20Overview.wmv>

<http://media.qad.com/qad/Play/c69f15137f90482081751cefc74d86c21d?catalog=229c8aec-7ef0-4643-be54-2a5b3227c04b>



BI Positioning



While decision making is moving down the organization, the day to day requirements of company representatives and the breadth of the analysis varies by a user's position in the corporate hierarchy. Operational representatives such as warehouse staff and planners regularly use very detailed capabilities such as inquiries and reports or browses and collections to manage their operational tasks. As we move up through the level of focus from Operational to Strategic, we move from detailed, daily requirements to higher levels of aggregation and a greater need for analysis by various attributes such as product family, product group, customer, or region. Another form of analysis is changes tactically and strategically over time. How have our inventory turns changed by product line or family over the past 6 or 12 months. Aggregations, attributes, and changes over time are all the domain of Business Intelligence.

QAD BI 3

QAD BI 3

- + **QAD BI Data Warehouse Designer (DWD)**

 - Easy to use
 - Consolidated / optimized – simple / fast access
 - QAD & non-QAD data sources
- + **Modules**

 - Easy to use
 - ERP “aware”
 - Extensible
- + **QAD BI Collaborative Portal**

 - Easy to use
 - Create & use queries, reports and dashboards



Key Concept:

QAD BI Components:

QAD BI Data Warehouse Developer (DWD)

Consolidate into a single warehouse optimized for simple, fast access

QAD and non-QAD data sources

Modules

ERP “aware” streamlines access to current and historical QAD ERP data

Extensible to support non-QAD data sources

QAD BI Collaborative Portal

Easy to use and create dashboard and reporting environment

QAD BI Components - Review

QAD BI Components - Review

- **QAD BI DWD** – ETL Tool
- **SQL Server** – Data Warehouse Database
- **BI Architecture** – Star Schema and Cubes
- **Modules** – Financials, Order Management, Operations, EAM and TMS
- **QAD BI Portal** – GUI Interface to the data



BI Glossary – From a Functional Standpoint

BI Glossary – From a Functional Standpoint

- **Facts** – Numerical information that can be analyzed
- **Dimensions** – Qualifiers of Facts
- **Star schema** – Method of organizing facts and dimensions
- **Metadata** – Data about data



Glossary:

Facts – Numerical information that can be analyzed by BI applications. E.g. total quantity sold, total sales dollars

Dimensions – Qualifiers of Facts. E.g. Transaction date, Customer name, Sales territory

Star schema - Method of organizing facts and dimensions for BI purposes. Star schemas can be deployed into Relational databases (in our case SQL Server) and into proprietary Cubes . Examples of the latter are Microsoft’s Analysis Services cubes or Cognos’ cubes.

Metadata – Data about data. For example: Columns in a fact table. Column names and formats.

BI Glossary –From a Technical Standpoint

BI Glossary –From a Technical Standpoint

- **Load tables** – Repository for data extracted from QAD's ERP progress tables.
- **Stage tables** – Tables used to transform data between the load tables and the fact and dimension tables.



BI Glossary –From a Technical Standpoint (cont.)

BI Glossary –From a Technical Standpoint (cont.)

- **Perm tables** – Repository of some ERP table's data prior to being sent to the fact tables
- **Cubes** – A cube contains dimensions, hierarchies, levels, and measures. Each individual point in a cube is referred to as a cell.

On-line Transaction Processing - OLTP

On-line Transaction Processing - OLTP

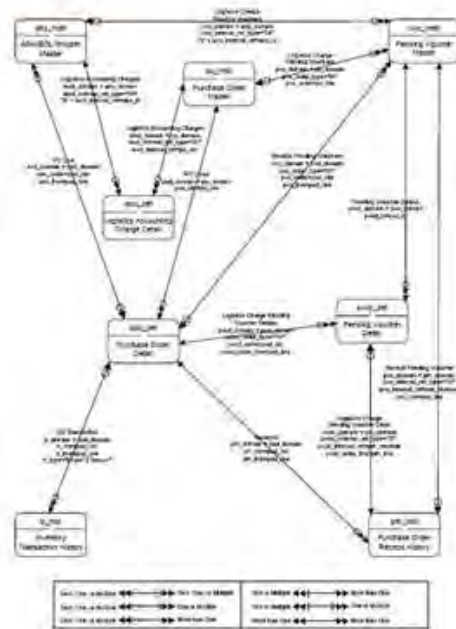
Pros:

- Good for transactions
- Interfaces to each table

Cons:

- Difficult for data analysis
- Slow query results

Purchasing III



On-line Analytical Processing - OLAP

On-line Analytical Processing - OLAP

Pros:

- Good for data analysis
- Supports complex queries

Cons:

- Partially de-normalized database design
- Larger database size




Facts & Dimensions

Facts & Dimensions

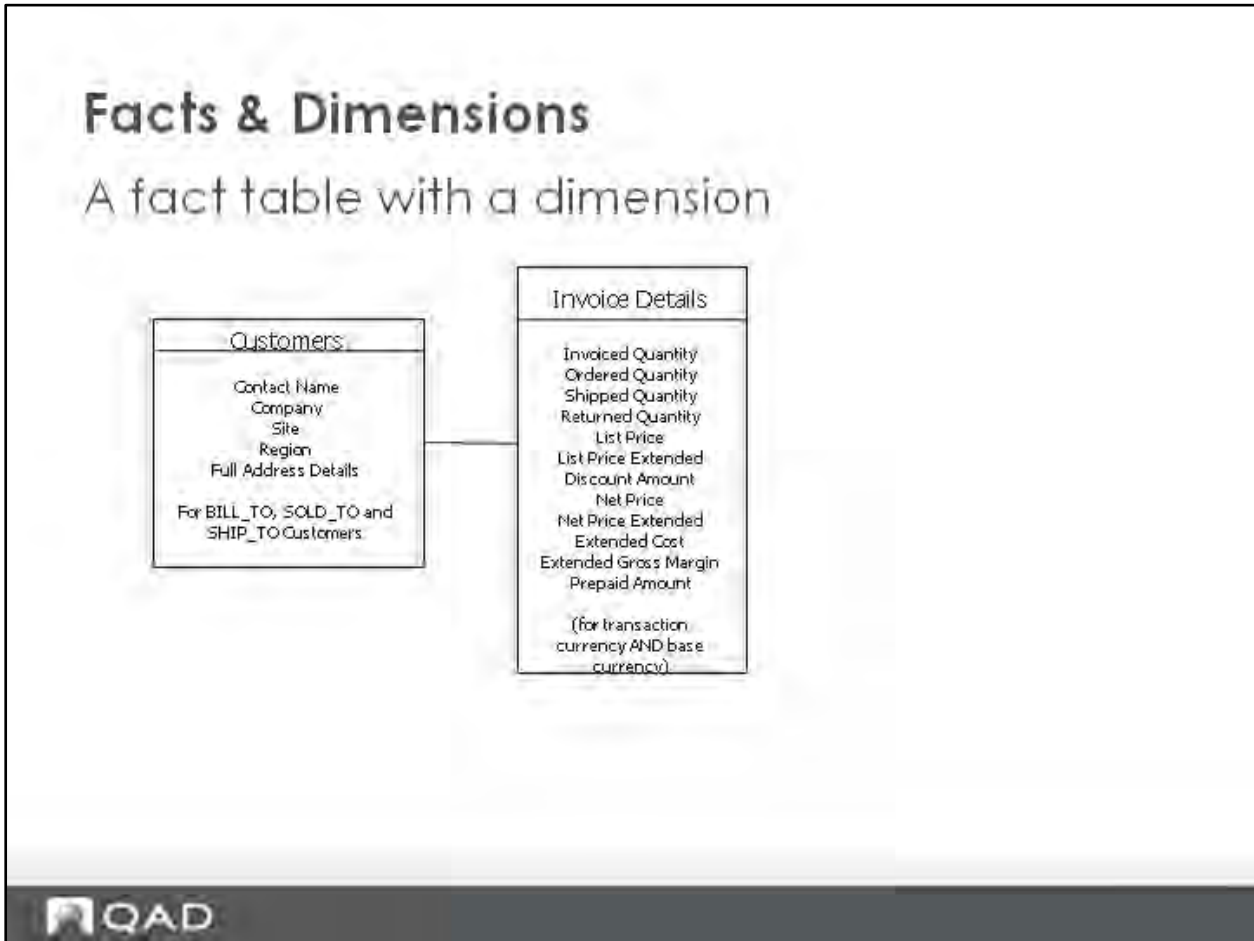
A fact table

| Invoice Details |
|--|
| Invoiced Quantity |
| Ordered Quantity |
| Shipped Quantity |
| Returned Quantity |
| List Price |
| List Price Extended |
| Discount Amount |
| Net Price |
| Net Price Extended |
| Extended Cost |
| Extended Gross Margin |
| Prepaid Amount |
| (for transaction currency AND base currency) |



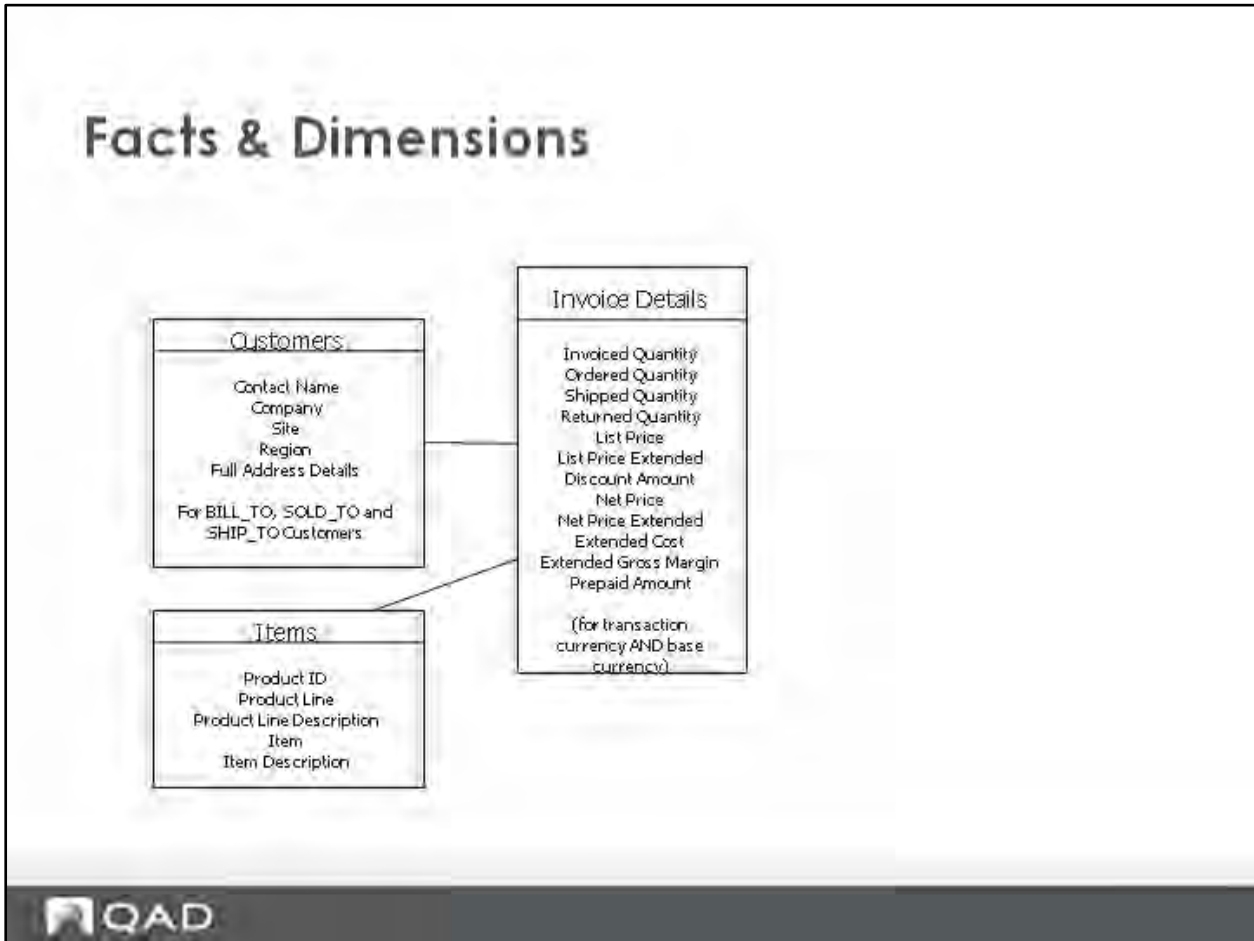
The object *Invoice Details* is a **Fact** table.

Facts & Dimensions



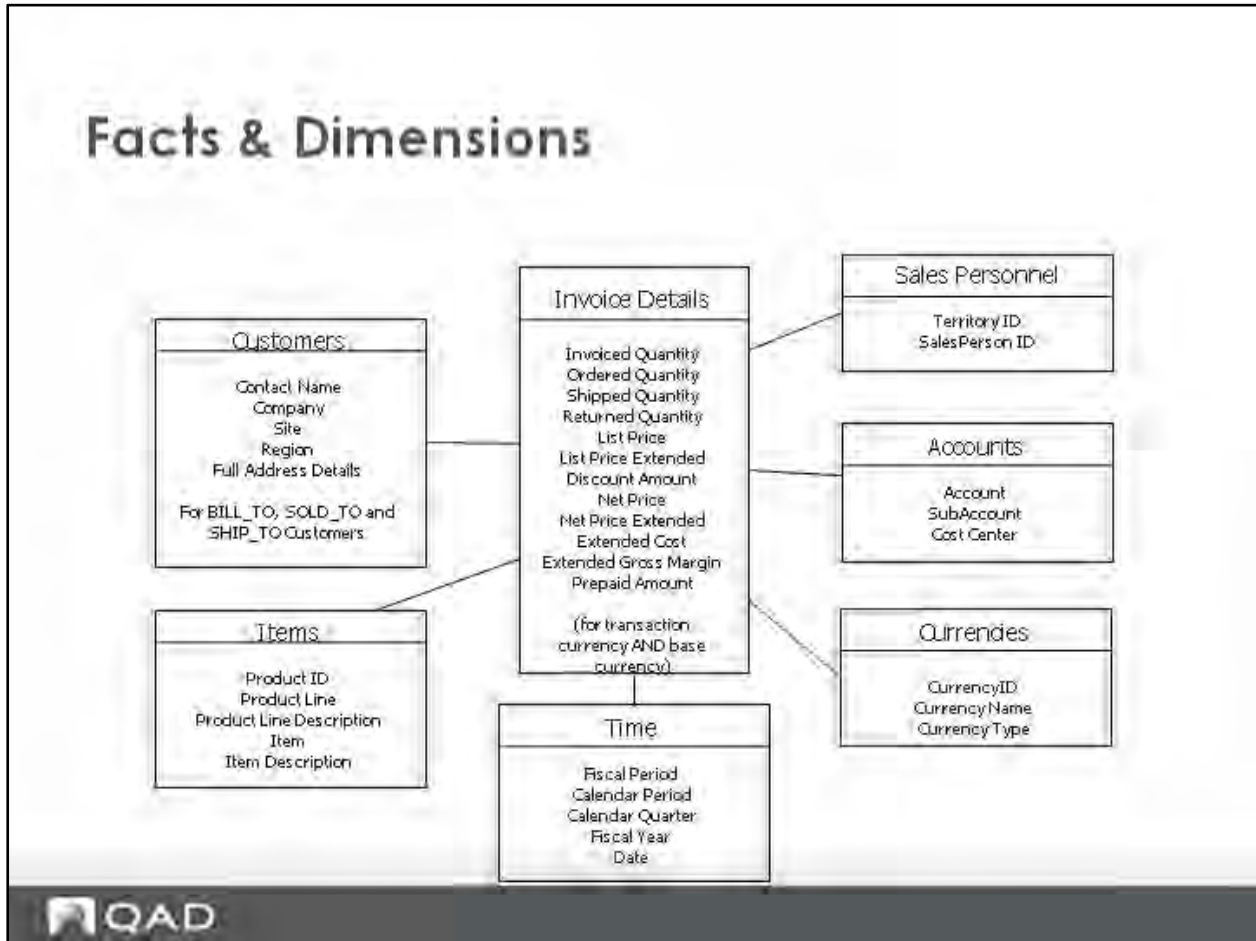
The object *Customers* is an example of a **Dimension** table.

Facts & Dimensions



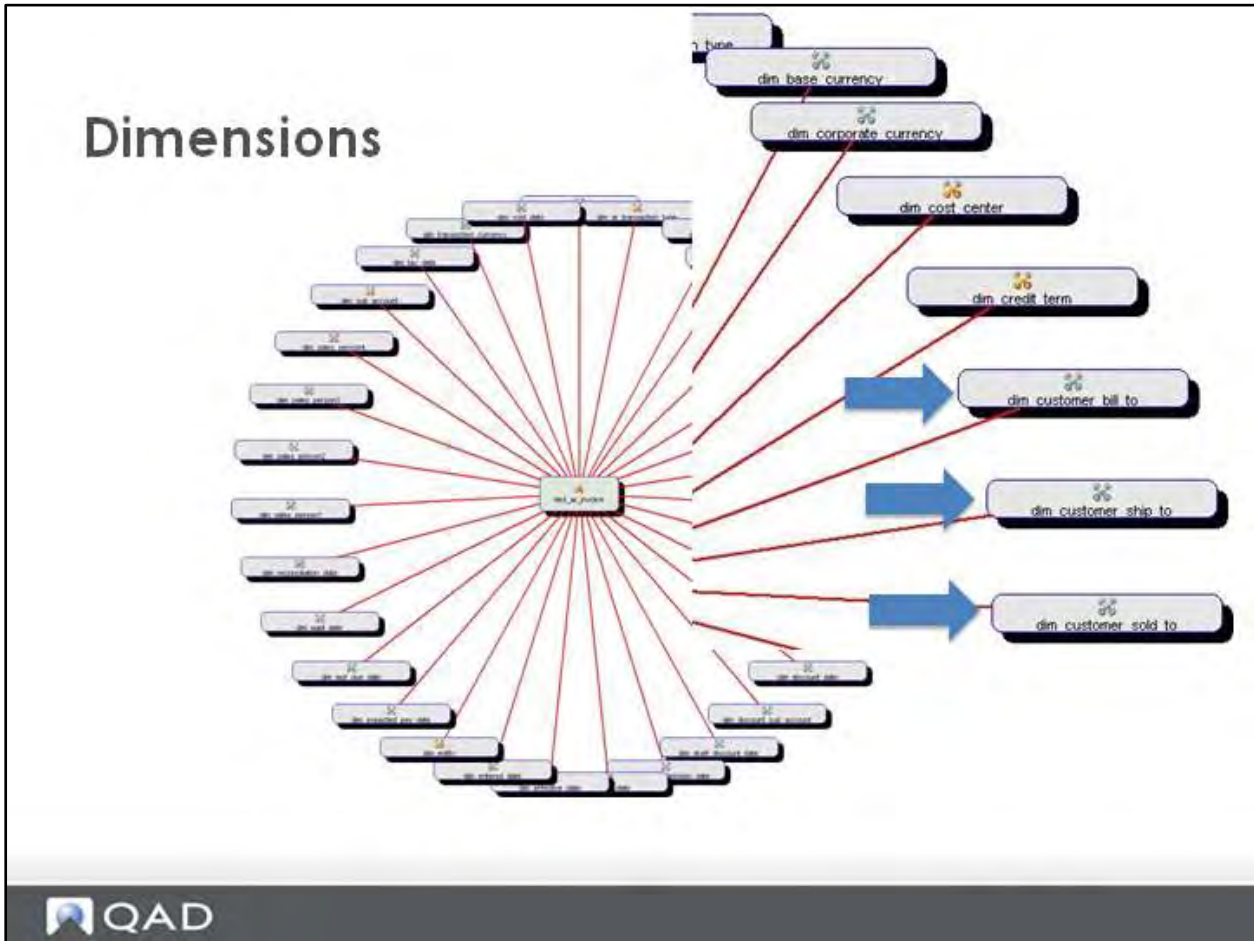
Items is another example of a **dimension** table.

Facts & Dimensions



Here are some other examples of dimension tables related to the fact table *Invoice Details*.

Dimensions



Dimensions are qualifiers of facts.

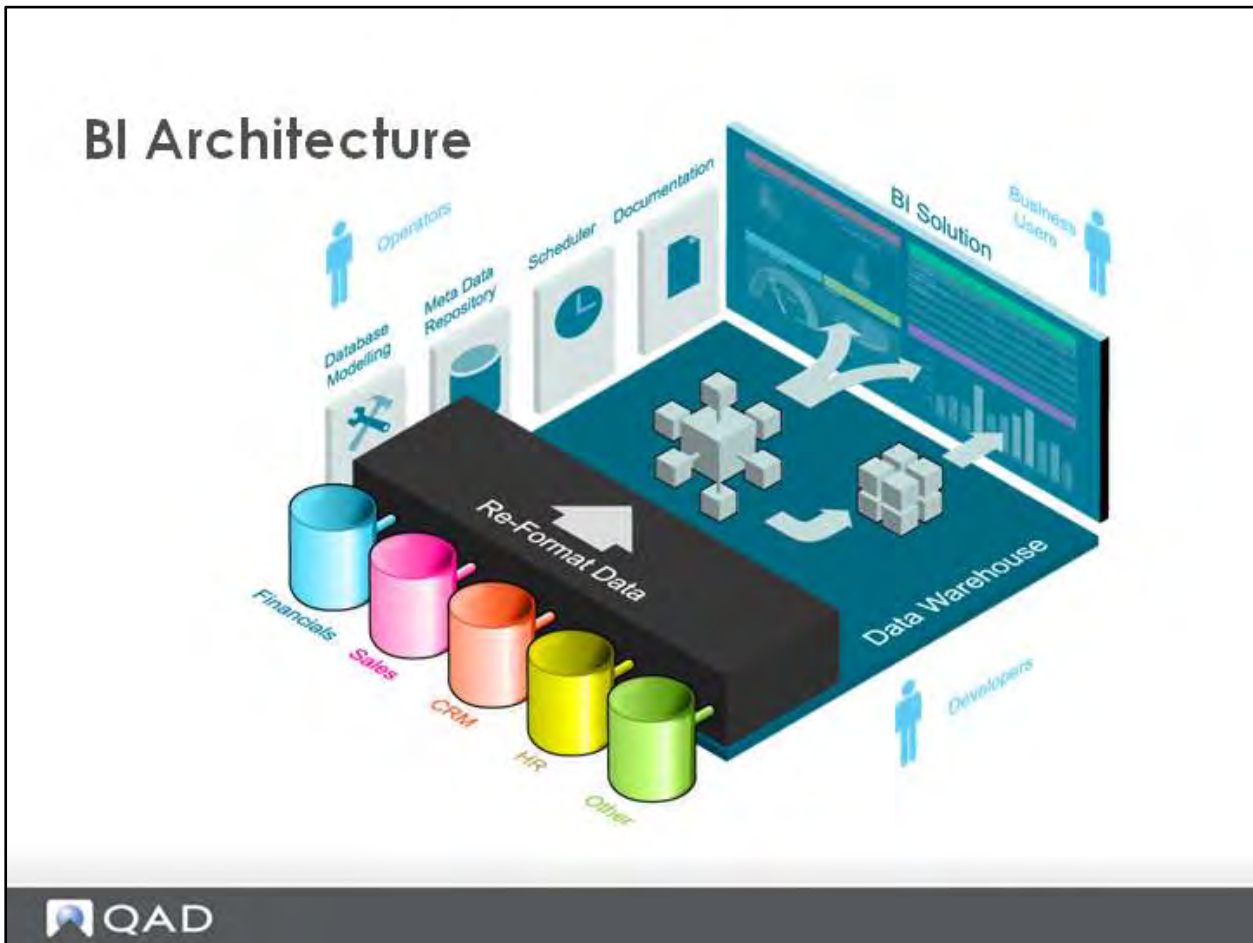
This diagram was produced using the QAD Data Warehouse Designer and one of the standard tools.

It is the fact_ar_invoice and all the related dimensions.

It is not important that you know which dimensions are available for each fact table. For standard QAD functionality, any dimension you need will generally be available.

The dimension tables shown in focus is an example of *dimension views*. A view makes it possible to look at a common file in different ways.

BI Architecture

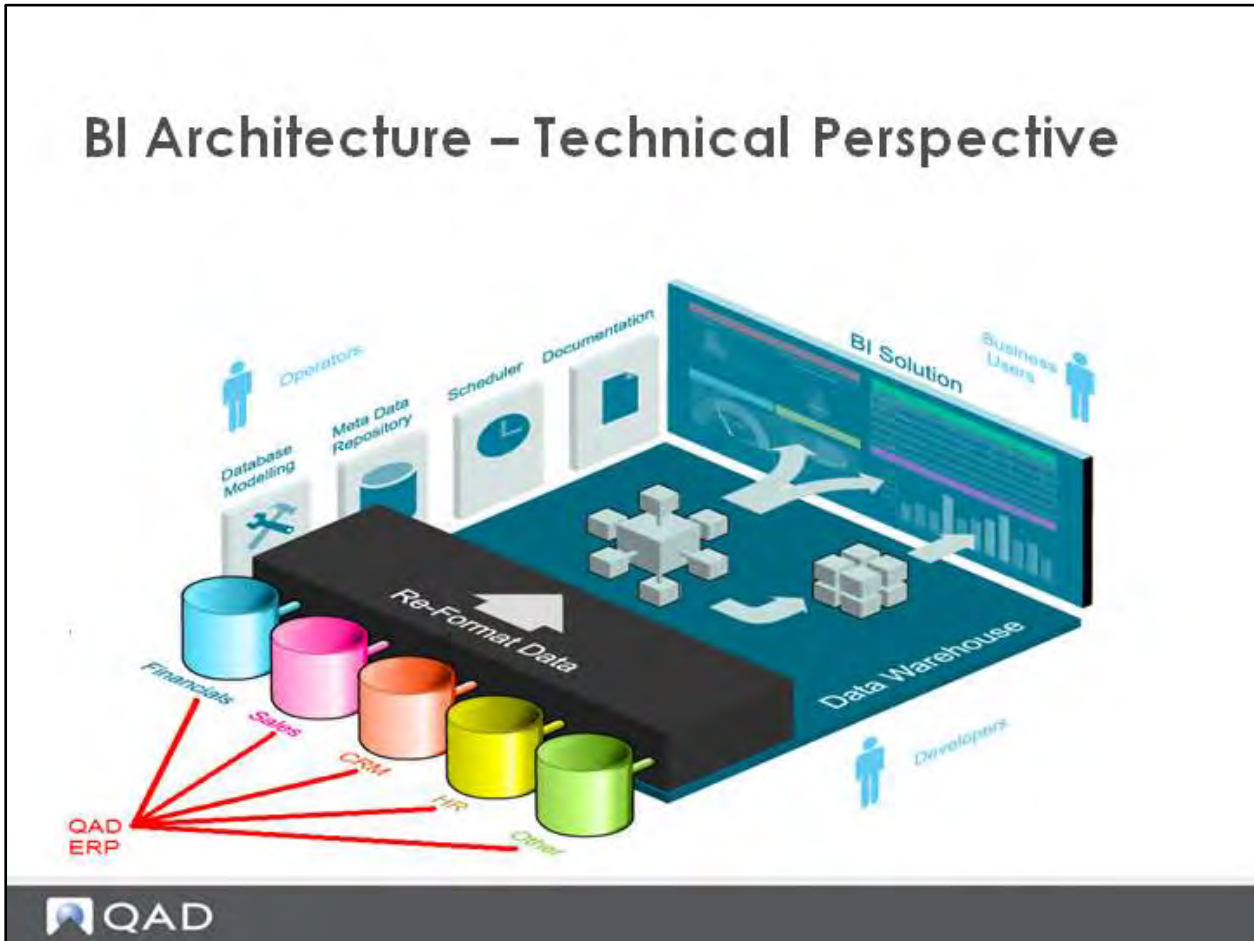


The last piece of the BI puzzle is Documentation.

The QAD DWD can automatically generate both technical and end-user documentation.

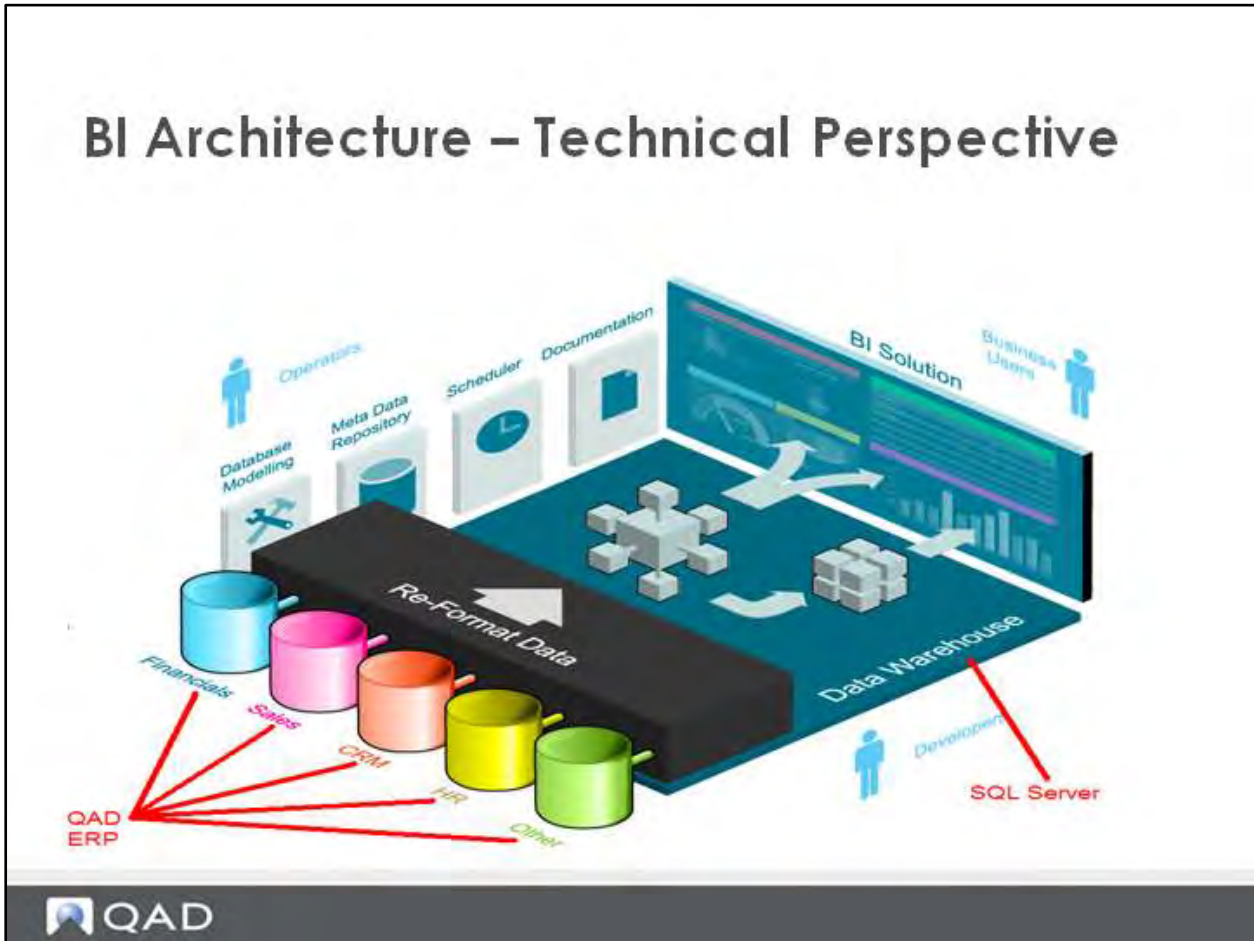
It also is our ETL tool. This is the structure of from the perspective of the DWD.

BI Architecture – Technical Perspective



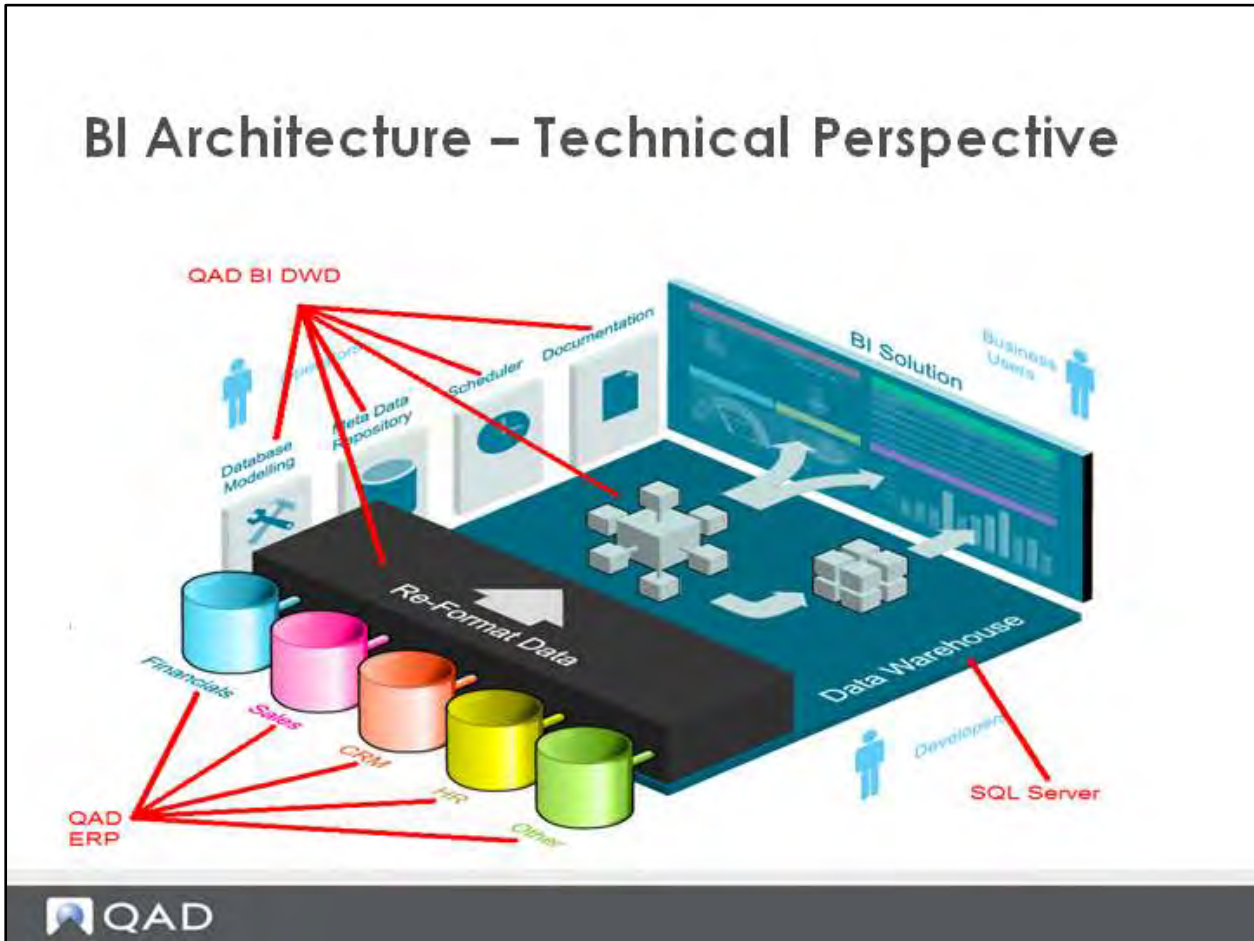
The QAD ERP Progress tables are the source of most of the data for our BI Modules.

BI Architecture – Technical Perspective



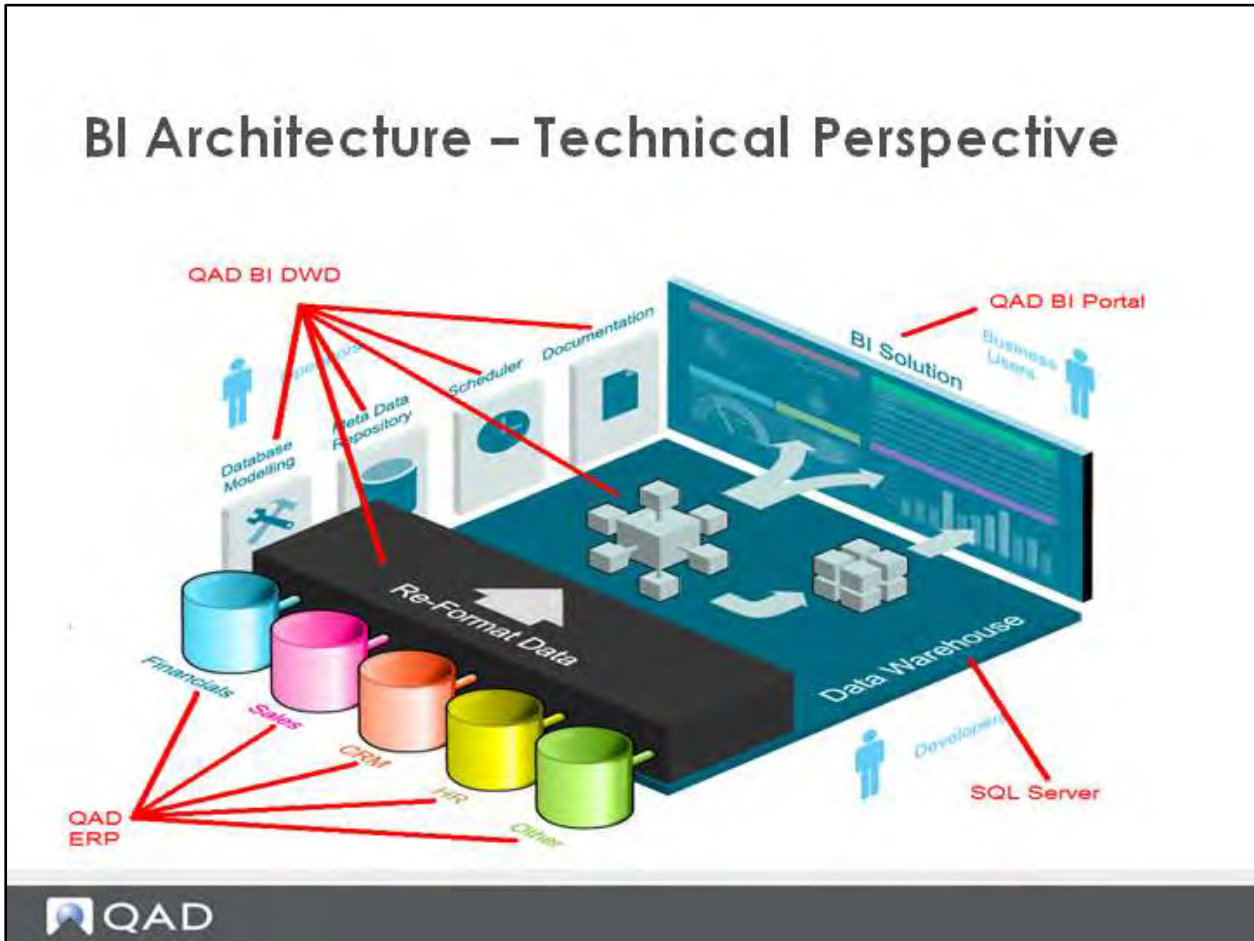
SQL Server is the underlying database that the data warehouse resides in.

BI Architecture – Technical Perspective



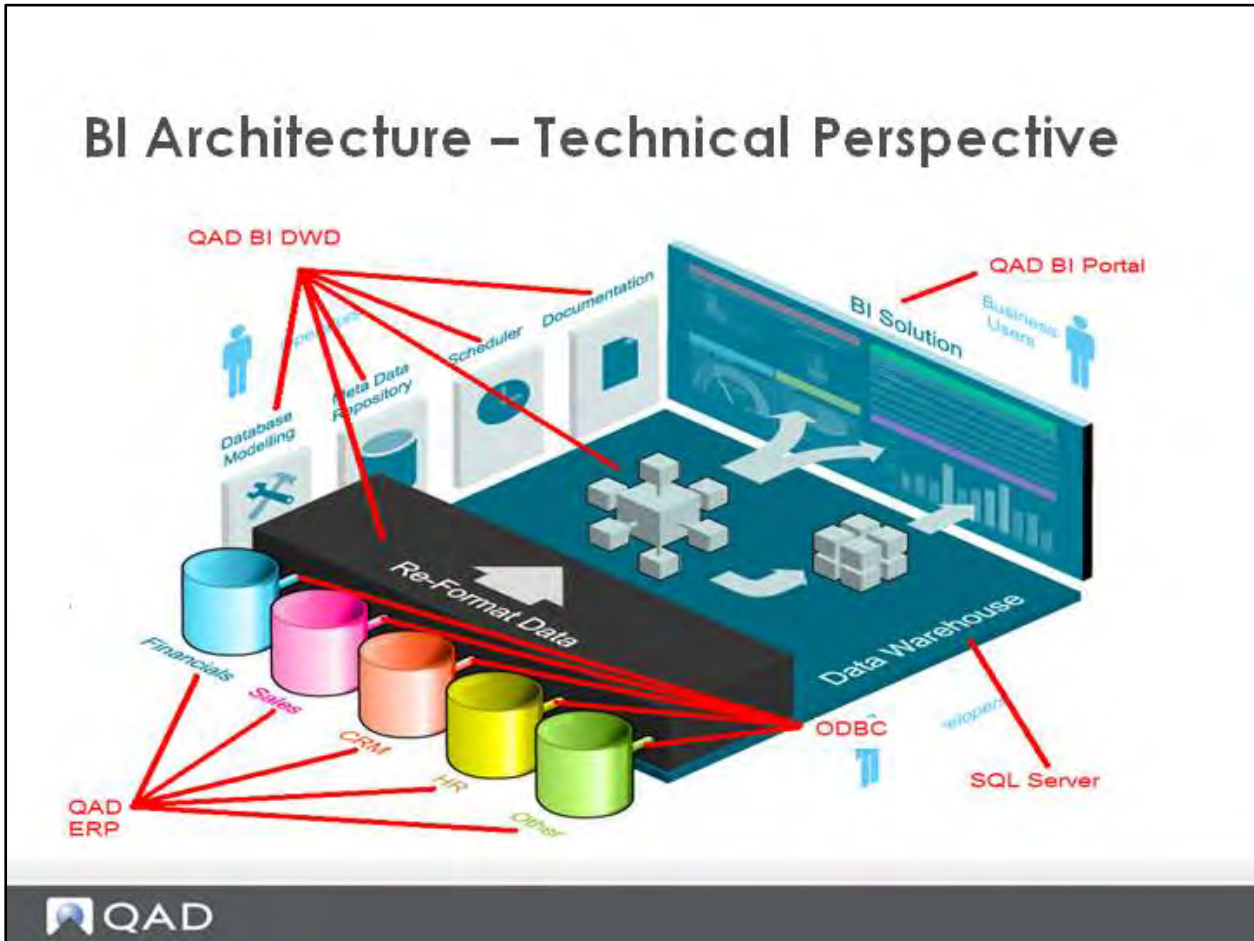
QAD BI DWD (Data Warehouse Designer) is the ETL tool that provides the means for Extraction, Transformation and Loading of data into the data warehouse. The DWD maintains all the metadata about the data warehouse tables as well as documentation, scheduling and anything else controlling the construction and building of the warehouses.

BI Architecture – Technical Perspective



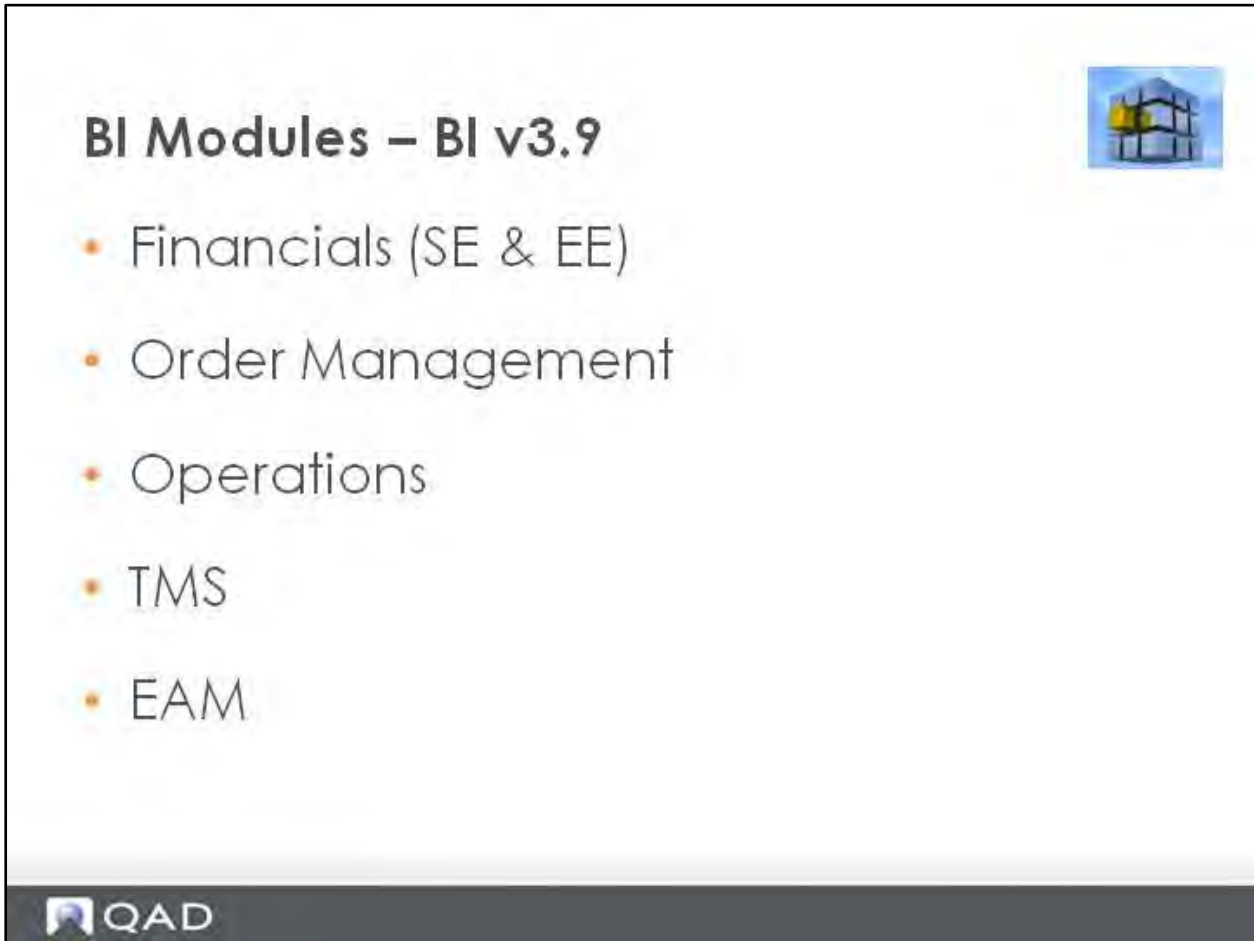
The QAD BI Portal is the front end web based portal GUI for Business Users to access the data in the data warehouse and see it in useful ways such as reports, charts, graphs and spreadsheets.

BI Architecture – Technical Perspective




The data flows from the ERP into the data warehouse via ODBC connections between SQL Server and Progress.

BI Modules – BI v3.9



BI Modules – BI v3.9

- Financials (SE & EE)
- Order Management
- Operations
- TMS
- EAM



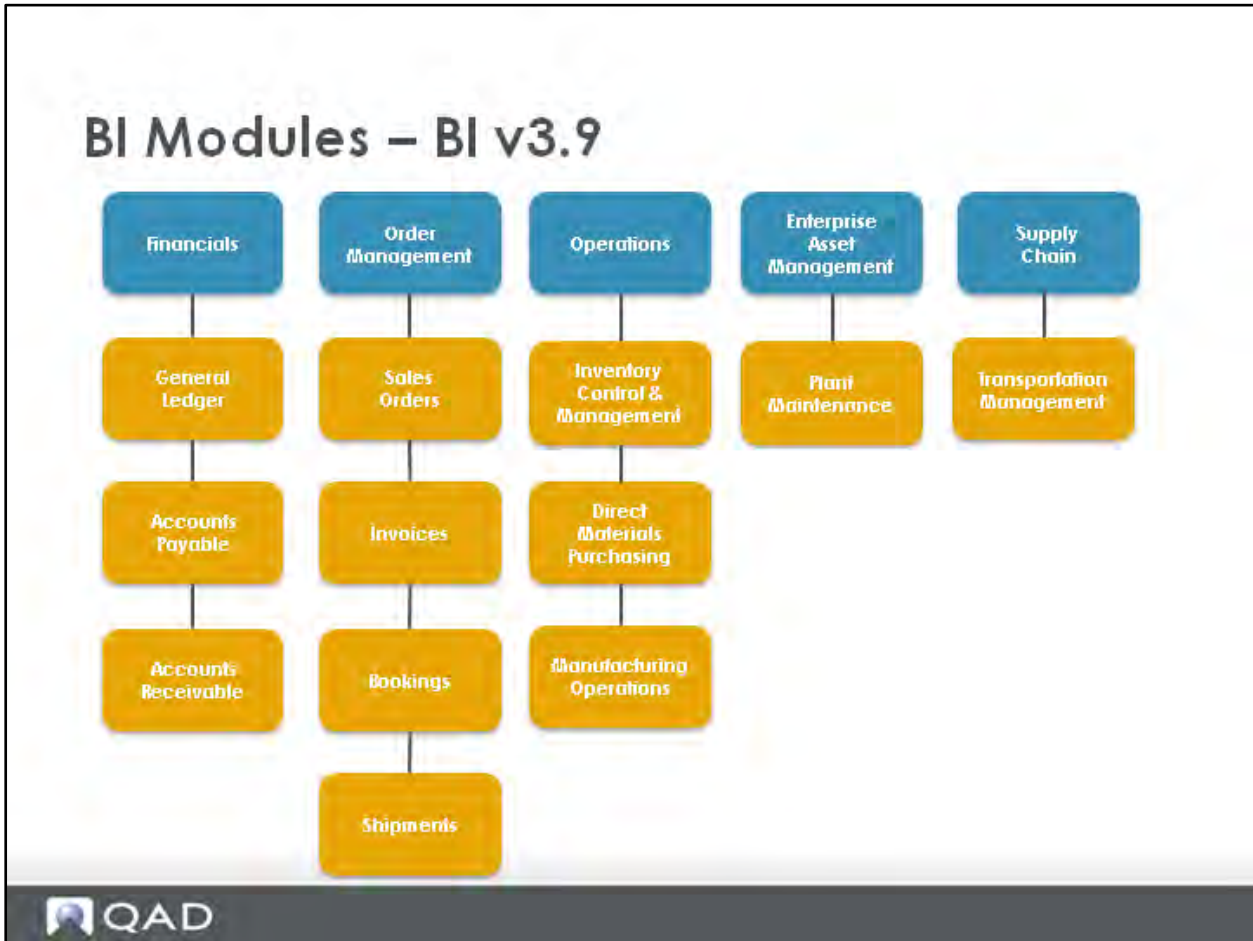
Key Concept:

The BI solution provides out-of-the box ERP and EAM “aware” modules. This streamlines access to current and historical QAD ERP data.

Current modules are:

- Financials (SE & EE)
- Order Management
- Operations
- TMS
- EAM

BI Modules – BI v3.9



Key Concept:

The BI solution provides out-of-the box QAD application “aware” modules. This streamlines access to current and historical QAD source data.

Current modules are as of the BI3.7 release are:

- Financials (SE & EE)
- Order Management
- Operations
- Enterprise Asset Management
- Supply Chain

Most modules have 1-4 submodules in the BI v3.7 release. Refer to the most recent Release Qbit for updated module information.

Review Question #1

Review Question #1

GL would be considered a component of which module?

- Financials
- Order Management
- Operations
- All of the above



Review Question #2

Review Question #2

The data warehouse is stored in which database?

- ERP Progress database
- BI Portal
- SQL Server database
- DWD Scheduler



Review Question #3

Review Question #3

By the end of this course what do you expect to be able to do?

- Install and use QAD's BI solution which includes:
 - SQL Server database
 - QAD BI Data Warehouse Designer (DWD)
 - QAD BI Modules
 - QAD BI Portal
 - Build cubes
 - Troubleshoot basic issues



In Review – BI Overview

In Review – BI Overview

- Explanation of BI
- BI structure
- BI's role in business
- QAD's BI plans



Installing the QAD BI Solution

Installing the QAD BI Solution



Overview - Installing the QAD BI Solution

Overview - Installing the QAD BI Solution

- Requirements
- Pre-install questions
- Validating data warehouse server
- Setting up a portal user
- Using the QAD BI Installer to install the ETL tool – Administrator & Data Warehouse Designer (DWD), and install the BI modules
- Parameter setup
- Languages install
- Historical load run



Installation - Getting Started

Installation - Getting Started

Customer pre-install requirements

- **64-bit Windows environment** (for installer)
- **Microsoft SQL Server (2008 or higher)** – We have discontinued support for 2005 for new releases
- **32-bit Progress OpenEdge for Windows** – Apparently it is not feasible to have both the 64-bit and the 32-bit ODBC drivers on the same server.
- **Large hard drive** – Database should be on a drive large enough to store a lot of data.



See sizing guidelines.

Keep in mind a table like tr_hist may have its data spread across a perm table and many multiple fact tables.

So one table may be represented across 10 or more fact tables.

It's not uncommon to see a data warehouse that is 10 times as large as the ERP it's being sourced from.

Pre-Install Customer Checks

Pre-Install Customer Checks

Sample parameters to review with Customer

- › CORP_CALENDAR_DOMAIN
- › CORP_CALENDAR_SOURCE
- › CORP_CURRENCY_CODE
- › DEFAULT_ACCOUNT
- › DEFAULT_COST_CENTER
- › DEFAULT_DOMAIN
- › DEFAULT_ENTITY
- › DEFAULT_SUB_ACCOUNT
- › INITIAL_JOB_SETUP_CONNECTION_0x_TYPE
- › INITIAL_JOB_SETUP_CONNECTION_0x_VERSION

See Reference Tables in Installation Guide



The customer will need to clarify the values for these parameters for us to be able finish the installation.

- **Corporate Calendar Information** hinges on the source and domain. Ask the customer which source and domain they want their corporate calendar driven from. If they have multiple source and multiple domains, they are picking the master calendar by choosing whichever source and domain they choose.
- **Corporate Currency Code** is the currency that the customer wants to see consolidated results across all of their data sources into.
- **Default Accounting Parameters** are values assigned for unknown values, if there isn't available information to match to. Someone in accounting at the customer site needs to confirm that these values don't conflict with existing used codes for other accounting values.
- **INITIAL_JOB_SETUP_CONNECTION_0x_TYPE** is used to determine if the source of the data is an SE or EE installation.
- **INITIAL_JOB_SETUP_CONNECTION_0x_VERSION** is used to specify the EE release number (ex: 2013 for 2013.1).

Login to Training Environment

Login to Training Environment

- **USER:** admin
- **PASSWORD:** qad

NOTE: Verify that the BI Installer files are mounted on the D:\ drive.



42

Prior to launching the training environment, it's critical to also attach the BI installer ISO image as part of the setup of that training environment. This may be already configured that way when the class is reserved.

Login to Training Environment

Login to Training Environment

- Start EE & SE Progress Databases
 - Double-click on C:\DBs\startdb_EE.bat
 - Double-click on C:\DBs\startdb_SE.bat



Confirm Database Settings

Confirm Database Settings

- Confirm that the **Microsoft SQL Server Management Studio (SSMS) Data Warehouse Database** points to the correct directory
- Set up a Portal user under Security
- Check tempdb files and change if necessary



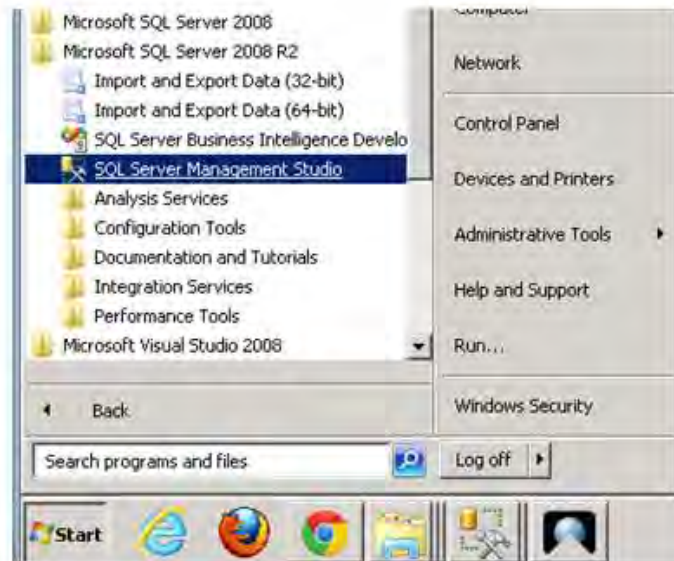
For the storage settings set up a directory in the E:\ drive.

E:\mssql\data

Confirming the Database Points to the Correct Directory

Confirming the Database Points to the Correct Directory

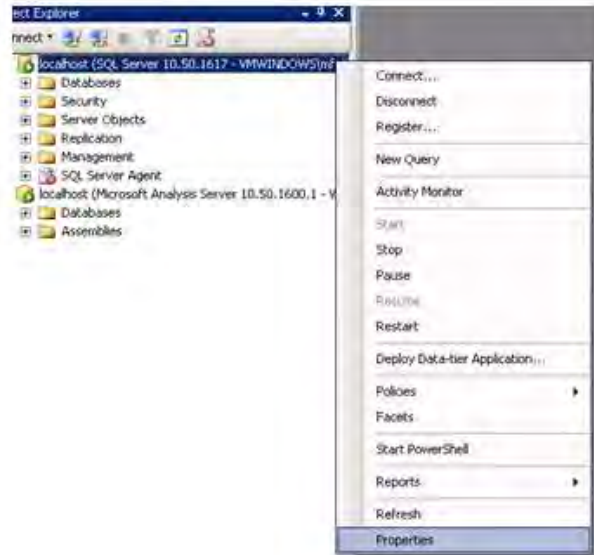
1. Launch SSMS.



Confirming the Database Points to the Correct Directory

Confirming the Database Points to the Correct Directory

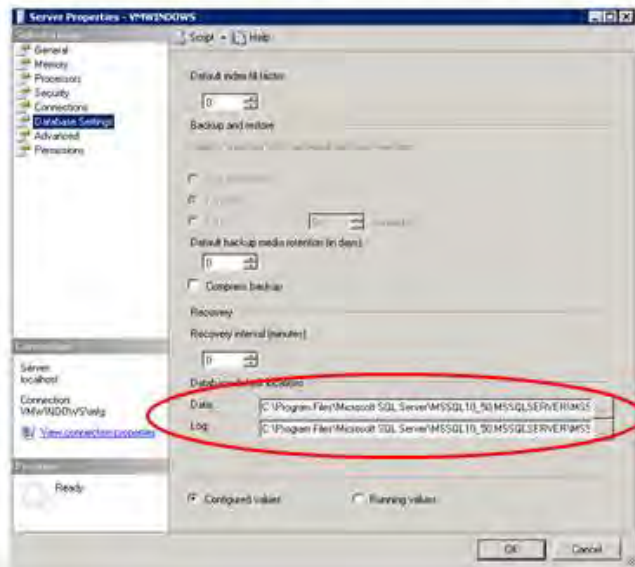
2. Right click on the server name and pick Properties.



Confirming the Database Points to the Correct Directory

Confirming the Database Points to the Correct Directory

- Choose Database Settings, and confirm that *Data* and *Log* directories are where they should be. Redirect if they are not.



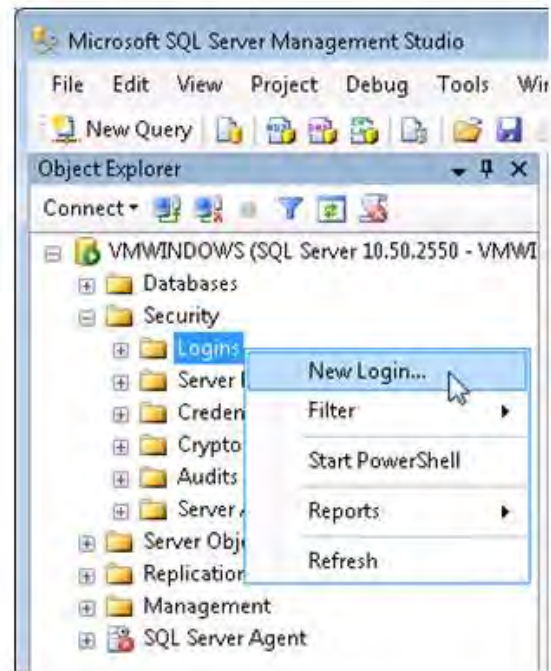
Usually, even if SQL Server is installed on the C drive, there is a large E (or some other labeled) drive available where the actual database and logs reside.

If for some reason the directories are pointing at the wrong server, remap those locations, otherwise you will quickly run out of room.

Define a Portal User

Define a Portal User

1. In SSMS, go to Security / Logins directory, right-click and choose *New Login...*

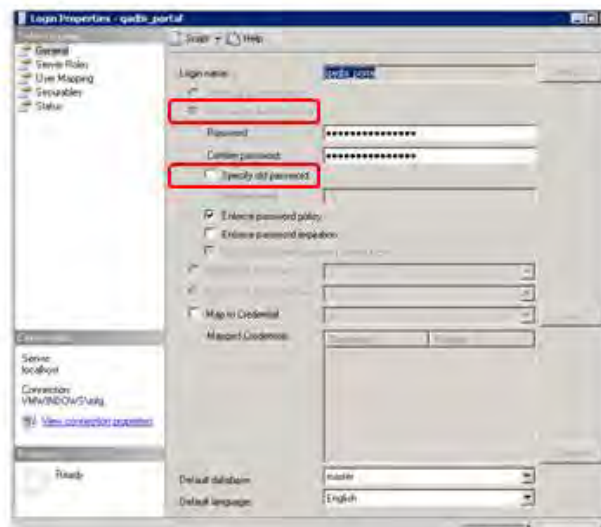


Define a Portal User

Define a Portal User

2. Enter a new *Login name* (qadbi_portal) and *Password* (qadbi_portal).

Make sure **SQL Server Authentication** is selected and **Specify old password** is not selected.



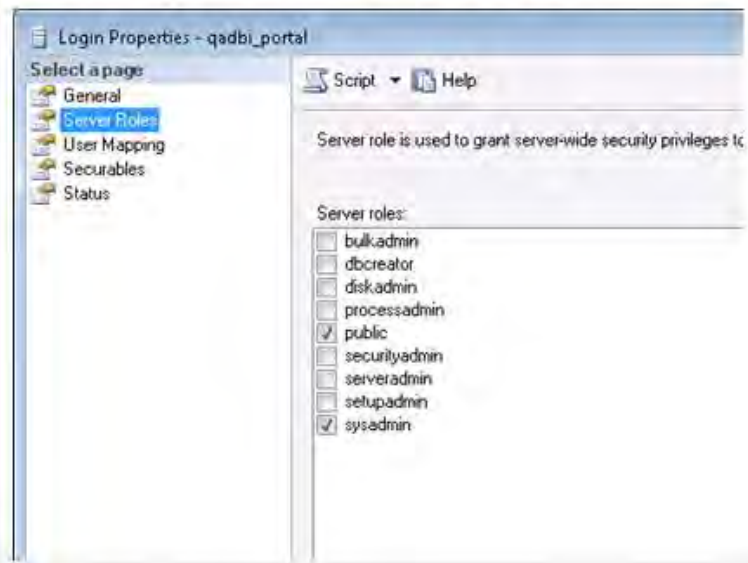
I usually also uncheck Enforce password expiration.



Define a Portal User

Define a Portal User

3. Go to the *Server Roles* tab and make sure that both *public* and *sysadmin* boxes are checked.







I usually also uncheck Enforce password expiration.

Install BI via Installer Program

Install BI via Installer Program

1. From the directory where the QAD BI Installer files were downloaded, double-click to launch the primary QAD BI Installer. (ex: qadbi-3.9.0.3.exe)

| Name | Date modified | Type | Size |
|--|-------------------|-------------|------------|
|  qadbi-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 140,760 KB |
|  qadbi-common-3.9.0.3.exe | 11/5/2013 4:54 PM | Application | 305,925 KB |
|  qadbi-financials-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 33,776 KB |
|  qadbi-order-management-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 33,068 KB |



Install BI via Installer Program

Install BI via Installer Program

2. The application will start extraction. If an *Open File – Security Warning* window appears, click *Run*.



Install BI via Installer Program

Install BI via Installer Program

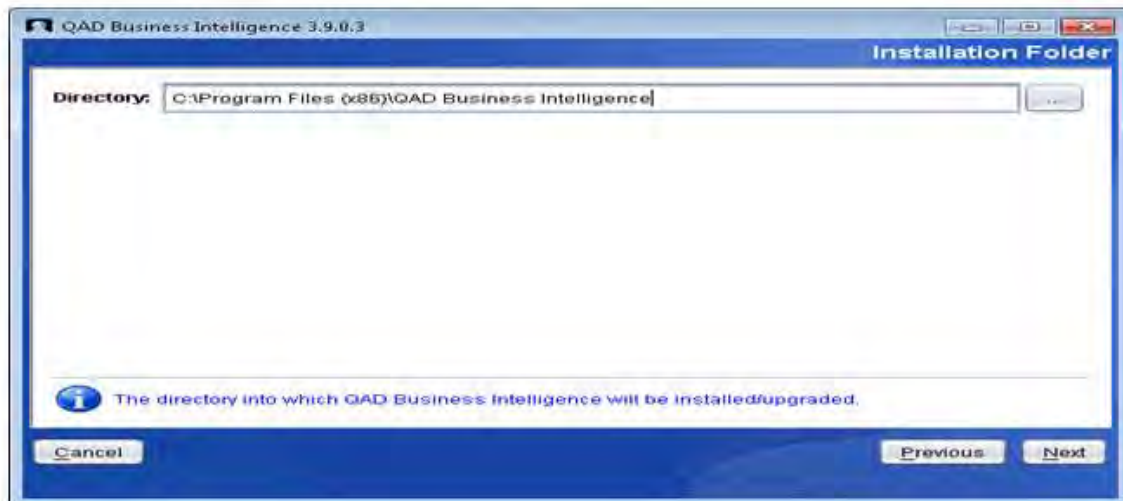
3. Click *Next*.



Install BI via Installer Program

Install BI via Installer Program

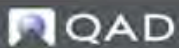
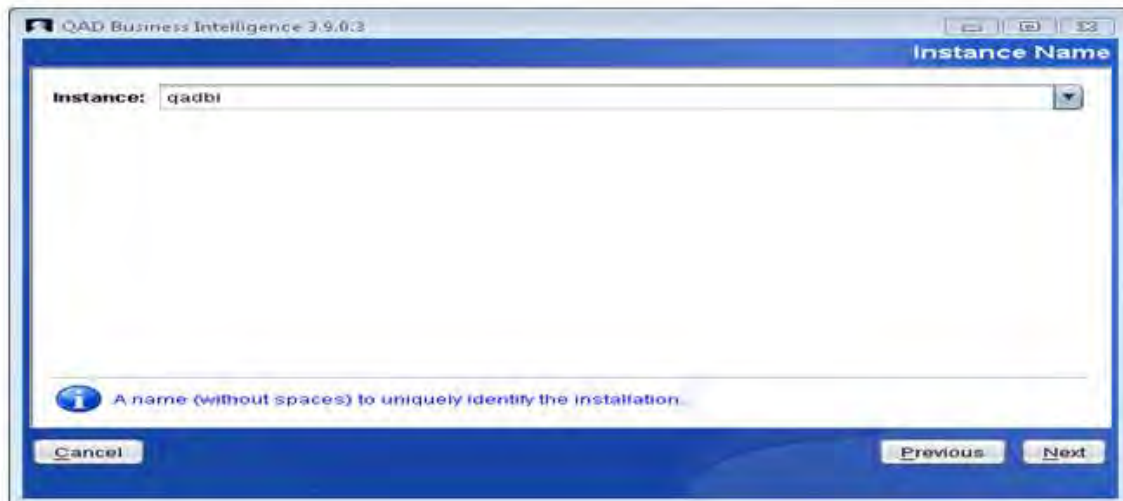
4. Set the directory or use the default.
Click *Next*.



Install BI via Installer Program

Install BI via Installer Program

5. Pick the database name or use the default.
Click *Next*.



Install BI via Installer Program

Install BI via Installer Program

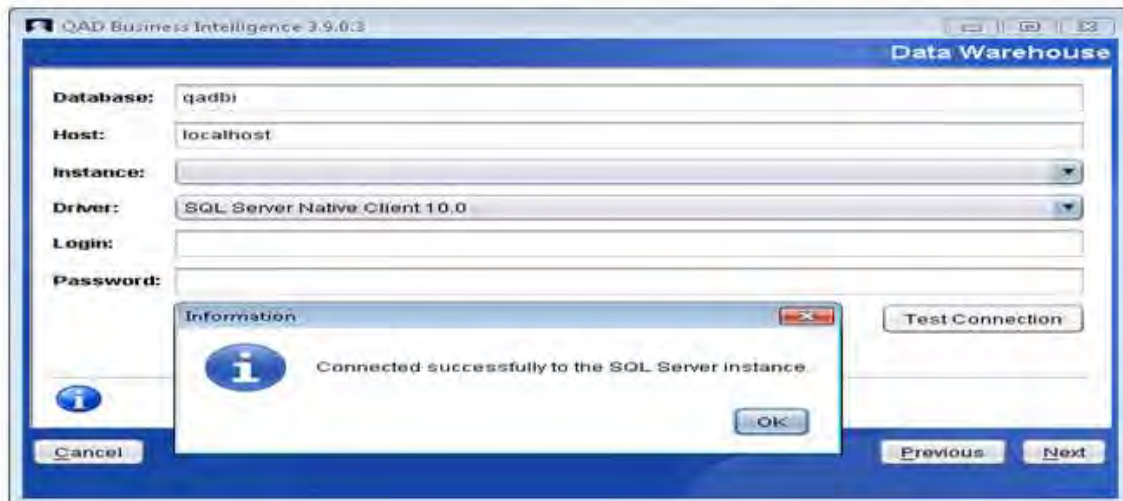
- To install everything, select *Complete*.
Click *Next*.



Install BI via Installer Program

Install BI via Installer Program

7. Click *Test Connection* to validate database connection. Click *OK*. Click *Next*.



Progress database(s) ODBC Pre-Setup

Progress database(s) ODBC Pre-Setup

You need to know the following information about the customer's Progress database(s).

Example:

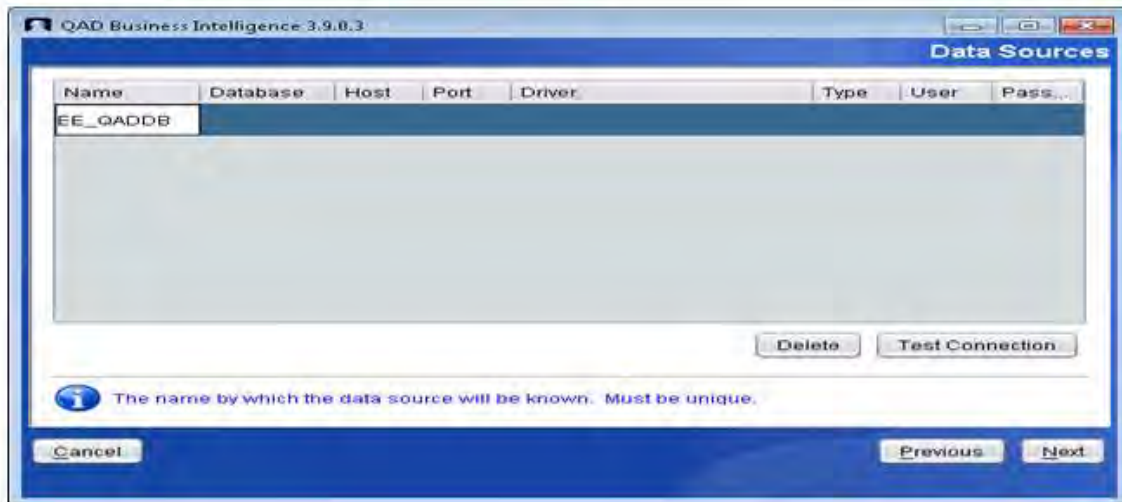
| Required Information for Progress Database(s) ODBC Pre-Setup | EE | SE |
|--|-------------------|-------------------|
| Preferred Connection Name | EE_QADDB | SE_QADDB |
| Database Name | qaddb | qaddb |
| Host Name | localhost | localhost |
| Port Number | 8811 | 8801 |
| Driver | Progress OE 10.2B | Progress OE 10.2B |
| Type | EE | SE |
| User ID | demo | demo |
| Password | | |

Install BI via Installer Program

Install BI via Installer Program

10. Define ODBC connections to the Progress databases.

- Name (a distinct name)



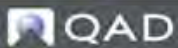
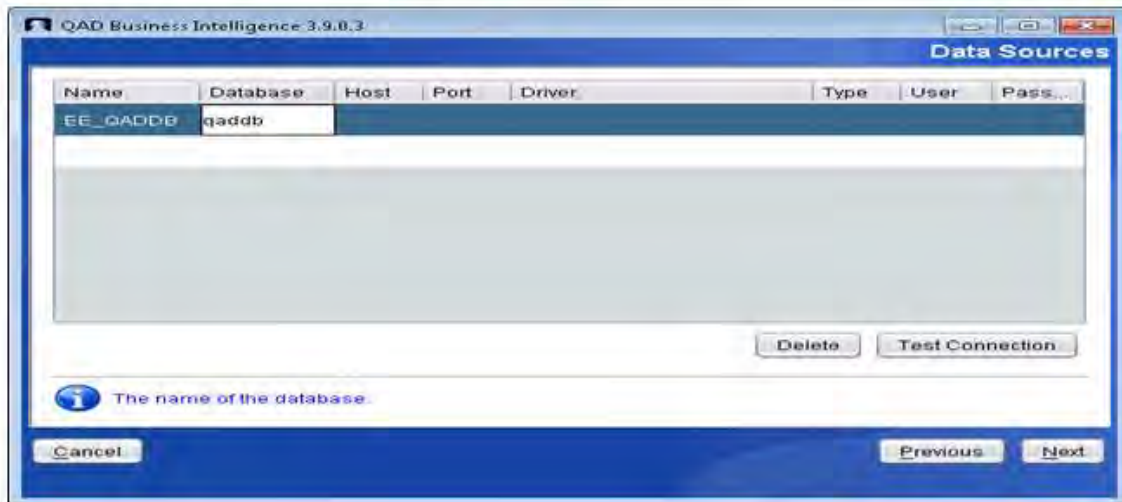
Name: Name of the ODBC connection (Note: If you are needing to set up multiple data warehouses that will source the same Progress databases, include a defining name extension for each ODBC Name such as SEDemoDev, SEDemoTest, SEDemoProd). The installer won't allow you to use the same ODBC connection NAME when multiple data warehouses are set up on the same server, even though the source may be exactly the same.

Install BI via Installer Program

Install BI via Installer Program

11. Define ODBC connections to the Progress databases.

- Source Database Name

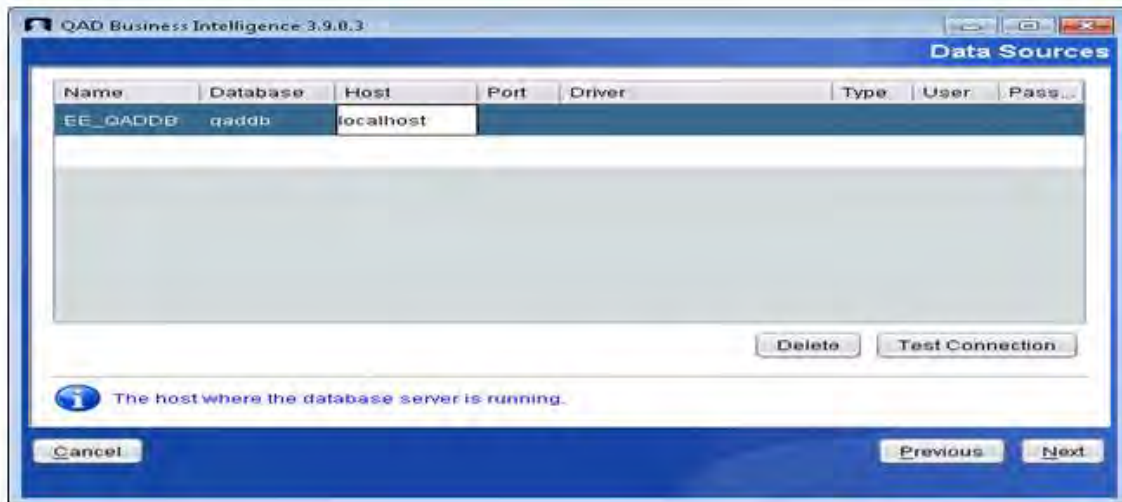


Install BI via Installer Program

Install BI via Installer Program

12. Define ODBC connections to Progress databases.

- Host Name

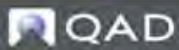
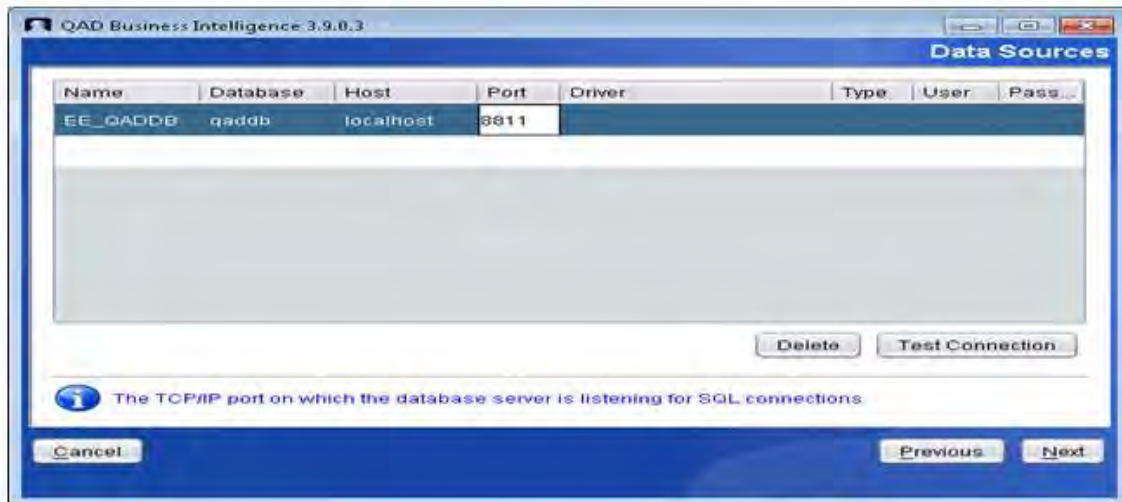


Install BI via Installer Program

Install BI via Installer Program

13. Define ODBC connections to Progress databases.

- Port Number

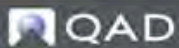
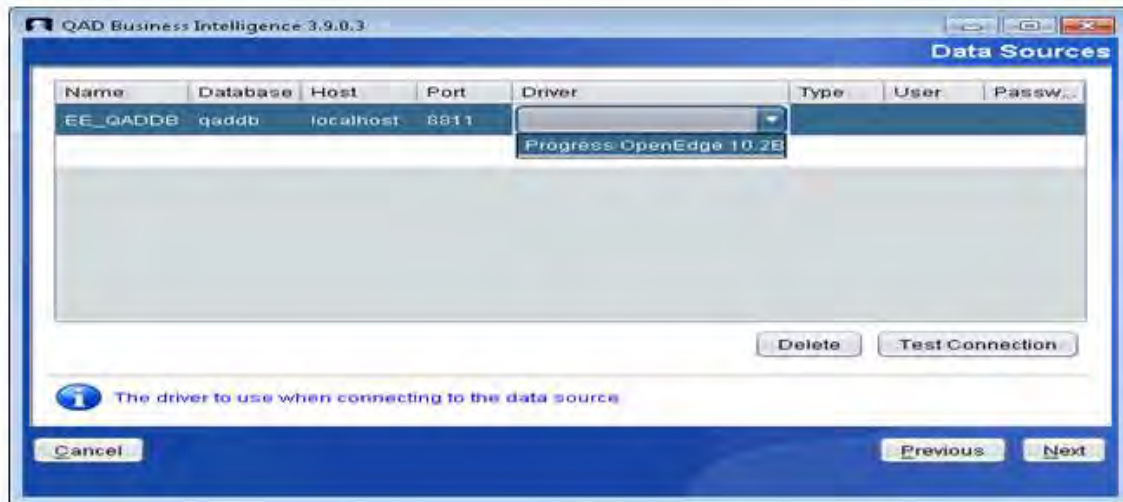


Install BI via Installer Program

Install BI via Installer Program

14. Define ODBC connections to Progress databases.

- ODBC Driver

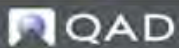
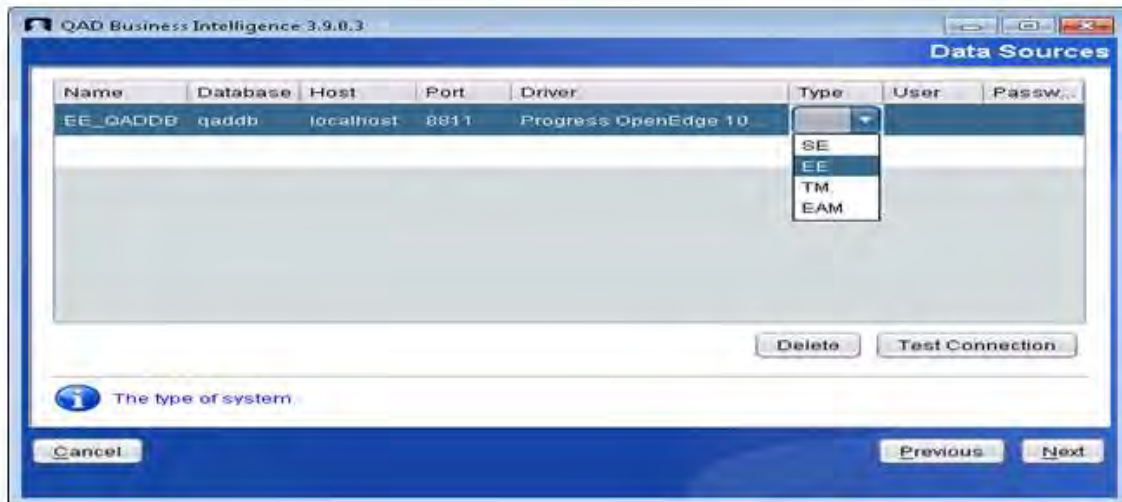


Install BI via Installer Program

Install BI via Installer Program

15. Define ODBC connections to Progress databases.

- Type

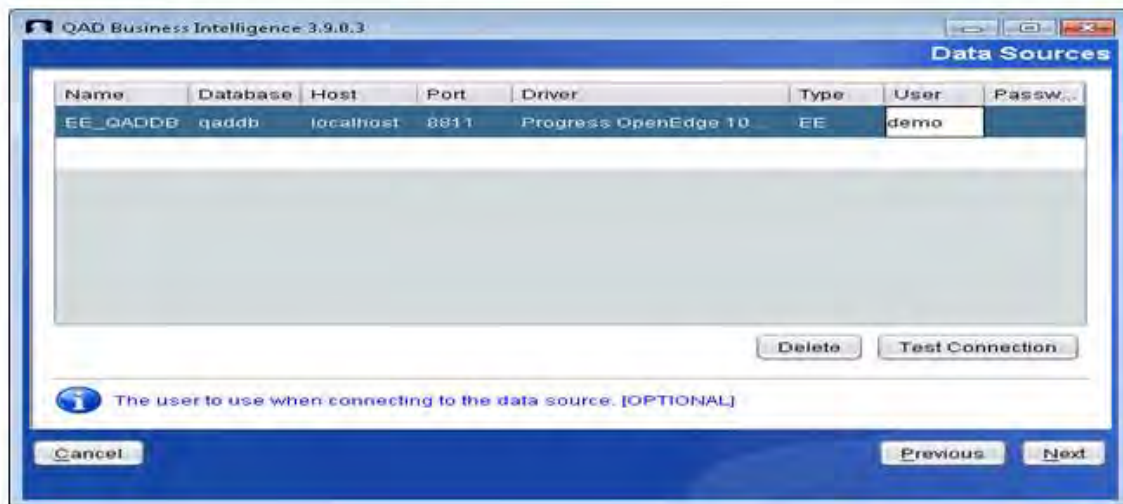


Install BI via Installer Program

Install BI via Installer Program

16. Define ODBC connections to Progress databases.

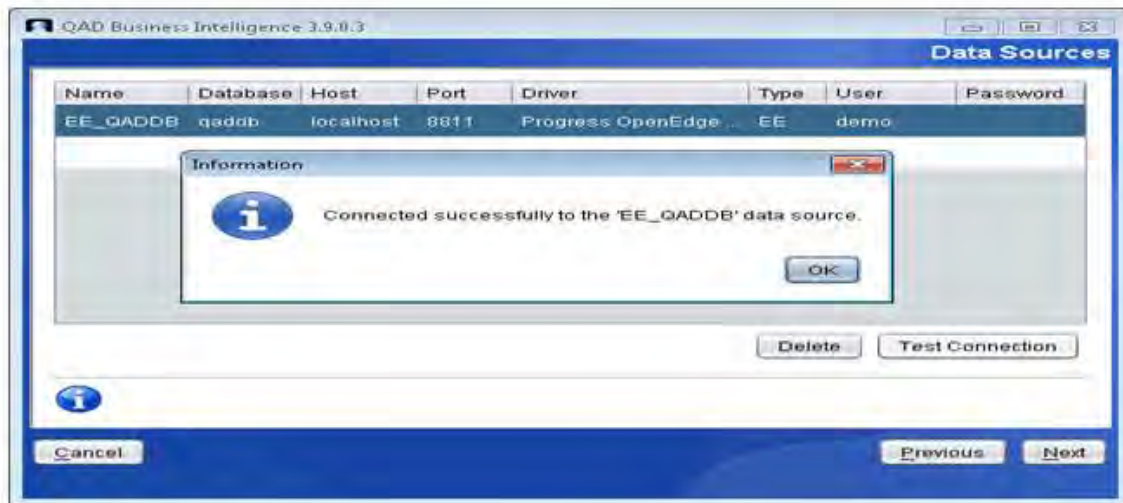
- User & Password



Install BI via Installer Program

Install BI via Installer Program

17. Click *Test Connection*. Click *OK* then *Next*.

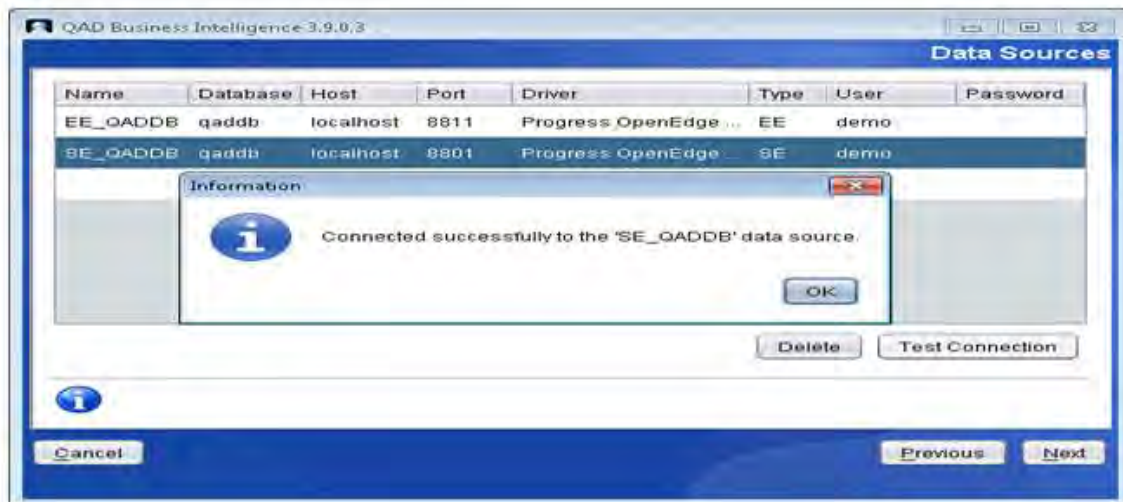


If test connection fails, confirm that the Progress database is running and entered values for database, host, port, user & password are correct.

Install BI via Installer Program

Install BI via Installer Program

18. Repeat for every Progress data source that you need to add.



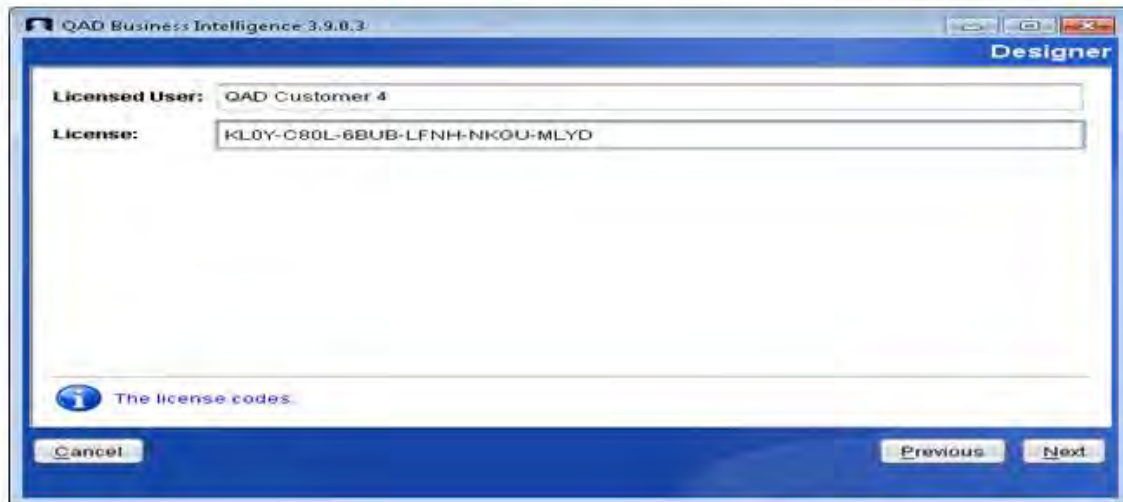
Extra non-Progress data sources can be added later via the DWD Admin tool.

Install BI via Installer Program

Install BI via Installer Program

19. Enter the DWD License Name and Number.

- **Licensed User:** QAD Customer 4
- **License:** KL0Y-C80L-6BUB-LFNH-NKGU-MLYD



The screenshot shows a window titled "QAD Business Intelligence 3.9.0.3" with a "Designer" tab. The window contains two text input fields. The first field is labeled "Licensed User:" and contains the text "QAD Customer 4". The second field is labeled "License:" and contains the text "KL0Y-C80L-6BUB-LFNH-NKGU-MLYD". At the bottom left of the window, there is an information icon and the text "The license codes:". At the bottom right, there are three buttons: "Cancel", "Previous", and "Next".

The license name number for a specific installation should be provided as part of the BI purchase. For the demo, we just use the values above.

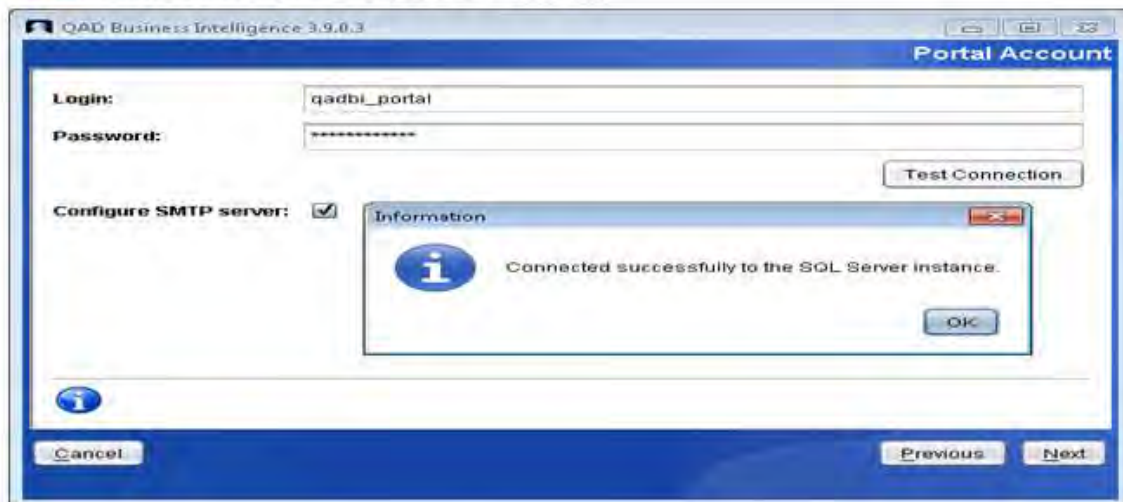
Install BI via Installer Program

Install BI via Installer Program

20. Enter the Portal *Login* and *Password*.

21. Check “*Configure SMTP server:*”

22. Click *Test Connection*.



This is for the SQL Server login and setup earlier.

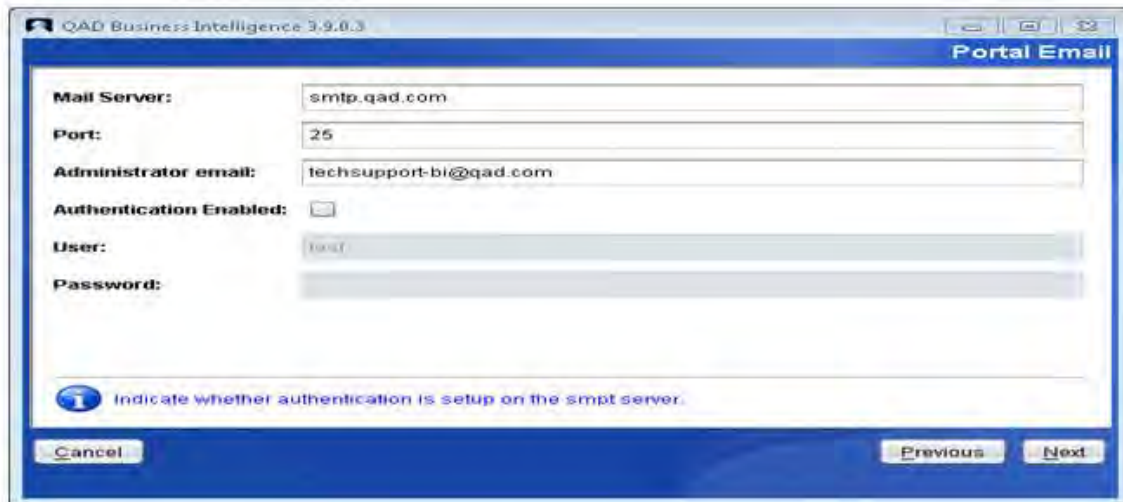
You can validate that the connection works.

Be sure to check SMTP Server configuration box. This is to ensure emails can be sent from the Portal.

Install BI via Installer Program

Install BI via Installer Program

23. Enter SMTP Server information. Click *Next*.



The screenshot shows a Windows-style dialog box titled "QAD Business Intelligence 3-9.0.3" with a "Portal Email" header. The dialog contains several input fields and a checkbox:

- Mail Server:** smtp.qad.com
- Port:** 25
- Administrator email:** techsupport-bi@qad.com
- Authentication Enabled:**
- User:** test
- Password:** (empty field)

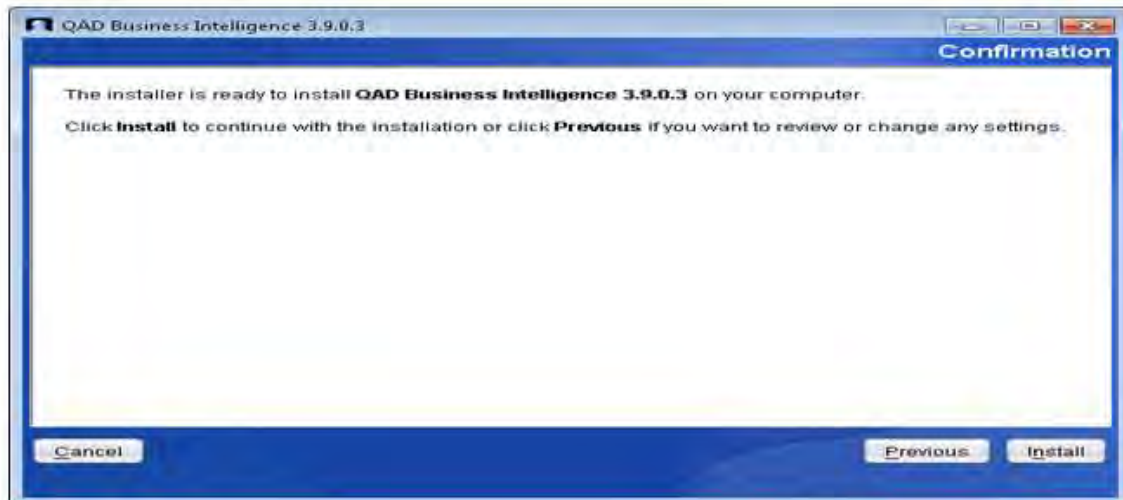
At the bottom, there is a blue information icon with the text: "Indicate whether authentication is setup on the smtp server." Below this are three buttons: "Cancel", "Previous", and "Next".

For the training environment, we don't use email authentication, but at some customer sites, it's necessary. Leave Authentication Enabled checkbox unchecked for this training.

Install BI via Installer Program

Install BI via Installer Program

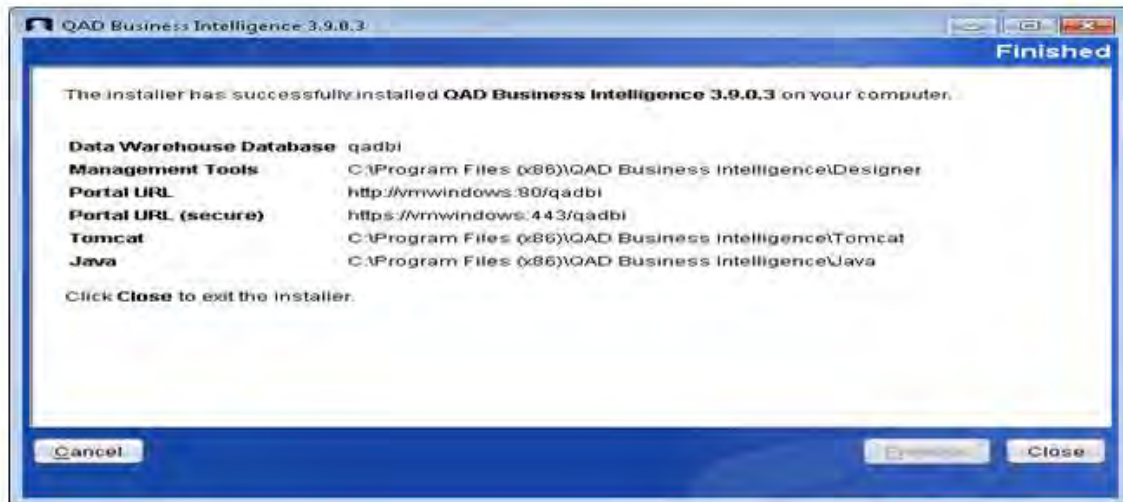
24. Click *Install*.



Install BI via Installer Program

Install BI via Installer Program

25. Once the installation is complete a successful installation message appears. Click *Close*.







Note the portal URLs.

Install BI Modules via Installer Program

Install BI Modules via Installer Program

1. Install the Common module next. Double-click the common exe file.
(ex: qadbi-common-3.9.0.3.exe)

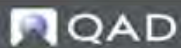
| Name | Date modified | Type | Size |
|--|-------------------|-------------|------------|
|  qadbi-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 140,760 KB |
|  qadbi-common-3.9.0.3.exe | 11/5/2013 4:54 PM | Application | 305,925 KB |
|  qadbi-financials-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 33,776 KB |
|  qadbi-order-management-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 33,068 KB |



Install BI Modules via Installer Program

Install BI Modules via Installer Program

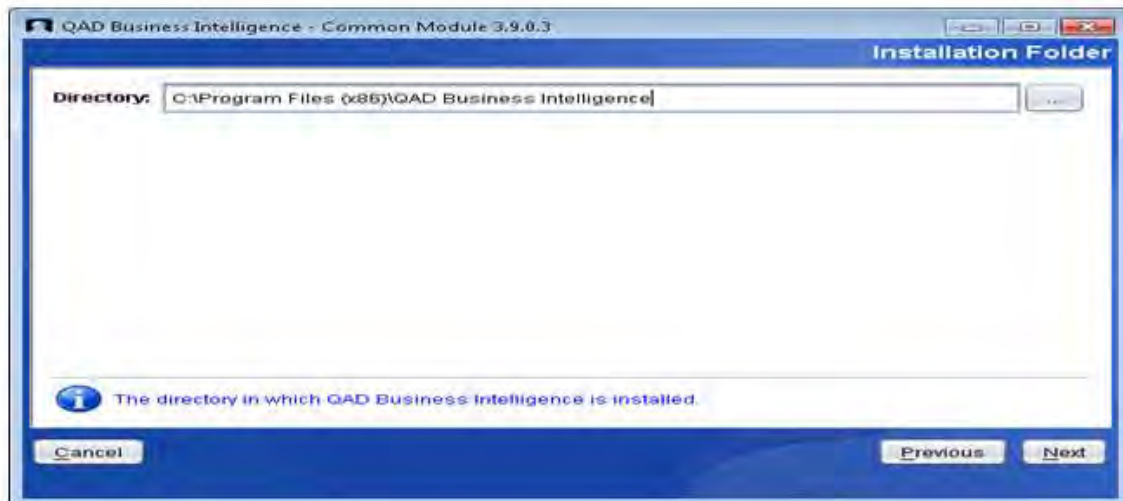
2. Click *Next*.



Install BI Modules via Installer Program

Install BI Modules via Installer Program

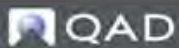
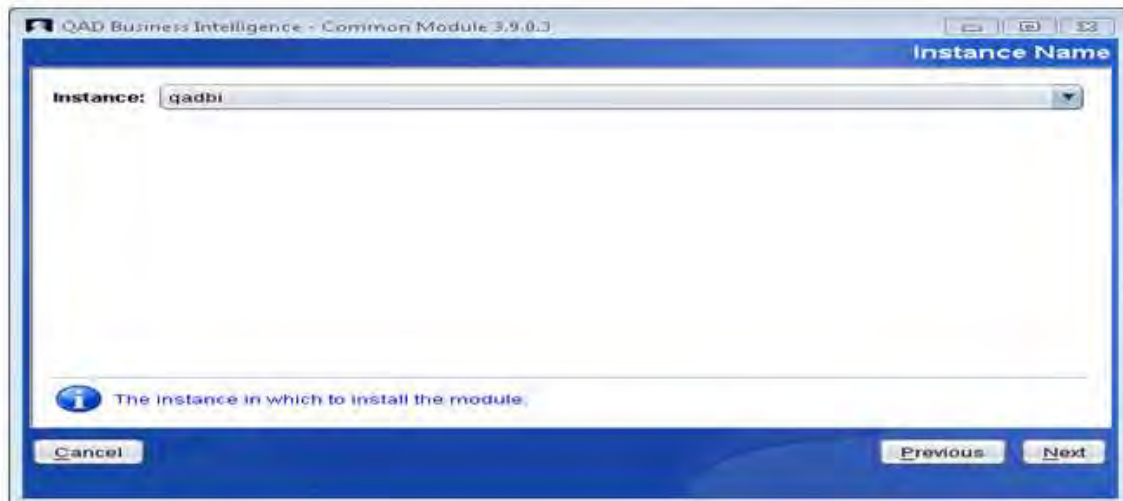
3. Choose the appropriate directory or accept the default. Click *Next*.



Install BI Modules via Installer Program

Install BI Modules via Installer Program

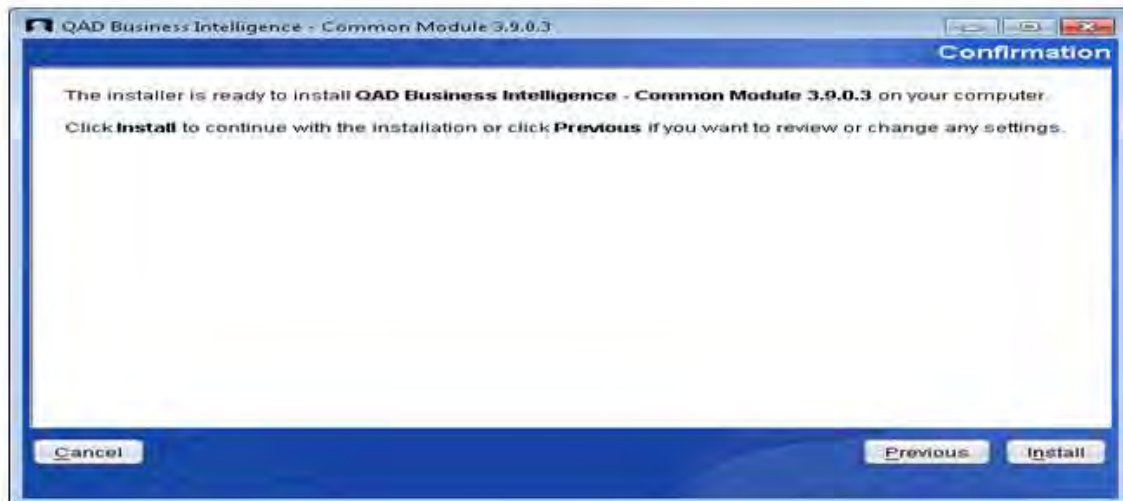
4. Choose the database instance. Click *Next*.



Install BI Modules via Installer Program

Install BI Modules via Installer Program

5. Click *Install*.



Install BI Modules via Installer Program

Install BI Modules via Installer Program

6. The installation will start.
- **Important Note:** It takes quite a while to load. If the screen says *Not Responding*, do not close it. Just wait until it finishes loading the Common module components.

| Output | |
|------------------------------|--------|
| Object | Status |
| fact_ee_gl_balance_idx_17 | New |
| fact_ee_gl_balance_idx_18 | New |
| fact_ee_gl_balance_idx_19 | New |
| fact_ee_gl_balance_idx_20 | New |
| fact_ee_gl_balance_idx_21 | New |
| fact_ee_gl_balance_idx_22 | New |
| fact_ee_gl_balance_idx_23 | New |
| fact_ee_gl_balance_idx_24 | New |
| fact_ee_gl_balance_idx_25 | New |
| fact_ee_gl_balance_idx_26 | New |
| fact_ee_gl_balance_idx_27 | New |
| fact_ee_gl_balance_idx_28 | New |
| fact_ee_gl_balance_idx_A | New |
| fact_ee_ar_customer_in_idx_1 | New |



Install BI Modules via Installer Program

Install BI Modules via Installer Program

7. Click *Close*.







Install BI Modules via Installer Program

Install BI Modules via Installer Program

- Repeat the same steps to install additional modules.

(ex: qadbi-financials-3.9.0.3.exe then
qadbi-order-management-3.9.0.3.exe)

| Name | Date modified | Type | Size |
|--|-------------------|-------------|------------|
|  qadbi-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 140,760 KB |
|  qadbi-common-3.9.0.3.exe | 11/5/2013 4:54 PM | Application | 305,925 KB |
|  qadbi-financials-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 33,776 KB |
|  qadbi-order-management-3.9.0.3.exe | 11/5/2013 4:55 PM | Application | 33,068 KB |



BI Designer – Set Parameters

BI Designer – Set Parameters

1. Open DWD located in Start → QAD Business Intelligence → Business Intelligence Designer.



While you wait for the modules to finish loading, work on the parameters.

Double click to launch the program.

BI Designer – Set Parameters

BI Designer – Set Parameters

2. Login to DWD. Enter your DWD User Name.

- **Note:** Since Windows Authentication is being used, no database username or password is necessary.



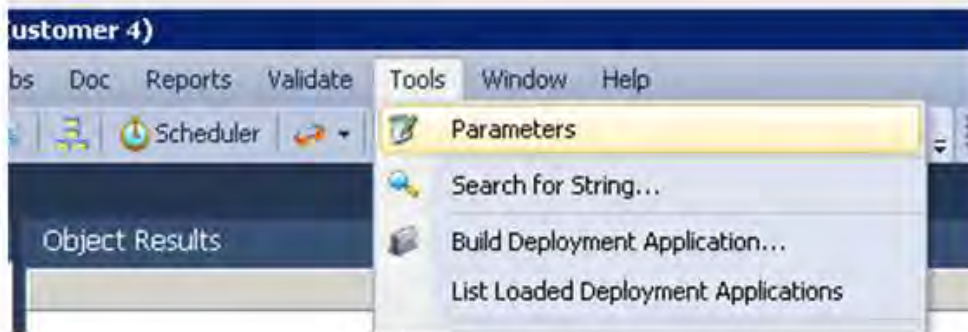
The screenshot shows a 'Repository Login' dialog box for 'QAD BI Data Warehouse Designer'. The dialog has a blue header with the text 'QAD BI Data Warehouse Designer' on the left and 'Repository Login' on the right. Below the header, it says 'Version 8.5 by QAD, Inc. Copyright (c) 2012 Licensed to QAD Customer 4'. The main area contains three input fields: 'Data Source' (a dropdown menu with 'qadbi' selected), 'Database Login ID' (an empty text box), and 'Password' (an empty text box). Below these is a section for 'METADATA REPOSITORY' with a 'DWD User Name' field containing 'Kurt Ehrler'. At the bottom right are three buttons: 'Help', 'Cancel', and 'Connect'.

Data Source is the name of the defined database instance.

BI Designer – Set Parameters

BI Designer – Set Parameters

3. Set parameters, from the customer feedback list and Installation Guide.
In DWD, go to Tools → Parameters to open the Parameters page.



Each customer may have different requirements that result in setting parameters differently for those customers.

BI Designer – Set Parameters

BI Designer – Set Parameters

Pre-install – Customer Feedback

- AP(AR)_EE_HISTORY_DATE_MIN = 20111101
- CORP_CALENDAR_DOMAIN = 10USA
- CORP_CALENDAR_SOURCE = EE_QADDB
- CORP_CURRENCY_CODE = USD
- DEFAULT_ACCOUNT = 00000
- DEFAULT_COST_CENTER = 0000
- DEFAULT_DOMAIN = 00000
- DEFAULT_ENTITY = 0000
- DEFAULT_PROJECT = 000000
- DEFAULT_SUB_ACCOUNT = 0000000
- DEFAULT_SOURCE = UNKNOWN



For EE financials, years back on AP/AR snapshot tables is dependent on the customer. Three years is typical, but snapshots take up a lot of space, so the customer may want to have less years. For the purpose of the class, we'll just do 2 years.

CORP values are the common denominator calendar and currency that will be used across all sources.

For the DEFAULT_ GL accounts values, check with the customer what values they want in the event that there are unknown or null values for the GL account fields. Sometimes they have already allocated our default values as valid numbers, so it's important to confirm these values are ok, or change them to whatever value they would prefer.

BI Designer – Set Parameters

BI Designer – Set Parameters

Initial Parameters

- INITIAL_JOB_SETUP_CONNECTION_01
- INITIAL_JOB_SETUP_CONNECTION_01_VERSION
 - Example: 2012 for EE 2012.1
 - Not currently used for SE data source (Optional)
- INITIAL_JOB_SETUP_CONNECTION_01_DAILY (HIST)_MODULES
 - COM, OM, OP, FIN, EAM
- INITIAL_JOB_SETUP_DATE
 - Set to today's date in YYYY-MM-DD format



For INITIAL_JOB_SETUP_CONNECTION there are the following:

- INITIAL_JOB_SETUP_CONNECTION_xx – The connection name.
- INITIAL_JOB_SETUP_CONNECTION_xx_DAILY_MODULES – The modules being installed that should be run daily. By default, all module options (except TM) are added by the Installer. If for some reason, a module is installed for one source, but shouldn't be used for another source, remove the module from it's related source. It's ok if modules are listed that are not installed, the startup program just ignores them.
- INITIAL_JOB_SETUP_CONNECTION_xx_HIST_MODULES – The modules being installed that should run for historical load. This part gives you an option in the future to install additional modules that hadn't been loaded before. For the initial installation, DAILY and HIST should be the same. By default, all module options (except TM) are added by the Installer. If for some reason, a module is installed for one source, but shouldn't be used for another source, remove the module from it's related source. It's ok if modules are listed that are not installed, the startup program just ignores them.
- INITIAL_JOB_SETUP_CONNECTION_xx_RUN – Defaults to N. Becomes Y after the installation is complete. Can be set back to N to rerun installation process if necessary.

- INITIAL_JOB_SETUP_CONNECTION_XX_TYPE – Pick whether this is an SE or EE data source. This is set via the installer.
- INITIAL_JOB_SETUP_CONNECTION_XX_VERSION – Pick which version of EE the code is for (2010,2011, 2012). It's only the EE source data that requires this parameter to be filled. For SE it doesn't matter what you put here.
- INITIAL_JOB_SETUP_DATE – Today's date in the format requested. This sets the current date for many of the parameters once the INITIAL_JOB_SETUP is run. It's also set by the Installer.

Running INITIAL_JOB_SETUP

Running INITIAL_JOB_SETUP

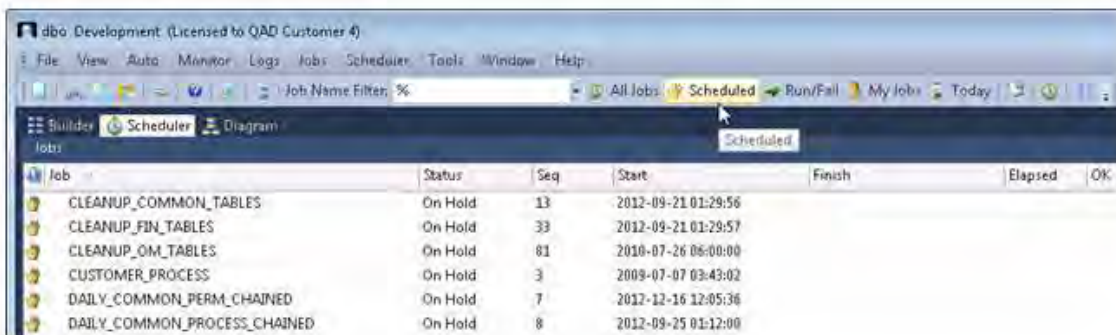
1. In the DWD, go to the *Scheduler* tab.



Running INITIAL_JOB_SETUP

Running INITIAL_JOB_SETUP

- Click on the *Scheduled* option, sort by *Job* name and find INITIAL_JOB_SETUP.



| Job | Status | Seq | Start | Finish | Elapsed | OK |
|------------------------------|---------|-----|---------------------|--------|---------|----|
| CLEANUP_COMMON_TABLES | On Hold | 13 | 2012-09-21 01:29:56 | | | |
| CLEANUP_FIN_TABLES | On Hold | 33 | 2012-09-21 01:29:57 | | | |
| CLEANUP_OM_TABLES | On Hold | 81 | 2010-07-26 08:00:00 | | | |
| CUSTOMER_PROCESS | On Hold | 3 | 2008-07-07 03:43:02 | | | |
| DAILY_COMMON_PERM_CHAINED | On Hold | 7 | 2012-12-16 12:05:36 | | | |
| DAILY_COMMON_PROCESS_CHAINED | On Hold | 8 | 2012-09-25 01:12:00 | | | |

The Scheduled tab is between All Jobs and Run Fail.

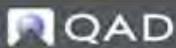
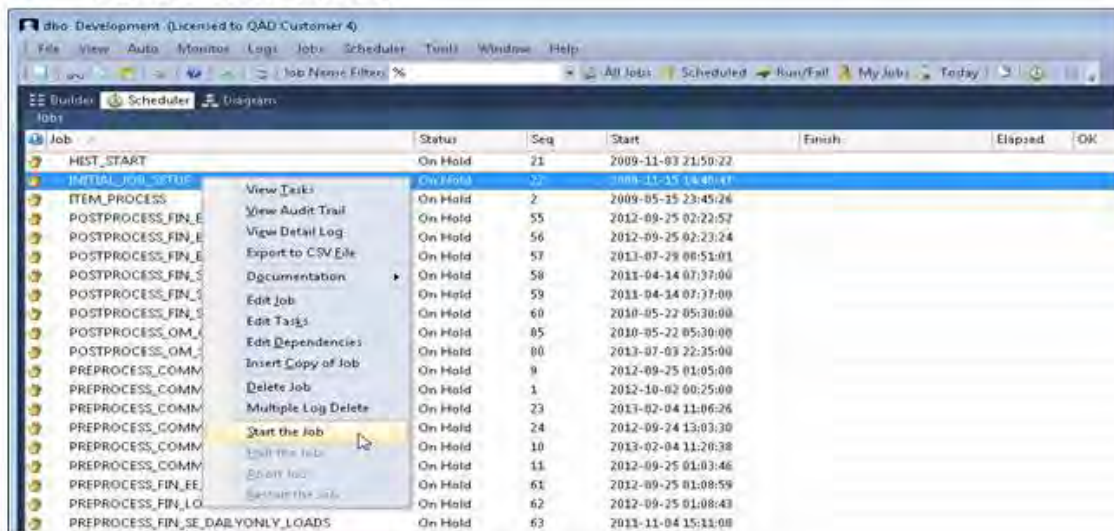
Then click on the tab called Job over the list of job names.

Then find the job INITIAL_JOB_SETUP.

Running INITIAL_JOB_SETUP

Running INITIAL_JOB_SETUP

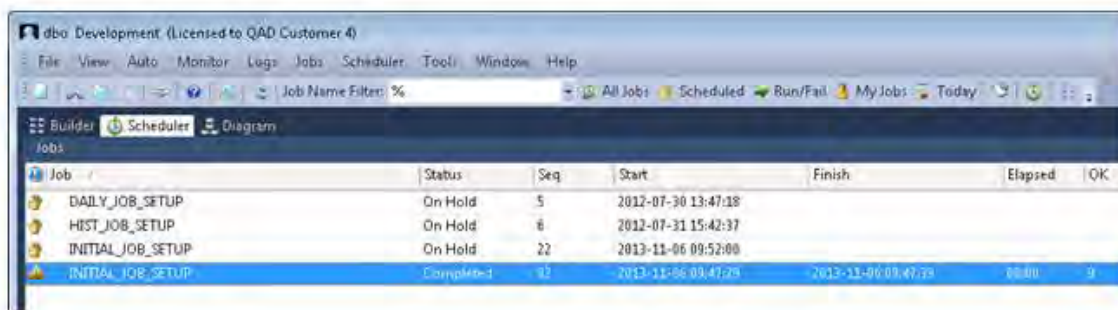
3. Right click INITIAL_JOB_SETUP, then click on *Start the Job*.



Running INITIAL_JOB_SETUP

Running INITIAL_JOB_SETUP

4. Click on the *Today* option to see that the job has run to completion.



The screenshot shows the QAD BI Scheduler interface. The 'Jobs' table is displayed with the following data:

| Job | Status | Seq | Start | Finish | Elapsed | OK |
|-------------------|----------|-----|---------------------|---------------------|---------|----|
| DAILY_JOB_SETUP | On Hold | 5 | 2012-07-30 13:47:18 | | | |
| HIST_JOB_SETUP | On Hold | 6 | 2012-07-31 15:42:37 | | | |
| INITIAL_JOB_SETUP | On Hold | 22 | 2013-11-06 09:52:00 | | | |
| INITIAL_JOB_SETUP | Complete | 1 | 2013-11-06 09:47:29 | 2013-11-06 09:47:29 | 00:00 | 9 |

This job runs very quickly.

Double-Check the Parameters

Double-Check the Parameters

Go back to Tools → Parameters.

| Parameter | Value | Comments |
|--------------------------------|-----------------------------------|---|
| CORP_CALENDAR_DOMAIN | 10USA | The Domain Code used for the corporate financial calendar |
| CORP_CALENDAR_SOURCE | EE_QADDB | The Source System Code used for the corporate financial calendar |
| CORP_CURRENCY_CODE | USD | Corporate currency code used for high level reporting |
| DAILY_DATE_ROLL_OVERRIDE | | YYYY-MM-DD Leave blank to keep dim_date_qad based on today's date |
| DAILY_HISTORY_DATE_SET_CURRENT | Y | Default Y indicates jobs will always set the SALES and PO HISTORY DATES |
| DAILY_LOAD_JOB_CURNO | 0 | Current Daily Load Job Number |
| DAILY_LOAD_JOB0001 | SET_CONNECTION_EE_QADDB | Inserted via WsParameterWrite |
| DAILY_LOAD_JOB0002 | PREPROCESS_COMMON_TRUNCATE | Inserted via WsParameterWrite |
| DAILY_LOAD_JOB0003 | PREPROCESS_OM_TRUNCATE | Inserted via WsParameterWrite |
| DAILY_LOAD_JOB0004 | PREPROCESS_FIN_TRUNCATE | Inserted via WsParameterWrite |
| DAILY_LOAD_JOB0005 | PREPROCESS_COMMON_LOADS | Inserted via WsParameterWrite |
| DAILY_LOAD_JOB0006 | PREPROCESS_COMMON_DAILYONLY_LOADS | Inserted via WsParameterWrite |
| DAILY_LOAD_JOB0007 | PREPROCESS_COMMON_EE2010UP_LOADS | Inserted via WsParameterWrite |

After the INITIAL_JOB_SETUP job runs, the DAILY_LOAD_JOB_ and HIST_LOAD_JOB_ lists will have a list of all the jobs that will run to populate all the various tables.

If this is not the case something went wrong with the INITIAL_JOB_SETUP.

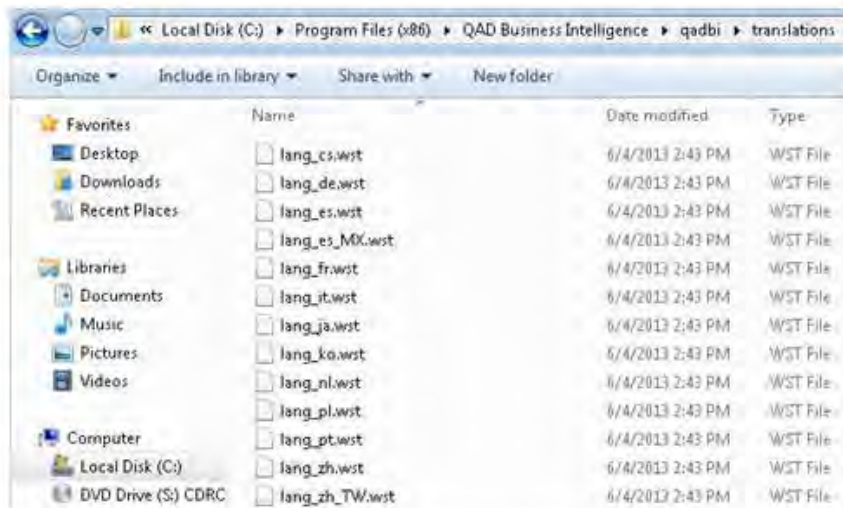
Also make sure that there are no date parameters with <> brackets around them. Easiest way to check this is to confirm that the AP_SE_HISTORY_PROCESS_DATE and INITIAL_JOB_SETUP_DATE have actual dates in them.

If there are any problems, correct them and run INITIAL_JOB_SETUP again.

Language Installations

Language Installations

1. Find the corresponding language file in the translations directory.



Look for this under the contents of the zipped file under C:\Program Files (x86)\QAD Business Intelligence\qadbi\translations

Language Installation – Pre DWD 6.2.2.0

Language Installation – Pre DWD 6.2.2.0

2. Add the two letter language codes and descriptions in the Tools → Options → Languages tab in the Designer.

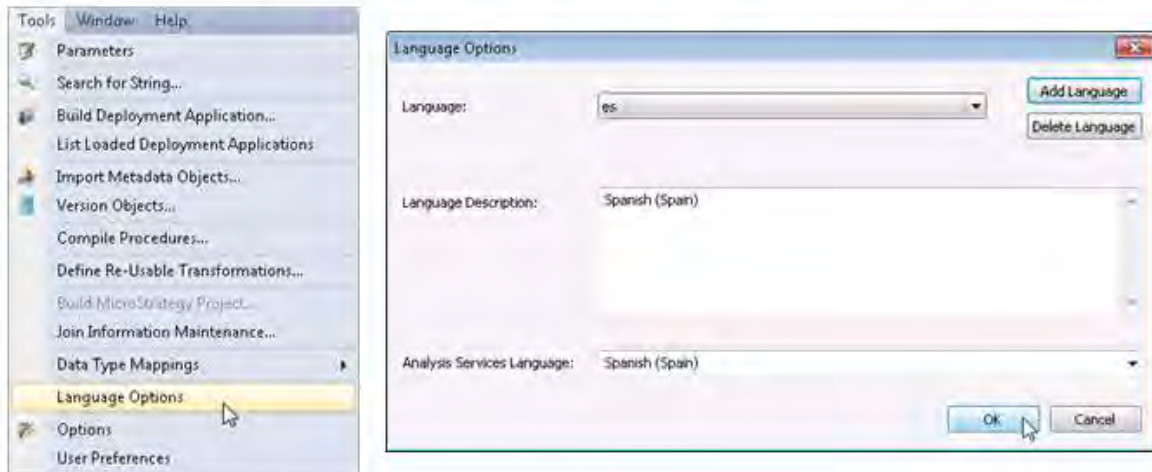


In QAD BI DWD (Data Warehouse Designer) tool, choose Tools/Options. Click on the Languages tab and add the codes for the language files you want to install. (E.g., 'es' for Spanish, 'fr' for French, 'zh' for simplified Chinese, 'zh_TW' for traditional Chinese. The complete set of files will be in the metadata package, under C:\Program Files (x86)\QAD Business Intelligence\qadbi\translations; the file names are in the format lang_ then the code then .wst - for example, lang_zh_TW.wst for traditional Chinese). The interface can be a little misleading here. It requires that you click the Add Language button, then type in a language or abbreviation in the Language drop down menu. What you type will help you find language options in the Analysis Services Languages list, but doesn't have to in any way match or limit what is in that list. What you type for the Language is simply used for reference purposes in later steps. If you type ES, the ISO abbreviation for Spanish, the Analysis Services Language drop down will display Estonian. You can just click on the drop down and scroll down to which Spanish option you want to select at that time.

Language Installation – DWD 6.2.2.0 and Up

Language Installation – DWD 6.2.2.0 and Up

3. Select the two letter language code in the Tools → Language Options. Click OK.



In QAD BI DWD (Data Warehouse Designer) tool, choose Tools -> Language Options. Click on the Add Language button to add the codes for the language files you want to install. (E.g., 'es' for Spanish, 'fr' for French, 'zh' for simplified Chinese, 'zh_TW' for traditional Chinese (or is it Taiwanese?). The complete set of files will be in the metadata package, under metadata/translations; the file names are in the format lang_ then the code then .wst - for example, lang_zh_TW.wst for traditional Chinese). The interface can be a little misleading here. It requires that you click the Add Language button, then type in a language or abbreviation in the Language drop down menu. What you type will help you find language options in the Analysis Services Languages list, but doesn't have to in any way match or limit what is in that list. What you type for the Language is simply used for reference purposes in later steps. If you type ES, the iso abbreviation for Spanish, the Analysis Services Language drop down will display Estonian. You can just click on the drop down and scroll down to which Spanish option you want to select at that time.

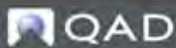
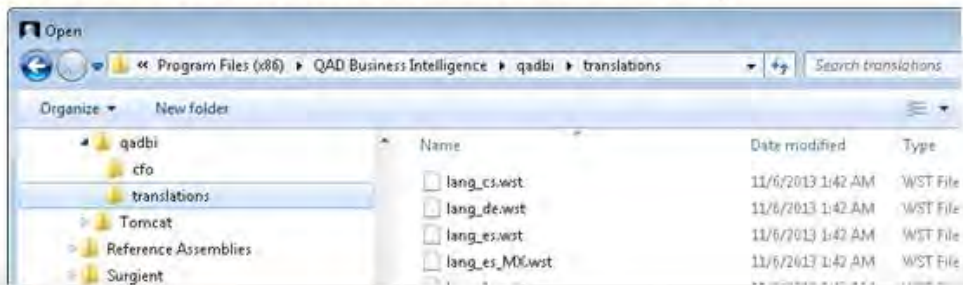
Language Installation

Language Installation

4. Launch QAD BI Administrator and select Languages → Change Languages Directory.



5. Select the translations directory



Language Installation

Language Installation

6. Right click on the languages you want to add and select *Install Language*. Make sure that the languages you pick match to the ones you setup in the languages option in the BI Designer Tool (DWD).

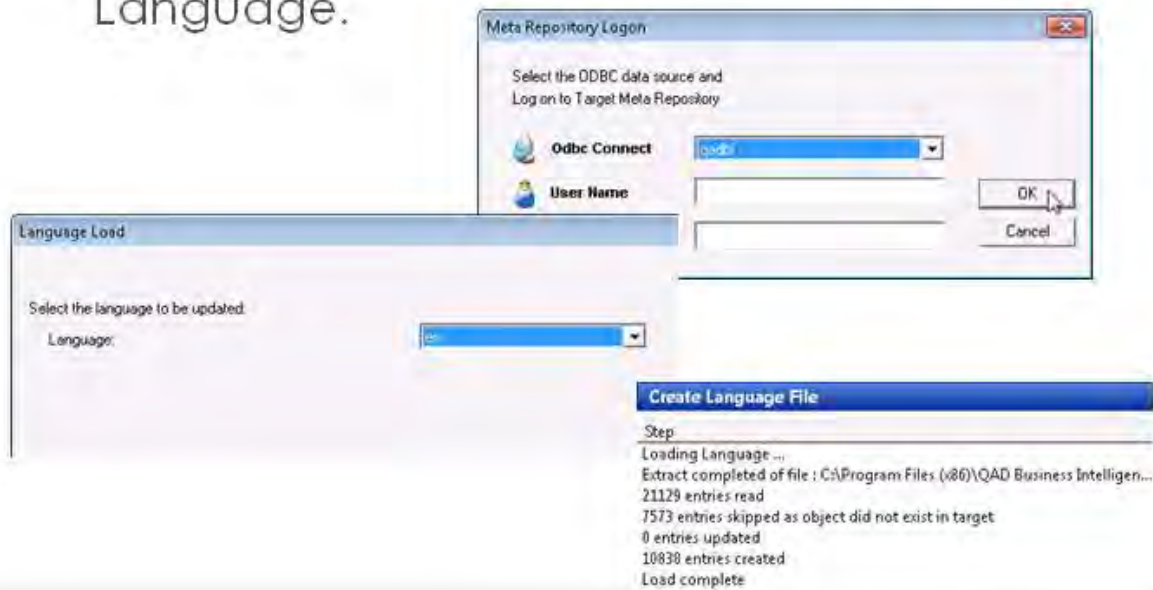
| Available Languages | | | | |
|---------------------|----------|---------------|------------|------------------------|
| File | Language | Skip Existing | Skip Empty | Source ODBC Connection |
| lang_cs.wst | Original | False | False | MASTER_39 |
| lang_de.wst | Original | False | False | MASTER_39 |
| lang_es.wst | Original | False | False | MASTER_39 |
| lang_... | Original | False | False | MASTER_39 |
| lang_... | Original | False | False | MASTER_39 |
| lang_... | Original | False | False | MASTER_39 |
| lang_ja.wst | Original | False | False | MASTER_39 |
| lang_ko.wst | Original | False | False | MASTER_39 |



Language Installation

Language Installation

7. Pick the ODBC Source then select the Language.



Language Installation

Language Installation

8. In DWD, confirm translations were installed.

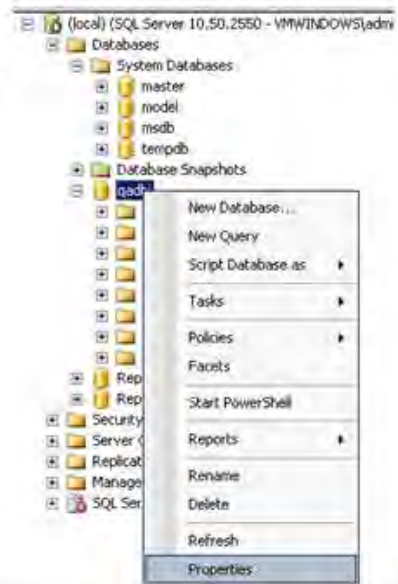


Double click on a fact table selected from the Browser tab. Pick the Language Mapping tab. Pick a language other than English. The Description should provide a translation below.

Database Size Validation Prior to Hist Load

Database Size Validation Prior to Hist Load

- Confirm database size and correct if necessary.
 - Go to SQL Server and right click on the data warehouse name and pick the *Properties* option.



Database Size Validation Prior to Hist Load

Database Size Validation Prior to Hist Load

- Confirm database size and correct if necessary. Keep in mind optimal settings should have the increase by a percentage such as 5% instead of by 1 MB. Also make sure that the directory of the database is the correct one. **Before:**

Database files:

| Logical Na... | File Ty... | Filegroup | Initial Size (MB) | Autogrowth | Path |
|---------------|------------|---------------|-------------------|----------------------------------|---------|
| qadbi | Rows... | PRIMARY | 34 | By 1 MB, unrestricted growth ... | C:\Prog |
| qadbi_ds | Rows... | DS | 2 | By 1 MB, unrestricted growth ... | C:\Prog |
| qadbi_is | Rows... | IS | 2 | By 1 MB, unrestricted growth ... | C:\Prog |
| qadbi_ms | Rows... | MS | 77 | By 1 MB, unrestricted growth ... | C:\Prog |
| qadbi_log | Log | Not Applic... | 5 | By 10 percent, restricted gro... | C:\Prog |

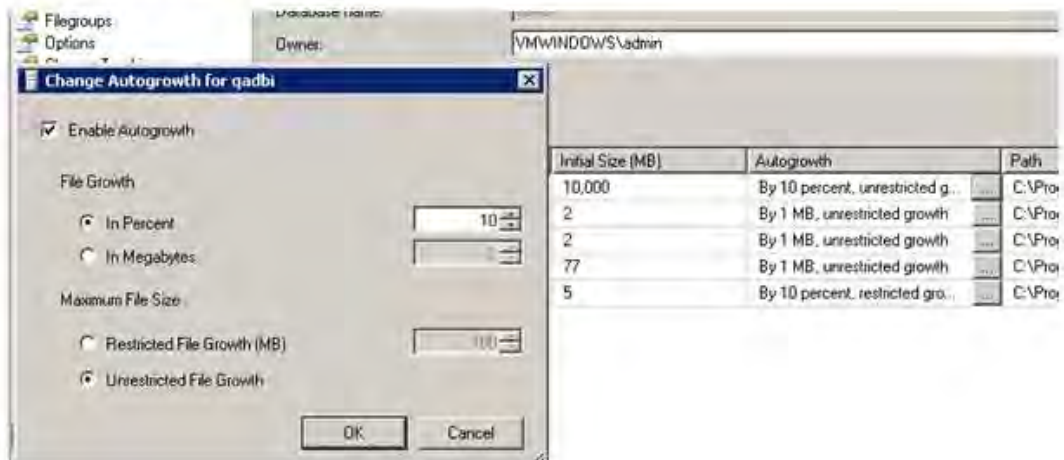


Refer to Sizing Guidelines in the QAD BI Installation Guide.

Database Size Validation Prior to Hist Load

Database Size Validation Prior to Hist Load

- Confirm database size and correct if necessary. **After:**

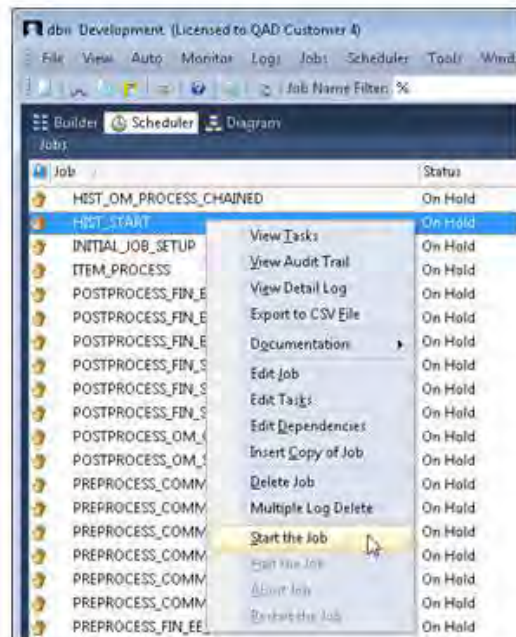


Here we've changed the initial database size to 10 GB (still relatively small for a data warehouse) and Autogrowth set to 10%.

Start the Historical Load

Start the Historical Load

- In DWD Scheduler, right-click on the job HIST_START and select *Start the Job*.
- Note:
 - HIST_LOAD_TR_HIST
 - HIST_LOAD_GLTR_HIST



Click the Run/Fail button to confirm that the job is running.

The Refresh button can be clicked to have the page keep refreshing to show the progress of the load.

Later in the run, explain what's going on with TR_HIST and GLTR_HIST repeatedly being displayed.

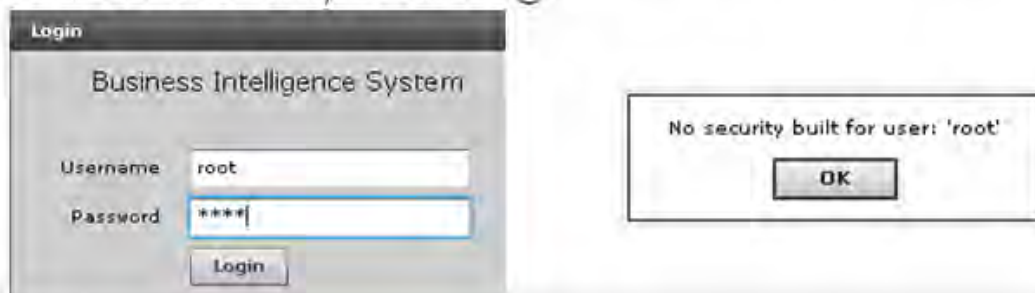
Also correlate the jobs that have run to the HIST_LOAD_JOB parameter list to explain job chaining.



BI Portal Web Application – Getting Started

BI Portal Web Application – Getting Started

1. In your web browser, go to `http://<server:port>/qadbi/Analytics.html`.
2. The initial login is root, with a password of root. You will be required to change the password immediately. For the purposes of the class, just change it back to root. Click *OK* on security warning.



The root user has no security, just click OK through that message.

<http://localhost/qadbi/Analytics.html>

BI Portal Web Application – Getting Started

BI Portal Web Application – Getting Started

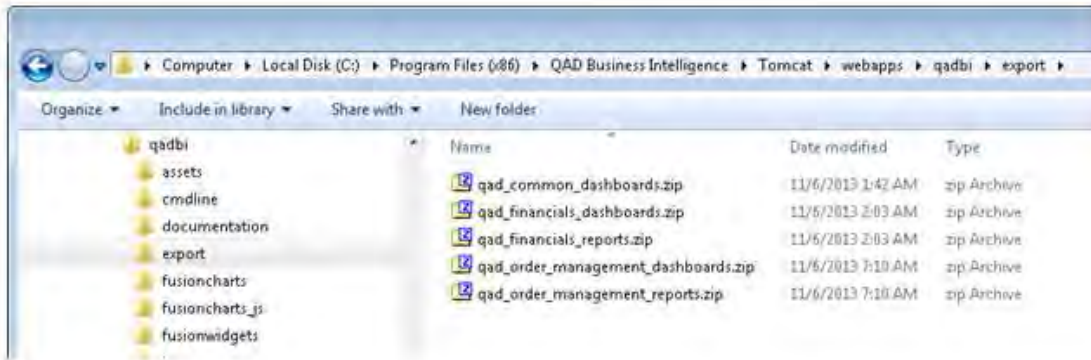
3. You need to build your model before you can do any analysis. Select Administration → Model Administration. Click on the *Rebuild Master* button.



Install Standard Dashboards and Reports

Install Standard Dashboards and Reports

1. Find the various portal .zip files in the export directory (like C:\Program Files (x86)\QAD Business Intelligence\Tomcat\webapps\qadbi\export).



Install Standard Dashboards and Reports

Install Standard Dashboards and Reports

2. In the Portal, go to Administration → Data Migration.
3. Click the *Import* tab, then click *Import*.

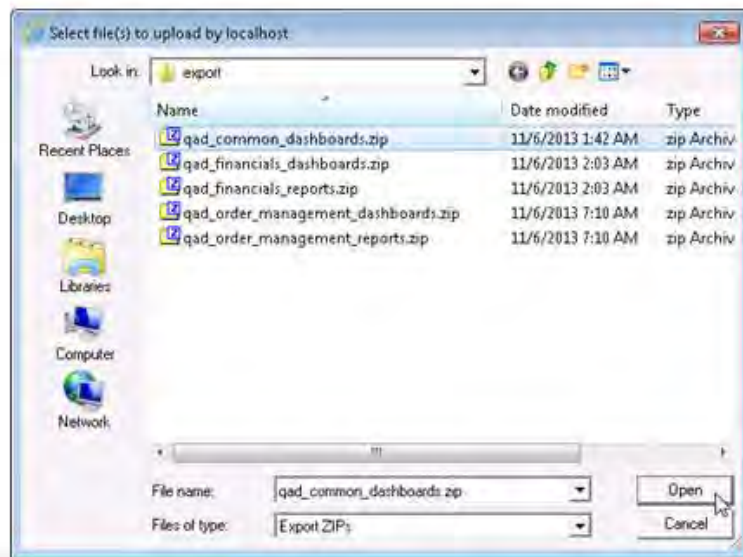


Load each module's reports or dashboards that are installed.

Install Standard Dashboards and Reports

Install Standard Dashboards and Reports

4. Select the .zip file.
5. Click *Open*.



Load each module's reports or dashboards that are installed.

Install Standard Dashboards and Reports

Install Standard Dashboards and Reports

6. Upon completion, click OK.
7. Repeat for each set of .zip files to be loaded.

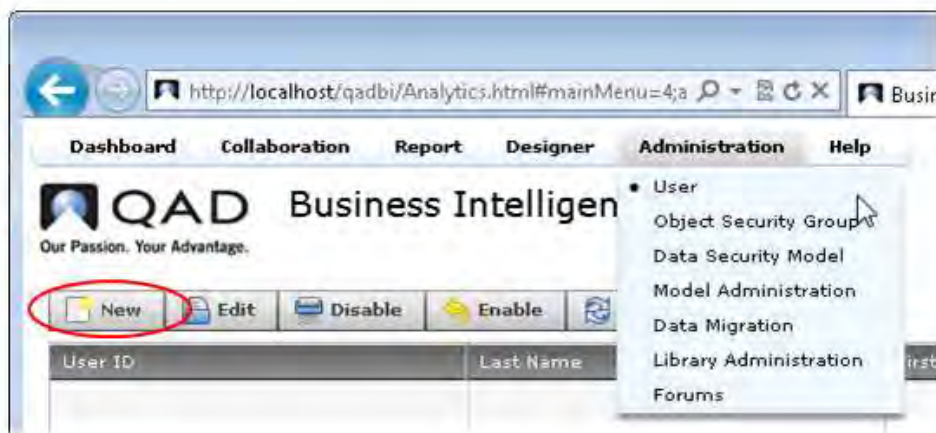


Load each module's reports or dashboards that are installed.

BI Portal – Create New Users

BI Portal – Create New Users

1. In the Portal, go to Administration → User.
2. Click *New* to create a user.



By default, root has no data access

BI Portal – Create New Users

BI Portal – Create New Users

3. Define *User ID*,
Common Name,
Email Address,
Application Role.
4. Check *Full Access*.
5. Click *Save*.

User Information

| | |
|------------------|-------------------------|
| User ID | mby |
| Password | |
| Confirm Password | |
| AD User | |
| AD Domain | |
| First Name | Michael |
| Last Name | Yap |
| Common Name | Michael Yap |
| Email address | mby@qad.com |
| Status | |
| Locale | English (United States) |
| Application Role | Administrator |

Object Security Groups

- Human Resources
- Sales Organization
- Management

Data Security Models

- Full Access

Save Cancel



BI Portal – Create New Users

BI Portal – Create New Users

6. Then go to Administration → Model Administration.
7. Click *Build Security*.
8. Logout as root.



BI Portal – Create New Users

BI Portal – Create New Users

9. Open your email and use the new User Id & Password log in.

From: techsupport-bi@qad.com
To: mby@qad.com

Welcome to QAD Business Intelligence.

A new account has been created for you:
User Id: mby
Password: mXr2g3

Click on the link below to begin: <http://localhost:80/qadbi/Analytics.html>

Regards
QAD Business Intelligence



Day 1: In Review

Day 1: In Review

- Overview of BI
- Ran the BI Installer
 - Setup data warehouse & installed DWD
 - Created ODBC connections to Progress ERP
- Set parameters & ran the INITIAL_JOB_SETUP
- Installed Language(s)
- Started the Historical Data Load
- Setup Portal
 - Dashboards & Reports
 - User(s)



Questions?

Questions?

- Customer Support is here to support you!
- Visit the QAD BI Forum at

<http://community.qad.com>



www.qad.com

© 2013 QAD Inc

CHAPTER 2

QAD BI Technical Level 1 Certification – Part 2

QAD BI v3 - Technical Level 1 Certification - Part 2

QAD Business Intelligence v3 - Technical Level 1 Certification Class - Part 2



Agenda - Day 2

Agenda - Day 2

- Day 1 Review
- Confirm historical data loads were successful.
- Discuss various failures & possibilities.
- Build and populate cubes.
- Connect ad hoc Excel to cubes.
- Review of the DWD and the Admin tool.

Day 1 Review

Day 1 Review

- Overview of BI
- Ran the BI Installer
 - Setup data warehouse & installed DWD
 - Created ODBC connections to Progress ERP
- Set parameters & ran the INITIAL_JOB_SETUP
- Installed Language(s)
- Started the Historical Data Load
- Setup Portal
 - Dashboards & Reports
 - User(s)

Confirm Historical Loads Have Run

Confirm Historical Loads Have Run

- Check for any long running jobs that are still running.
- Are there any failed jobs?
- Various steps to troubleshoot and restart jobs.

Review of Day 1 - Problems Encountered

Review of Day 1 - Problems Encountered

- ODBC connection issues
 - /F switch failures
- Restarting the hold job vs. the failed job
- Date parameters
- Source connection parameters
- Starting over if the historical job failed
- "Over-packed" columns
- Other issues

Review of Day 1 – Setting Connections and Scheduler

Review of Day 1 – Setting Connections and Scheduler

- SET_CONNECTION_<connection_name>
 - Example: SET_CONNECTION_EE_QADDB
- QAD_MASTER
 - Connection
 - Parameter
- setp_job_<job_name>
 - Example: setp_job_sales_process_start
- <JOB>_PROCESS_RUNNING
 - updated by setp_job_<job_name>
 - Example: SALES_PROCESS_RUNNING



setp_job_<job_name> sets a <JOB>_PROCESS_RUNNING parameter to Y, which will be set back to N at the end of the job. If another job starts that checks the parameter, it will fail as it believes there is already a running version of the job.

Review of Day 1 – Setting Connections and Scheduler

Review of Day 1 – Setting Connections and Scheduler

- If a Failed job is not restarted and the same On Hold job is instead started, the Failed job gets set to Failed – Aborted. The new job checks the parameter to see if the job is running. If the <JOB>_PROCESS_RUNNING parameter is still Y from the prior job, the new job will fail. The parameter will need to be manually reset for the job to continue.

Review of Day 1 – Setting Connections and Scheduler

Review of Day 1 – Setting Connections and Scheduler

- If load tables fail due to some issue with the Connection setup for a given source. If that Connection gets corrected, the load job could still fail because it uses QAD_MASTER connection to load. The corrections made in the fixed Connection also need to be applied to QAD_MASTER either manually or by running the proper setp_job_<job_name> job.

DWD Scheduler

DWD Scheduler



Scheduling the DAILY Job Run

Scheduling the DAILY Job Run

- Go to DWD Scheduler
- Right-click on DAILY_START job
- Select *Edit Job*
 - Set Frequency from Hold to Daily or Custom
 - Specify Start Date
 - Specify Start Time
 - Update Custom Settings (if Frequency = Custom)

Scheduler – Jobs Inside Jobs

Scheduler – Jobs Inside Jobs

- A job can run inside a job
- Can only run tasks consecutively
- Do not update the Start Date
- Sample jobs inside jobs
 - ITEM_PROCESS
 - ADDRESS_PROCESS
 - CUSTOMER_PROCESS
 - DAILY_xx_PERM_EXTRACT



11

A job can run inside a job, but there are some important things to know.

- Jobs that run as children of other jobs will only be able to have its Tasks run consecutively. This is why our jobs are set up to be run via the Job Parameter list.

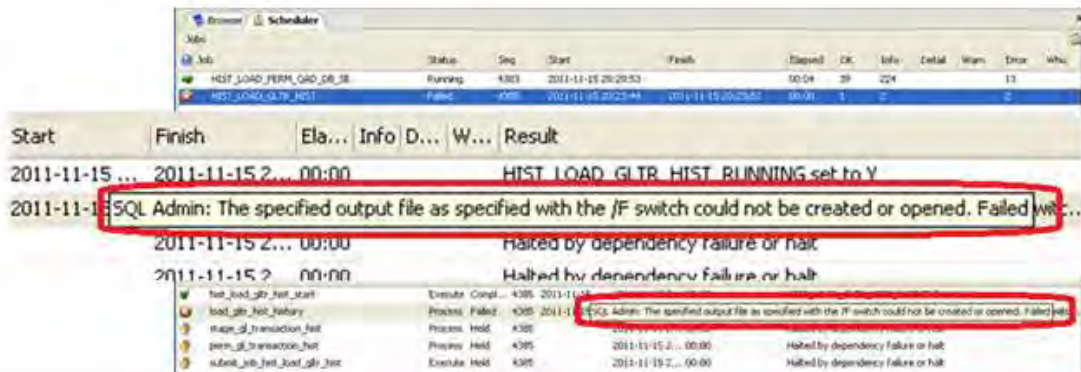
The following are examples of jobs that run inside jobs:

- ITEM_PROCESS
- ADDRESS_PROCESS
- CUSTOMER_PROCESS
- DAILY_xx_PERM_EXTRACT
- Jobs run from another job do not update their Start Date in the Scheduler. They will look like they haven't run for a long time.

Scheduler – /F Switch Failure

Scheduler – /F Switch Failure

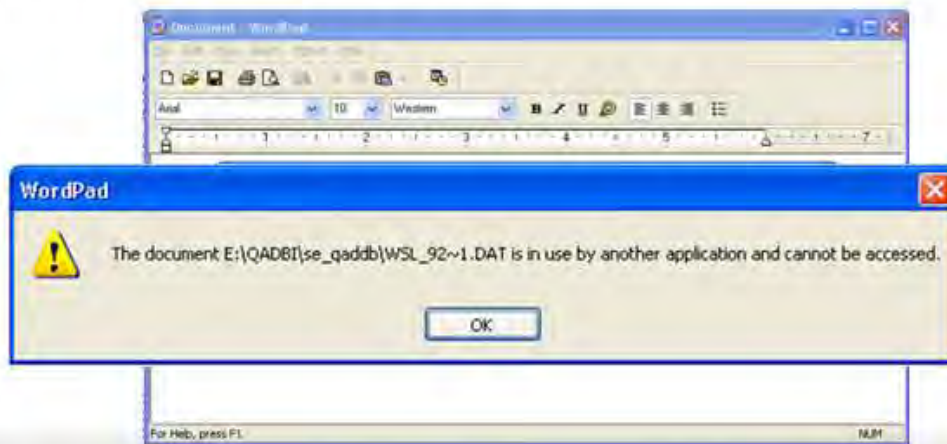
- Sometimes a job fails with a mysterious /F switch error message
- SQL Admin: The specified output file as specified with the /F switch could not be created or opened



Scheduler – /F Switch Failure

Scheduler – /F Switch Failure

- This is due to a job failure when there is a hung job that locks a .dat file.
- If you try to look at the .dat file, it will tell you that the file is already in use.



Scheduler – /F Switch Failure

Scheduler – /F Switch Failure

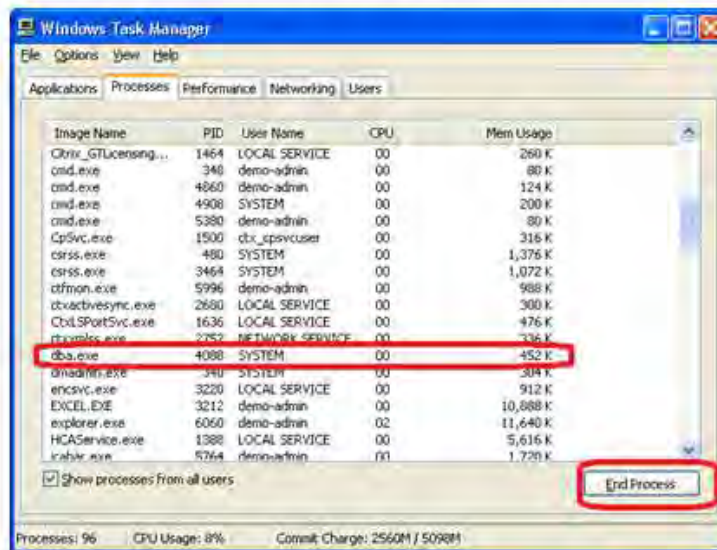
- To unlock the .dat file, terminate the *dba.exe* process.
- Launch the Windows Task Manager.



Scheduler – /F Switch Failure

Scheduler – /F Switch Failure

- In the Task Manager, go to the Processes tab and select the *dba.exe* process. Click *End Process*.



Click on Image Name to sort by name. Look for a job called *dba.exe*. It's possible there may be multiple running (if there are, make sure that other load tables aren't actually loading before proceeding). This is the job that is holding the file. Click on the *dba.exe*, then click *End Process*.

Scheduler – /F Switch Failure

Scheduler – /F Switch Failure

- Once the dba.exe file(s) have been halted, restart the job via the Scheduler. It should now run to completion.

Scheduler – “Over-packed” Columns

Scheduler – “Over-packed” Columns

- Jobs running unusually long (hung)
- When data in a Progress database field exceeds the length defined for its SQL-width
- Progress dbtool Utility



17

When the ODBC connection, which the DWD uses to connect from SQL Server to Progress, tried to pull data from a table with an over-packed column, the load table job hangs, with no error message. A job can sit there for days or even months with nothing happening. To fix this issue, the over-packed column in the Progress table needs to be fixed by a Progress DBA. After the fix occurs, restart the aborted job.

The program provided in Progress for fixing issues like this is called dbtool. Even if the over-packed column is redefined, if there is more characters in a column than the load table’s mapped column, it may require a modification to the load table column by either using a substring command or by making the column wider (although if the latter is chosen it will require that subsequent staging tables also be changed to accommodate the wider column).

More information on the Progress dbtool Utility within Extras PowerPoint slides or BI Site Prep Guide.

The Pieces So Far- How This All Ties Together

The Pieces So Far- How This All Ties Together

- SQL Server
 - Data warehouse
 - Portal database
 - Metadata database
- DWD (Data Warehouse Designer)
 - Datasource
 - Data load procedures
 - Data transformation
 - Metadata
 - Scheduler

The Pieces So Far- How This All Ties Together

The Pieces So Far- How This All Ties Together

- Portal
 - Web interface to the fact tables
- Administrator
 - License key
 - ODBC connections
 - Metadata implementation
 - Scheduler setup
 - Applications/patches/languages installation



19

The Installer should be used to control any standard changes to the data in the Admin tool, but it will be needed for some other things.

We'll review the Admin and DWD further shortly.

Cubes

Cubes



Cubes – What are they?

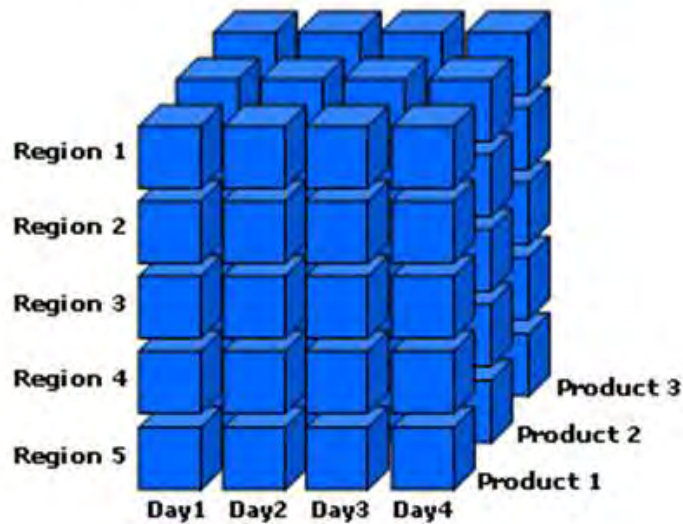
Cubes – What are they?

- An **OLAP cube** (for [online analytical processing](#)) is a [data structure](#) that allows fast analysis of data. It can also be defined as the capability of manipulating and analyzing data from multiple perspectives.
- In our BI environment, our OLAP cubes are considered **MOLAP** ([multidimensional online analytical processing](#)) cubes.

Cube Structure

Cube Structure

- Visual of a cube



In this example, the cube is three dimensional, but cubes can be many more dimensions than just three. It's just very difficult to display them graphically.

DWD Cube Types

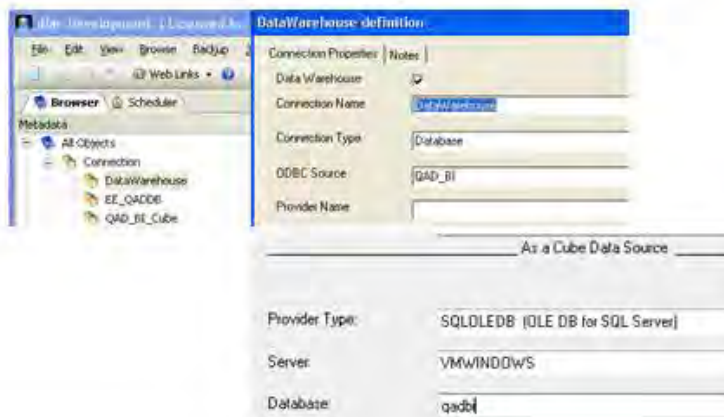
DWD Cube Types

- Olap Cube
 - Sales, Inventory, Purchasing
 - Measure Groups
- *Cube*
 - *cube_om_booking, cube_om_invoice*
 - *cube_om_order, cube_om_shipment*
 - *cube_ap_check, cube_ar_payment*
 - *Specific Measures*
 - *Old Style (WhereScape v5)*

Cubes Setup – Set Connections

Cubes Setup – Set Connections

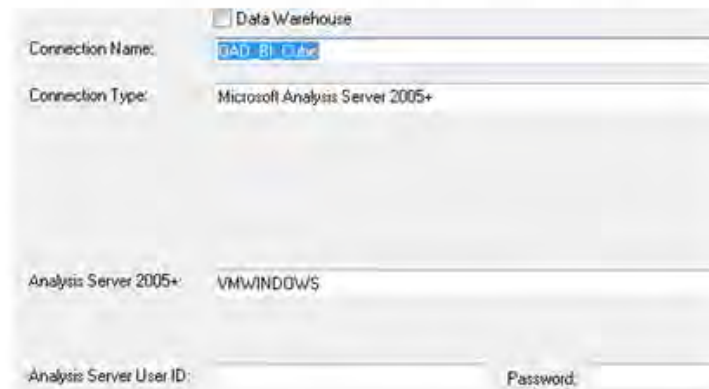
1. In DWD, double-click on the *DataWarehouse Connection*
2. Set the *Provider Type, Server & Database*



Cubes Setup – Set Connections

Cubes Setup – Set Connections

3. Double-click on the *QAD_BI_Cube* Connection
4. Set hostname in *Analysis Server 2005+* box



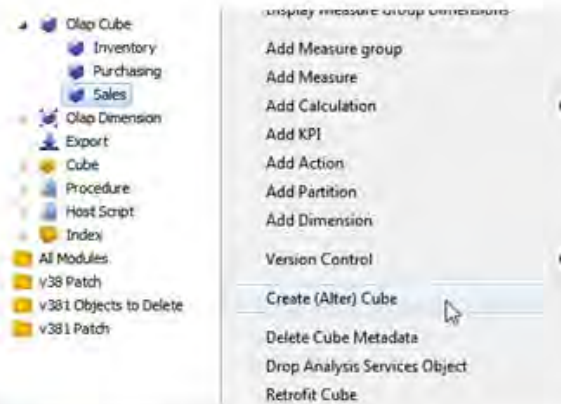
The screenshot shows a dialog box for setting up a connection. At the top, there is a checkbox labeled "Data Warehouse" which is checked. Below this, the "Connection Name" field contains "QAD_BI_Cube" and is highlighted in blue. The "Connection Type" is set to "Microsoft Analysis Server 2005+". The "Analysis Server 2005+" field contains "VMWINDOWS". At the bottom, there are fields for "Analysis Server User ID:" and "Password:".

Localhost works if the server is on the same machine as the DWD. It's typically better to use the actual host name.

Cubes Setup – Build the Olap Cubes

Cubes Setup – Build the Olap Cubes

5. Expand the *Olap Cube* folder. Right-click on an Olap Cube and select the *Create (Alter) Cube* option.
6. Do this for every Olap Cube.



Cubes Setup – Build the Cubes (v5)

Cubes Setup – Build the Cubes (v5)

7. Expand the *Cube* folder. Right-click on a *Cube* and select the *Create Cube* option.
8. Do this for every cube.



Cubes Processing – Run the Job

Cubes Processing – Run the Job

- In DWD Scheduler, run the job(s)
POSTPROCESS_XX_CUBES

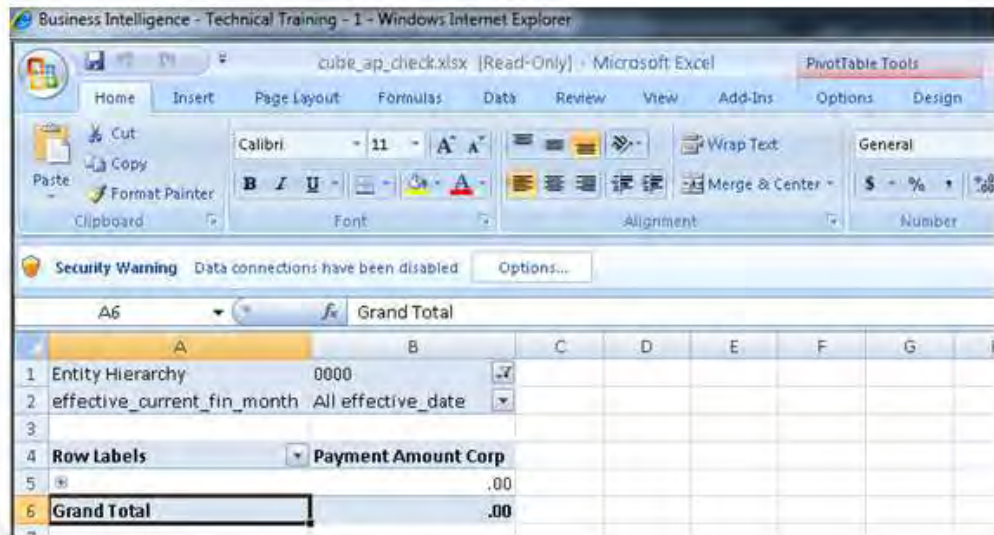
| Job | Status | Seq | Star |
|-----------------------------------|---------|-----|------|
| POSTPROCESS_FIN_SE_CUBES | On Hold | 60 | 2011 |
| POSTPROCESS_OM_CUBES | | | |
| POSTPROCESS_OM_SUMMARY | | | |
| PREPROCESS_COMMON_DAILYONLY_LOADS | | | |
| PREPROCESS_COMMON_EE2010UP_LOADS | | | |
| PREPROCESS_COMMON_EEONLY_LOADS | | | |
| PREPROCESS_COMMON_HISTONLY_LOADS | | | |
| PREPROCESS_COMMON_LOADS | | | |
| PREPROCESS_COMMON_TRUNCATE | | | |
| PREPROCESS_FIN_EE_LOADS | | | |
| PREPROCESS_FIN_LOADS | | | |
| PREPROCESS_FIN_SE_DAILYONLY_LOADS | | | |
| PREPROCESS_FIN_SE_HISTONLY_LOADS | | | |
| PREPROCESS_FIN_SE_LOADS | | | |
| PREPROCESS_FIN_TRUNCATE | | | |
| PREPROCESS_OM_DAILYONLY_LOADS | | | |
| PREPROCESS_OM_LOADS | | | |
| PREPROCESS_OM_TRUNCATE | | | |
| SET_CONNECTION_EE_QADDE | | | |

| |
|----------------------|
| View Tasks |
| View Audit Trail |
| View Detail Log |
| Export to CSV File |
| Documentation |
| Edit Job |
| Edit Tasks |
| Edit Dependencies |
| Insert Copy of Job |
| Delete Job |
| Multiple Log Delete |
| Start the Job |
| Hide the Job |
| About Job |
| Restart the Job |

Setup Microsoft Excel for Cubes

Setup Microsoft Excel for Cubes

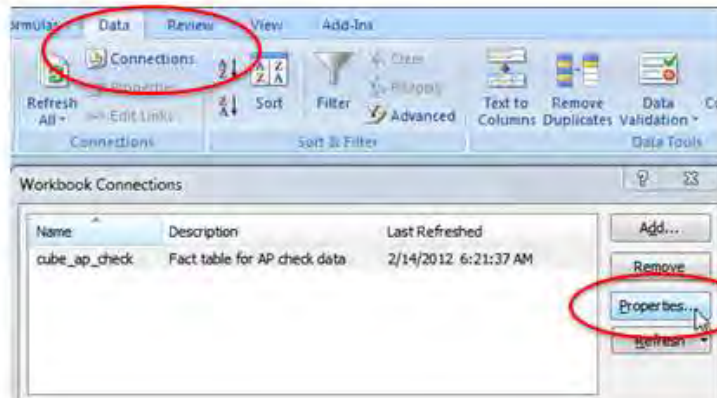
1. Open an Excel spreadsheet from the /published directory.



Setup Microsoft Excel for Cubes

Setup Microsoft Excel for Cubes

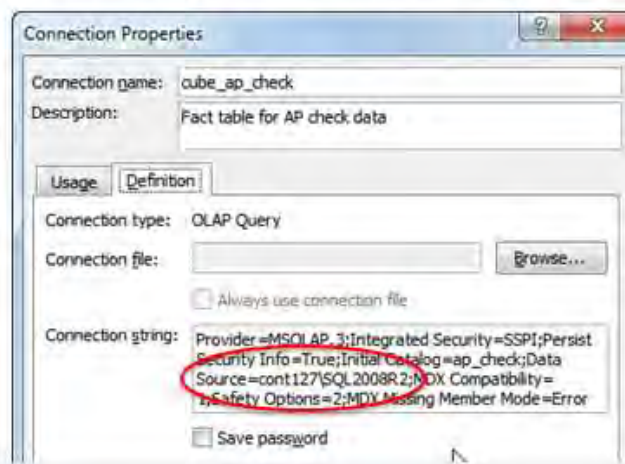
2. Go to the *Data* tab. Click on *Connections* then *Properties*.



Setup Microsoft Excel for Cubes

Setup Microsoft Excel for Cubes

3. In *Connection Properties*, update the *Data Source* to the hostname of the SQL Server (Ex: VMWINDOWS)



Setup Microsoft Excel for Cubes

Setup Microsoft Excel for Cubes

- The pivot table can now be refreshed to retrieve new data.

Note: change filter values to produce results

The screenshot shows a pivot table with the following structure:

| | A | B |
|---|-----------------------------|----------------------------|
| 1 | Entity Hierarchy | All dim_entity |
| 2 | effective_current_fin_month | All effective_date |
| 3 | | |
| 4 | Row Labels | Payment Amount Corp |
| 5 | | 6,030.00 |
| 6 | Grand Total | 6,030.00 |

A Complete Review of the DWD Tool

A Complete Review of the DWD Tool

- Now that we've installed all the important components, it's time to become familiar with the DWD.

QAD Administrator

QAD Administrator

- Most elements are now set via the Installer. The QAD Administrator can still be used for:
 - Additional non Progress sourced ODBC setup

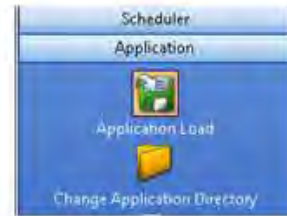


Add Odbc entry

- Scheduler Management



- Installing applications and hot fixes



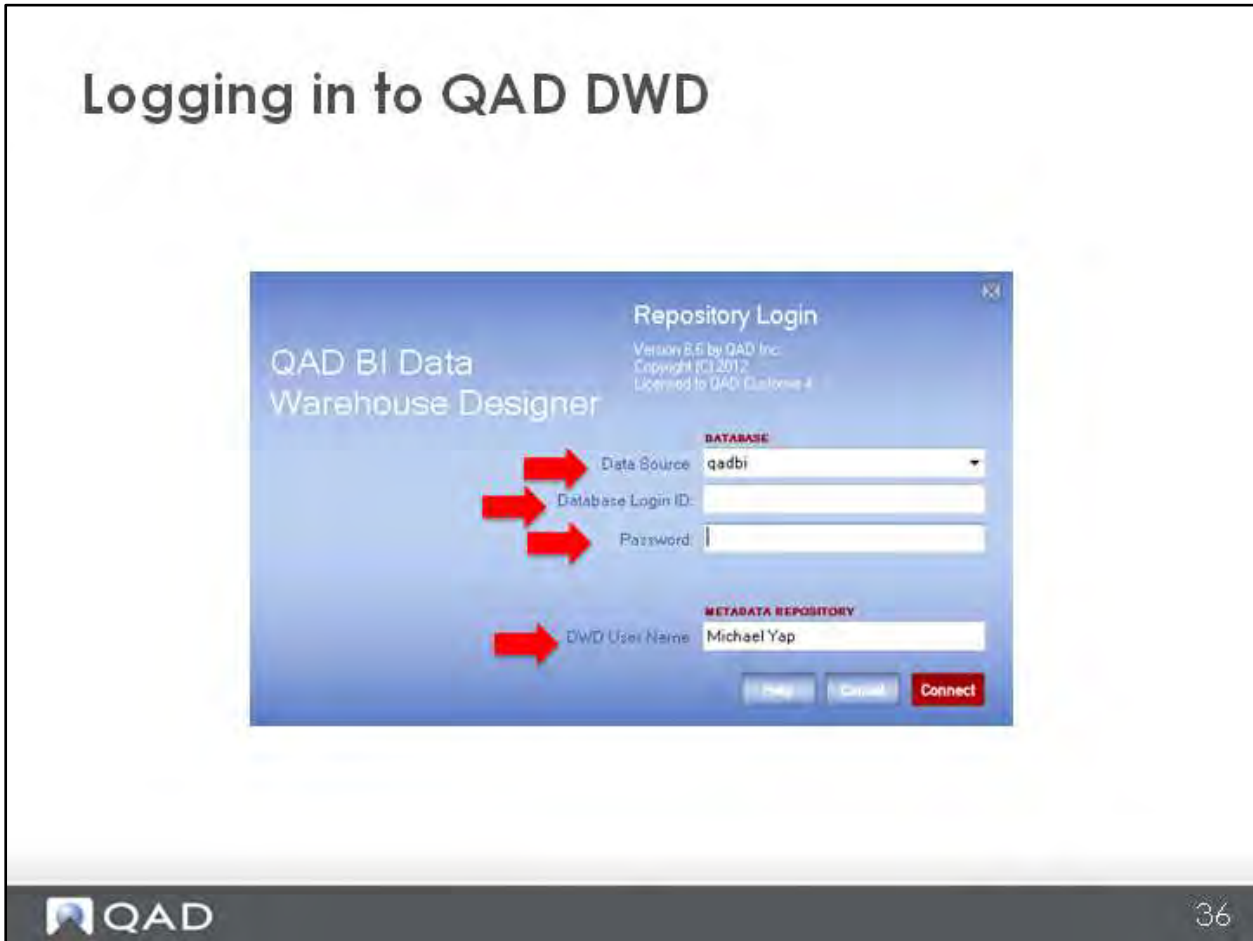
ODBC - If flat file sources or other non-Progress system sources need to be set up, this can be done here.

DWD Interface

DWD Interface



Logging in to QAD DWD



Data Warehouse Repository includes both the Data IN the Data Warehouse and the Metadata.

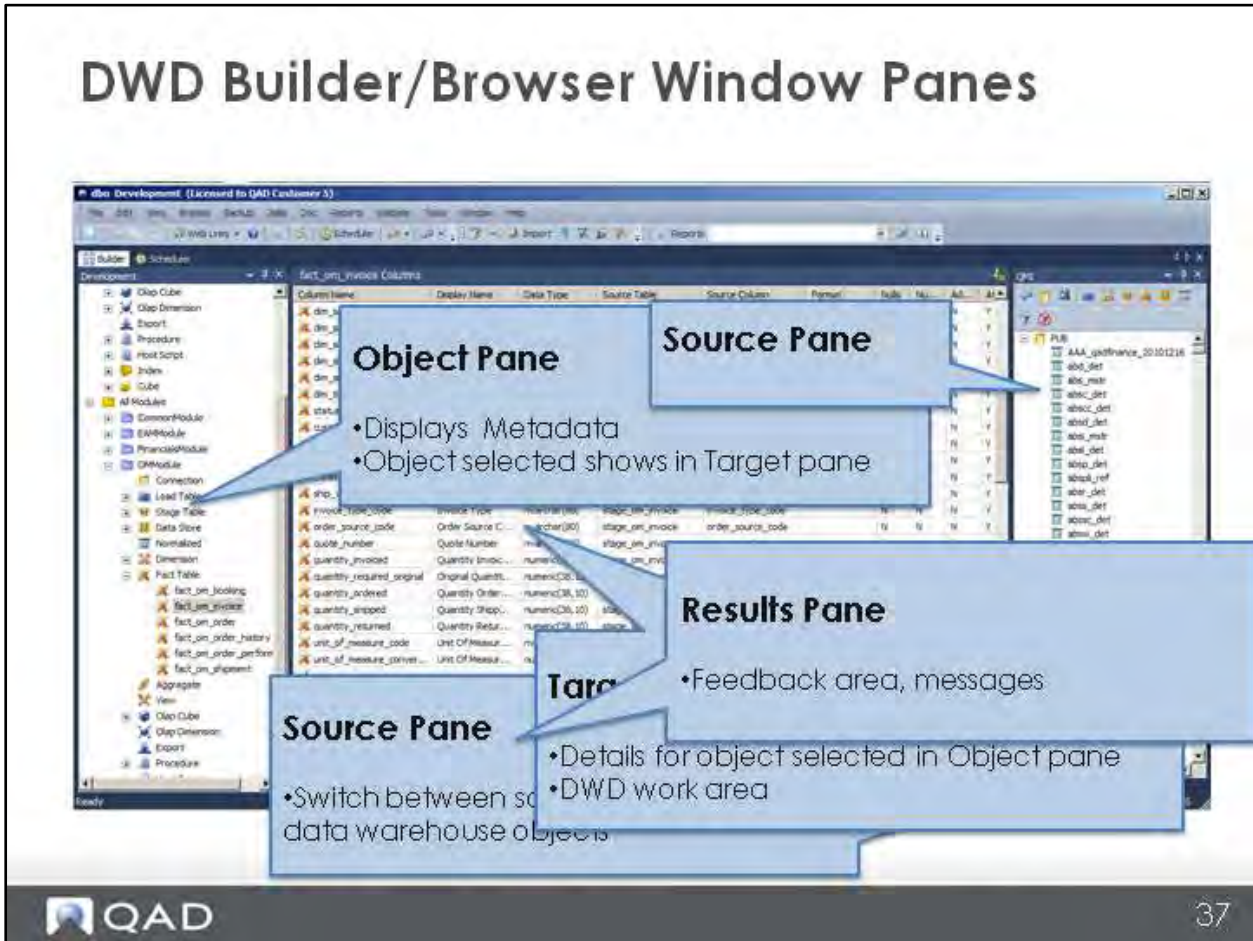
Metadata is the information that describes the data. That is, the data about the Data.

The Repository is a SQL Server database and Is accessed via an ODBC connection.

To login to the DWD:

1. Select the *Data Source* or ODBC connection to your Data Warehouse from the drop down list. If the environment includes multiple data warehouses, this field will default to the last data warehouse you accessed using the DWD.
2. Enter a valid Database Login ID and password.
3. The Full User Name is the name that will be associated to any procedures, tables, etc. and scheduled jobs that are created from within the DWD. Normally, you should enter your first and last name.

DWD Builder/Browser Window Panes



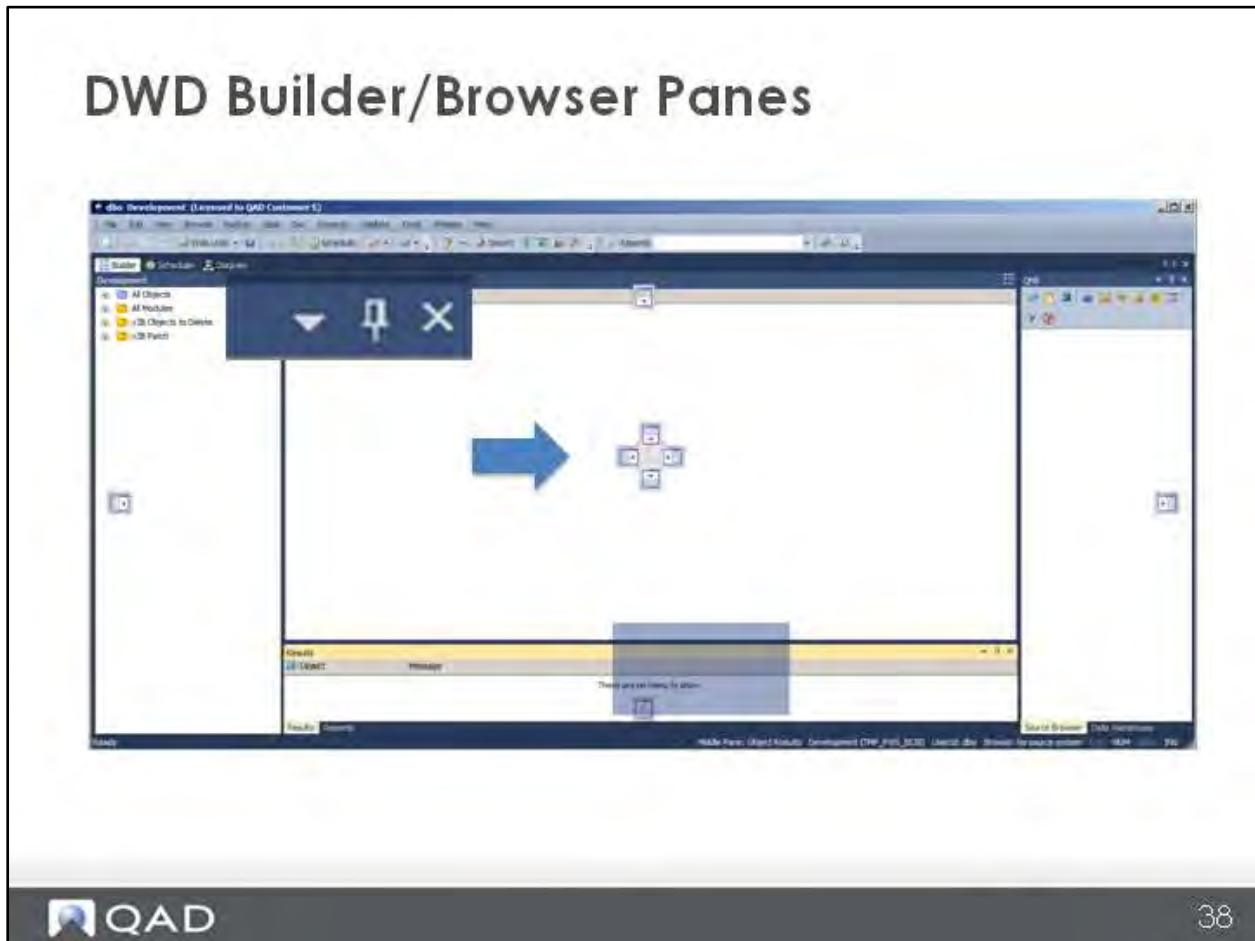
There are 4 main window panes in the DWD:

- Builder (formerly Browser)
- Scheduler
- Diagram
- Procedure Editor

Within each window, there may be multiple panes. In the Builder/Browser Window, there are 4 panes.

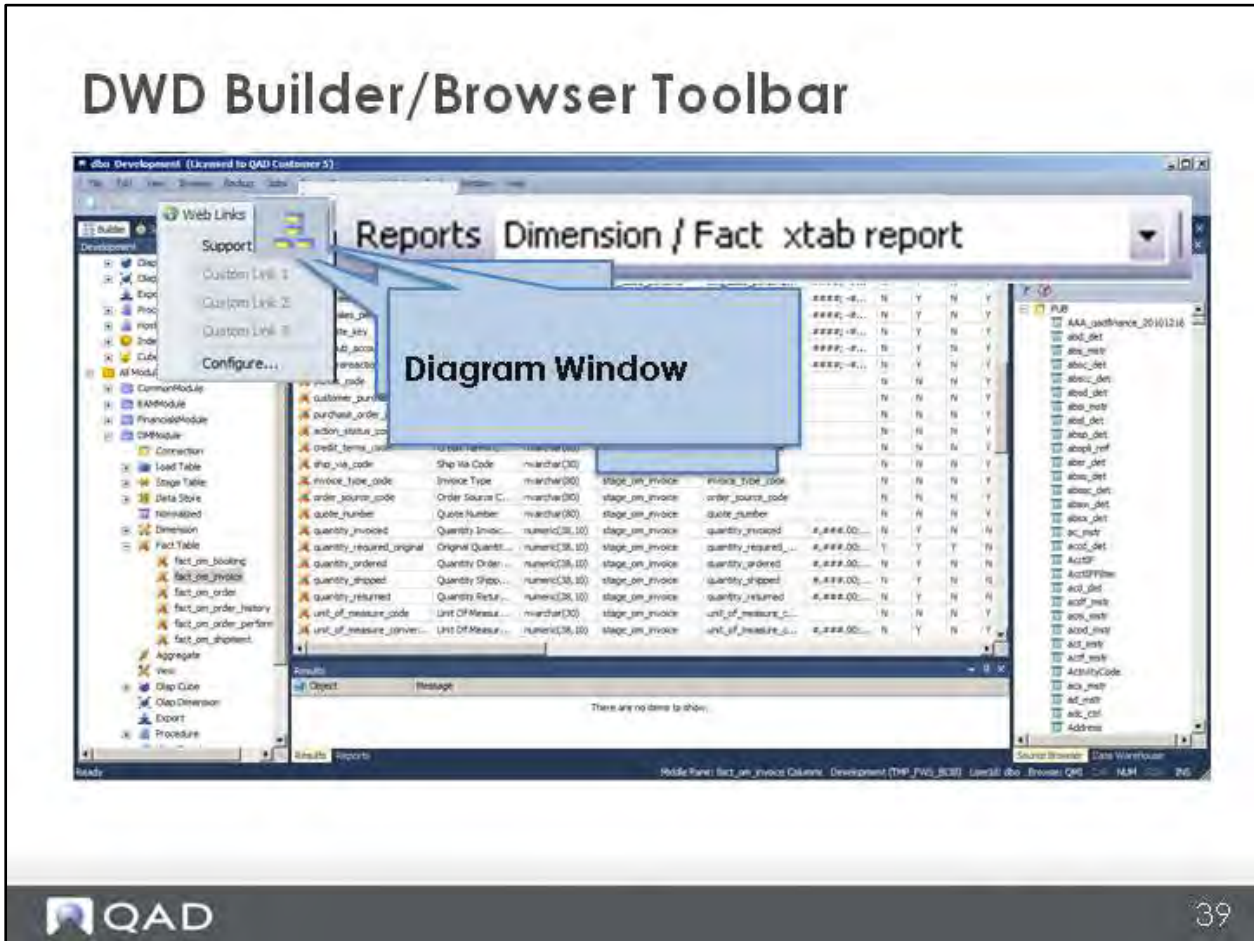
- **Object pane.** Displays the metadata
- **Target pane.** Item selected in the object pane has detailed information shown in the target pane
- **Results pane.** Feedback area
- **Source pane.** Two tabs – Source system and Data Warehouse

DWD Builder/Browser Panes



In the Builder (Browser) the individual panes can be moved, resized and hidden. To move the pane back to the original position, double click on the pane title bar.

DWD Builder/Browser Toolbar



The Builder (Browser) menu bar provides access to the following:

- QAD Support
- Diagram Window
- Scheduler Window
- Reports

WhereScape RED®

WhereScape RED®

- QAD DWD is a re-branded WhereScape RED
- www.wherescape.com
- Access to Forum with registered email

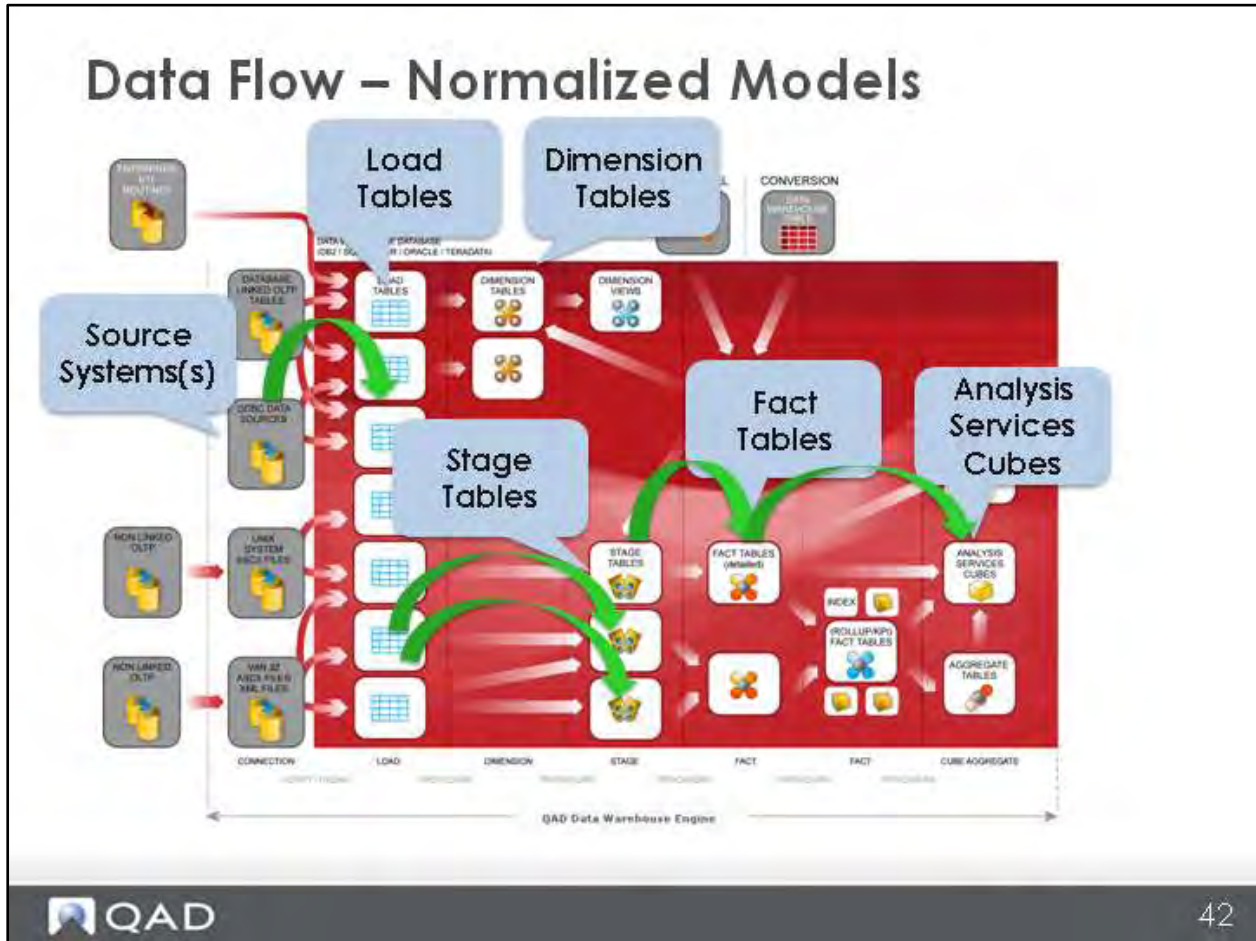
The QAD DWD is a rebranded version of a product from WhereScape Inc. named “Red”.

DWD Concepts

DWD Concepts



Data Flow – Normalized Models



The expected flow of data is as follows:

- Source systems (ERP, EAM, TMS)
- Load tables
- Stage tables
- Dimension & Fact tables
- Microsoft Analysis Services Cubes

DWD Objects

DWD Objects



Our Passion. Your Advantage.

Data Objects Types

Data Objects Types

- Connection
- Load table
- Stage table
- Dimension
- Fact table
- Data Store

Data Object Types (continued)

Data Object Types (continued)

- OLAP cube
- OLAP dimension
- Cube
- Procedure
- Index

DWD Object Builder

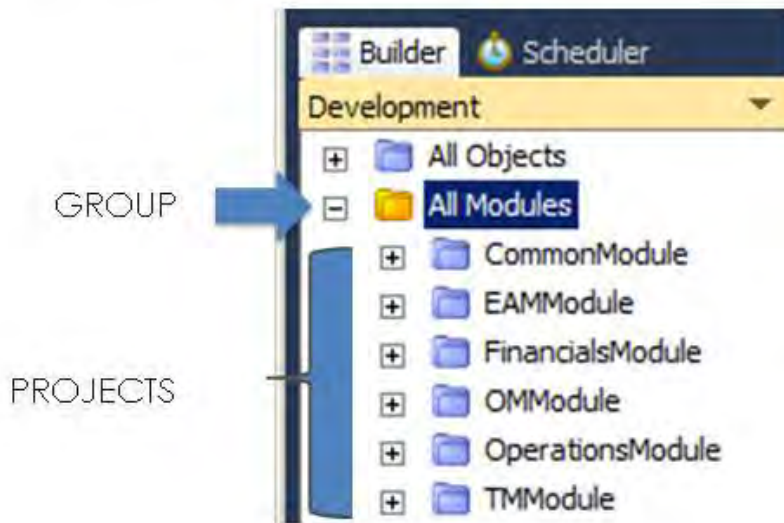


Objects displayed in the object pane are organized into Projects.

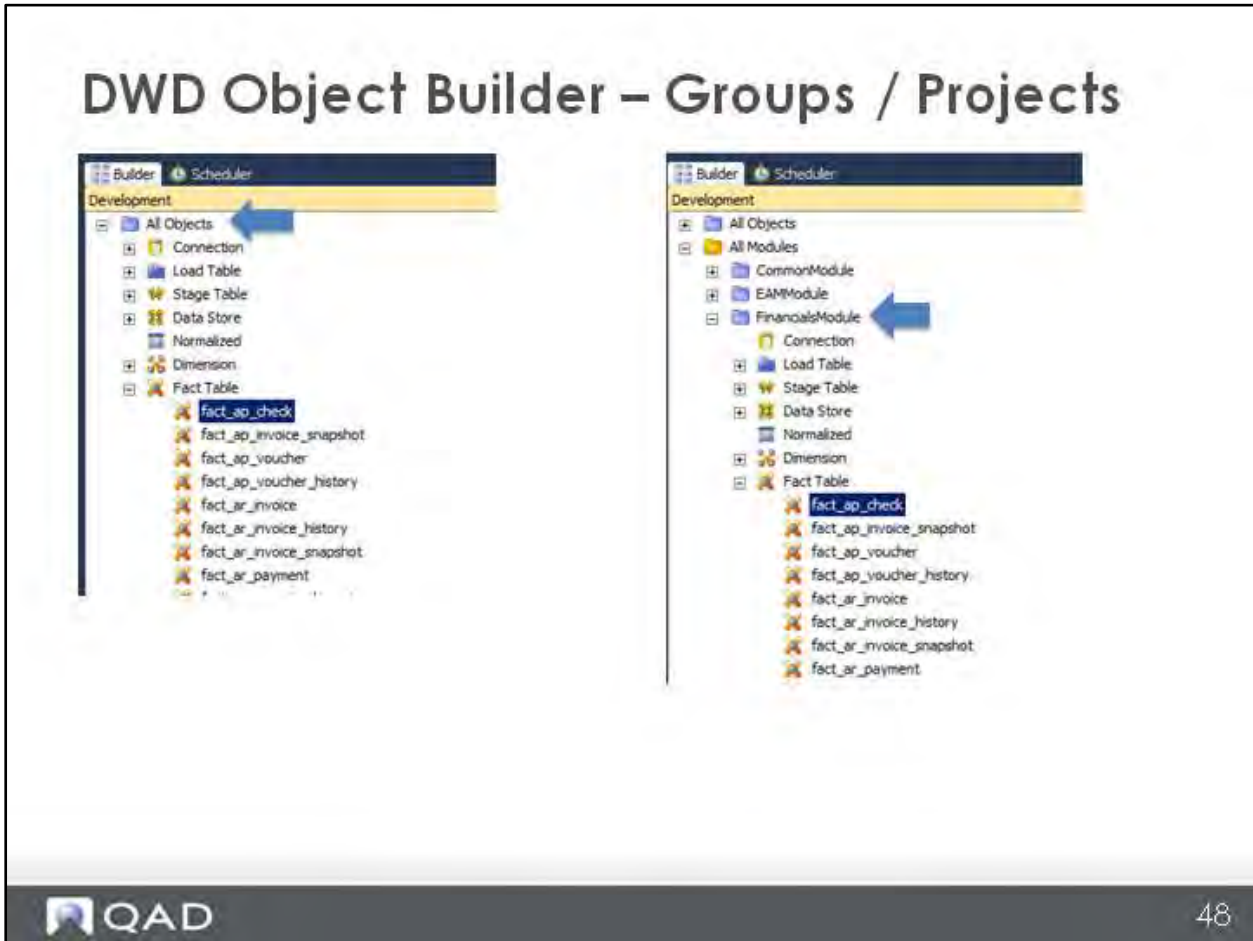
A single object can be assigned to more than one project.

DWD Object Builder – Groups / Projects

DWD Object Builder – Groups / Projects



DWD Object Builder – Groups / Projects



The All Objects project contains every object in the data warehouse. To expand the set, click on the plus sign in front of the set name.

Here we see that the object named *fact_ap_check* is linked to both the *All Objects* project, and the *FinancialModules* project. Further, the *FinancialsModules* project is part of the *All Modules* group.

Connections



Connections define the path to external objects, such as source data.

Also, a connection is defined to the BI Data Warehouse itself, the SQL database that contains the BI data.

QAD_MASTER is a template connection for Enterprise Application (ERP) and EAM Source systems. The connections named *2010SE* and *QMI* above were defined during the implementation of this specific data warehouse. They are links to 2 different QAD Enterprise Application environments.

There is also a link shown to the Microsoft Analysis Services Cubes (*QAD_BI_Cube*)

Load Tables

Load Tables

| Table Name | Table Type | Table Description | Table Status | Table Owner | Table Location | Table Size | Table Date | Table Time | Table User | Table Action |
|------------|------------|---------------------------|--------------|-------------|----------------|------------|------------|------------|------------|--------------|
| 1. 10100 | A | Plant MynkinnellEquipment | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 2. 10115 | A | Acc Dep Plant Mnth & Exp | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 3. 10120 | A | Furniture | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 4. 10125 | A | Accum Dep Furniture | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 5. 10130 | A | Computer Hardware | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 6. 10135 | A | Acc Dep Computer Hardware | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 7. 10140 | A | Computer Software | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 8. 10145 | A | Land | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 9. 10150 | A | Acc Depreciation Land | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 10. 10155 | A | Office Equipment | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 11. 10160 | A | Building | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 12. 10165 | A | Auto Dep Building | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 13. 10170 | A | Building Improvement | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 14. 10175 | A | Acc Dep Building Improve | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 15. 10180 | A | Transportation | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 16. 10185 | A | Acc Dep Transport Equip | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 17. 10190 | A | Other Long Term Assets | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 18. 11110 | A | Raw Material Inventory | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 19. 11115 | A | Work In Process Inventory | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 20. 11120 | A | Work In Process Adjust | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 21. 11125 | A | Sub-Assemblies | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 22. 11130 | A | Finished Goods Inventory | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 23. 11135 | A | Finished Goods Adjust | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 24. 11140 | A | Spares | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |
| 25. 11145 | A | Inventory Reserve | DMR | 1.1 | 1 | 1 | 1 | 1 | 1 | AUTO |

Context Menu Options:

- Load
- Process Table via Scheduler
- Object Documentation
- Remove from Project
- Add to Project

QAD 50

Load tables are the first entry point of data into the data repository. They hold the latest set of change data.

Standard QAD modules include Load tables that are ODBC “type”.

Right click on the table name for a drop down list of options. Options include *Display Data* and *Export Data via Excel*.

When using the *Export Data via Excel*, be aware of the potential number of rows to be extracted. You can add a SQL Where clause to the extract, to limit the number of rows extracted.

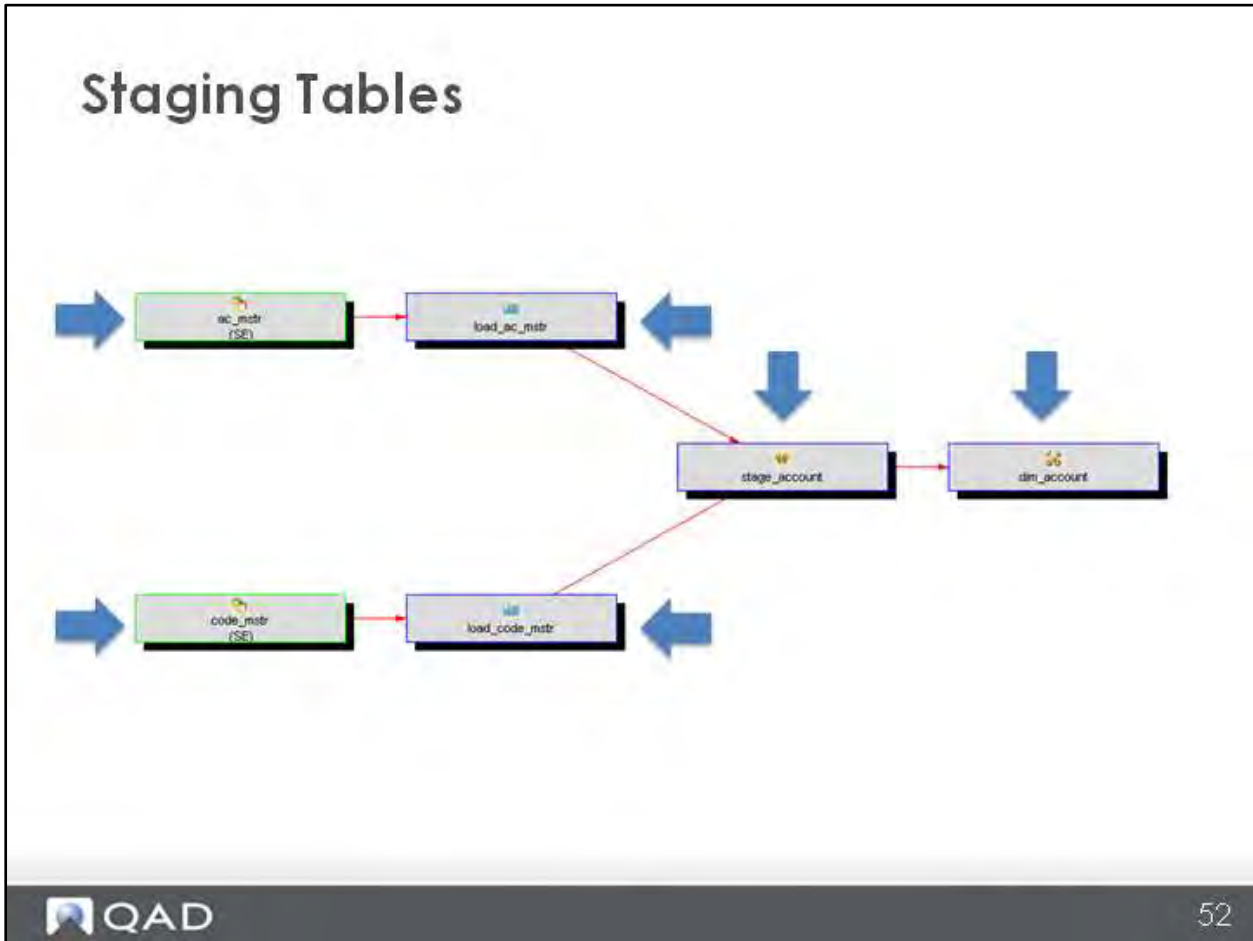
Load Tables - Properties



Table *Properties* show key elements related to the table.

Load tables can include pre-load actions. Here we see an example of a *Truncate* pre-load action.

Staging Tables



Staging tables are used in the transformation of raw data into a star schema format. They are **temporary** storage.

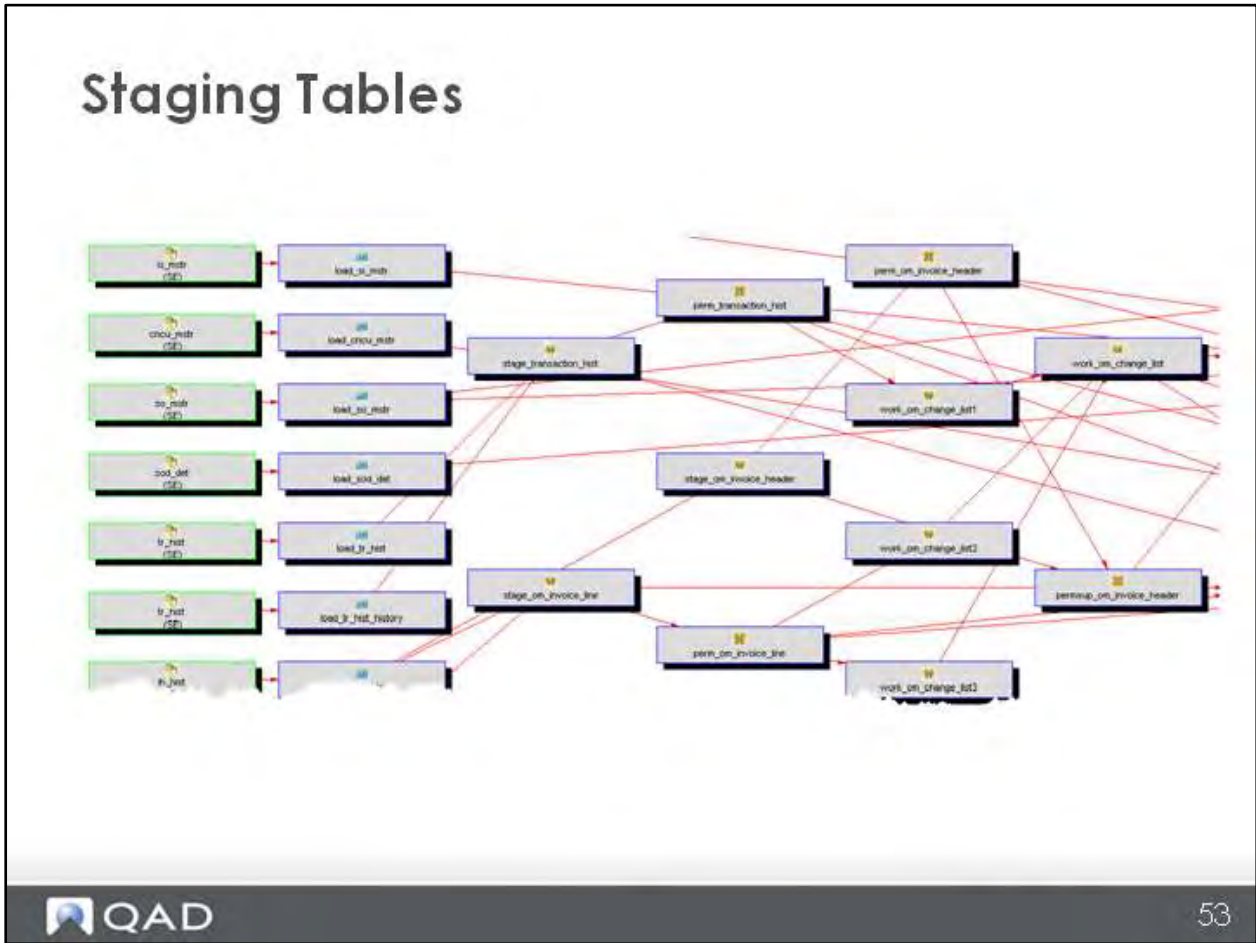
This slide is an example of a *Track Back Diagram*.

From left to right:

The two source tables shown in this diagram are *ac_mstr* and *code_mstr*. Note that the boxes are in green, there are outside the data Warehouse.

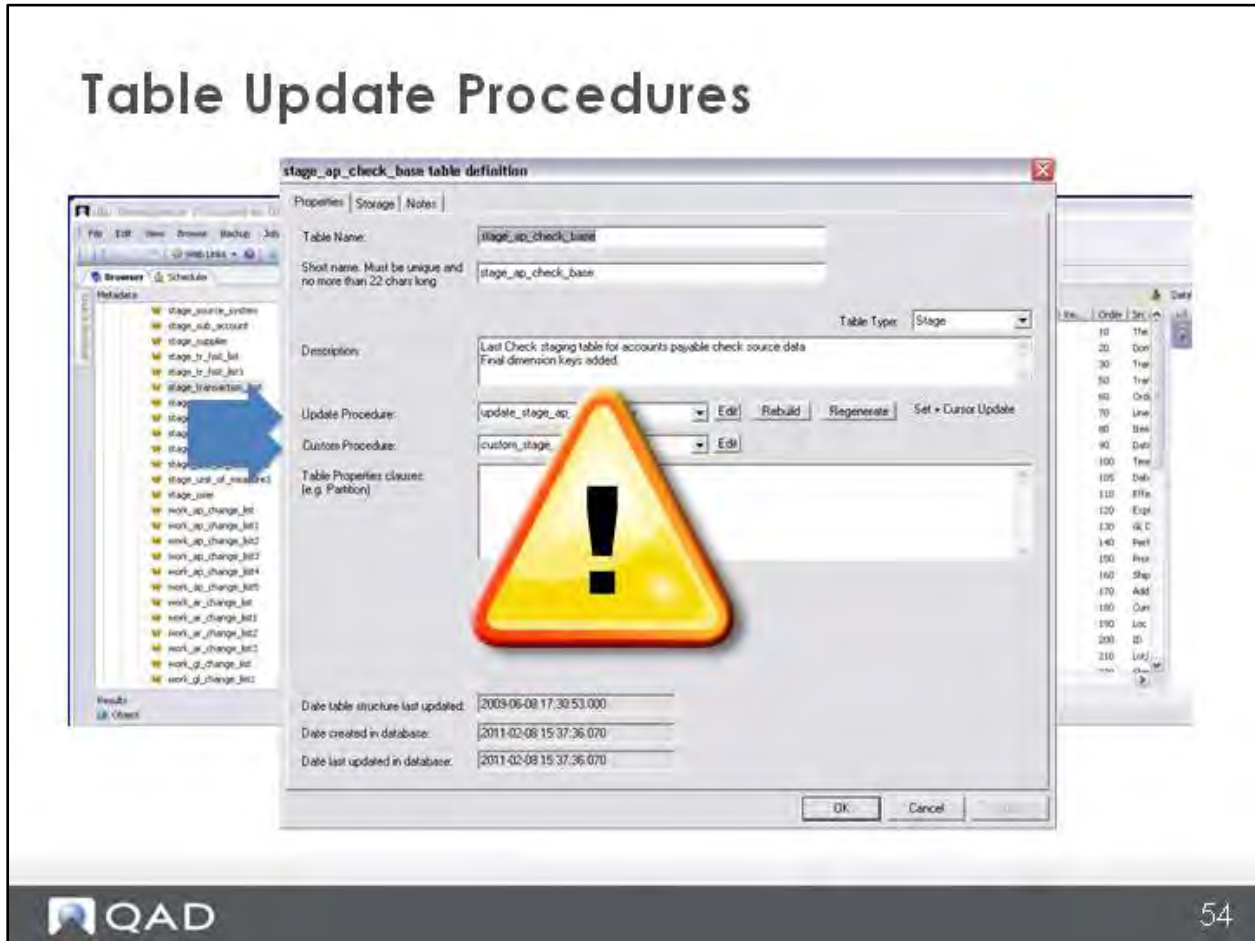
The source tables feed the load tables. The load tables are combined to a single stage table. Finally, we see that the stage table feeds the account dimension table.

Staging Tables



This slide is an example of a portion of a much more complex track-back diagram.

Table Update Procedures

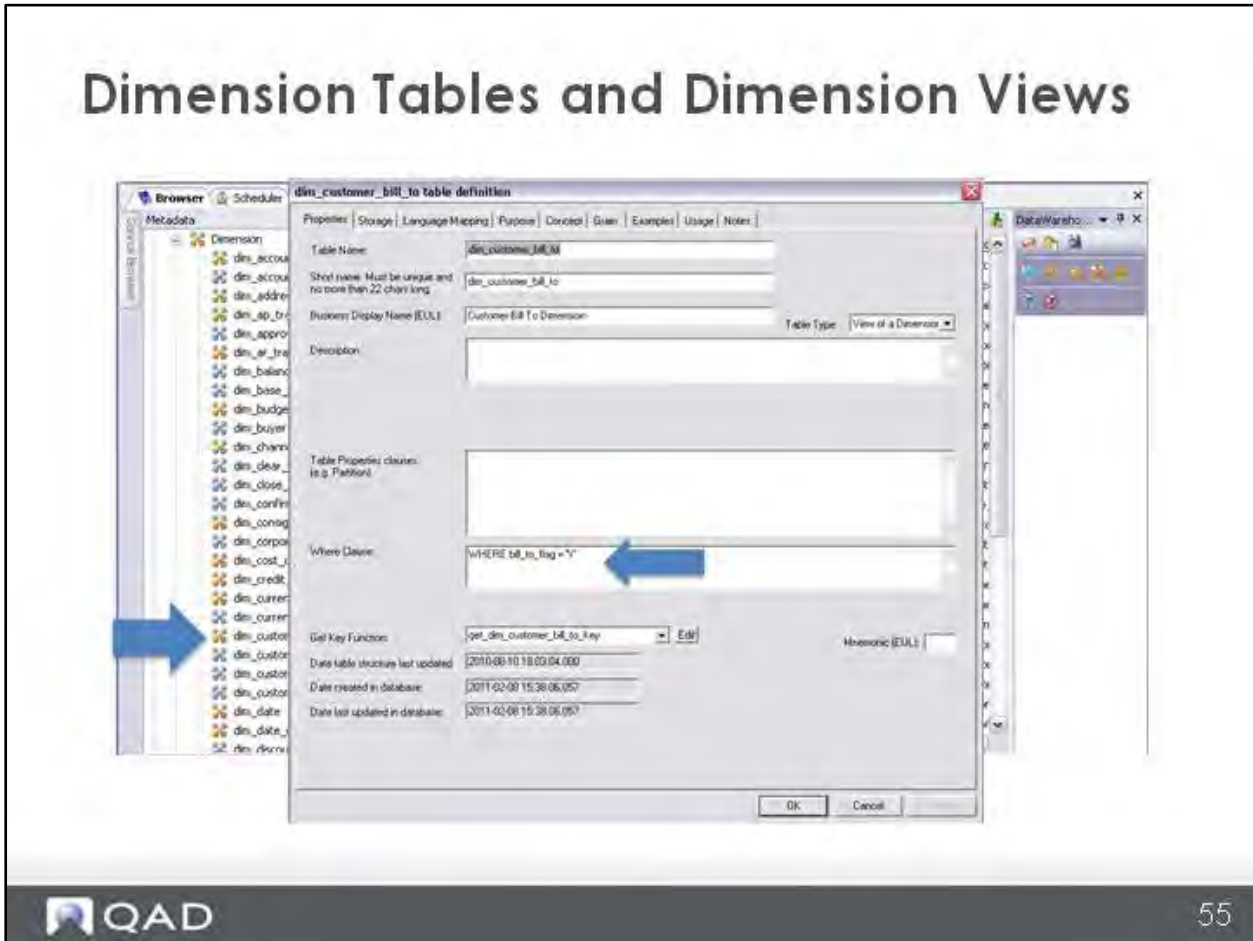


As with other tables within the data warehouse, the processing of data in the staging tables is controlled using procedures.

In this example, we see two procedures: *Update* and *Custom*.

These procedures should NOT be modified, except by trained personnel.

Dimension Tables and Dimension Views



Dimensions are qualifiers of facts.

Views are created from dimensions.

As an example: The dim_customer contains all rows for all customers. There are 3 views for this dimension: bill_to, sold_to and ship_to.

The Bill_to dimension view will select only rows in the customer dimension where the bill-to flag is “Y”.

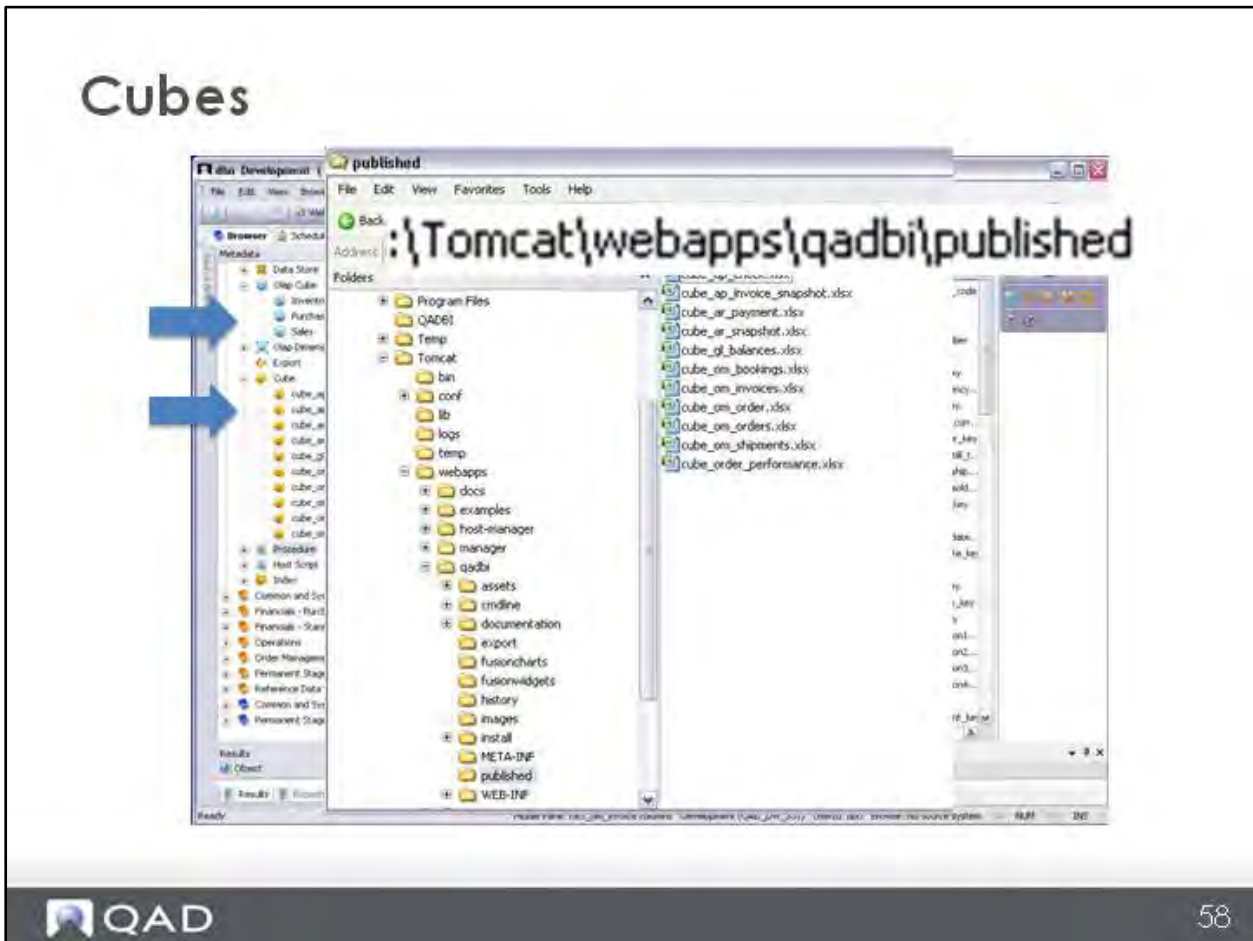
Fact Tables

Fact Tables

| Col name | Display | Data type | No. table | Col name | Measure | Sub | No... | Att... | Att... |
|----------------------------|--------------------|---------------|------------------------|----------------------------|---------|-----|-------|--------|--------|
| source_system_code | Source System | varchar(4) | stage_on_invoice | source_system_code | N | N | N | Y | |
| domain_code | Domain | varchar(8) | stage_on_invoice | domain_code | N | N | N | Y | |
| order_number | Order Number | varchar(80) | stage_on_invoice | order_number | N | N | N | Y | |
| order_line_number | Line | Integer | stage_on_invoice | order_line_number | ### | N | Y | N | Y |
| invoice_number | Invoice Number | varchar(80) | stage_on_invoice | invoice_number | N | N | N | Y | |
| dim_account_key | Dim Account Key | Integer | dim_account | dim_account_key | ### | N | Y | N | Y |
| dim_base_currency_key | Dim Base Currency | Integer | dim_channel | dim_base_currency | ### | N | Y | N | Y |
| dim_channel_key | Dim Channel Key | Integer | dim_channel | dim_channel_key | ### | N | Y | N | Y |
| dim_corporate_currency_key | Dim Corporate... | Integer | dim_corporate_currency | dim_corporate_currency_key | ### | N | Y | N | Y |
| dim_cost_center_key | Dim Cost Center | Integer | dim_cost_center | dim_cost_center_key | ### | N | Y | N | Y |
| dim_customer_bill_to_key | Dim Customer B... | Integer | dim_customer_bill_to | dim_customer_bill_to_key | ### | N | Y | N | Y |
| dim_customer_bill_to_key | Dim Customer B... | Integer | dim_customer_bill_to | dim_customer_bill_to_key | ### | N | Y | N | Y |
| dim_customer_bill_to_key | Dim Customer B... | Integer | dim_customer_bill_to | dim_customer_bill_to_key | ### | N | Y | N | Y |
| dim_customer_bill_to_key | Dim Customer B... | Integer | dim_customer_bill_to | dim_customer_bill_to_key | ### | N | Y | N | Y |
| dim_due_date_key | Dim Due Date Key | decimal(16,0) | dim_due_date | dim_due_date_key | ### | N | Y | N | Y |
| dim_entity_key | Dim Entity Key | Integer | dim_entity | dim_entity_key | ### | N | Y | N | Y |
| dim_effective_date_key | Dim Effective D... | decimal(16,0) | dim_effective_date | dim_effective_date_key | ### | N | Y | N | Y |
| dim_invoice_date_key | Dim Invoice Date | decimal(16,0) | dim_invoice_date | dim_invoice_date_key | ### | N | Y | N | Y |
| dim_item_key | Dim Item Key | Integer | dim_item | dim_item_key | ### | N | Y | N | Y |
| dim_location_key | Dim Location Key | Integer | dim_location | dim_location_key | ### | N | Y | N | Y |
| dim_order_date_key | Dim Order Date | decimal(16,0) | dim_order_date | dim_order_date_key | ### | N | Y | N | Y |
| dim_project_key | Dim Project Key | Integer | dim_project | dim_project_key | ### | N | Y | N | Y |
| dim_sales_person1_key | Dim Sales Perso... | Integer | dim_sales_person1 | dim_sales_person1_key | ### | N | Y | N | Y |
| dim_sales_person2_key | Dim Sales Perso... | Integer | dim_sales_person2 | dim_sales_person2_key | ### | N | Y | N | Y |
| dim_sales_person3_key | Dim Sales Perso... | Integer | dim_sales_person3 | dim_sales_person3_key | ### | N | Y | N | Y |
| dim_sales_person4_key | Dim Sales Perso... | Integer | dim_sales_person4 | dim_sales_person4_key | ### | N | Y | N | Y |
| dim_site_key | Dim Site Key | Integer | dim_site | dim_site_key | ### | N | Y | N | Y |
| dim_sub_account_key | Dim Sub Account | Integer | dim_sub_account | dim_sub_account_key | ### | N | Y | N | Y |

Fact tables are the central table in a star schema design.

Cubes



2 types of cube objects in the DWD:

- **Older type** - “Cubes”
- **Newer type** - “OLAP Cubes” and “OLAP Dimensions”.

QAD standard provides Excel workbooks that access the older type of cube.

Procedures

The screenshot displays a SQL development environment with a central window showing the code for a stored procedure. The code is as follows:

```

CREATE PROCEDURE custom_stage_ap_check_base
    @p_sequence integer,
    @p_job_name varchar(256),
    @p_task_name varchar(256),
    @p_job_id integer,
    @p_task_id integer,
    @p_return_msg varchar(256) OUTPUT,
    @p_status integer OUTPUT
AS
SET XACT_ABORT OFF -- Turn off auto abort on errors
SET NOCOUNT ON -- Turn off row count messages

-- Control variables used in each procedure
DECLARE
    @v_msgtext varchar(256) -- Inst for audit trail
    @v_step integer -- return code
    @v_update_count integer -- no of records updated
    @v_insert_count integer -- no of records inserted
    @v_count integer -- General counter
    @v_return_status integer -- Query result status
    @v_row_count integer -- Query returned row count
    @v_db_code varchar(10) -- Database error code
    @v_db_msg varchar(100) -- Database error message

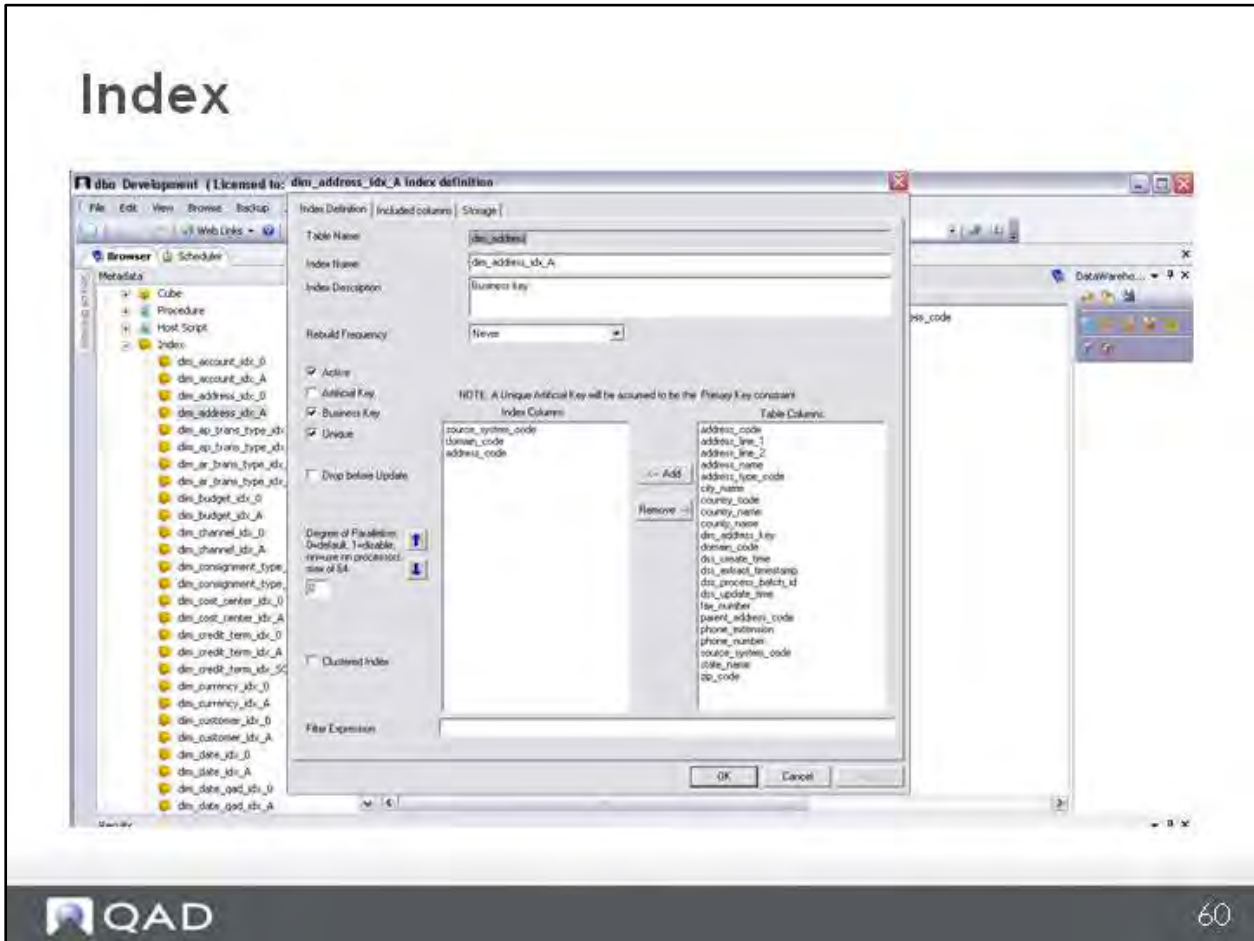
-- Main
SET @v_step = 100
SET @v_update_count = 0
SET @v_insert_count = 0

UPDATE stage_ap_check_base
SET missing_currency_conv_flag = 'Y'
WHERE (amount_trans + amount_base + amount_corp = 0
AND (amount_trans = 0
OR amount_base = 0
OR amount_corp = 0))
    
```

The interface also shows a metadata browser on the left with a tree view of procedures, and a right-hand pane with project information. The QAD logo is visible in the bottom left corner, and the page number '59' is in the bottom right corner.

Procedures are an object type in the DWD.

Index



Tables may have indexes.

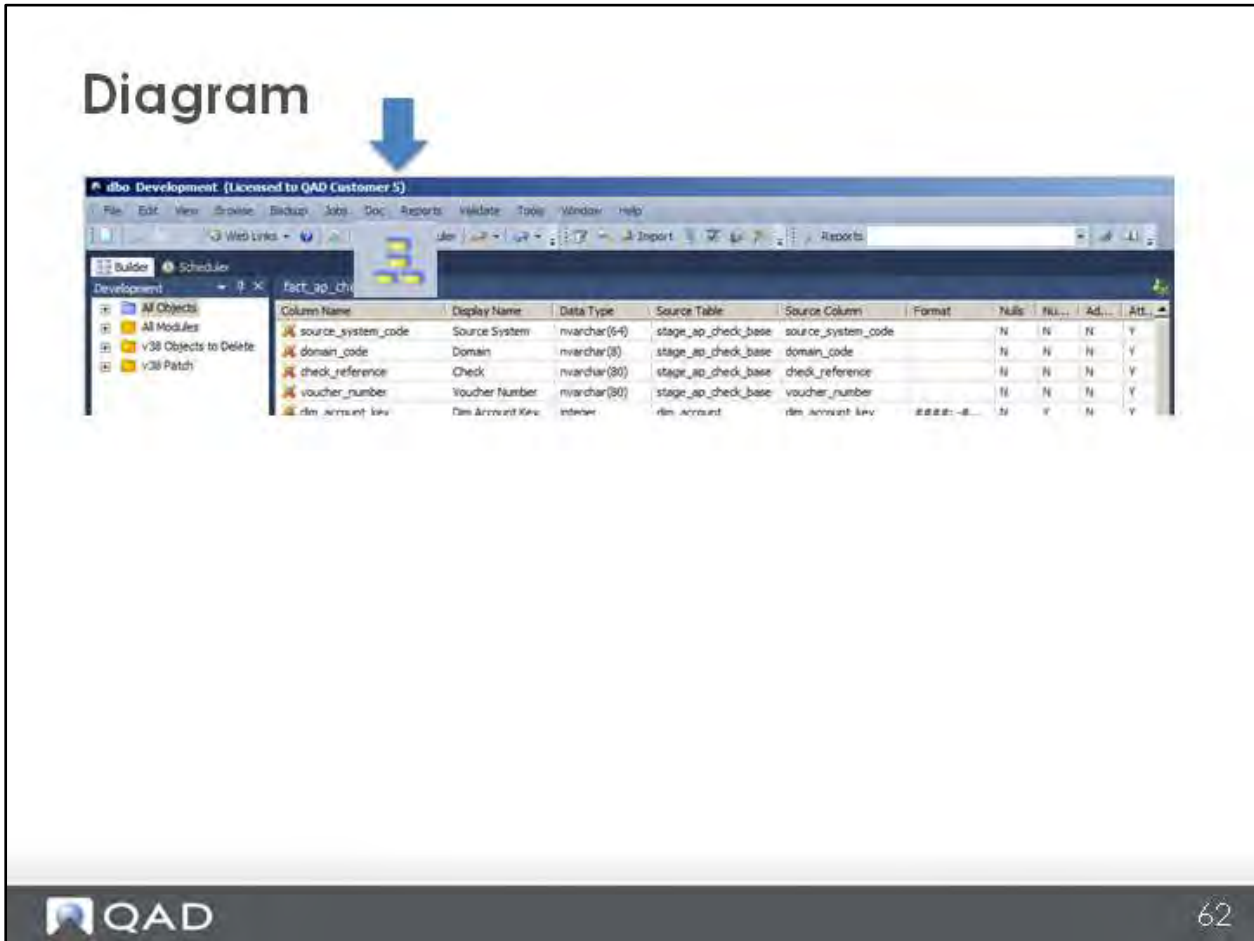
To see which indexes are defined for a table, right click on the table object and select *Display Indexes*

Diagrams

Diagrams

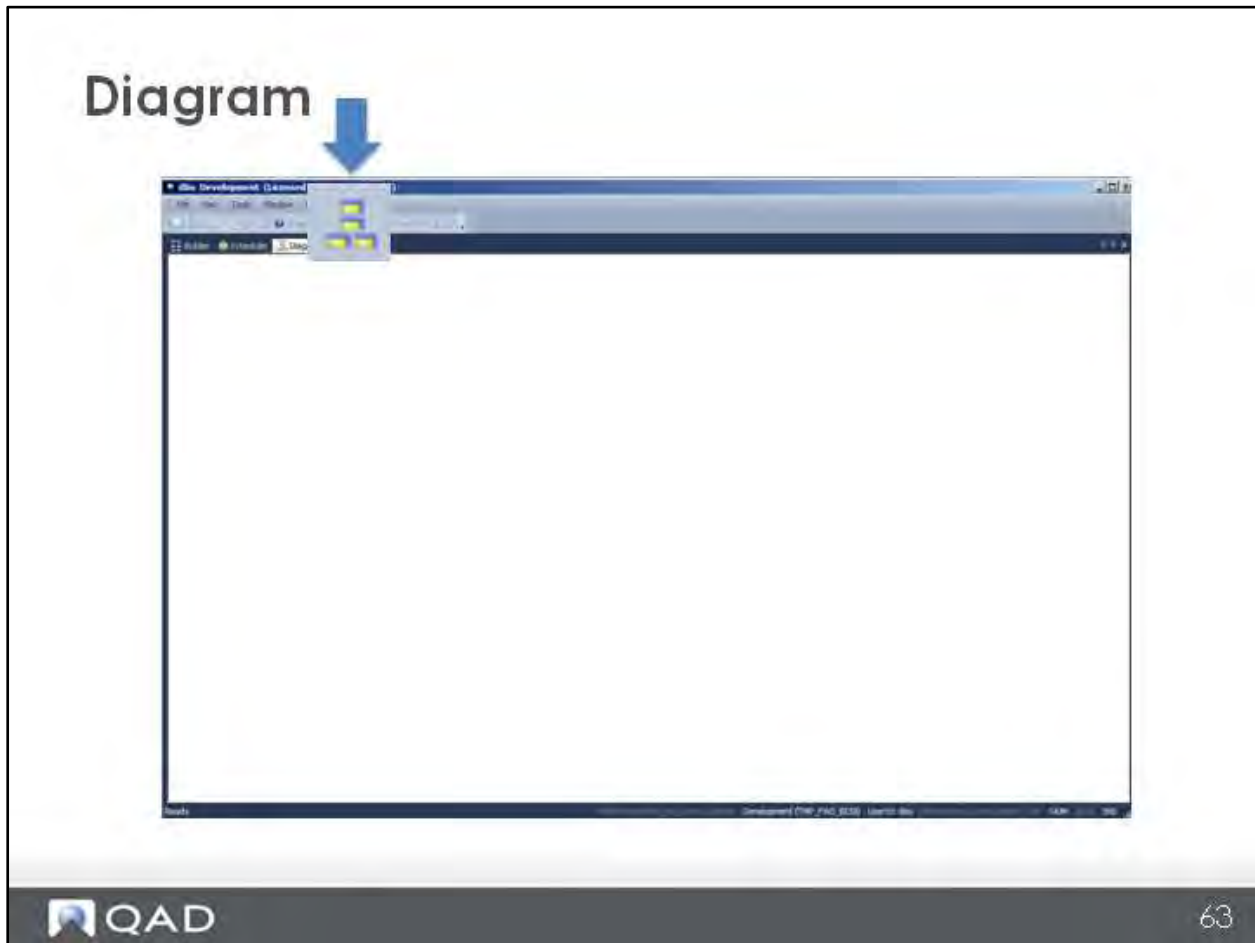


Diagram



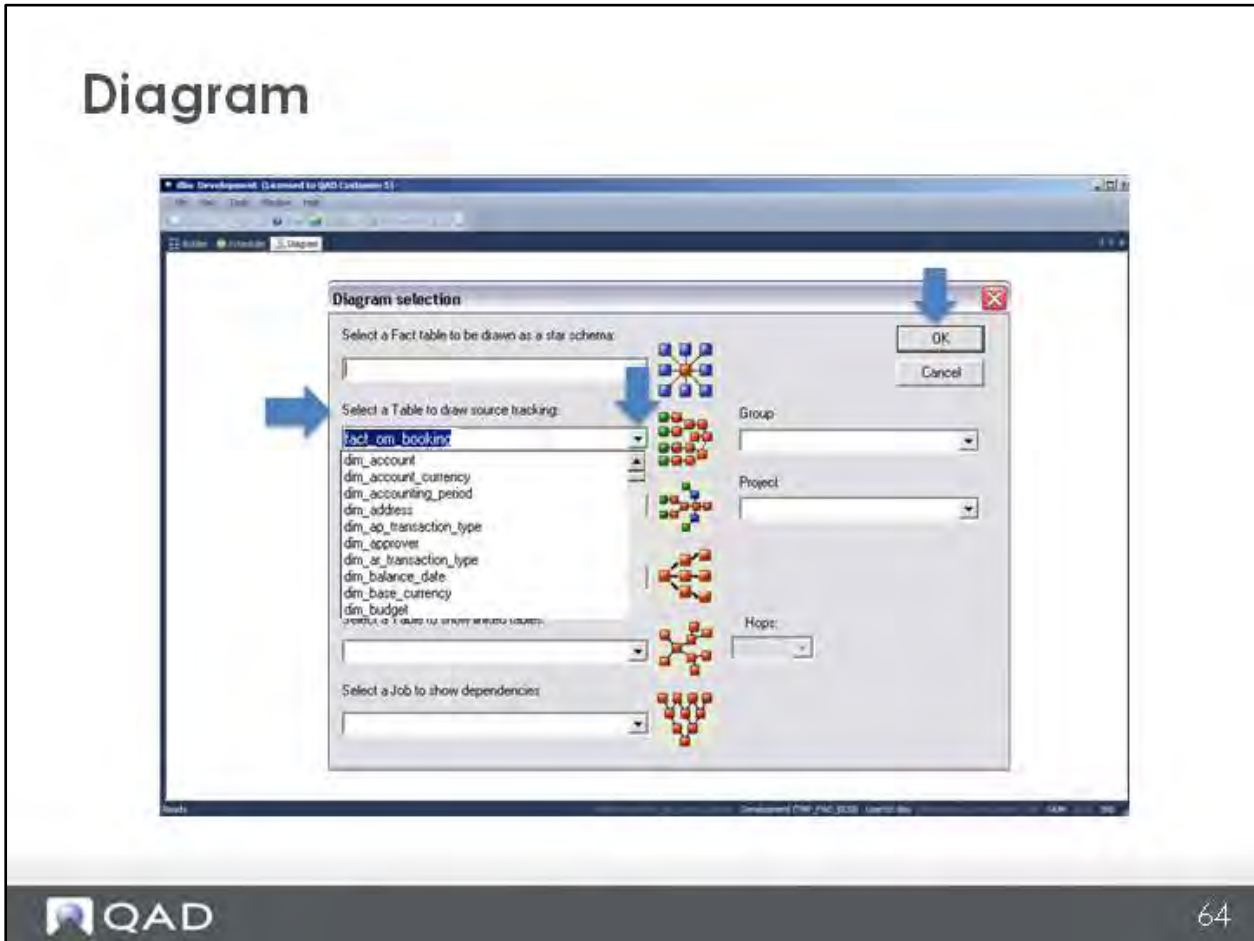
Open the Diagram window by clicking on the Diagram button.

Diagram



When the Diagram window is displayed, click the Diagram button again.

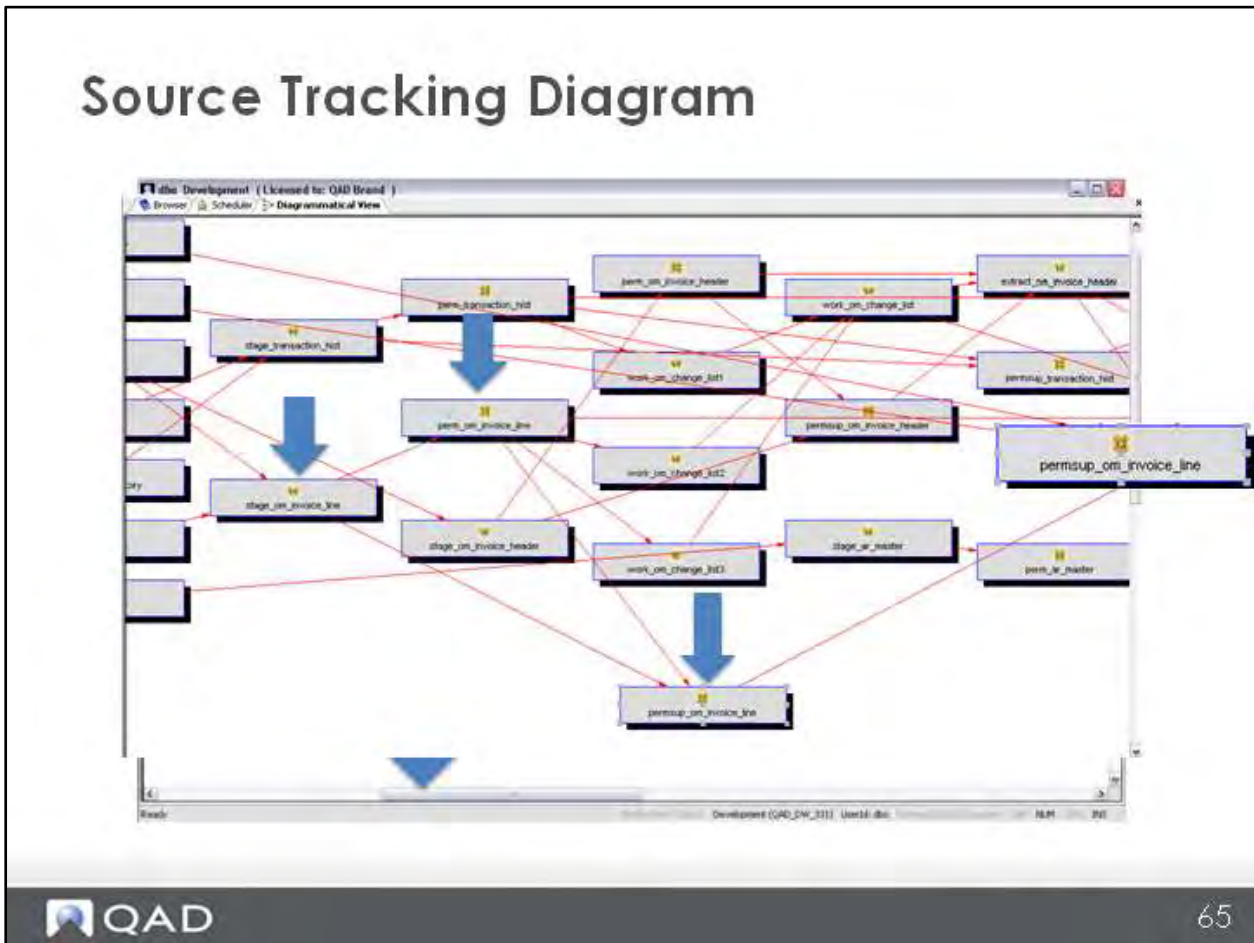
Diagram



The diagram selection window lets the user select the type of diagram and which object to use.

The user **MUST** select the object from the drop down list.

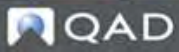
Source Tracking Diagram



The diagram is interactive. The user can click and drag objects within the window.

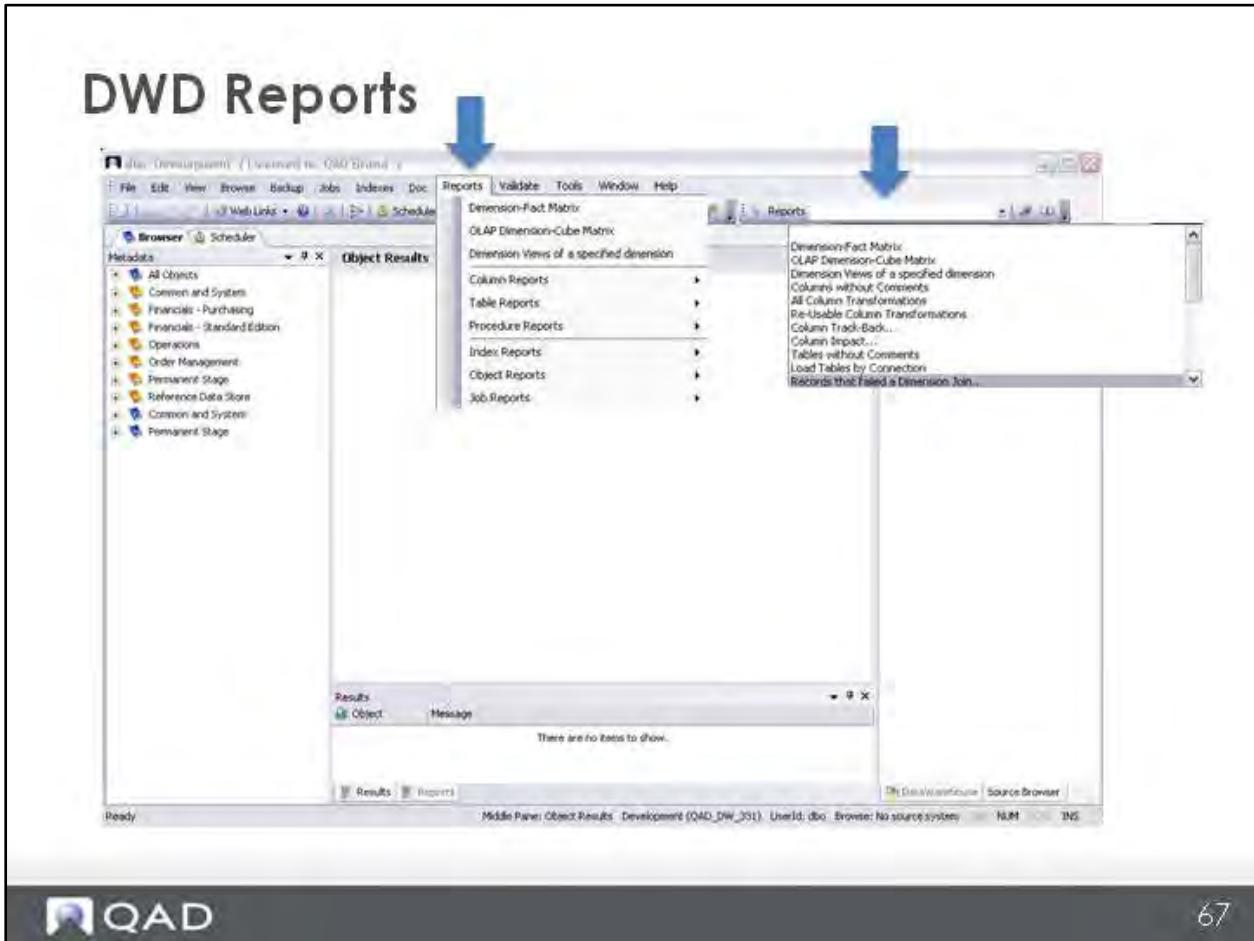
Reports

Reports



Our Passion. Your Advantage.

DWD Reports



Reports can be run from a drop down menu available on the main Builder/Browser window.

A subset of all the reports is available in the “fast access” bar in the window.

Fact to Dimension Matrix

Fact to Dimension Matrix

Dimension cross reference

| Dimension | fact_ap_bill | fact_ap_invoice | fact_ap_vendor | fact_ap_credit | fact_ap_reversal | fact_ap_payment | fact_ap_bill | fact_ap_bill | fact_ap_bill |
|---------------------------|--------------|-----------------|----------------|----------------|------------------|-----------------|--------------|--------------|--------------|
| dim_account | | | | | | | | | |
| dim_account_category | | | | | | | | | |
| dim_address | | | | | | | | | |
| dim_ap_transaction_type | | | | | | | | | |
| dim_applier | | | | | | | | | |
| dim_ap_transaction_type | | | | | | | | | |
| dim_balance_date | | | | | | | | | |
| dim_base_currency | | | | | | | | | |
| dim_budget | | | | | | | | | |
| dim_base | | | | | | | | | |
| dim_channel | | | | | | | | | |
| dim_date_date | | | | | | | | | |
| dim_date_date | | | | | | | | | |
| dim_confirmed_date | | | | | | | | | |
| dim_engagement_date | | | | | | | | | |
| dim_currency_currency | | | | | | | | | |
| dim_cost_center | | | | | | | | | |
| dim_credit_term | | | | | | | | | |
| dim_currency | | | | | | | | | |
| dim_currency_date | | | | | | | | | |
| dim_customer | | | | | | | | | |
| dim_customer_bill_to | | | | | | | | | |
| dim_customer_bill_to | | | | | | | | | |
| dim_customer_bill_to | | | | | | | | | |
| dim_date | | | | | | | | | |
| dim_date_per | | | | | | | | | |
| dim_discount_account | | | | | | | | | |
| dim_discount_bill_to | | | | | | | | | |
| dim_discount_date | | | | | | | | | |
| dim_discount_bill_account | | | | | | | | | |
| dim_bill_discount_date | | | | | | | | | |
| dim_bill_discount_date | | | | | | | | | |

This is an example of a report – the *Fact to Dimension Matrix*.

Export Report To Excel

Export Report To Excel

The screenshot shows a Microsoft Excel spreadsheet with a table of dimensions and fact tables. The table has columns labeled with dimension names and fact table names. The data is as follows:

| Dimension | fact_ap | fact_ap_invfact_ap | fact_ap_invfact_ar | fact_ar | fact_ar_invfact_ar | fact_ar_invfact_in | fact_gen | fact_gi | fact_gi_invfact_gi | fact_inv | fact_inv_invfact_inv |
|-------------------------|---------|--------------------|--------------------|---------|--------------------|--------------------|----------|---------|--------------------|----------|----------------------|
| dim_account | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| dim_account_currency | | | | | | | 1 | 1 | 1 | 1 | 1 |
| dim_address | | | | | | | | | | | |
| dim_ap_inv | 1 | 1 | 1 | 1 | | | | | | | |
| dim_approver | | | | | | | | | | | |
| dim_ar_transaction_type | | | | | 1 | 1 | 1 | 1 | | | |
| dim_balance_date | | | | | | | 1 | 1 | 1 | | |
| dim_base | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| dim_budget | | | | | | | 1 | 1 | 1 | 1 | 1 |
| dim_buyer | | | | | | | | | | | |
| dim_channel | | | | | | | | | | | |
| dim_clear | 1 | | | | | | | | | | |
| dim_close_date | | | | | | | | | | | |
| dim_confirmed_date | | | | | | | | | | | |
| dim_consignment_type | | | | | | | | | | | 1 |
| dim_corp | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| dim_cost | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| dim_credit_term | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| dim_currency | | | | | | | | | | | |
| dim_current_date | | | | | | | | | | | 1 |
| dim_customer | | | | | | | | | | 1 | |
| dim_customer_bill_to | | | | | 1 | 1 | 1 | 1 | | | |
| dim_customer_ship_to | | | | | 1 | 1 | 1 | 1 | | | |
| dim_customer_sold_to | | | | | 1 | 1 | 1 | 1 | | | |
| dim_date | | | | | | | | | | | |
| dim_date_qad | | | | | | | | | | | |
| dim_desc | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

The user can export the report to Excel. Right click within the *Results* and select *Excel It*.

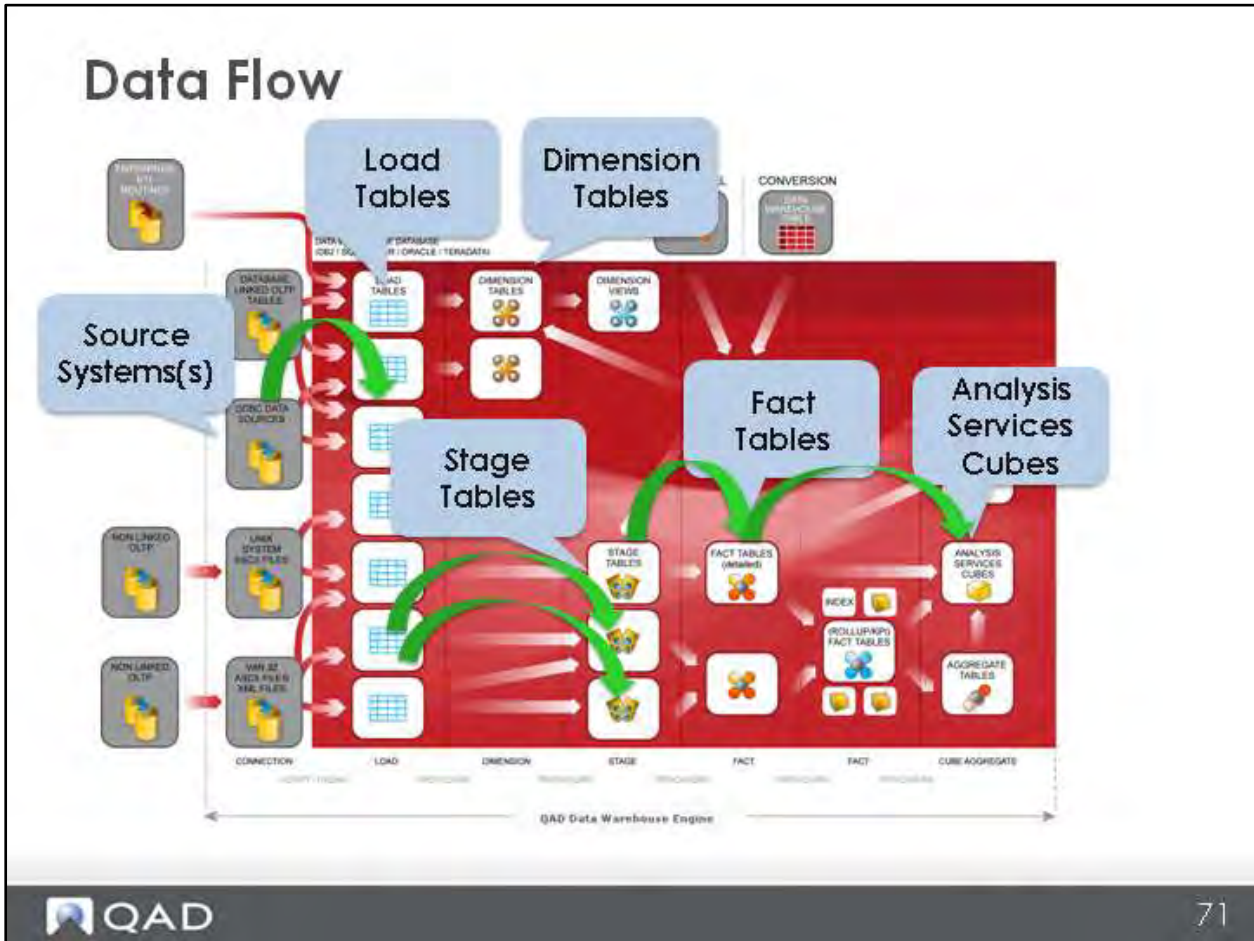
Scheduler Components

Scheduler Components



Our Passion. Your Advantage.

Data Flow

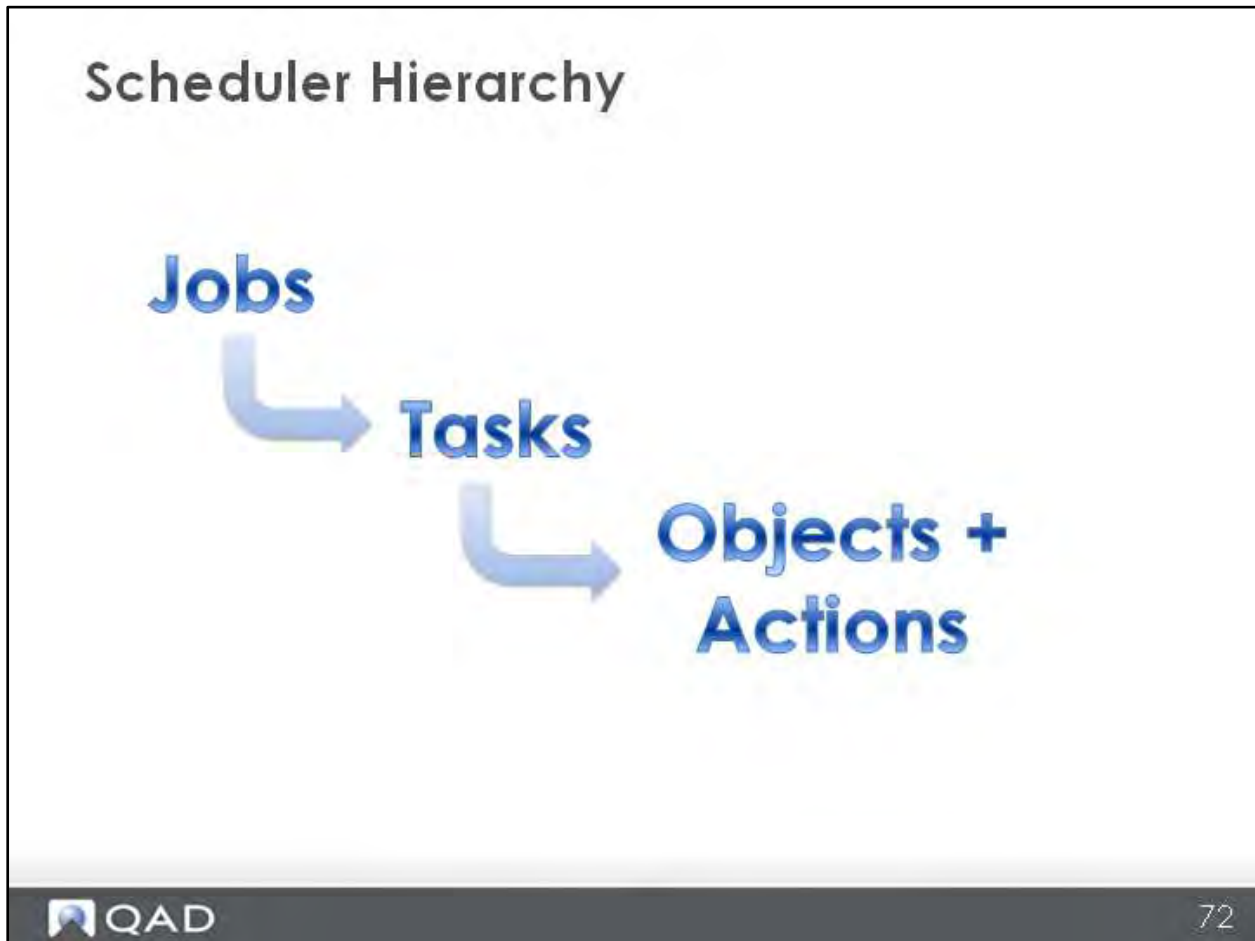


Data objects in the standard data flow include:

- Source systems
- Load tables
- Staging tables
- Dimension tables
- Fact tables

The DWD Scheduler is the mechanism that controls the Extract/Transformation/Load process.

Scheduler Hierarchy



The scheduler component hierarchy:

- Jobs
 - Tasks
 - Objects and Actions associated to the object

Common Task Actions



Common Task Actions

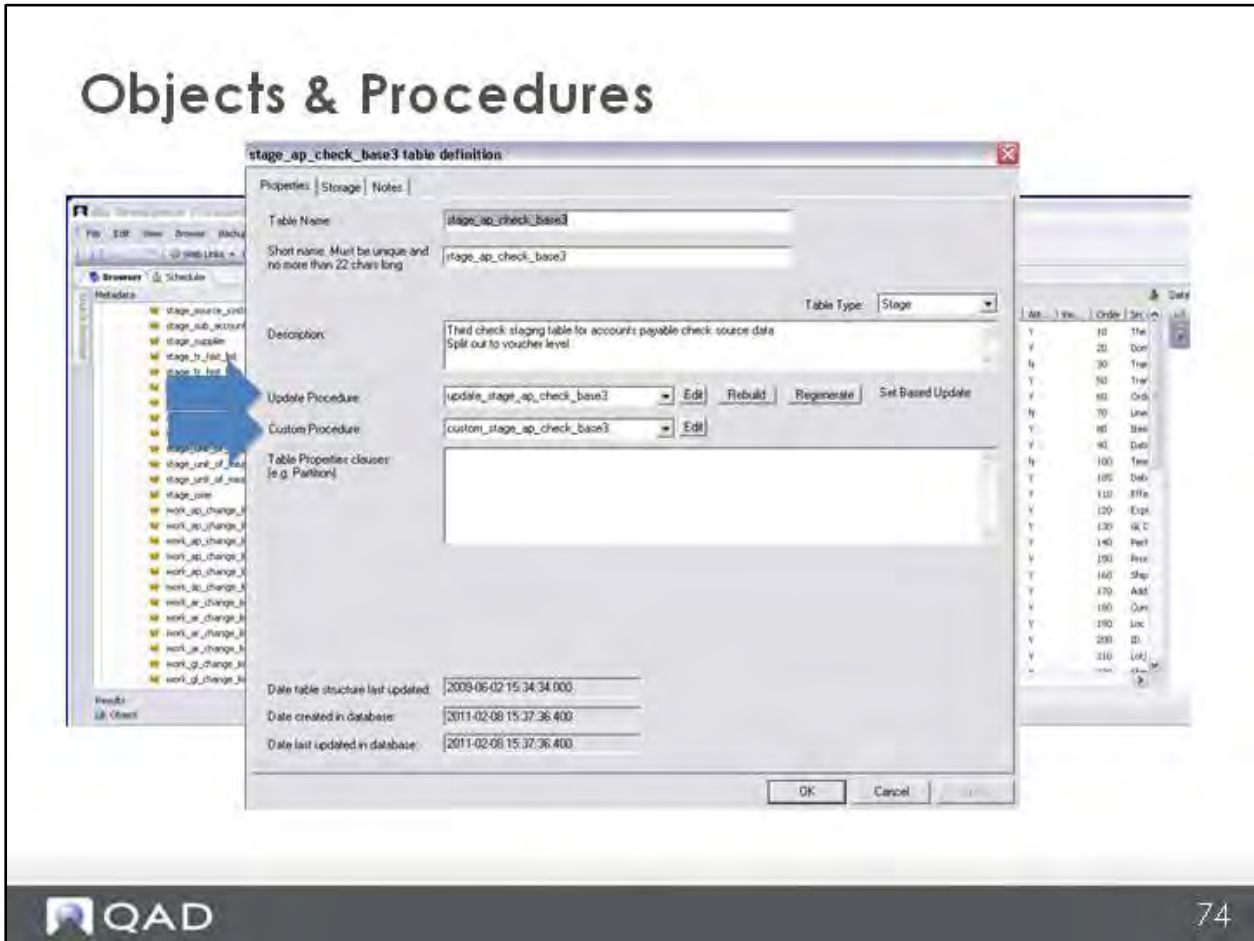
- Drop
- Create
- Truncate
- Load
- Custom
- Update
- Execute
- Process

QAD 73

Common Task Actions include:

- **Drop.** Removes table from SQL data base.
- **Create.** Creates table in SQL data base.
- **Truncate.** Deletes all rows (data) in a SQL table.
- **Load.** Executes standard DWD load procedure and any defined pre and post load procedures.
- **Custom.** Executes custom procedure associated to table/object.
- **Update.** Execute update procedure associated to table/object.
- **Execute.** Execute the procedure or host scripts.
- **Process.** Drops all indexes on the table, performs the update procedure and rebuilds the indexes.

Objects & Procedures



This is the *Properties* page for a staging table in the system. It has two procedures associated to the table/object. One is the standard Update Procedure, the other is a Custom Procedure.

Jobs and Tasks

Jobs and Tasks

Job Name: DAILY_AP_PROCESS_QMI



| Task Name | Action | Order | Depends On |
|-------------------------------|---------|----------|--|
| setp_job_ap_process_start | Execute | 10.10.1 | |
| stage_customer_ar_bot | Process | 11.10.2 | setp_job_ap_process_start (10.10.1) |
| CUSTOMER_PROCESS | Process | 12.10.3 | stage_customer_ar_bot (11.10.2) |
| stage_ap_voucher_base1 | Process | 13.10.4 | CUSTOMER_PROCESS (12.10.3) |
| stage_ap_voucher_base2 | Process | 14.10.5 | stage_ap_voucher_base1 (13.10.4) |
| stage_ap_voucher_base3 | Process | 15.10.6 | stage_ap_voucher_base2 (14.10.5) |
| stage_ap_voucher_base3 | Custom | 16.10.7 | stage_ap_voucher_base3 (15.10.6) |
| stage_ap_check_base1 | Process | 17.10.8 | stage_ap_voucher_base3 (15.10.7) |
| stage_ap_check_base2 | Process | 18.10.9 | stage_ap_check_base1 (17.10.8) |
| stage_ap_vch_chk_match_adj | Process | 19.10.10 | stage_ap_check_base2 (18.10.9) |
| stage_ap_vch_chk_match_vch | Process | 20.10.11 | stage_ap_vch_chk_match_adj (19.10.10) |
| stage_ap_vch_chk_match_vch_cp | Process | 21.10.12 | stage_ap_vch_chk_match_vch (20.10.11) |
| stage_ap_vch_chk_match_net_rl | Process | 22.10.13 | stage_ap_vch_chk_match_vch_cp (21.10.12) |
| stage_ap_vch_chk_match_chk | Process | 23.10.14 | stage_ap_vch_chk_match_net_rl (22.10.13) |
| stage_ap_voucher_chk_match | Custom | 24.10.15 | stage_ap_vch_chk_match_chk (23.10.14) |
| stage_ap_voucher_check_sum | Process | 25.10.16 | stage_ap_voucher_chk_match (24.10.15) |
| stage_ap_check_voucher_sum | Process | 26.10.17 | stage_ap_voucher_check_sum (25.10.16) |
| stage_ap_voucher_base1 | Process | 27.10.18 | stage_ap_check_voucher_sum (26.10.17) |
| stage_ap_check_base3 | Process | | |
| stage_ap_check_base3 | Custom | | |
| fact_ap_check | Process | 33.10.24 | stage_ap_check_base (32.10.23) |
| fact_ap_voucher | Process | 33.10.25 | stage_ap_check_base (32.10.23) |
| stage_ap_voucher_history | Process | 34.10.26 | fact_ap_check (33.10.24) |
| stage_ap_voucher_history | Process | 34.10.26 | fact_ap_voucher (33.10.25) |
| fact_ap_voucher_history | Process | 35.10.27 | stage_ap_voucher_history (34.10.26) |



This is one job in the system.

The object and it's related actions are shown here as tasks.

The object is stage_ap_check_base3 and the actions are *Process* and *Custom*.

Scheduler Navigation

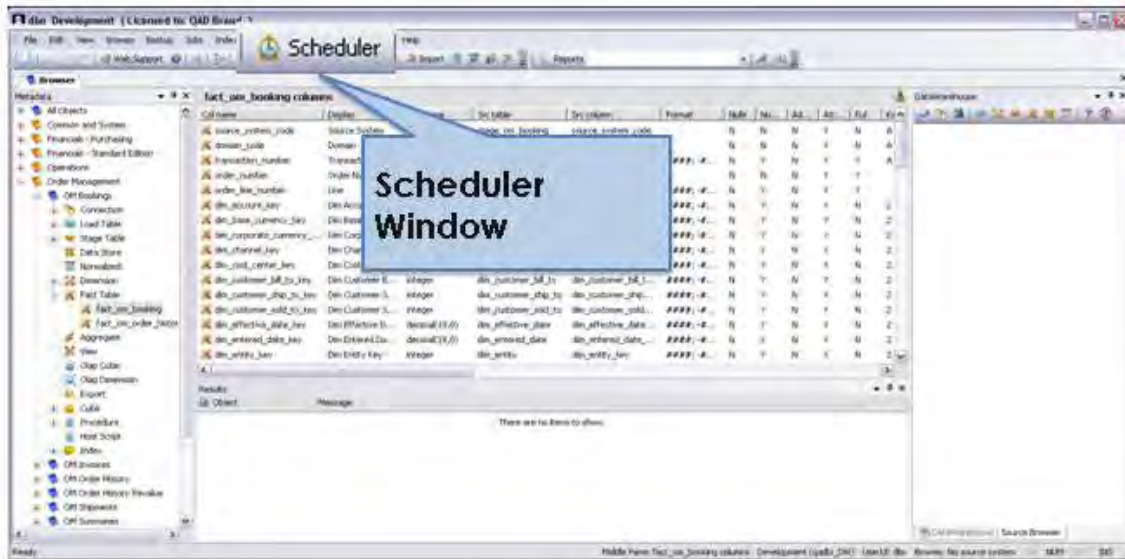
Scheduler Navigation



Our Passion. Your Advantage.

Scheduler Access

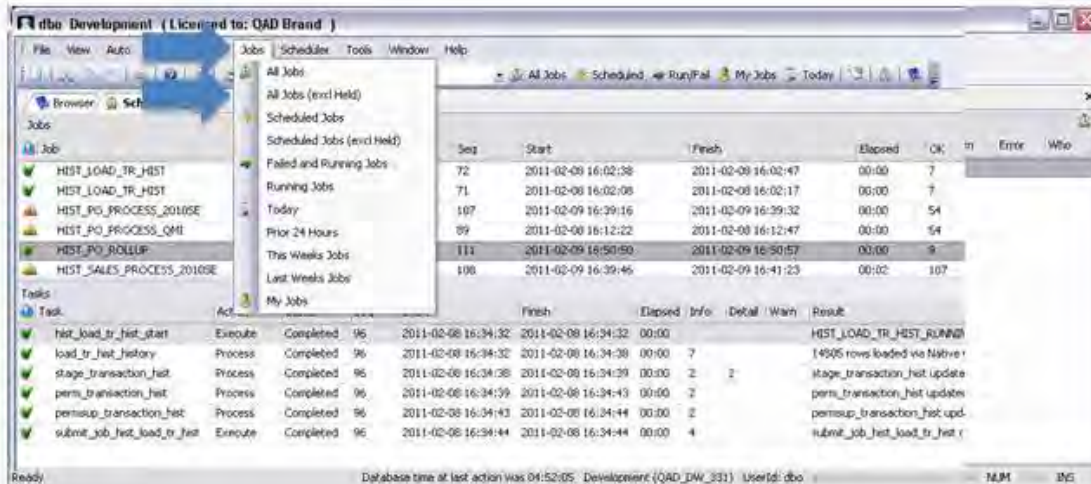
Scheduler Access



The scheduler is accessed via a button on the builder/browser toolbar.

Scheduler Toolbars and Menus

Scheduler Toolbars and Menus



The Scheduler toolbar includes a drop down menu to run various views of the jobs.

Similar functionality is available via buttons on the toolbar. The user can enter a search string, along with wildcards in the box on the toolbar, prior to running one of the options.

For example, to see all jobs with “start” in the name, the user would enter “%start%” in the search window.

Scheduler Panes

Scheduler Panes

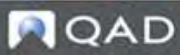
The screenshot shows the 'Scheduler' window with three distinct panes. Blue arrows on the left indicate the location of each pane:

- Jobs Pane:** A table listing various jobs such as 'HST_AP_PROCESS_OPE', 'HST_LOAD_TR_HIST', and 'HST_LOAD_TR_HIST'. Columns include Job, Status, Seq, Start, Finish, Elapsed, OK, Info, Detail, Warn, Error, and Who.
- Tasks Pane:** A table listing actions like 'load_tr_hist_start', 'load_tr_hist_history', 'stage_transaction_hist', 'permux_transaction_hist', and 'advant_job_load_load_tr_hist'. Columns include Task, Action, Status, Seq, Start, Finish, Elapsed, Info, Detail, Warn, and Error.
- Audit and Scheduler Log Pane:** A table showing log entries for 'load_tr_hist_history'. Columns include Seq, Status, Seq, Timestamp, Message, DBMessage, and Job.

In the standard Scheduler window, the Job pane appears at the top. Followed by Tasks, followed by the auditor/scheduler log.

Scheduler States

Scheduler States



Our Passion. Your Advantage.

Scheduler States - JOBS

Scheduler States - JOBS

- Hold
- Waiting
- Blocked
- Pending
- Running
- Failed
- Failed – Aborted
- Completed


81

Scheduler States for Jobs:

- **Hold** . Job is on hold. It can be edited and its state changed in order to release the job.
- **Waiting** . Job is waiting to start, or waiting for it's scheduled time to arrive, or waiting for a scheduler agent to become available.
- **Blocked** . Job is blocked because another instance of the same job is running. When the currently running job completes. This instance of the job will start.
- **Pending** . This is the first stage of a running job. The scheduler has identified the job as ready to start and allocated a thread, or sub task to process the job. A job is in this state until the thread or sub task begins processing. If a job stays in this state, then the scheduler thread has failed for some reason. The logs can be check on which the scheduler is running.
- **Running** . The jobs is currently running. Double click on the job name in the right hand pane to drill down into specific tasks.

- **Failed** . A failed job is one that had a problem. It can be restarted from the point of failure and is considered to be running unless subsequently aborted.
- **Failed – Aborted** . The job has been aborted after having failed. Once in this state, a job cannot be restarted. The job exists only as a log of what occurred and essentially is now not regarded as a job.
- **Completed** . The job has successfully completed, possibly with warnings. Once in this state a job cannot be restarted. The job exists then only as a log of what occurred and essentially is now not regarded as a job.

Scheduler States - TASKS

Scheduler States - TASKS

- Held
- Waiting
- Running
- Failed
- Completed
- Error Completion
- Bad Return Status

QAD 82

Tasks also have a status associated to them:

- **Held.** The task has been held due to a prior dependency failure. The problem must be rectified and the job restarted.
- **Waiting (Blank).** Tasks that are waiting to run either due to a shortage of threads, or prior dependencies normally have a blank status.
- **Running.** The task is currently running.
- **Failed.** The task has had a fatal error. Any dependencies on this task will be held. Double click on the task to see more detail error information or review the audit and error/detail log for the job.
- **Completed .** The task has completed successfully.
- **Error Completion.** The task has completed with a handled error. Any dependent tasks will be held, and the job must be restarted when the problem is rectified.
- **Bad Return Status.** The task has returned and unknown status. This normally occurs with script files that produce unexpected information.

Scheduler Status

Scheduler Status

dbo Development (Licensed to: QAD Brand)

File View Auto Monitor Logs Jobs Scheduler Tools Window Help

Job Name Filter: % All Jobs Scheduled Run/Pal My Jobs

Jobs

| Job | Status | Seq | Start | Finish | Elapsed | OK | Info | Detail | Warn | Error | Who |
|------------------------------|---------|-----|---------------------|--------|---------|----|------|--------|------|-------|-----|
| HIST_AP_PROCESS_00000000 | On Hold | 19 | 2009-06-10 18:25:00 | | | | | | | | |
| DAILY_SALES_PROCESS_00000000 | On Hold | 15 | 2009-06-09 19:15:00 | | | | | | | | |

Tasks

| Task | Action | Status | Seq | Start | Finish | Ela... | Info | D... | Warn | Result |
|------------------------|---------|--------|-----|-------|--------|--------|------|------|------|--------|
| start_daily_load_ch... | Execute | | 17 | | | | | | | |
| submit_next_daily_job | Execute | | 17 | | | | | | | |

Audit and Schedule Log

| Type | Name | Host | Status | Started | Last Status | Stopped | Sample Se... | Version | Message |
|---------|---------|----------|---------|-----------------------|-----------------------|---------|--------------|---------|---------|
| Windows | BIS.3.1 | VWCTRDM5 | Running | 2011-02-12 10:58:3... | 2011-02-15 08:00:5... | | 30 | 6006... | |

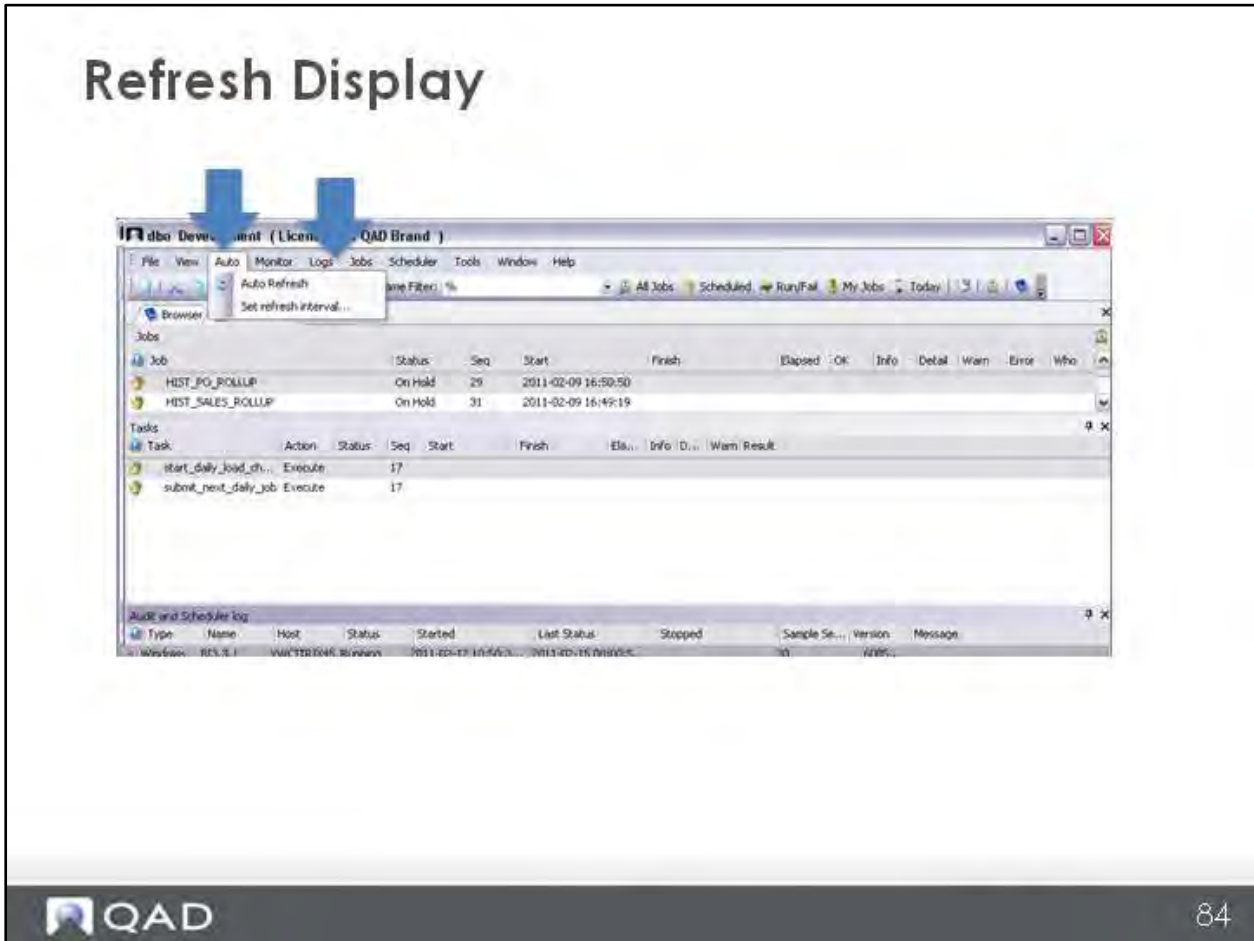
Rejdy Database time of last action was 06:42:35 Development (QAD_DW_331) UserId: dbo NUM INF

The scheduler itself has a status, that is, the overall process that is executing the scheduler functions.

To see the current status of the Scheduler process, select the Scheduler Status icon, which is this little Clock icon on the Browser toolbar.

The current status is shown in the bottom pane.

Refresh Display



If you are monitoring the jobs using the Scheduler Browser, you will want to refresh the screen from time to time.

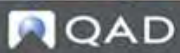
Select the Refresh button for a one-time refresh.

You can also set the screen to refresh automatically

By Selecting Auto on the Menu bar.

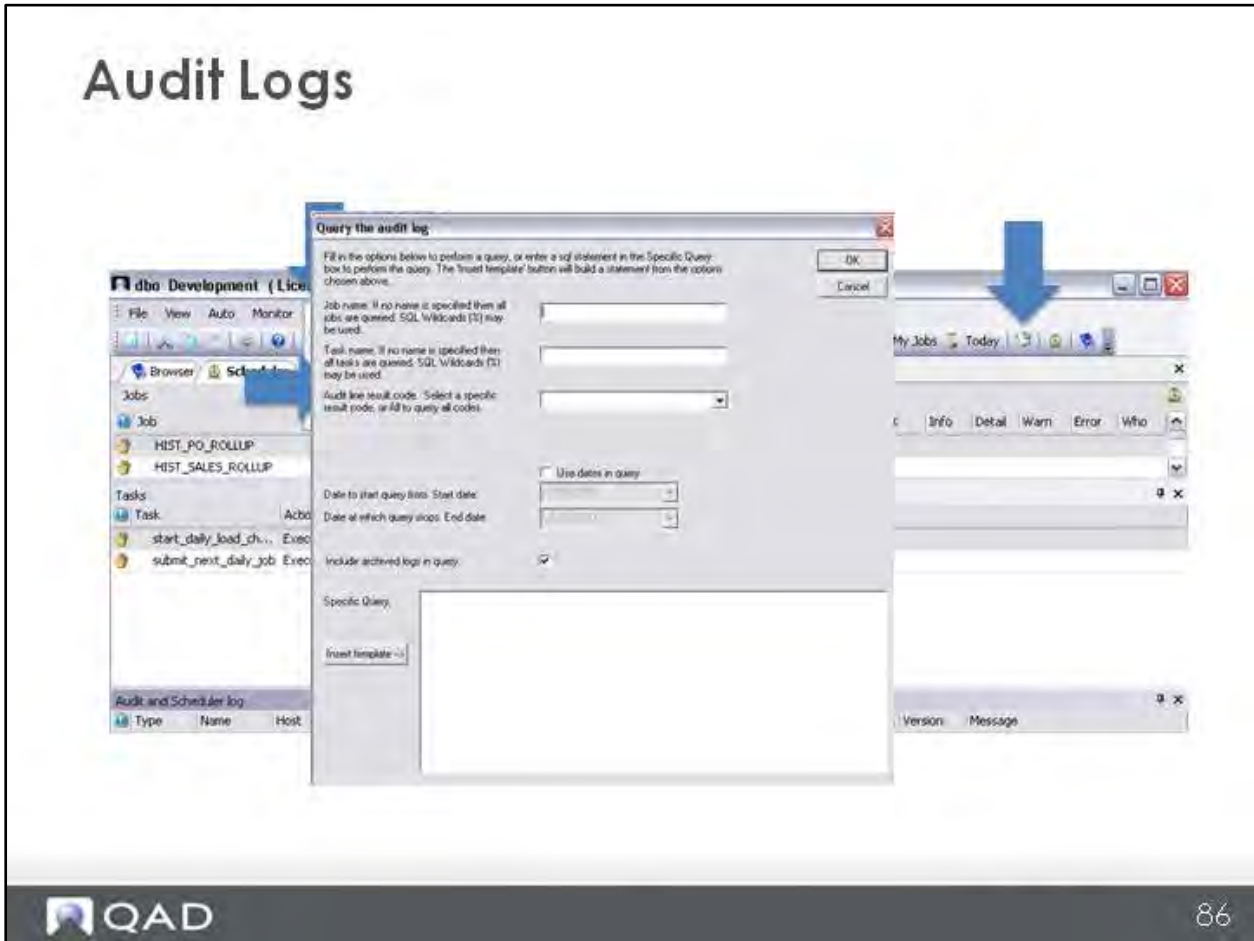
Troubleshooting Scheduler Issues

Troubleshooting Scheduler Issues



Our Passion. Your Advantage.

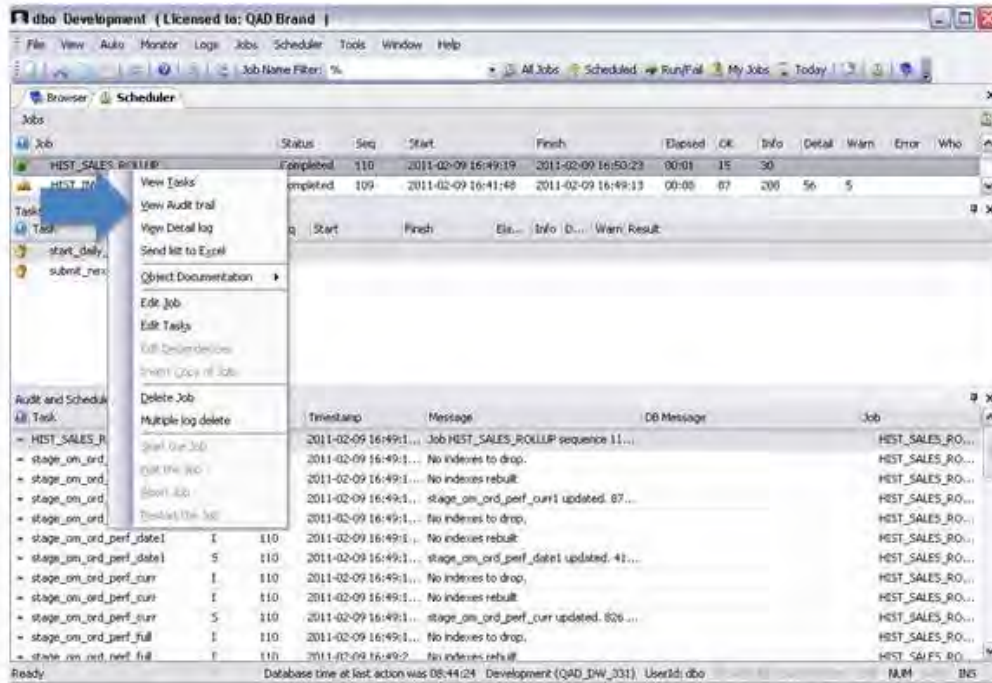
Audit Logs



If a job fails and drilling down does not show any errors against the tasks, right click the job and select “View Audit Trail”. The job may have failed because of an error in the JOB level.

Audit Trail at Job Level

Audit Trail at Job Level



Manually Starting a Job

Manually Starting a Job

The screenshot displays the QAD BI Scheduler interface. The 'Jobs' table is as follows:

| Job | Status | Seq | Start | Finish | Elapsed | OK | Info | Detail | Warn | Error | Who |
|----------------|-----------|-----|---------------------|---------------------|---------|----|------|--------|------|-------|-----|
| HIST_PO_ROLLUP | On Hold | 29 | 2011-02-17 07:33:09 | | | | | | | | |
| HIST_PO_ROLLUP | Completed | 115 | 2011-02-17 07:33:09 | 2011-02-17 07:33:26 | 00:00 | 9 | 14 | | | | |

The 'Tasks' table below shows the internal steps of the job:

| Task | Action | Status | Seq | Start | Finish | Elap... | Info | De... | Warn | Result |
|-------------------------|---------|--------|-----|-------|--------|---------|------|-------|------|--------|
| stage_po_order_snap3 | Process | | | | | | | | | |
| stage_po_order_snap1 | Process | | 29 | | | | | | | |
| stage_po_order_snap... | Process | | 29 | | | | | | | |
| fact_po_order | Process | | 29 | | | | | | | |
| submit_next_history_job | Execute | | 29 | | | | | | | |

88

If during the course of troubleshooting system problems, a job needs to be started, or re-started, the operator would highlight the job, then right click and select *Start the Job*.

DAILY_START

DAILY_START

The screenshot displays the QAD BI Scheduler interface. The main window shows a list of jobs with columns for Job, Status, Seq, Start, Finish, Elapsed, OK, Info, Detail, Warn, Error, and Info. The 'DAILY_START' job is selected and highlighted. Below the job list, the 'Tasks' section shows two tasks: 'start_daily_load_chained' and 'submit_next_daily_job'.

| Job | Status | Seq | Start | Finish | Elapsed | OK | Info | Detail | Warn | Error | Info |
|------------------------------|-----------|-----|---------------------|---------------------|---------|----|------|--------|------|-------|------|
| FINANCIAL_REPORT_GENERATOR | On Hold | 18 | 2009-12-11 13:00:04 | | | | | | | | |
| DAILY_SALES_ROLLUP | On Hold | 16 | 2009-11-11 06:13:00 | | | | | | | | |
| DAILY_START | On Hold | 17 | 2009-10-09 14:29:00 | | | | | | | | |
| DAILY_AP_PROCESS_00000000 | On Hold | 4 | 2009-08-02 14:43:00 | | | | | | | | |
| TEST_AP_PROCESS_00000000 | On Hold | 20 | 2009-07-23 22:10:00 | | | | | | | | |
| CUSTOMER_PROCESS | On Hold | 3 | 2009-07-07 00:43:02 | | | | | | | | |
| DAILY_GL_PROCESS_00000000 | On Hold | 8 | 2009-06-10 19:18:00 | | | | | | | | |
| TEST_GL_PROCESS_00000000 | On Hold | 22 | 2009-06-10 19:18:00 | | | | | | | | |
| TEST_SALES_PROCESS_00000000 | On Hold | 30 | 2009-06-10 19:08:00 | | | | | | | | |
| TEST_AP_PROCESS_00000000 | On Hold | 19 | 2009-06-10 18:20:00 | | | | | | | | |
| DAILY_SALES_PROCESS_00000000 | On Hold | 15 | 2009-06-09 19:15:00 | | | | | | | | |
| DAILY_AP_PROCESS_00000000 | On Hold | 5 | 2009-06-09 17:41:00 | | | | | | | | |
| ITEM_PROCESS | On Hold | 6 | 2009-05-18 23:45:26 | | | | | | | | |
| TEST_PO_ROLLUP | Completed | 145 | 2011-02-17 07:23:09 | 2011-02-17 07:23:26 | 00:00 | 0 | 14 | | | | |
| TEST_ORL | Completed | 114 | 2011-02-15 06:56:23 | 2011-02-15 06:56:25 | 00:00 | 2 | 1 | | | | |
| TEST_PO_ROLLUP | Completed | 111 | 2011-02-09 14:52:50 | 2011-02-09 14:52:57 | 00:00 | 0 | 14 | | | | |
| TEST_SALES_ROLLUP | Completed | 103 | 2011-02-09 16:40:14 | 2011-02-09 16:50:23 | 00:01 | 15 | 30 | | | | |
| TEST_INV_PROCESS_201106 | Completed | 109 | 2011-02-09 16:41:48 | 2011-02-09 16:49:12 | 00:00 | 07 | 208 | 36 | 5 | | |

| Task | Action | Status | Seq | Start | Finish | Elapsed | Info | D... | Warn | Result |
|--------------------------|---------|--------|-----|-------|--------|---------|------|------|------|--------|
| start_daily_load_chained | Execute | 27 | | | | | | | | |
| submit_next_daily_job | Execute | 17 | | | | | | | | |

The job that executes all the daily processing of the QAD BI is called DAILY START.

It contains only two tasks:

Stage_daily_load_chained

Submit_next_daily_job

Both of these tasks are custom procedures. The Stage Daily Load Chained job will kick off the first job in the string of jobs necessary to Extract, Transform and Load the data.

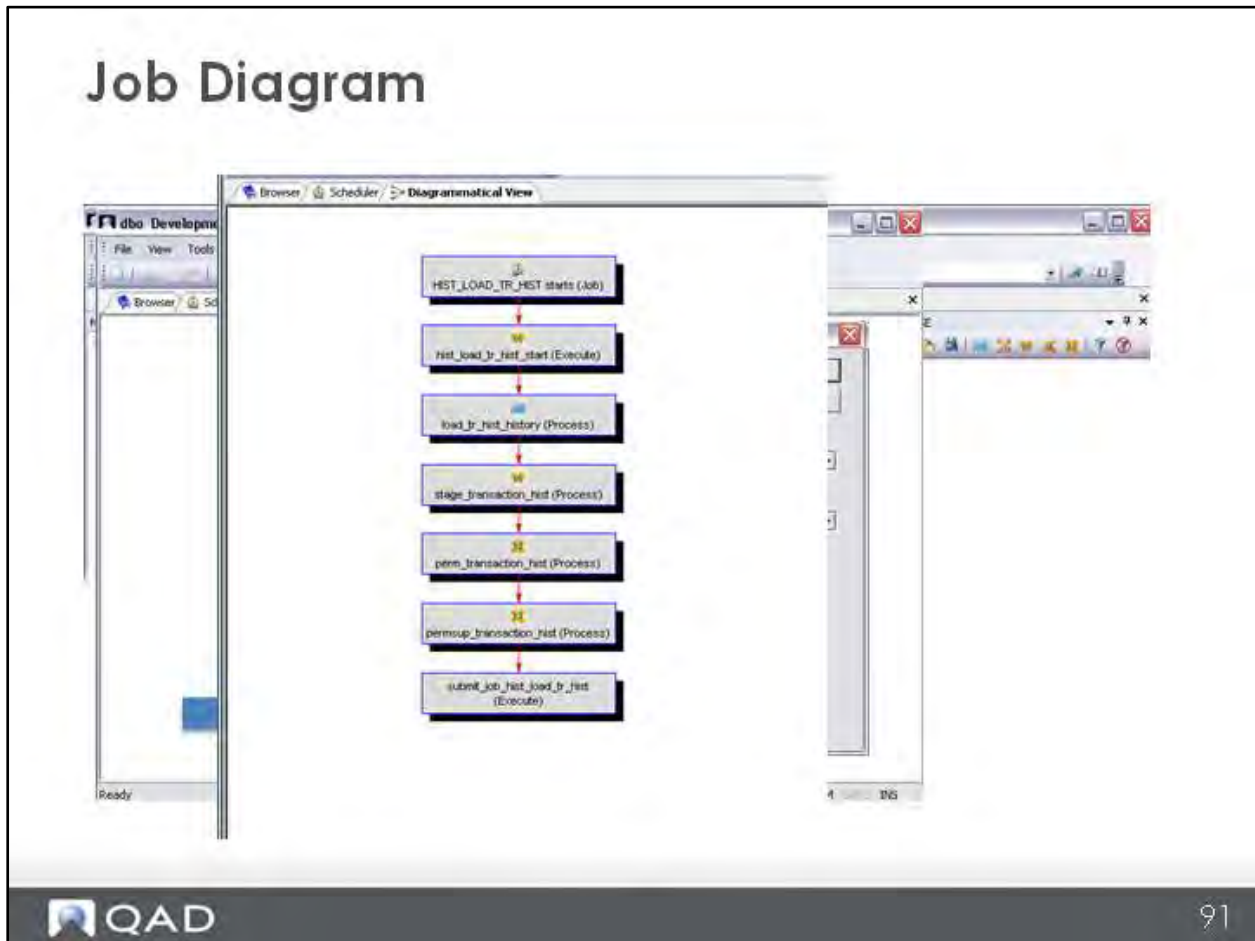
Scheduler Documentation

Scheduler Documentation



Our Passion. Your Advantage.

Job Diagram



From the main Data Warehouse Designer Browser window, elect Diagrammatical View.

When the Diagram window appears, select the Diagram button again.

When the Diagram Selection window appears, one of the options is *Select a Job to show dependencies*

Select the job from the drop down list and click *OK*.

A Dependency diagram will be displayed.

Job Documentation

Job Documentation

HIST_LOAD_TR_HIST

Technical Description of HIST_LOAD_TR_HIST within MASTER_33 QAD BI 3.3 Data Warehouse at 20-Oct-2010 06:44:25

| | |
|----------------------|---------------------|
| Job Frequency: | H24 |
| Job Owner: | Isat Elm- |
| Maximum Threads: | 1 |
| Scheduler Type: | Workflow |
| Description: | Job number 238 |
| Start Date and time: | 2010/10/20 06:44:25 |

Dependencies
 The job HIST_LOAD_TR_HIST is dependent on the following jobs:
 No job dependencies found.

Tasks
 The job HIST_LOAD_TR_HIST contains the following tasks:

| Task Name | Type |
|------------------------------|---------|
| hist_load_tr_hist_start | Execute |
| load_tr_hist_history | Process |
| stage_transaction_hist | Process |
| perm_transaction_hist | Process |
| permisp_transaction_hist | Process |
| submit_job_hist_load_tr_hist | Execute |

Dependencies
 The job HIST_LOAD_TR_HIST contains the following dependencies:

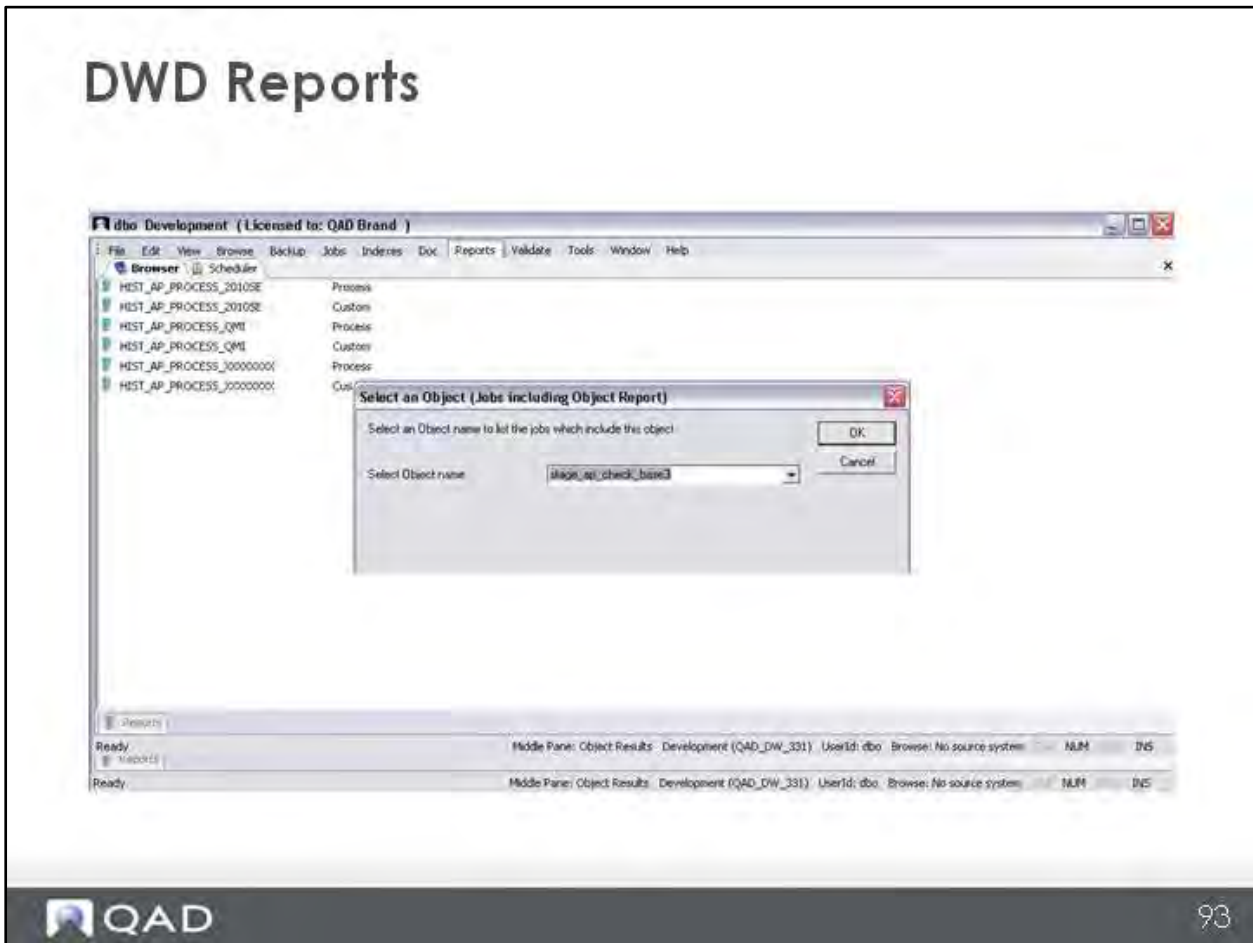
| Parent Task | Parent Action | Child Task | Child Action |
|-------------------------|---------------|--------------------------|--------------|
| hist_load_tr_hist_start | Execute | load_tr_hist_history | Process |
| load_tr_hist_history | Process | stage_transaction_hist | Process |
| stage_transaction_hist | Process | perm_transaction_hist | Process |
| perm_transaction_hist | Process | permisp_transaction_hist | Process |


92

Another way to see Job Documentation is from within the Technical Documentation, key information about a job is displayed.

This includes the Tasks included in the job and the dependencies for that job.

DWD Reports



Scheduler reports are available in the Builder/Browser Reports menu.

From the DWD Builder/Browser menu bar, select *Reports*.

One of the submenu options is *Job Reports*.

Course Review

Course Review



Course Review – DWD Scheduler

Course Review – DWD Scheduler

- Components
 - Jobs
 - Tasks
 - Objects
 - Actions
- Navigation
- States
- Troubleshooting Issues - Auditing
- Documentation

DWD – Features – Data Tools

DWD – Features – Data Tools

- The DWD has several means of seeing how data flows from its source to its final destination tables for the purpose of trouble shooting and figuring out how to best customize systems.
 - Diagram track back views
 - Column track back
 - Search for string

DWD – Features – Diagram Views

DWD – Features – Diagram Views

- The Diagram track back view option is generated from the DWD and is launched by clicking on the Diagram view button and when the blank screen comes up, click on it a second time which brings up the Diagram selection window.



DWD – Features – Diagram Views – Options

DWD – Features – Diagram Views – Options

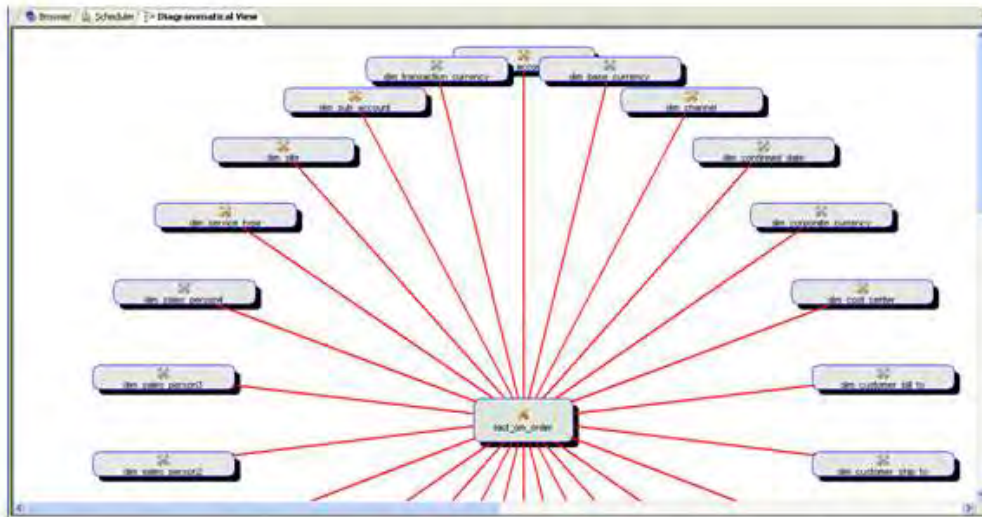
- Fact table star schema
- Two track back reports
- Track forward
- Linked tables
- Job dependencies

Note: You can start to type the name of the table in the drop down menu to find it faster, but must be selected from drop down menu.

DWD – Features – Diagram Views – Star Schema View

DWD – Features – Diagram Views – Star Schema View

- Fact table star schema



DWD – Features – Diagram Views – Source Tracking

DWD – Features – Diagram Views – Source Tracking

- Does NOT display where dim_keys are obtained



DWD – Features – Diagram Views – Track Forward

DWD – Features – Diagram Views – Track Forward

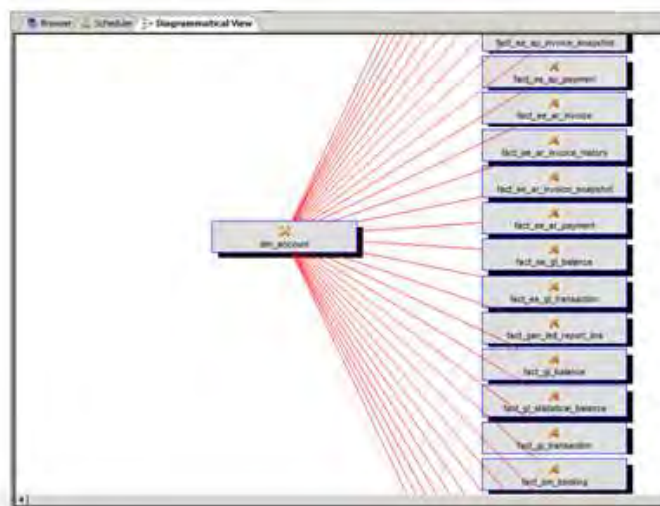
- Displays all the tables fed by selected table



DWD – Features – Diagram Views – Linked Tables

DWD – Features – Diagram Views – Linked Tables

- Displays all the dim (or fact) tables linked to a fact (or dim) table



DWD – Features – Diagram Views – Jobs

DWD – Features – Diagram Views – Jobs

- Displays the order of tasks in a job

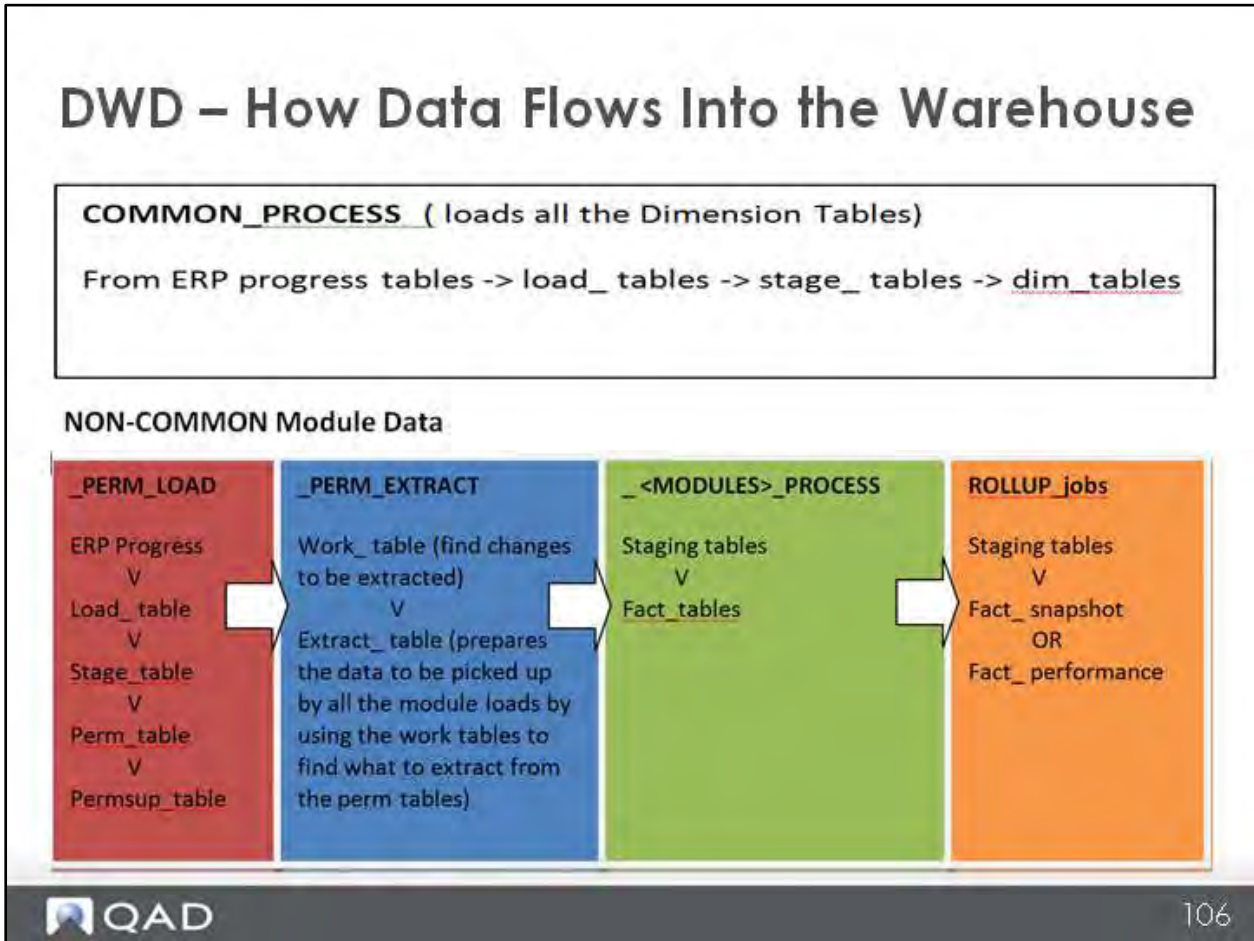


DWD – Features – Diagram Views – Exercises

DWD – Features – Diagram Views – Exercises

- What fact tables does load_so_mstr populate?
- How many dimensions are linked to fact_ee_ap_invoice?
- Is load_tr_hist a source table of fact_om_booking?

DWD – How Data Flows Into the Warehouse



Common Process plays a VERY important role of building the dimension tables which have all the keys, the fact tables will refer to when they are built.

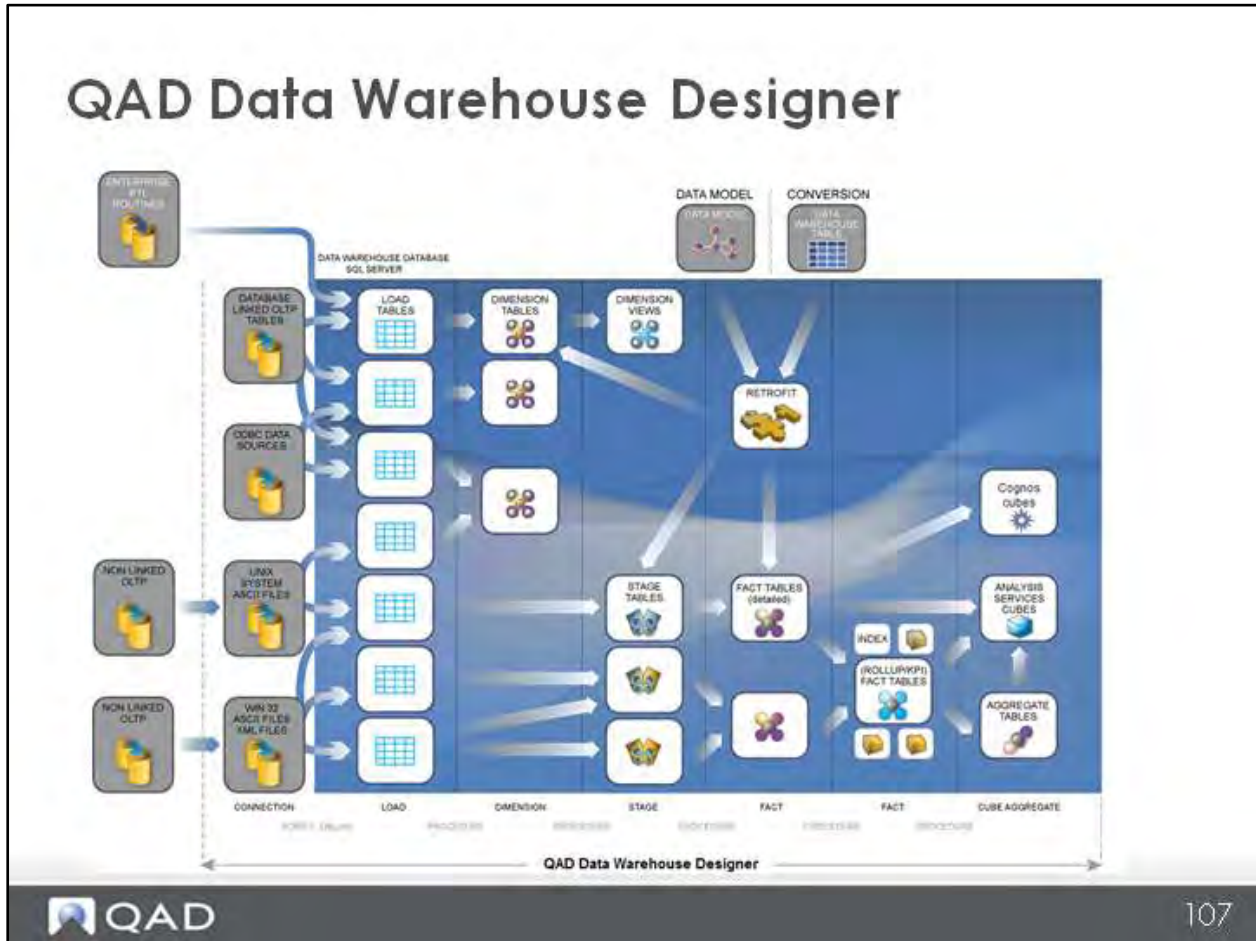
Perm_load loads any tables that store permanent data (large tables that shouldn't change once created).

Perm_extract “primes the pump” for the module loads by looking for changes to perm_tables and preparing the data to be loaded in extract tables.

The various modules processes loads for their respective modules.

The Rollup jobs populate the snapshots and the performance tables.

QAD Data Warehouse Designer

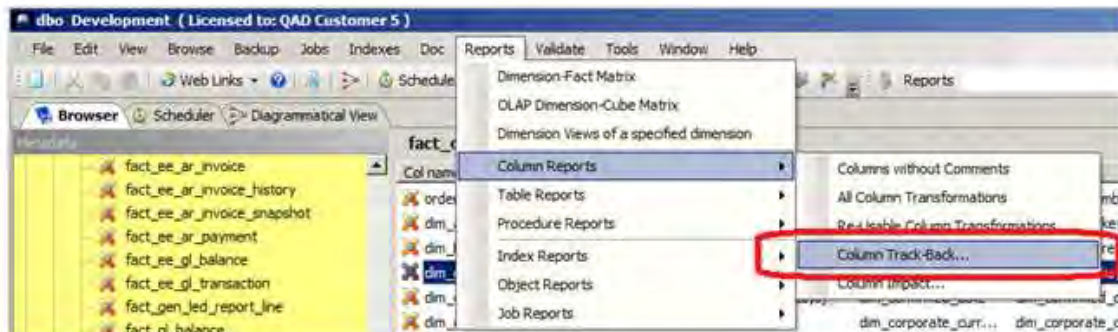


DWD – Features – Column Track Back

DWD – Features – Column Track Back

Very useful reporting option. Displays all tables and columns backwards from the table and column in question to its source.

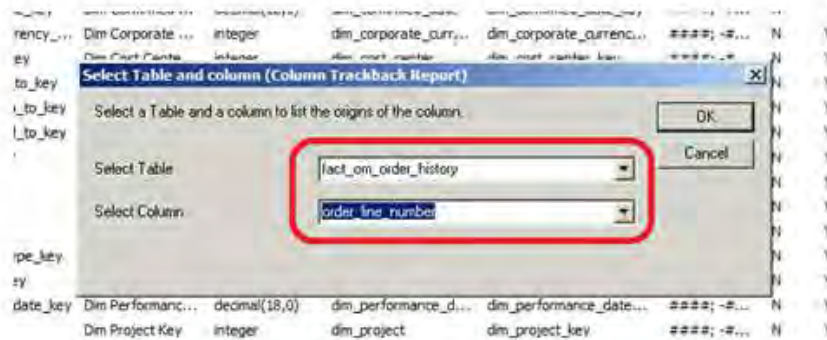
1. From Browser tab, Go to Reports/Column Reports/Column Track-Back



DWD – Features – Column Track Back

DWD – Features – Column Track Back

- From Select Table and column window, pick the Table and the Column to be tracked back on and click OK.



DWD – Features – Column Track Back

DWD – Features – Column Track Back

- In the results pane, each table and column that the data has passed through starting at the table in question

| Report: Column track back | |
|--|-------------------------------------|
| Source table (fact_om_order_history) | Source column (order_line_number) |
| fact_om_order_history | order_line_number |
| stage_om_order | order_line_number |
| stage_om_order3 | order_line_number |
| stage_om_order2 | order_line_number |
| stage_om_order1 | order_line_number |
| stage_om_order_transaction | order_line_number |
| stage_om_order_transaction3 | order_line_number |
| stage_om_order_transaction2 | order_line_number |
| stage_om_order_transaction1 | order_line_number |
| extract_om_transaction_hist | order_line_number |
| perm_transaction_hist | order_line_number |
| stage_transaction_hist | order_line_number |
| load_tr_hist | tr_line |
| tr_hist | tr_line |

DWD – Features – Column Track Back

DWD – Features – Column Track Back

4. However, an attempt to track back on a dimension key, will only bring data back for that dimension.

Select Table and column (Column Trackback Report)

Select a Table and a column to list the origins of the column.

Select Table: fact_om_order_history

Select Column: dim_entity_key

OK Cancel

Report: Column track back

| Source table (fact_om_order_history) | Source column (dim_entity_key) | Transform Code |
|--------------------------------------|--------------------------------|----------------|
| fact_om_order_history | dim_entity_key | |
| dim_entity | dim_entity_key | |

DWD – Features – Column Track Back

DWD – Features – Column Track Back

- To find the source of dimension keys for facts, start at the prior staging table and track back instead on the relevant code

The screenshot shows a dialog box titled "Select Table and column (Column Trackback Report)". The "Select Table" dropdown is set to "stage_om_order" and the "Select Column" dropdown is set to "entity_code". Below the dialog, a report titled "Report: Column track back" is displayed. The report has three columns: "Source table (stage_om_order)", "Source column (entity_code)", and "Transform Code".

| Source table (stage_om_order) | Source column (entity_code) | Transform Code |
|-------------------------------|-----------------------------|---|
| stage_om_order | entity_code | |
| stage_om_order3 | entity_code | |
| stage_om_order2 | entity_code | |
| stage_om_order1 | entity_code | ISNULL(NULLIF(load_si_mstr.si_entity,''),@v_DEFAULT_ENTITY) |
| load_si_mstr | si_entity | REPLACE(REPLACE(REPLACE(si_mstr.si_entity,' ',' '),CHR(10),' '),CHR(13),'') |
| si_mstr | si_entity | |

DWD – Features – Column Track Back - Exercise**DWD – Features – Column Track Back - Exercise**

- What is the source of order_line_number column in table fact_om_order_history?

DWD – Features – Search For String

DWD – Features – Search For String

If a certain column or procedure or function or any type of string is sought in a procedure or in a table, this can easily be achieved using the Search for String option under Tools.

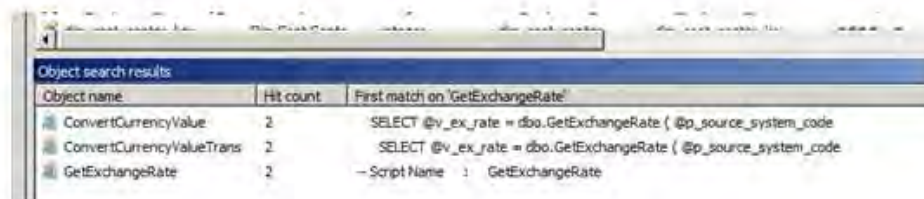
1. Click the check boxes for where the string will be searched.



DWD – Features – Search For String

DWD – Features – Search For String

- The relevant procedures and tables are displayed with the number of times that the string appears in that procedure/table.



The screenshot shows a window titled 'Object search results' with a table containing the following data:

| Object name | Hit count | First match on 'GetExchangeRate' |
|---------------------------|-----------|---|
| ConvertCurrencyValue | 2 | SELECT @v_ex_rate = dbo.GetExchangeRate (@p_source_system_code |
| ConvertCurrencyValueTrans | 2 | SELECT @v_ex_rate = dbo.GetExchangeRate (@p_source_system_code |
| GetExchangeRate | 2 | -- Script Name : GetExchangeRate |

DWD – Features – Column Transformations

DWD – Features – Column Transformations

In many instances, columns are transformed to display some other value than the column being referenced.

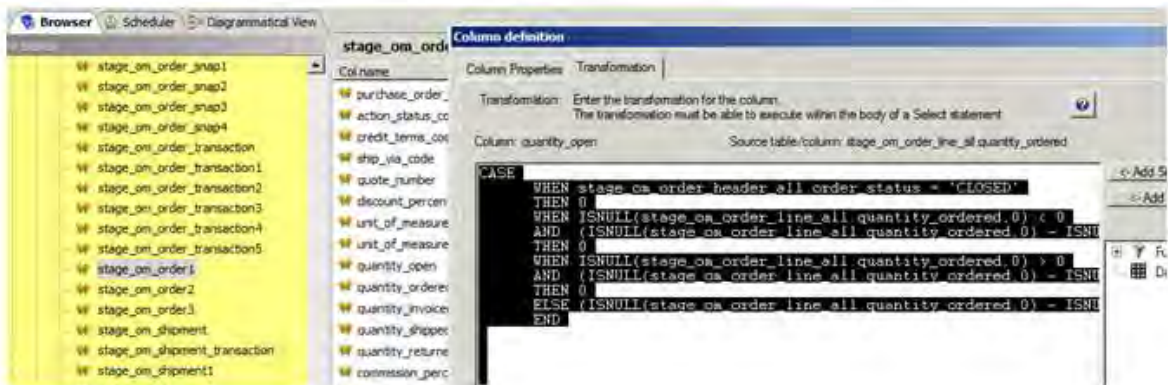
1. In the browser pane, click on a table name to see the columns from the table.

| | | | | | | | | | | | | | |
|----|-------------------------|----------------------------|--------------------|----------------|------------------|----------------------|-------|---|---|---|---|-----|---------------|
| 14 | stage_order_transaction | quote_number | Quote Number | nvarchar(50) | stage_order_j... | quote_number | | Y | N | N | Y | 440 | Quote |
| 15 | stage_order_transaction | discount_percent | Discount Percent | numeric(28,10) | stage_order_j... | discount_percent | 4,440 | Y | Y | Y | N | 450 | Disc % |
| 16 | stage_order_transaction | unit_of_measure_code | Unit Of Measu... | nvarchar(30) | stage_order_j... | unit_of_measure_c... | | Y | N | N | Y | 460 | UM |
| 17 | stage_order_transaction | unit_of_measure_conversion | Unit Of Measu... | numeric(28,10) | stage_order_j... | unit_of_measure_c... | 4,440 | Y | Y | Y | N | 470 | UM Conversion |
| 18 | stage_order | quantity_open | Quantity Open | numeric(28,10) | stage_order_j... | quantity_ordered | 4,440 | Y | Y | Y | N | 480 | Qty Ordered |
| 19 | stage_order | quantity_ordered | Quantity Ordered | numeric(28,10) | stage_order_j... | quantity_ordered | 4,440 | Y | Y | Y | N | 490 | Qty Ordered |
| 20 | stage_order | quantity_invoiced | Quantity Invoiced | numeric(28,10) | stage_order_j... | quantity_invoiced | 4,440 | Y | Y | Y | N | 500 | Qty Invoiced |
| 21 | stage_order | quantity_shipped | Quantity Shipped | numeric(28,10) | stage_order_j... | quantity_shipped | 4,440 | Y | Y | Y | N | 510 | Qty Shipped |
| 22 | stage_order | quantity_returned | Quantity Return... | numeric(28,10) | stage_order_j... | quantity_returned | 4,440 | Y | Y | Y | N | 520 | Qty Returned |

DWD – Features – Column Transformations

DWD – Features – Column Transformations

2. Double click on the column name of interest and when the Column Definition window pops up, switch to the Transformation tab to see, change or create a transformation.



Use stage_om_order1 table and quantity_open column as a good example.

HISTORY TABLES

HISTORY TABLES

- Tracks the history of records over time. Is a base table for other rollup type facts (snapshots and performance).

Example: Record created for 100 widgets ordered due July 1, 2011

| order | ln | dss_hist_start_date | dss_hist_end_date | vers | curr |
|-------|----|---------------------|-------------------|------|------|
| 1000 | 1 | 2011-01-01 | 2999-12-31 | 1 | Y |

HISTORY TABLES

HISTORY TABLES

- Tracks the history of records over time. Is a base table for other rollup type facts (snapshots and performance).

| order | ln | dss_hist_start_date | dss_hist_end_date | vers | curr |
|-------|----|---------------------|-------------------|------|------|
| 1000 | 1 | 2011-01-01 | 2999-12-31 | 1 | Y |

- Customer changes mind on Feb 2. Changes qty ordered 1 50
- Old record end dated set current N, new record created.

| order | ln | dss_hist_start_date | dss_hist_end_date | vers | curr |
|-------|----|---------------------|-------------------|------|------|
| 1000 | 1 | 2011-01-01 | 2011-02-01 | 1 | N |
| 1000 | 1 | 2011-02-02 | 2999-12-31 | 2 | Y |

HISTORY TABLES

HISTORY TABLES

- Tracks the history of records over time. Is a base table for other rollup type facts (snapshots and performance). For example:

| order | In | dss_hist_start_date | dss_hist_end_date | vers | curr |
|-------|----|---------------------|-------------------|------|------|
| 1000 | 1 | 2011-01-01 | 2999-12-31 | 1 | Y |

| order | In | dss_hist_start_date | dss_hist_end_date | vers | curr |
|-------|----|---------------------|-------------------|------|------|
| 1000 | 1 | 2011-01-01 | 2011-02-01 | 1 | N |
| 1000 | 1 | 2011-02-02 | 2999-12-31 | 2 | Y |

| order | In | dss_hist_start_date | dss_hist_end_date | vers | curr |
|-------|----|---------------------|-------------------|------|------|
| 1000 | 1 | 2011-01-01 | 2011-02-01 | 1 | N |
| 1000 | 1 | 2011-02-02 | 2011-04-14 | 2 | N |
| 1000 | 1 | 2011-04-15 | 2999-12-31 | 3 | Y |



The customer then changes their mind again on April 15th. They change the due date of the record. The end date for the prior current record which is set to Y changes to current of N and the end date is set to the day prior to the new records begin date.

SNAPSHOT FROM HISTORY

SNAPSHOT FROM HISTORY

- Creates snapshots of the data from certain points in time (daily, weekly, monthly or both monthly and weekly).

| order | In | dss_hist_start_date | dss_hist_end_date | vers | curr |
|-------|----|---------------------|-------------------|------|------|
| 1000 | 1 | 2011-01-01 | 2011-02-01 | 1 | N |
| 1000 | 1 | 2011-02-02 | 2011-04-14 | 2 | N |
| 1000 | 1 | 2011-04-15 | 2999-12-31 | 3 | Y |

- Snapshot would show.

| Month | order | In | dss_hist_start_date | dss_hist_end_date | vers |
|----------|-------|----|---------------------|-------------------|------|
| January | 1000 | 1 | 2011-01-01 | 2011-02-01 | 1 |
| February | 1000 | 1 | 2011-02-02 | 2011-04-14 | 2 |
| March | 1000 | 1 | 2011-02-02 | 2011-04-14 | 2 |
| April | 1000 | 1 | 2011-04-15 | 2999-12-31 | 3 |
| May | 1000 | 1 | 2011-04-15 | 2999-12-31 | 3 |

DWD – Jobs

DWD – Jobs

- **DAILY_/HIST_START** - Kicks off all the other jobs in the parameter list
- **PREPROCESS** - Truncate and load
- **COMMON** - For load of dimension tables
- **PERM** - For load of data store tables and extract (PERM_EXTRACT) of data in preparation for modules
- **AP, AR, GL, INV, PO, OM, TM, EAM** - Post PERM stage tables leading to FACT tables
- **POSTPROCESS** - ROLLUP and SNAPSHOT jobs

DWD – Jobs and Parameters

DWD – Jobs and Parameters

- The jobs are run from the parameter list – DAILY_LOAD_JOB parameters for the DAILY load.

| Parameter | Value | Comments |
|--------------------|-----------------------------------|--------------|
| DAILY_LOAD_JOB0001 | SET_CONNECTION_QM1 | Inserted via |
| DAILY_LOAD_JOB0002 | PREPROCESS_COMMON_TRUNCATE | Inserted via |
| DAILY_LOAD_JOB0003 | PREPROCESS_OM_TRUNCATE | Inserted via |
| DAILY_LOAD_JOB0004 | PREPROCESS_OP_TRUNCATE | Inserted via |
| DAILY_LOAD_JOB0005 | PREPROCESS_FIN_TRUNCATE | Inserted via |
| DAILY_LOAD_JOB0006 | PREPROCESS_COMMON_LOADS | Inserted via |
| DAILY_LOAD_JOB0007 | PREPROCESS_COMMON_DAILYONLY_LOADS | Inserted via |
| DAILY_LOAD_JOB0008 | PREPROCESS_COMMON_EE2010UP_LOADS | Inserted via |
| DAILY_LOAD_JOB0009 | PREPROCESS_COMMON_EEONLY_LOADS | Inserted via |
| DAILY_LOAD_JOB0010 | PREPROCESS_OM_LOADS | Inserted via |
| DAILY_LOAD_JOB0011 | PREPROCESS_OM_DAILYONLY_LOADS | Inserted via |
| DAILY_LOAD_JOB0012 | PREPROCESS_OP_LOADS | Inserted via |
| DAILY_LOAD_JOB0013 | PREPROCESS_OP_DAILYONLY_LOADS | Inserted via |
| DAILY_LOAD_JOB0014 | PREPROCESS_FIN_LOADS | Inserted via |
| DAILY_LOAD_JOB0015 | PREPROCESS_FIN_EE_LOADS | Inserted via |
| DAILY_LOAD_JOB0016 | DAILY_COMMON_PROCESS_CHAINED | Inserted via |
| DAILY_LOAD_JOB0017 | DAILY_COMMON_PERM_CHAINED | Inserted via |
| DAILY_LOAD_JOB0018 | DAILY_OM_PERM_CHAINED | Inserted via |
| DAILY_LOAD_JOB0019 | DAILY_OM_PROCESS_CHAINED | Inserted via |
| DAILY_LOAD_JOB0020 | DAILY_OP_PERM_CHAINED | Inserted via |

Important Notes – DWD JOBS

Important Notes – DWD JOBS

- When a job fails, in order to find the problem, double click the job to see the tasks and find the task that failed.
- Whenever possible, after fixing a task, **restart the failed job** instead of starting a new job.
- Jobs can control how many concurrent threads are run in the underlying task list. Right click on job, Edit Job, Max Threads.
- A job run inside a job can only have its tasks run consecutively, no matter how many threads are allowed.

Important Notes – DWD JOBS (continued)

Important Notes – DWD JOBS (continued)

- To edit the sequence of a task list, right click on the Job and Edit Tasks. Tasks can be grouped together to run at the same time or not.
- Each Job for a module has a first setp_ task at the beginning that sets a JOB_RUNNING parameter to Y. If a new job starts and another job hasn't finished or has failed so that parameter is still set to Y, the new job will fail on the setp_ task.

Important Notes – Other Key DWD Components

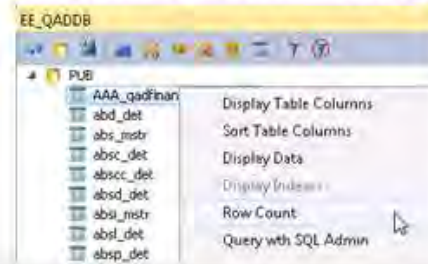
Important Notes – Other Key DWD Components

- To do a quick spot check of data in a Data Warehouse table to see data inside that table, go to the Browser column, right click on the table and click Display Data.
- To do a quick spot check of data in the Data Warehouse or Progress table to see data inside that table, go the Source Browser, right click on the table and click Display Data.

Important Notes – Other Key DWD Components

Important Notes – Other Key DWD Components

- To run a query against Progress or Data Warehouse tables, go to the Source Browser, ensure the connection is against a Progress source, find the table to be queried and right click on it and pick Query with SQL Admin.
- Same logic can work to count rows in table by using Row Count.



Important Notes – Other Key DWD Components

Important Notes – Other Key DWD Components

- Parameter `JOB_CHAINING_ENABLED` allows the chained jobs to run in sequence. If one of the chained jobs needs to be run stand-alone and not launch the next job in the chain, then this parameter needs to be set to N.

BI Portal – Administration



Show Fran's video about BI Portal Administration. <https://share.qad.com/gm/document-1.9.727125/BI-9%20BI%20Portal%20Administrator.wmv>

BI Portal – User



BI Portal – User
(by Fran Shannon)

QAD Business Intelligence
Level 1 Course - 1-3 BI Portal User

Presented by: Fran Shannon

QAD

QAD 130

Show Fran’s video about Portal Users. <https://share.qad.com/gm/document-1.9.788453/BI-3%20BI%20Portal%20User.wmv>

Day 2: In Review

Day 2: In Review

- Troubleshooting Historical Load Issues
- Building and Populating Cubes
- Using Excel to Query the Cubes
- DWD and Scheduler
- BI Administrator
- BI Portal

Parking Lot Review ?

Parking Lot Review ?



CHAPTER 3

QAD BI Technical Level 1 Certification – Part 3

QAD BI v3 - Technical Level 1 Certification - Part 3

QAD Business Intelligence v3 - Technical Level 1 Certification Class - Part 3



Agenda - Day 3

Agenda - Day 3

- Day 2 Review
- Setup and Build the CFO Dashboard
- Create Documentation from the Designer
- Review for Exam

Day 2 Summary

Day 2 Summary

- Troubleshooting Historical Load Issues
- Building and Populating Cubes
- Using Excel to Query the Cubes
- DWD and Scheduler
- BI Administrator
- BI Portal

Building the CFO Dashboard

Building the CFO Dashboard

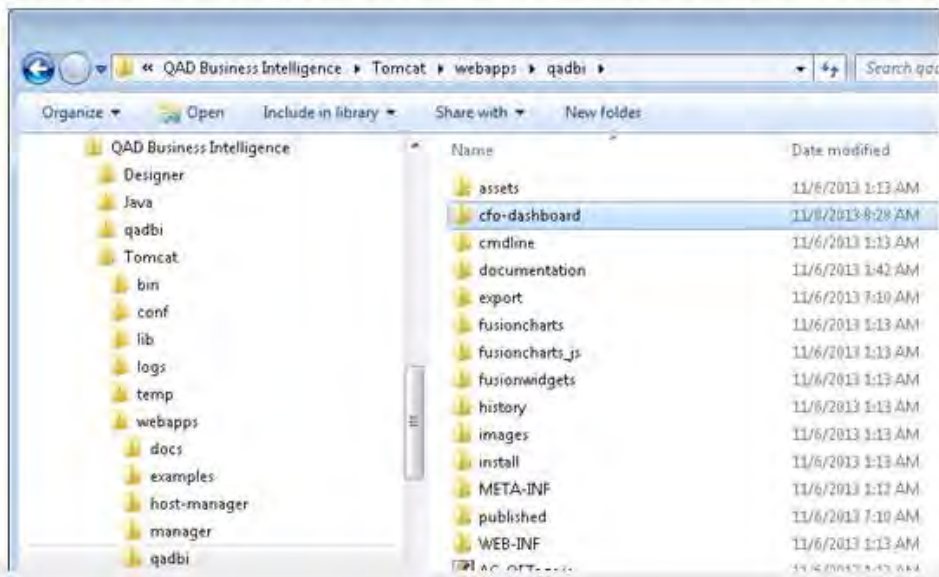
- About the CFO Dashboard
 - The CFO Dashboard is a standard element that ships with the QAD BI Portal.
 - It is designed to collect in one easy-to-view place the important financial metrics for a customer organization so that the financial health of the organization can be quickly understood.

Building the CFO Dashboard

Building the CFO Dashboard

1. Add a *cfo-dashboard* directory in Tomcat

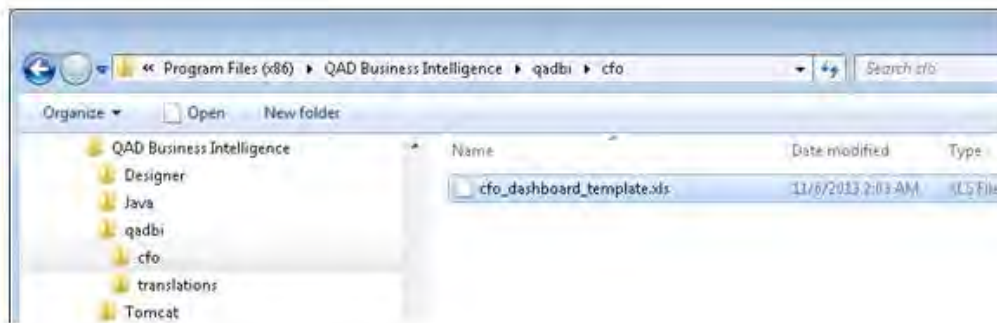
C:\Program Files (x86)\QAD Business Intelligence\Tomcat\webapps\qadbi



Building the CFO Dashboard

Building the CFO Dashboard

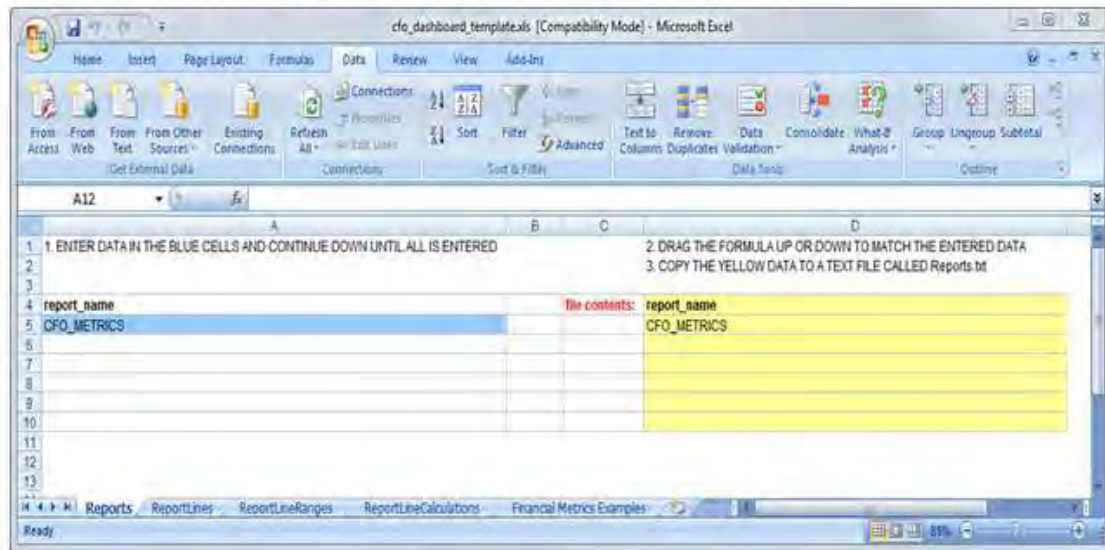
2. Copy the *cfo_dashboard_template.xls* from the installation directory.



Building the CFO Dashboard

Building the CFO Dashboard

3. Open the cfo_dashboard_template.xls
 - Note the worksheets at the bottom

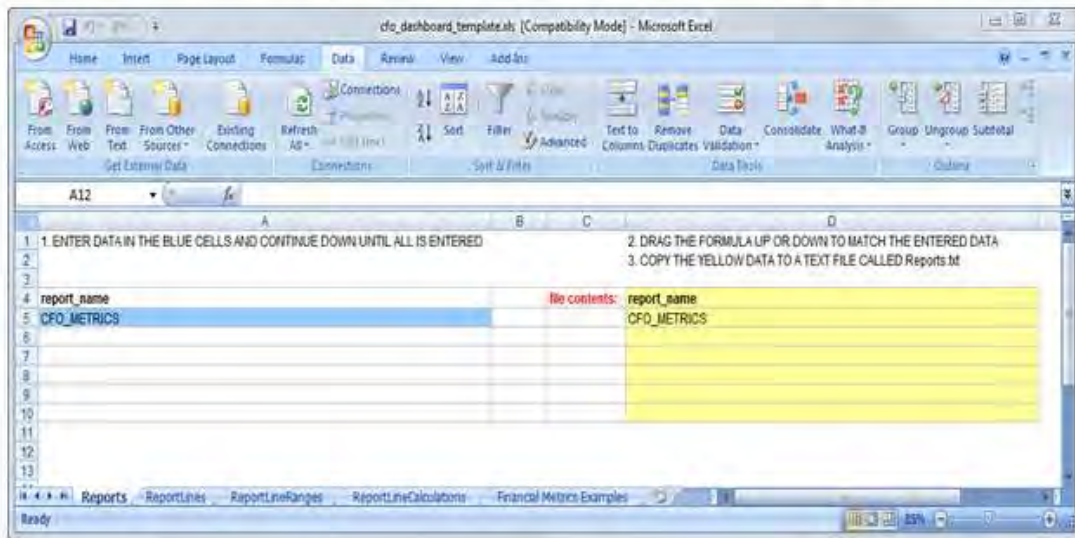


Each worksheet in the template represents a different level of the dashboard.

Building the CFO Dashboard

Building the CFO Dashboard

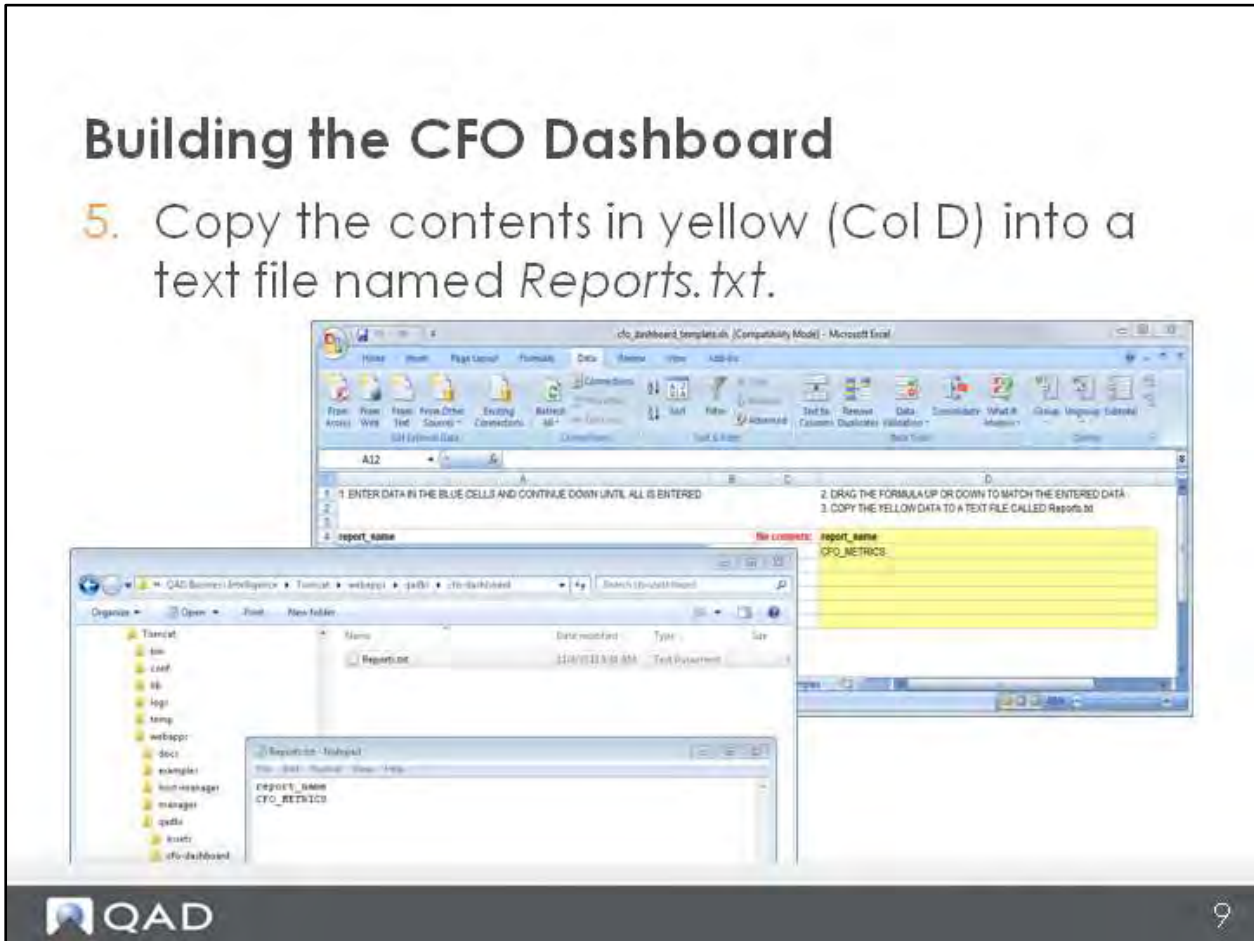
4. Reports worksheet stores the report name. Additional reports can be added in Col A.



Building the CFO Dashboard

Building the CFO Dashboard

5. Copy the contents in yellow (Col D) into a text file named *Reports.txt*.

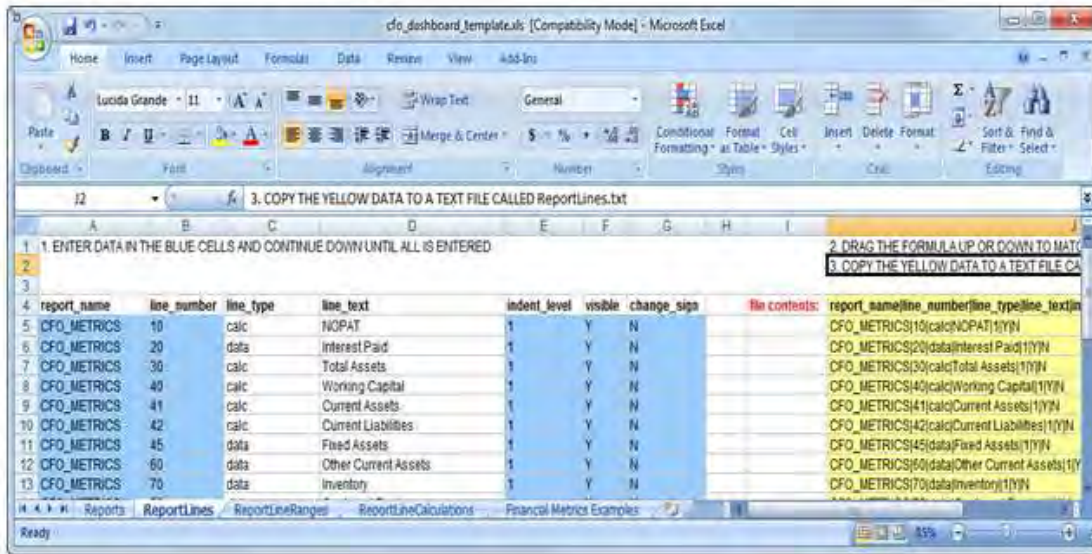


When all reports to be generated are listed, copy all the contents in Yellow starting at D4 down until the last entry in the D column. Paste the contents into notepad and save the file as Report.txt into the cfo-dashboard folder you just created. Make sure to not leave any spaces or carriage returns at the end of the document.

Building the CFO Dashboard

Building the CFO Dashboard

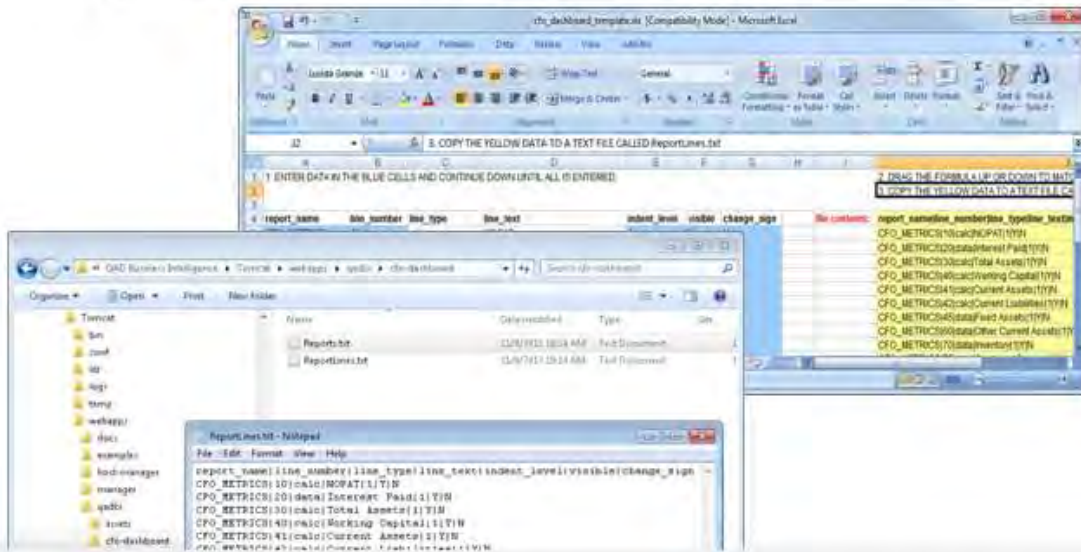
6. *ReportLines* worksheet stores the report line item definition.



Building the CFO Dashboard

Building the CFO Dashboard

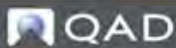
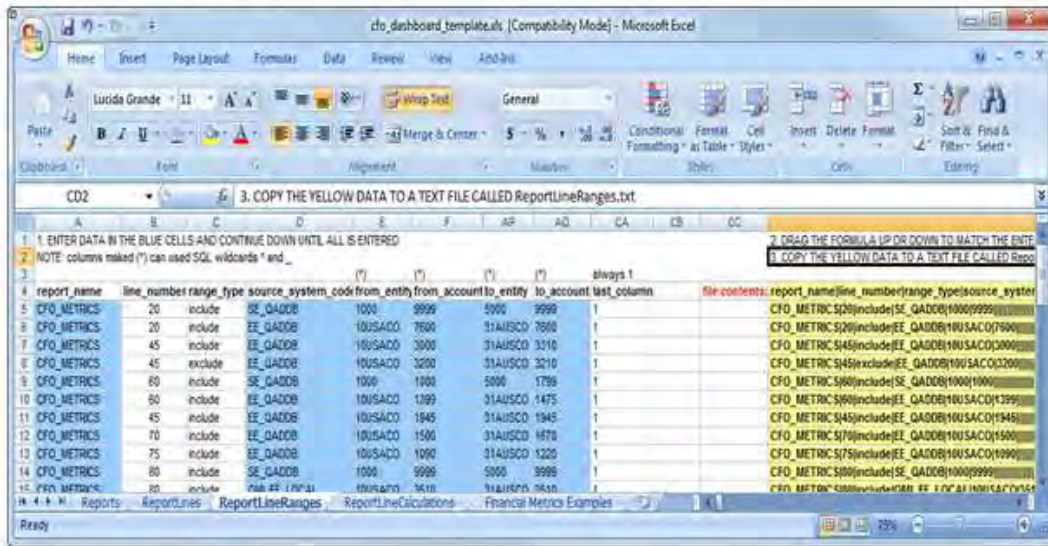
- Copy the contents in yellow (Col J) into a text file named *ReportLines.txt*.



Building the CFO Dashboard

Building the CFO Dashboard

8. *ReportLineRanges* maps Report Lines to accounts in the General Ledger.

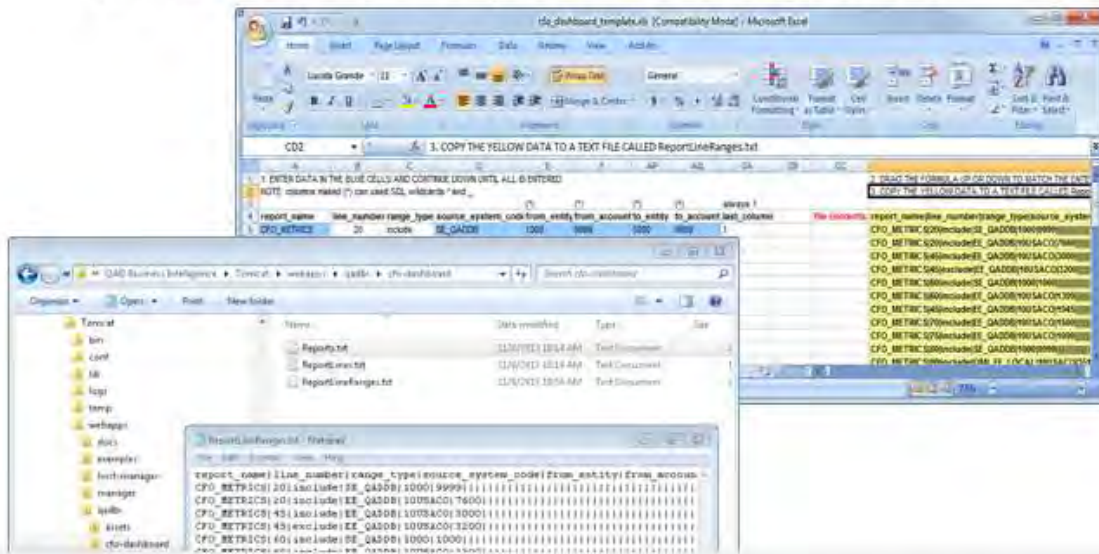


By default, the report values are already listed, but the **source_system_code** and the relevant range values will need to be modified to the correct values. This information should be provided by someone, normally from the Finance Department, who understands their Chart of Accounts.

Building the CFO Dashboard

Building the CFO Dashboard

- Copy the contents in yellow (Col CD) into a text file named *ReportLineRanges.txt*.



Building the CFO Dashboard

Building the CFO Dashboard

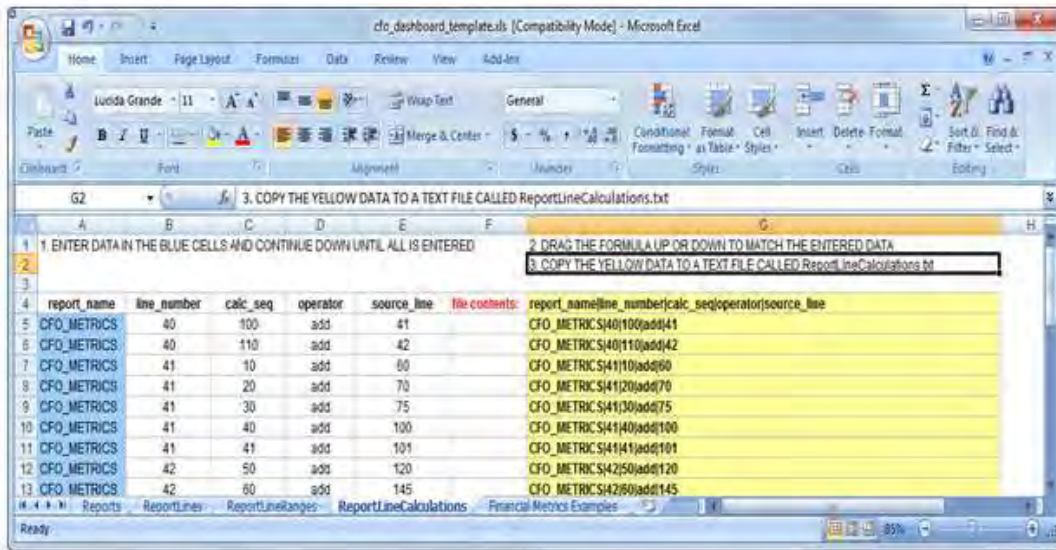
- For the training environment, you can query the Account Dimension and pick out the range of accounts.
- In SSMS, execute the following query

```
SELECT DISTINCT
    source_system_code,
    account_code,
    account_description
FROM [qadbi].[dbo].[dim_account]
WHERE source_system_code = 'SE_QADDB'
-- AND account_description LIKE '%Inventory%'
```

Building the CFO Dashboard

Building the CFO Dashboard

10. *ReportLineCalculations* combine line items to create derived metrics.



Building the CFO Dashboard

Building the CFO Dashboard

1. Copy the contents in yellow (Col G) into a text file named *ReportLineCalculations.txt*.

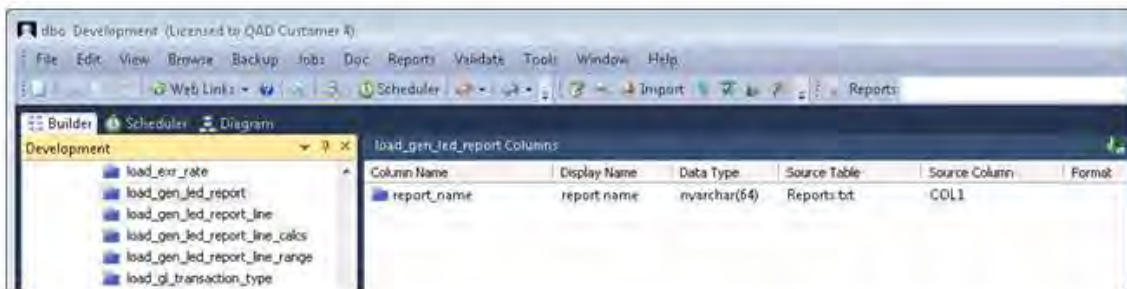
The screenshot shows two overlapping windows. The top window is Microsoft Excel, titled 'cfo_dashboard_template.xls (Compatibility Mode) - Microsoft Excel'. It displays a spreadsheet with columns 'report_name', 'line_number', 'calc_seq', 'operator', and 'source_line'. Row 2 contains the values 'CFO_METRICS', '43', '100', '954', and '41'. Column G is highlighted in yellow and contains a list of report names: 'report_name|line_number|calc_seq|operator|source_line', 'CFO_METRICS|40|100|954|41', 'CFO_METRICS|41|100|954|42', 'CFO_METRICS|42|100|954|43', 'CFO_METRICS|43|100|954|44', 'CFO_METRICS|44|100|954|45', 'CFO_METRICS|45|100|954|46', 'CFO_METRICS|46|100|954|47', 'CFO_METRICS|47|100|954|48', 'CFO_METRICS|48|100|954|49', 'CFO_METRICS|49|100|954|50', 'CFO_METRICS|50|100|954|51', 'CFO_METRICS|51|100|954|52', 'CFO_METRICS|52|100|954|53', 'CFO_METRICS|53|100|954|54', 'CFO_METRICS|54|100|954|55', 'CFO_METRICS|55|100|954|56', 'CFO_METRICS|56|100|954|57', 'CFO_METRICS|57|100|954|58', 'CFO_METRICS|58|100|954|59', 'CFO_METRICS|59|100|954|60', 'CFO_METRICS|60|100|954|61', 'CFO_METRICS|61|100|954|62', 'CFO_METRICS|62|100|954|63', 'CFO_METRICS|63|100|954|64', 'CFO_METRICS|64|100|954|65', 'CFO_METRICS|65|100|954|66', 'CFO_METRICS|66|100|954|67', 'CFO_METRICS|67|100|954|68', 'CFO_METRICS|68|100|954|69', 'CFO_METRICS|69|100|954|70', 'CFO_METRICS|70|100|954|71', 'CFO_METRICS|71|100|954|72', 'CFO_METRICS|72|100|954|73', 'CFO_METRICS|73|100|954|74', 'CFO_METRICS|74|100|954|75', 'CFO_METRICS|75|100|954|76', 'CFO_METRICS|76|100|954|77', 'CFO_METRICS|77|100|954|78', 'CFO_METRICS|78|100|954|79', 'CFO_METRICS|79|100|954|80', 'CFO_METRICS|80|100|954|81', 'CFO_METRICS|81|100|954|82', 'CFO_METRICS|82|100|954|83', 'CFO_METRICS|83|100|954|84', 'CFO_METRICS|84|100|954|85', 'CFO_METRICS|85|100|954|86', 'CFO_METRICS|86|100|954|87', 'CFO_METRICS|87|100|954|88', 'CFO_METRICS|88|100|954|89', 'CFO_METRICS|89|100|954|90', 'CFO_METRICS|90|100|954|91', 'CFO_METRICS|91|100|954|92', 'CFO_METRICS|92|100|954|93', 'CFO_METRICS|93|100|954|94', 'CFO_METRICS|94|100|954|95', 'CFO_METRICS|95|100|954|96', 'CFO_METRICS|96|100|954|97', 'CFO_METRICS|97|100|954|98', 'CFO_METRICS|98|100|954|99', 'CFO_METRICS|99|100|954|100'. The bottom window is QAD Business Intelligence, showing a file explorer view of the 'cfo_dashboard' folder. The 'ReportLineCalculations.txt' file is selected, and its contents are displayed in a preview window, matching the yellow-highlighted data in the Excel spreadsheet.

Building the CFO Dashboard

Building the CFO Dashboard

12. In DWD, map the text files to the load tables

- load_gen_led_report
- load_gen_led_report_line
- load_gen_led_report_line_calcs
- load_gen_led_report_line_range

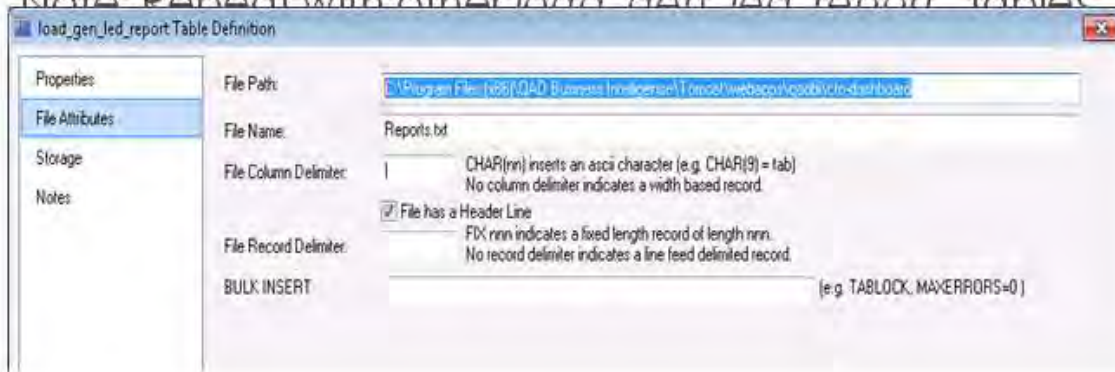


Building the CFO Dashboard

Building the CFO Dashboard

13. Open the *Properties* for the load table, click on *File Attributes* tab and update the *File Path*.

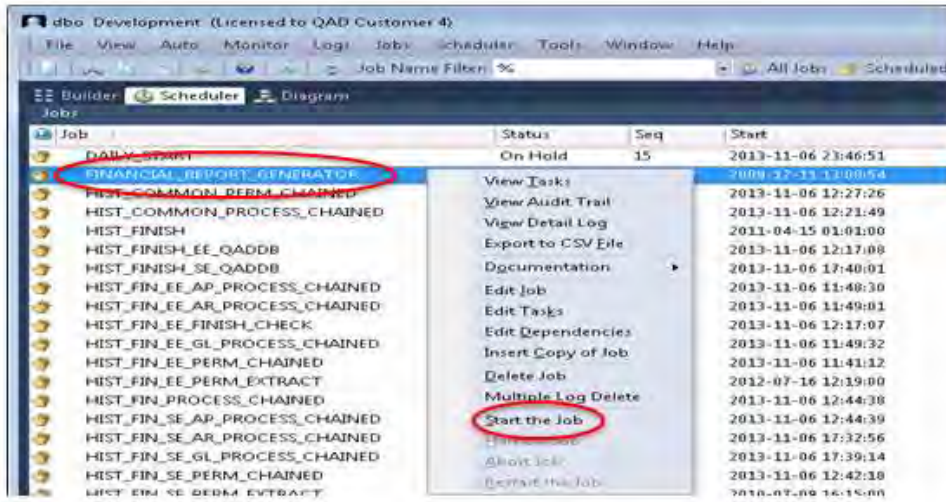
Note: Repeat with other load gen led report tables.



Building the CFO Dashboard

Building the CFO Dashboard

14. In DWD Scheduler, run the job FINANCIAL_REPORT_GENERATOR



Building the CFO Dashboard

Building the CFO Dashboard

15. Data can be reviewed using the dashboards and reports in the BI Portal.

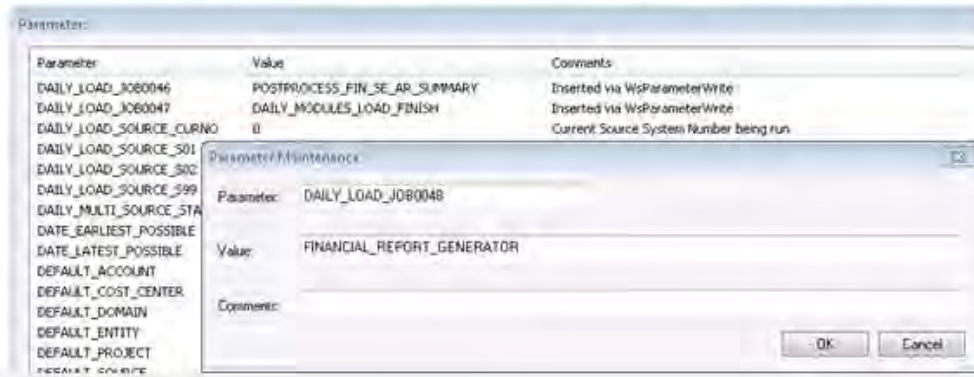


Building the CFO Dashboard

Building the CFO Dashboard

1.6. Schedule FINANCIAL_REPORT_GENERATOR

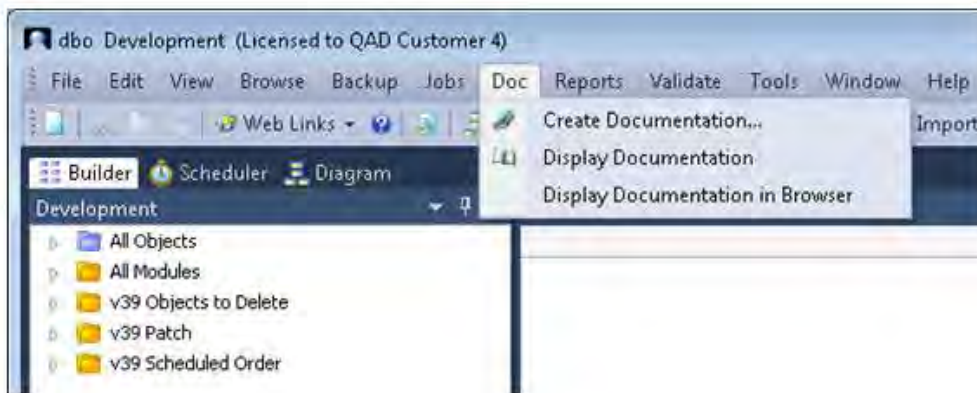
- Option 1: Setup stand-alone schedule
 - Use *Edit Job* to specify *Frequency*, *Start Date & Time*
- Option 2: Add to DAILY_LOAD_JOB chain
 - Add a new parameter - DAILY_LOAD_JOB:xxxx incremented by 1 from the last entry in the job list



DWD – Creating Documentation

DWD – Creating Documentation

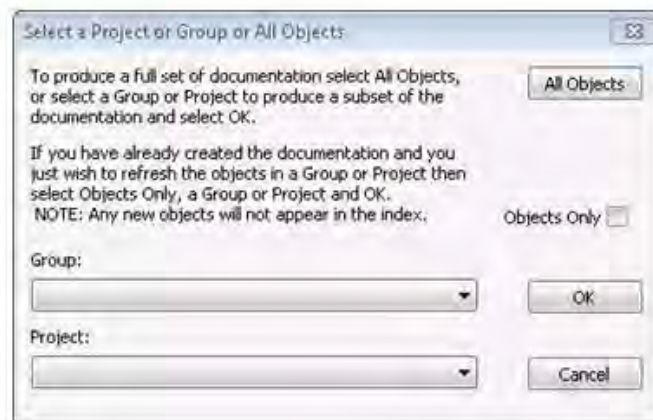
1. Click on *Builder* tab then from top menu select *Doc* → *Create Documentation*.



DWD – Creating Documentation

DWD – Creating Documentation

2. Select *All Objects*, or pick a *Group* or a *Project* within a *Group*. Click *OK*.

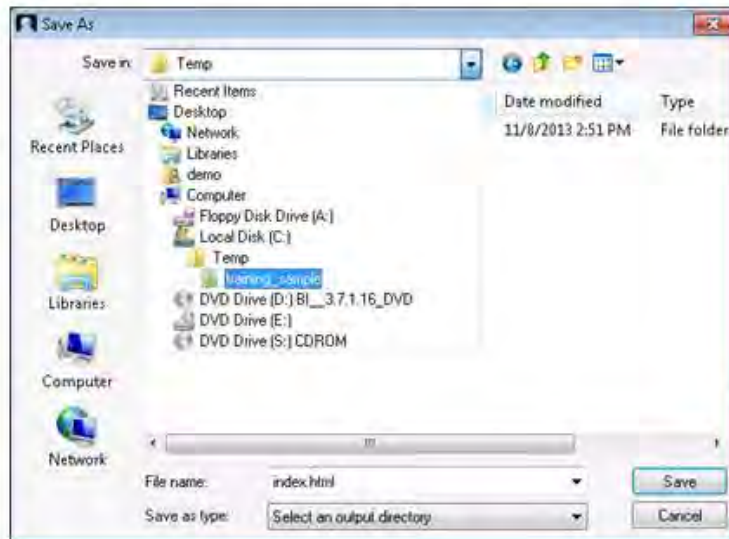


Click OK if presented with flowchartx32.dll error in the Training Environment.

DWD – Creating Documentation

DWD – Creating Documentation

3. Select or create a new folder. Click *Save*.



DWD – Creating Documentation

DWD – Creating Documentation

4. Provide a *Documentation Title*. Click *OK*.

Documentation Creation Options

User and technical documentation will be created in HTML format in the destination directory.

To use a custom look and feel add your own MainStyle.css file into the destination directory.

Documentation Title (e.g. Data Warehouse)

Do you want to link in any custom HTML pages? Links

Do you want to include current file group usage?
It will take longer to create the documentation. Sizes

How would you like the columns sorted? Column Order Column Name Business Name

Do you want to include shadows on the diagram boxes? Shadow

Do you want to create impact analysis on load tables?
It will take longer to create the documentation. Impact

Do you want to replace the existing style sheet?
Do not tick this box if you utilize a custom style sheet. Replace Style Sheet

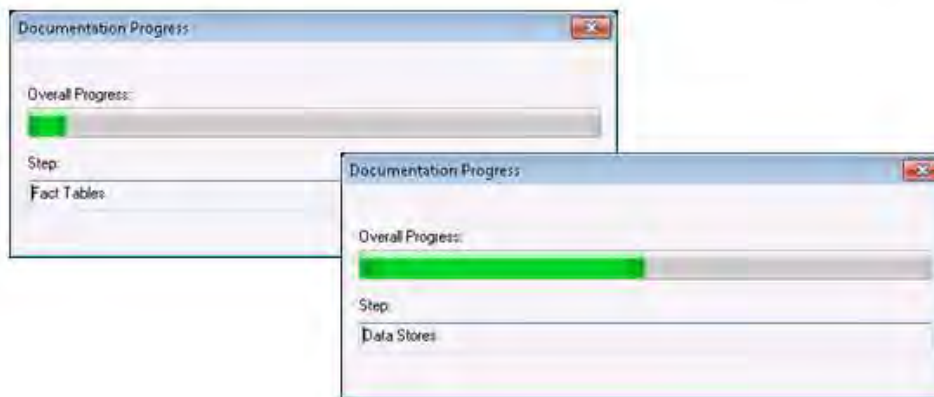
Do you wish to limit the complexity of the diagrams?
Select the maximum number of process steps to display in the source diagrams.

DWD – Creating Documentation

DWD – Creating Documentation

5. A progress bar window will display.

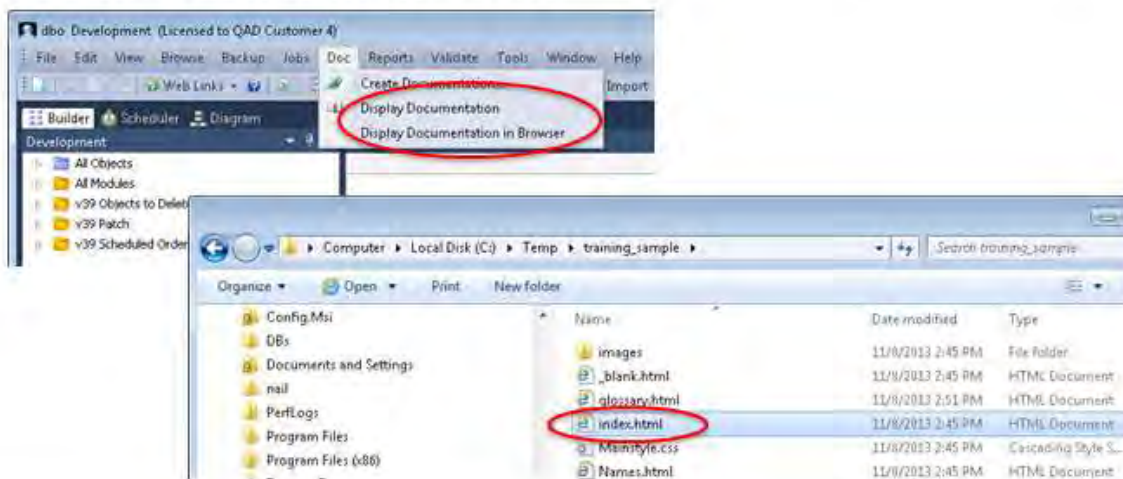
Note: This process may be a while based on selection.



DWD – Creating Documentation

DWD – Creating Documentation

6. Upon completion, the documentation can be viewed in DWD or selecting the *index.html* in the destination folder.



Standard documentation is stored in C:\Program Files (x86)\QAD Business Intelligence\Tomcat\webapps\qadbi\documentation\suite

Day 3 - In Review

Day 3 - In Review

- Setup for CFO Dashboard
- Generating Documentation
- Review for Exam

Parking Lot Review ?



Upon completion of Level 1 Certification you will be able to do the following:

1. **Demonstrate core functionality.** While you will not yet be a SME knowing all the subtle nuances of the product, you will be able to utilize the product being taught and demonstrate its high level functionality.
2. **Guide customers to Best Practices.** This program will help you to understand what are best practices around the particular product you are studying and why they are best practices. This will enable you to speak confidently in making recommendations for business improvements.
3. **Recommend an appropriate Solution.** Perhaps worse than never recommending one of our products is recommending a product to address an issue it was never designed to address. Following your certification, you should be able to identify appropriate opportunities for solutions.
4. **Execute upon a SME Guided Project Plan.** Following Level 1 certification you should not just start working with EAM all by yourself. You should be able, though, to take direction and guidance from a SME and then execute upon that mentoring.

5. **Begin customer facing activities.** Whether you are in support, pre-sales, services, or some other area of expertise, following your Level 1 Certification you will be equipped to begin working face-to-face with customers and properly represent the solution. You should begin using your education immediately! In general:
- Support Resources will be able to address T1 and T2 issues with some assistance from a SME
 - Services Resources will be able to be primary resources on Implementations working alongside a Mentor
 - Pre-Sales Resources will be able to provide customer-specific product demonstrations with support from a Mentor

CHAPTER 4

QAD BI Technical Level 1 Certification – Extras

QAD BI v3 - Technical Level 1 Certification - Extras

QAD Business Intelligence v3 - Technical Level 1 Certification Class - Extras



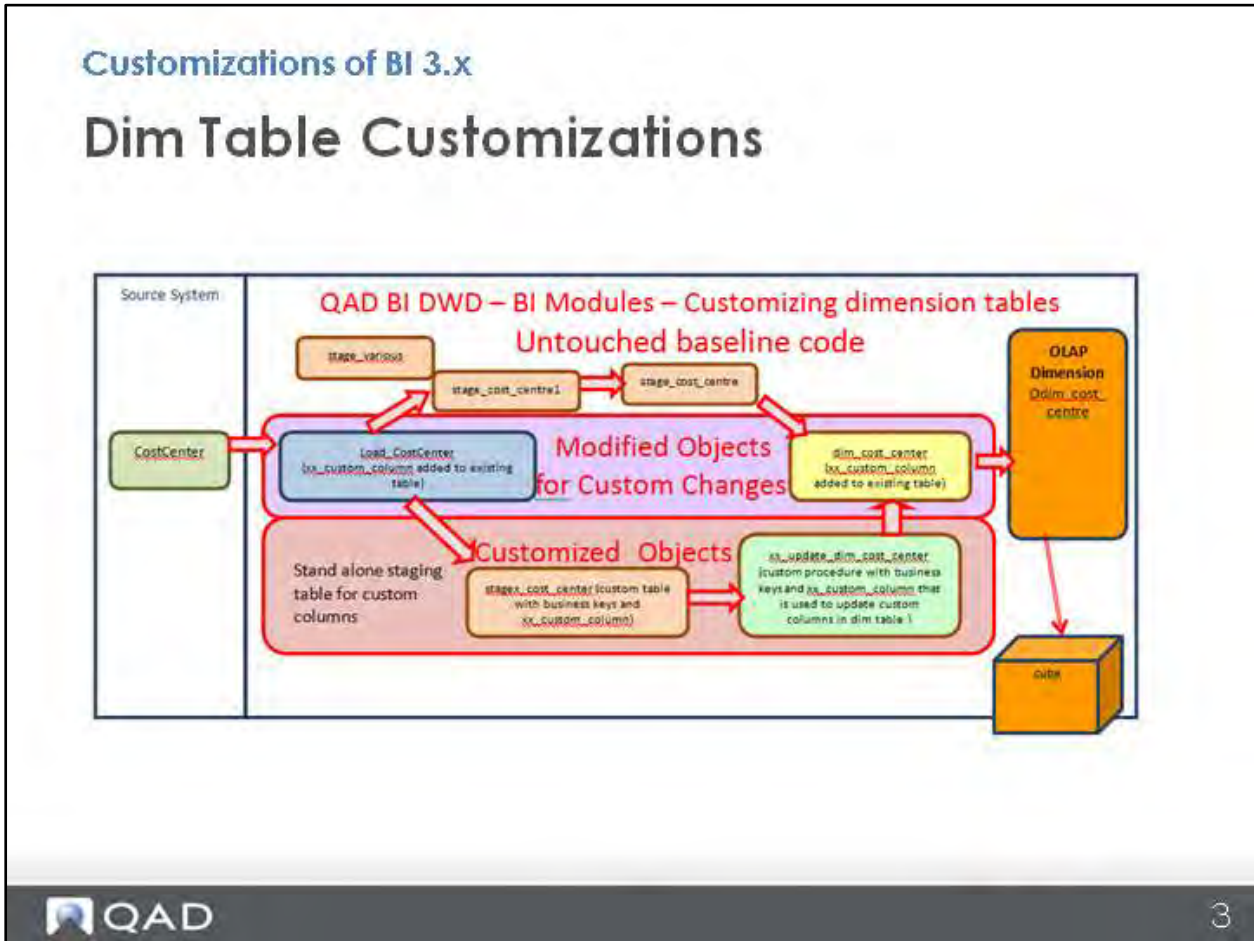
QAD BI 3.x Customization Methodology

QAD BI 3.x Customization Methodology

There are three primary approaches to customizing dim and fact tables using our Custom Path approach.

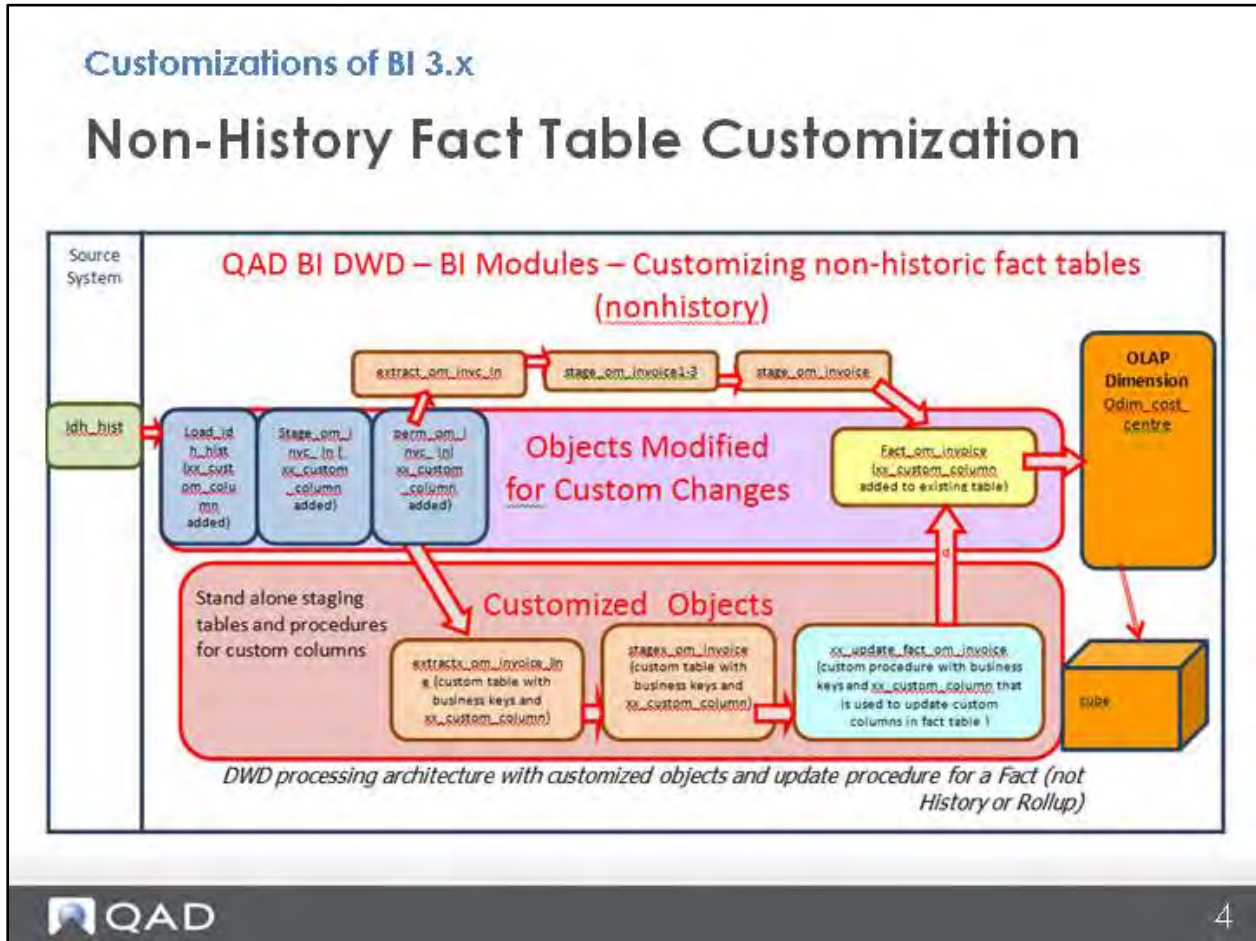
- Dim Table Customizations
- Non-History Fact Table Customization
- History Fact Table Customization

Dim Table Customizations



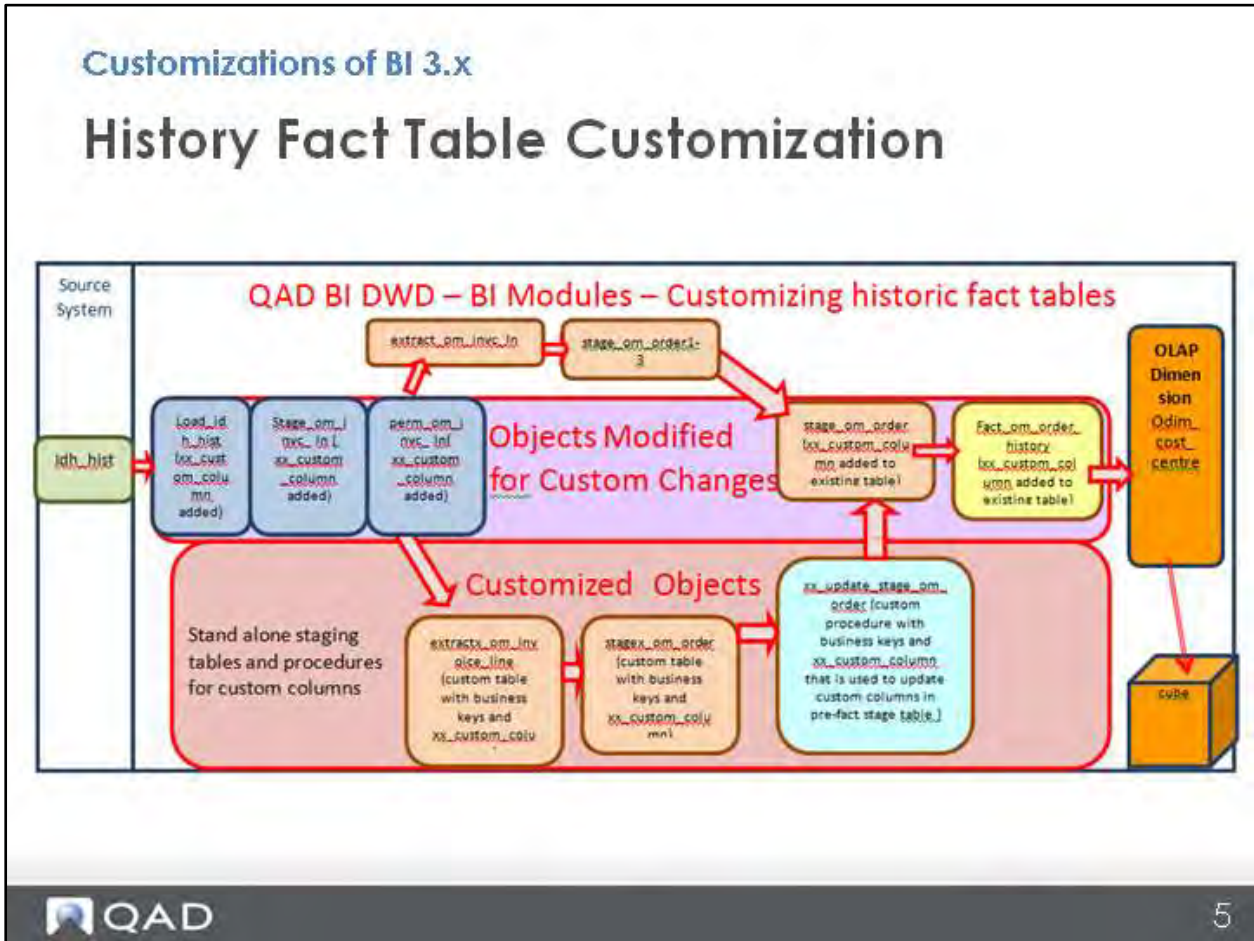
For dimension tables, the only tables that should get modified are load tables and dimension tables. Load tables can have additional columns added and so can dimension tables, but the new column in a dimension should only be updated by an outside procedure that is run after the table is loaded, not the code built by the wizard.

Non-History Fact Table Customization



For non-history fact tables, the only tables that should get modified are load tables, the stage table prior to a perm table, perm tables and the fact tables. Any new columns in the fact table should only be updated by an outside procedure that is run after the table is loaded, not the code built by the wizard.

History Fact Table Customization



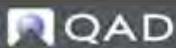
For history fact tables, the only tables that should get modified are load tables, the stage table prior to a perm table, perm tables, the last stage table prior to the fact and the fact tables. Any new columns in the fact table should only be updated by the prior staging table. The new column in the prior staging table should be updated by an outside procedure that is run after the table is loaded, not the code built by the wizard. The reason we allow this is that the logic involved in getting data right in the fact_om_order_history table is incredibly complex and by inserting it into the staging table prior, we can insure that the logic for the fact does it's thing as it should and people building customizations don't have to attempt to emulate all that logic with an update procedure for a customization.

Customizations – Why We Chose Our Methodology

Customizations – Why We Chose Our Methodology

- As an example, let's look at a customized dim_item table that then will have a patch (without the customizations) applied to it.
- Here's the layout for the dim_item table from the 3.4.1 release. The last several columns are circled for reference purposes.

| | | | | | | | | | | | |
|-------------------------|-----------------------|--------------------|----------------|------------|-----------------------|---------------|---|---|---|---|---|
| dim_due_date | promotion_group | Promotion Group | varchar(30) | stage_item | promotion_group | Y | N | N | Y | Y | |
| dim_effective_date | current_cost | Current Cost T... | numeric(38,10) | stage_item | current_cost | #,###(00)... | Y | Y | Y | N | Y |
| dim_end_of_month_date | standard_cost | Standard Cost ... | numeric(38,10) | stage_item | standard_cost | #,###(00)... | Y | Y | Y | N | Y |
| dim_entered_date | latest_standard_cost | Latest Standar... | numeric(38,10) | stage_item | latest_standard_cost | #,###(00)... | Y | Y | N | N | Y |
| dim_entity | net_weight | Net Weight | numeric(38,10) | stage_item | net_weight | #,##0 | Y | Y | Y | N | Y |
| dim_exchange_rate | net_weight_uoi | Net Weight Unit... | varchar(30) | stage_item | net_weight_uoi | | Y | N | N | Y | Y |
| dim_expected_pay_date | latest_current_cost | Latest Current ... | numeric(38,10) | stage_item | latest_current_cost | #,###(00)... | Y | Y | N | N | Y |
| dim_gl_transaction_type | latest_current_cost | Latest Current ... | numeric(38,10) | stage_item | latest_current_cost | #,###(00)... | Y | Y | N | N | Y |
| dim_invoice_date | dss_extract_timestamp | Extract Timesta... | datetime | stage_item | dss_extract_timest... | MM/dd/yyyy... | Y | N | N | Y | Y |
| dim_item | dss_process_batch_id | Process Batch Id | numeric(18) | stage_item | dss_process_batch_id | #####-#... | Y | Y | Y | N | Y |
| dim_last_due_date | dss_update_time | Update Time | datetime | stage_item | dss_update_time | MM/dd/yyyy... | Y | N | N | Y | N |



Customizations – Why We Chose Our Methodology (cont. 2)

Customizations – Why We Chose Our Methodology (cont. 2)

- Here is a customer's installation with different columns, some almost properly labeled.

| | | | | | | |
|---------------------------|----------------------------|---------------------|----------------|------------|------------------------|------|
| dim_discount_sub_account | configuration | Configuration T... | varchar(30) | stage_item | configuration | |
| dim_draft_discount_date | promotion_group | Promotion Group | varchar(80) | stage_item | promotion_group | |
| dim_draft_submission_date | current_cost | Current Cost T... | numeric(38,10) | stage_item | current_cost | ###0 |
| dim_due_date | standard_cost | Standard Cost ... | numeric(38,10) | stage_item | | ###0 |
| dim_effective_date | latest_standard_cost | Latest Standar... | numeric(38,10) | stage_item | latest_standard_cost | ###0 |
| dim_end_of_month_date | latest_current_cost | Latest Current ... | numeric(38,10) | stage_item | latest_current_cost | ###0 |
| dim_entered_date | promotion_group_descrip... | Promotion Grou... | varchar(80) | stage_item | promotion_group_d... | |
| dim_entity | supplier_description | Supplier Descrip... | varchar(30) | stage_item | supplier_description | |
| dim_exchange_rate | buyer_description | Buyer Description | varchar(80) | stage_item | buyer_description | |
| dim_expected_pay_date | abc_class | ABC Class | varchar(1) | stage_item | x_abc_class | |
| dim_gl_transaction_type | safety_time | safety time | numeric(20) | stage_item | safety_time | ###0 |
| dim_invoice_date | dss_extract_timestamp | extract timestamp | datetime | stage_item | dss_extract_timest... | |
| dim_item | dss_process_batch_id | process batch id | numeric(18) | stage_item | dss_process_batch_id | ###0 |
| dim_last_due_date | dss_update_time | dss update time | datetime | | dss_update_time | |
| dim_last_receipt_date | x_reporting_unit_of_mea... | reporting unit o... | varchar(64) | stage_item | x_reporting_unit_of... | |
| dim_layer | x_ship_day_tolerance | ship day tolerance | integer | stage_item | x_ship_day_tolerance | ###0 |
| dim_location | x_ship_quantity_tolerance | ship quantity to... | decimal(13,2) | stage_item | x_ship_quantity_tol... | ###0 |
| dim_order_date | | | | | | |

Here's the layout for the dim_item table for a customer that has a 3.3 customized version of the modules that not only has the easily identifiable custom columns that begin with an x_ prefix (our preferred prefix for custom columns now is xx_), but also note that they have some other columns that don't match our 3.4.1 dim_item table. These had been added at time of installation with the belief that these same columns would be added to the code base. That ended up not yet happening, so an upgrade to 3.4.1 is likely going to require even more work and investigation. Unless columns are already in the code base, treat any additions as custom.

Customizations – Why We Chose Our Methodology (cont. 3)

Customizations – Why We Chose Our Methodology (cont. 3)

- In SQL Server you can see that the table matches the meta-data observed via the DWD.

```
configuration (varchar(30), null)
promotion_group (varchar(80), null)
current_cost (numeric(38,10), null)
standard_cost (numeric(38,10), null)
latest_standard_cost (numeric(38,10), null)
latest_current_cost (numeric(38,10), null)
promotion_group_description (varchar(80), null)
supplier_description (varchar(30), null)
buyer_description (varchar(80), null)
abc_class (varchar(1), null)
safety_time (numeric(20,0), null)
dss_extract_timestamp (datetime, null)
dss_process_batch_id (numeric(18,0), null)
dss_update_time (datetime, null)
x_reporting_unit_of_measure (varchar(64), null)
x_ship_day_tolerance (int, null)
x_ship_quantity_tolerance (decimal(13,2), null)
```

Customizations – Why We Chose Our Methodology (cont. 4)

Customizations – Why We Chose Our Methodology (cont. 4)

- And if we open the affiliated procedure, in the DWD, we can see what columns are being inserted into and that they match the columns listed in the DWD,

```

current_cost
0
latest_standard_cost
latest_current_cost
promotion_group_description
supplier_description
buyer_description
x_abc_class
safety_time
dss_extract_timestamp
dss_process_batch_id
x_reporting_unit_of_measure
x_ship_day_tolerance
x_ship_quantity_tolerance

current_cost
standard_cost
latest_standard_cost
latest_current_cost
promotion_group_description
supplier_description
buyer_description
abc_class
safety_time
dss_extract_timestamp
dss_process_batch_id
x_reporting_unit_of_measure
x_ship_day_tolerance
x_ship_quantity_tolerance

FROM
stage_item
WHERE NOT EXISTS (
SELECT 1 FROM dim_item
WHERE ( stage_item.source_system_code = dim_item.source_system_code
OR ( stage_item.source_system_code IS NULL AND dim_item.source_system_code IS NULL ))
AND ( stage_item.domain_code = dim_item.domain_code
OR ( stage_item.domain_code IS NULL AND dim_item.domain_code IS NULL ))
)

```

Customizations – Why We Chose Our Methodology (cont. 5)

Customizations – Why We Chose Our Methodology (cont. 5)

- A patch is then applied via the Administrator tool to fix some issue with the table or to upgrade the table to the 3.4.1 version of the code.

| Object | Status |
|------------------------------|---|
| Application load begins | D:\bjart_work\custom_tests\app_obj_Patch_Test_1.wst |
| get_dim_item_key | Will replace existing object |
| update_dim_item | Will replace existing object |
| update_stage_item | Will replace existing object |
| update_stage_item_site | Will replace existing object |
| update_stage_om_order3 | Will replace existing object |
| update_stage_om_order2 | Will replace existing object |
| update_stage_om_order1 | Will replace existing object |
| custom_stage_om_order1 | Will replace existing object |
| update_fact_om_order_history | Will replace existing object |
| custom_fact_om_order_history | Will replace existing object |
| dim_item | Will replace existing object |
| stage_item | Will replace existing object |
| stage_item_site | Will replace existing object |
| stage_om_order3 | Will replace existing object |
| stage_om_order2 | Will replace existing object |
| stage_om_order1 | Will replace existing object |
| fact_om_order_history | Will r |
| dim_item_idx_0 | Will r |
| dim_item_idx_A | Will r |
| stage_om_order1_idx_A | Will r |
| fact_om_order_history_idx_1 | Will r |
| fact_om_order_history_idx_10 | Will r |

Setup Administrator

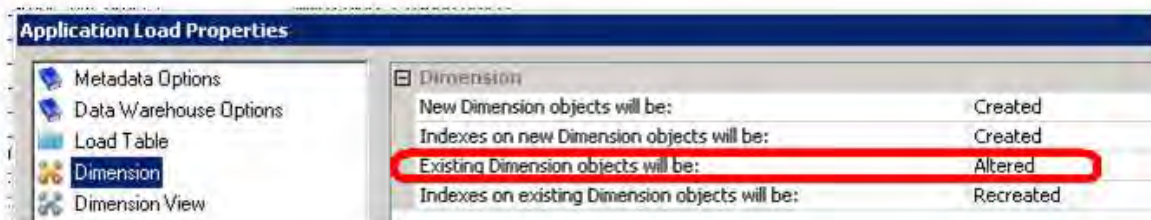
47 Object/s will be replaced.
You may cancel at this point and review the objects to be replaced and/or added.

You can see that it initially says that existing objects will be replaced.

Customizations – Why We Chose Our Methodology (cont. 6)

Customizations – Why We Chose Our Methodology (cont. 6)

- However, the default setting for each object during application installation ensures that the dim, perm and fact tables are only altered by changes and the updates proceed successfully.



Customizations – Why We Chose Our Methodology (cont. 7)

Customizations – Why We Chose Our Methodology (cont. 7)

- A refresh of the DWD Browser pane to review the changes to dim_item then reveals that the data in the tables looks exactly like 3.4.1 version of dim_item. Any other columns that had existed for this table in the DWD's meta-data are no longer shown.

| Table Name | Column Name | Data Type | Stage Item | Column Name | Format | Y | N | N | Y | N |
|-----------------------|--------------------|----------------|------------|-----------------------|------------------|---|---|---|---|---|
| promotion_group | Promotion Group | varchar(50) | stage_item | promotion_group | | Y | N | N | Y | N |
| current_cost | Current Cost T... | numeric(38,10) | stage_item | current_cost | #,###,000,000... | Y | Y | Y | N | Y |
| standard_cost | Standard Cost ... | numeric(38,10) | stage_item | standard_cost | #,###,000,000... | Y | Y | Y | N | Y |
| latest_standard_cost | Latest Standard... | numeric(38,10) | stage_item | latest_standard_cost | #,###,000,000... | Y | Y | N | N | Y |
| net_weight | Net Weight | numeric(38,10) | stage_item | net_weight | #,###0 | Y | Y | Y | N | Y |
| net_weight_um | Net Weight Unit... | varchar(30) | stage_item | net_weight_um | | Y | N | N | Y | Y |
| latest_current_cost | Latest Current ... | numeric(38,10) | stage_item | latest_current_cost | #,###,000,000... | Y | Y | N | N | Y |
| dss_extract_timestamp | Extract TimeSta... | datetime | stage_item | dss_extract_timestamp | MM/dd/yyyy... | Y | N | N | Y | Y |
| dss_process_batch_id | Process Batch Id | numeric(18) | stage_item | dss_process_batch_id | #####-#... | Y | Y | Y | N | Y |
| dss_update_time | Update Time | datetime | stage_item | dss_update_time | MM/dd/yyyy... | Y | N | N | Y | N |

Customizations – Why We Chose Our Methodology (cont. 8)

Customizations – Why We Chose Our Methodology (cont. 8)

- Going to SQL Server reveals that the columns still exist in the database and that any new columns that were found when the patch was applied were placed at the end of the table column list.

```

configuration (nvarchar(30), null)
promotion_group (nvarchar(80), null)
current_cost (numeric(38,10), null)
standard_cost (numeric(38,10), null)
latest_standard_cost (numeric(38,10), null)
latest_current_cost (numeric(38,10), null)
promotion_group_description (varchar(80), null)
supplier_description (varchar(30), null)
buyer_description (varchar(80), null)
abc_class (varchar(1), null)
safety_time (numeric(20,0), null)
dss_extract_timestamp (datetime, null)
dss_process_batch_id (numeric(18,0), null)
dss_update_time (datetime, null)
x_reporting_unit_of_measure (varchar(64), null)
x_ship_day_tolerance (int, null)
x_ship_quantity_tolerance (decimal(13,2), null)
net_weight (numeric(38,10), null)
net_weight_um (nvarchar(30), null)
    
```



When the patch was applied, we had selected to not change existing table, only alter them. Why are the other columns gone?

Customizations – Why We Chose Our Methodology (cont. 9)

Customizations – Why We Chose Our Methodology (cont. 9)

- Reviewing the affiliated procedure in the DWD shows that all the logic referencing the custom columns has been removed and the new logic only reflects the columns that exist in the 3.4.1 version that was applied.

```

configuration
promotion_group
current_cost
0
latest_standard_cost
net_weight
net_weight_um
latest_current_cost
dss_extract_timestamp
dss_process_batch_id
FROM
stage_item
WHERE NOT EXISTS (
SELECT 1 FROM dim_item
WHERE ( stage_item.source_system_code = dim_item.source_system_code
OR ( stage_item.source_system_code IS NULL AND dim_item.source_system_code IS NULL ))
AND ( stage_item.domain_code = dim_item.domain_code
OR ( stage_item.domain_code IS NULL AND dim_item.domain_code IS NULL ))
)

```

Note - Since the new columns were added to the table and the table was not dropped, the custom columns still would contain their data, but would no longer be updated by the new procedure. A SQL Query against the table would confirm this.

Customizations – Why We Chose Our Methodology (cont. 10)

Customizations – Why We Chose Our Methodology (cont. 10)

- From this example, it is clear that implementing patches and upgrades to tables with customizations can be a difficult process, if customizations occur to the baseline code.
- Our preferred solution is to touch as little of the baseline code as possible for customizations.

As one can see, implementing patches and upgrades to tables that have any custom work done to them is not a straight forward process and likely requires a fair amount of follow up work to get the customizations back in place in the DWD meta-data (the underlying SQL table can remain unchanged). It is for this reason that **ideally** custom work will not be done directly on standard release tables in the data warehouse or at the very least, the footprint of the customized work will mostly fall outside of the baseline code.

Adding a Column to a Table

Adding a Column to a Table

- New columns can be added to tables quite easily. The most common method is to drag and drop the column into place from a prior table, but a column can just be added. The new column can be added to reference a column from an existing source table, a new source table, or no source table at all (for static value or parameter fed columns).

Adding a Column to a Table

Adding a Column to a Table

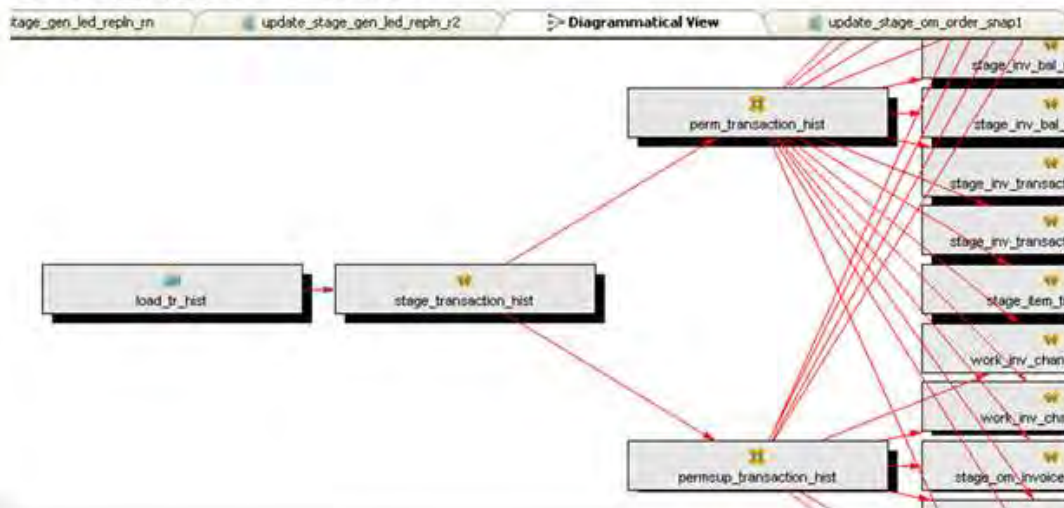
- Add tr_prod_line to the load_tr_hist table. Here's a track forward diagram of that load_tr_hist table.



Adding a Column to a Table

Adding a Column to a Table

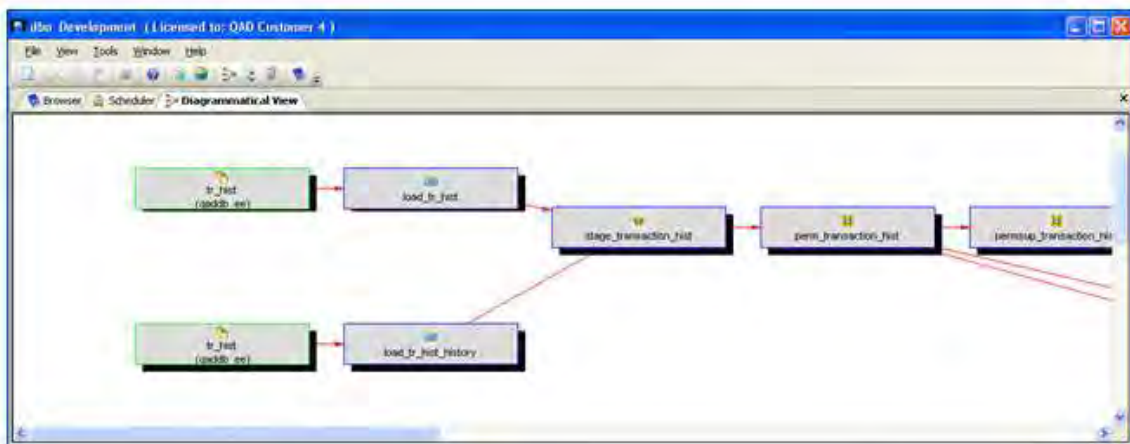
- Add tr_prod_line to the load_tr_hist table. Here's a track forward diagram of that load_tr_hist table.



Adding a Column to a Table

Adding a Column to a Table

- Add tr_prod_line to the load_tr_hist table. Here's a track back diagram of one the downstream tables.

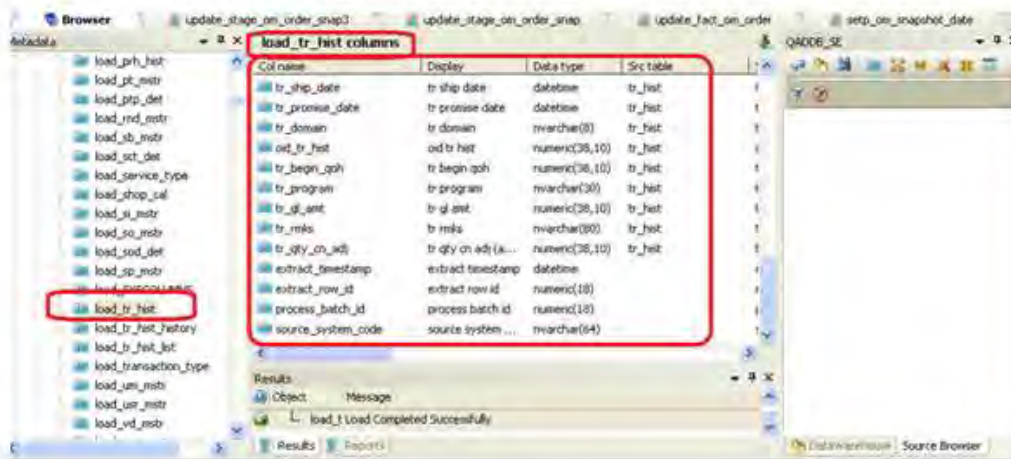


Stage_transaction_hist is a set merge table, so the change needs to be made to both load_tr_hist, load_tr_hist_history. Show the class the source columns from stage_transaction_hist, also properties and the stored procedure of the Set Merge for the insert into each table.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

1. In the Browser pane, click on the table the column is to be added to so that all its columns appear in the Target pane.

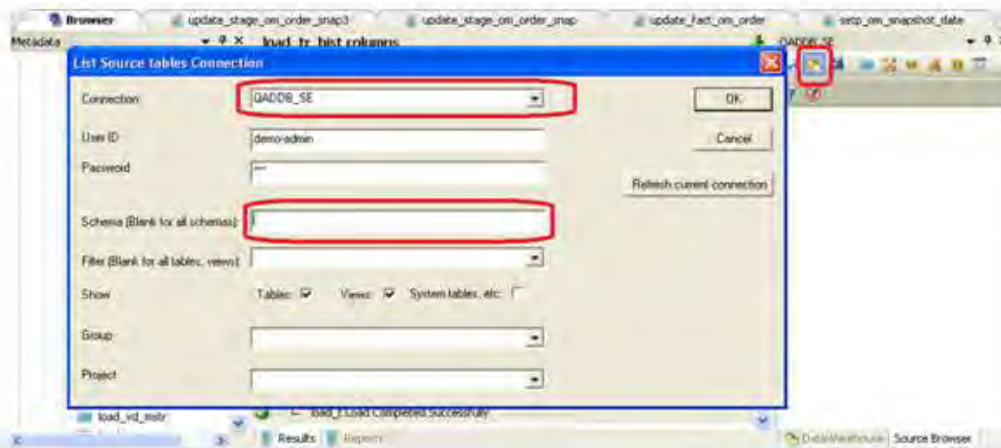


For this example use tr_hist which will have the tr_prod_line added.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

2. In the Source pane, connect to a source database that has the source table and column.



First in the source pane, click the connection button.

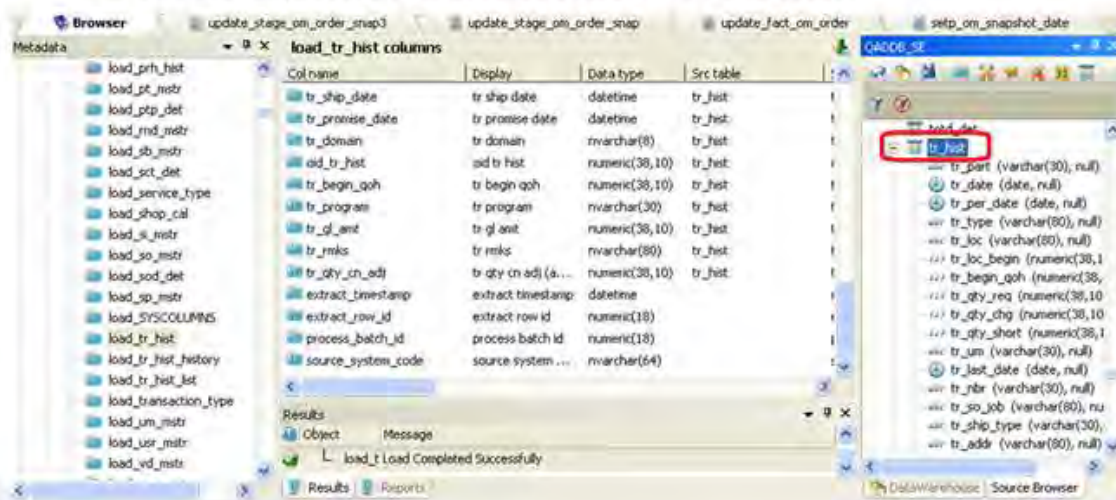
When the List Source table Connection window opens, change to the Connection.

Remove the Schema if there is one. It needs to be blank.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

- When the list of tables appears in the browser pane, find the table to be sourced. A filter option can make it easier.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

4. Double click the table in the Source pane and find the column to be added.

The screenshot shows the QAD BI Source Browser interface. The 'load_tr_hist columns' table is displayed with the following columns:

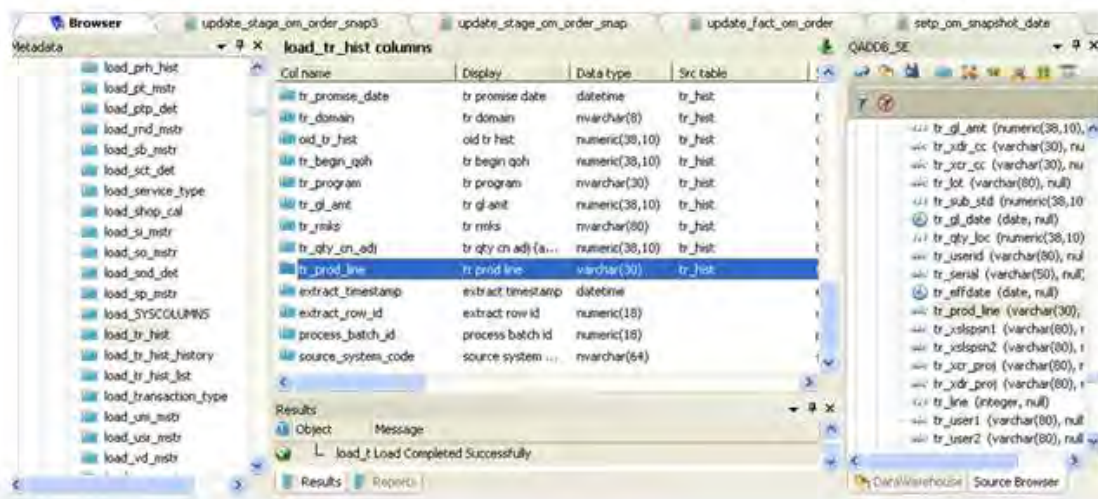
| Col name | Display | Data type | Src table |
|--------------------|--------------------|----------------|-----------|
| tr_ship_date | tr ship date | datetime | tr_hist |
| tr_promise_date | tr promise date | datetime | tr_hist |
| tr_domain | tr domain | nvarchar(8) | tr_hist |
| oid_tr_hist | oid tr hist | numeric(38,10) | tr_hist |
| tr_begin_qoh | tr begin qoh | numeric(38,10) | tr_hist |
| tr_program | tr program | nvarchar(30) | tr_hist |
| tr_gl_amt | tr gl amt | numeric(38,10) | tr_hist |
| tr_rnks | tr rnks | nvarchar(80) | tr_hist |
| tr_qty_on_ad | tr qty on ad (a... | numeric(38,10) | tr_hist |
| extract_timestamp | extract timestamp | datetime | |
| extract_row_id | extract row id | numeric(18) | |
| process_batch_id | process batch id | numeric(18) | |
| source_system_code | source system ... | nvarchar(64) | |

The 'tr_prod_line (varchar(50))' column is highlighted in red in the source pane on the right. The 'Results' pane at the bottom shows the message: 'load_tr_hist Load Completed Successfully'.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

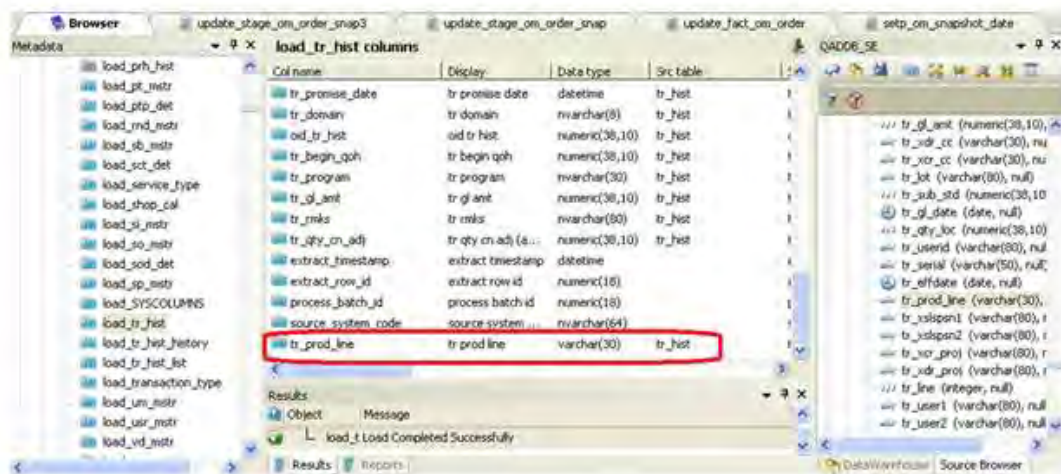
5. Drag the column from the Source pane table to the Target pane.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

6. Make sure that the new custom columns are listed last for the table.



25

Do this by dragging the column to the end of the column list. If having trouble either move the columns before it up, or manually number the column with a number higher than the last one.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

7. Rename the Column using the customization naming convention to `xx_tr_procline`. If the column is char or varchar, switch the data type to nvarchar.

The screenshot shows the 'Column Properties' dialog box with the 'Transformation' tab selected. The following fields are visible:

- Table: load_tr_hist
- Column name: xx_tr_procline (highlighted with a red box)
- Business display name: tr prod line
- Data type: nvarchar(30) (highlighted with a red box)
- Format: (empty)

Buttons on the right side include 'Update', 'Update <', and 'Update >'.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

8. In the transformation tab, for all character type columns (varchar, char, nvarchar), put in a transformation to strip out carriage returns or ' | ' symbols. Use any other varchar column in the table as a reference.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

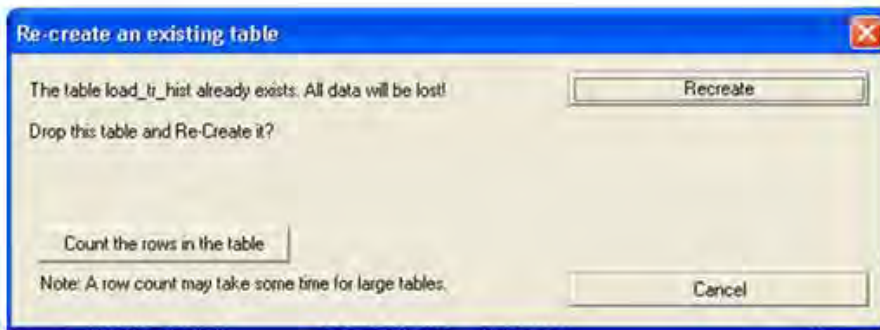
9. Because this is a load table, it is fine to Create (Recreate) the table. In the Browser pane, right click on the table name and pick Create (Recreate).



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

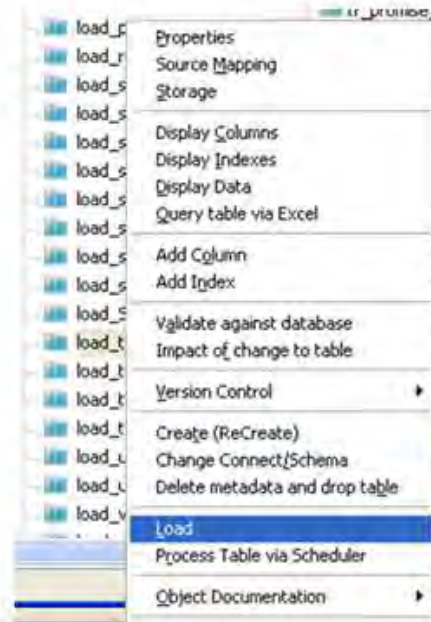
10. When the pop up window opens, select Recreate.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

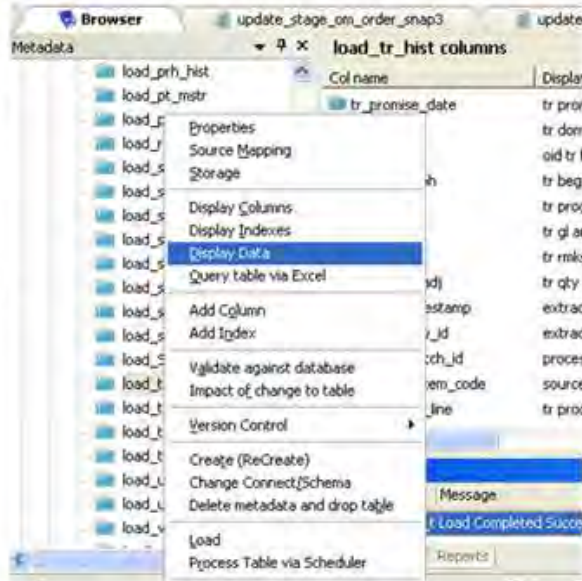
11. Right click on the load_table and pick the Load option to confirm that the table loads ok.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

12. Validate that the new column shows up and is being populated in the load table by using Display Data.



Scroll to the end of the results pane to see if the column is there and the data is populated.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

13. Display Data shows the new column.

Data display for load_tr_hist

| tr_qty_cn_adj | extract_tm... | extract_ro... | process_ba... | source_sys... | cx_tr_prod... |
|---------------|---------------|---------------|---------------|---------------|---------------|
| .0000000000 | 2011-11-16... | 48709 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48710 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48711 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48712 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48713 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48714 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48715 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48716 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48717 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48718 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48768 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48769 | 2862 | QADDB_SE | NP |
| .0000000000 | 2011-11-16... | 48770 | 2862 | QADDB_SE | NP |

Scroll to the end of the results pane to see if the column is there and the data is populated.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

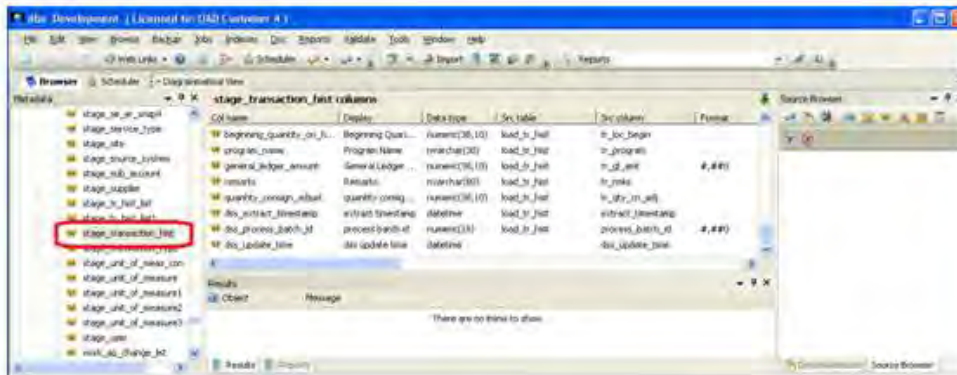
14. Because the next table, the `stage_transaction_hist` table is a SET MERGE table, go through the same steps 1-10 for `load_tr_hist_history` (don't do the Load steps).

Scroll to the end of the results pane to see if the column is there and the data is populated.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

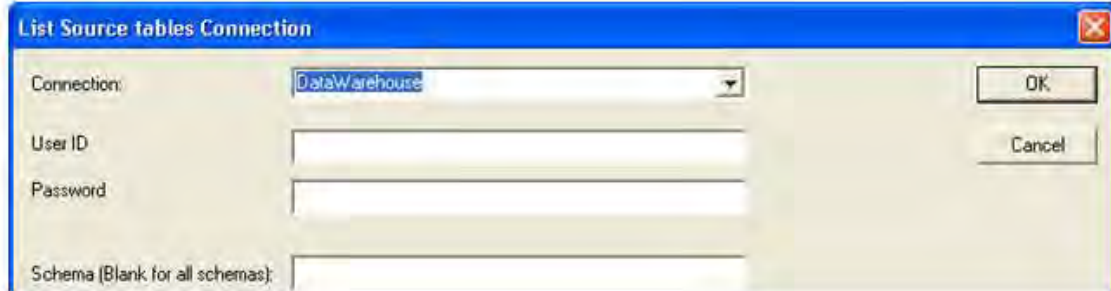
- To add the column to the staging table prior to the perm_table, double click the stage_transaction_hist file in the Browser pane so that all column show in the Target pane.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

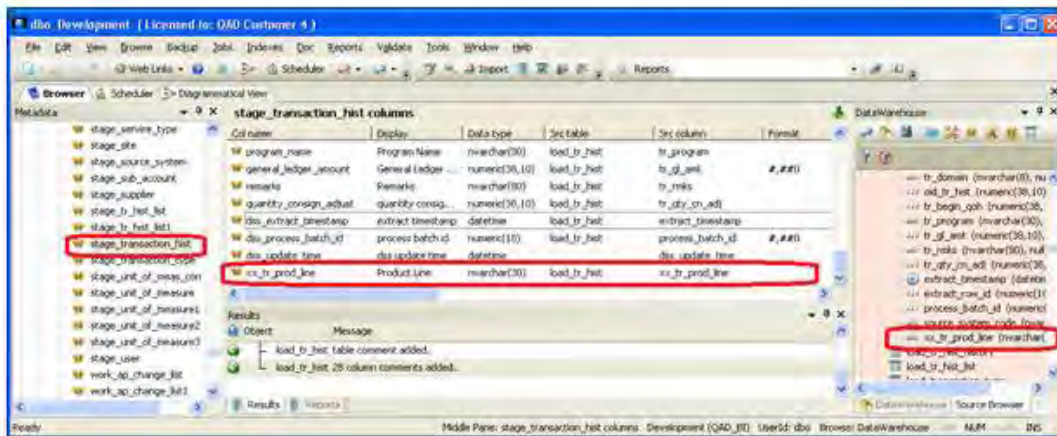
16. In the Source pane, change the Connection to Data Warehouse. Make sure that the Schema is blank.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

17. Find the load_tr_hist table, double click it and find the xx_tr_prod_line and drag it over to the stage_transaction_hist table.

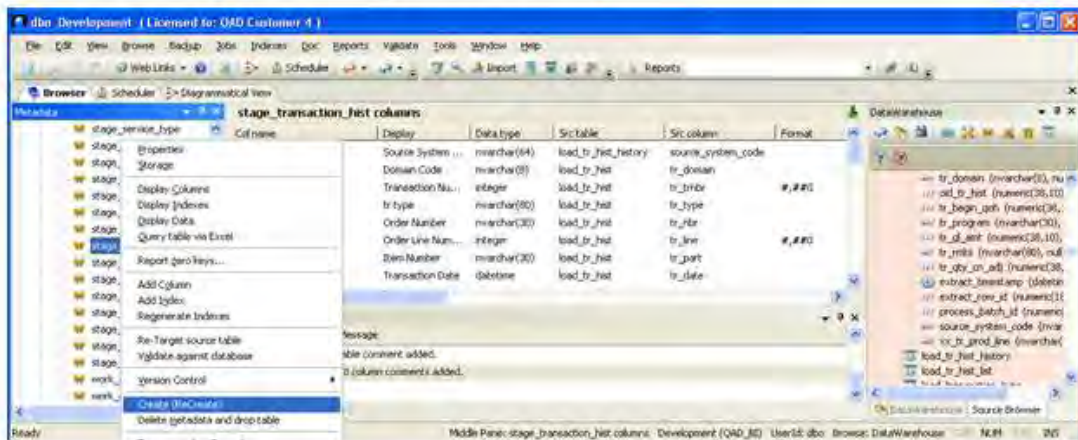


Make sure the column is last in the column list (see Step 7).

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

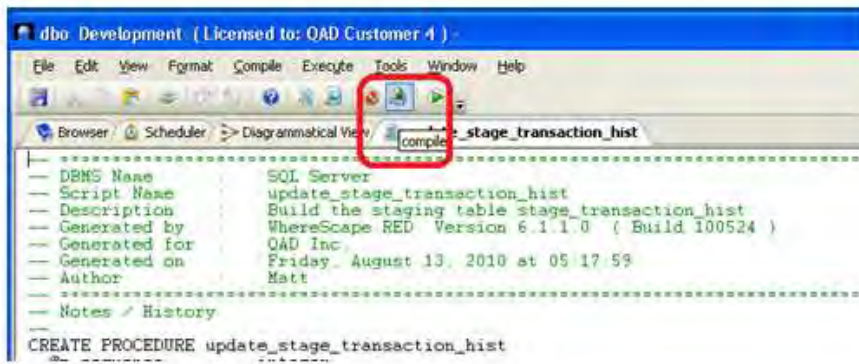
18. This being a staging table, Create(Recreate) can be used to rebuild the table with the new column (see Steps 10 & 11).



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

19. Save a versioned copy of the stored procedure for the table.



Double click the table in the Browser table. When the pop-up window appears, click the Edit button. When the procedure window opens, click the Compile button. Close the window.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

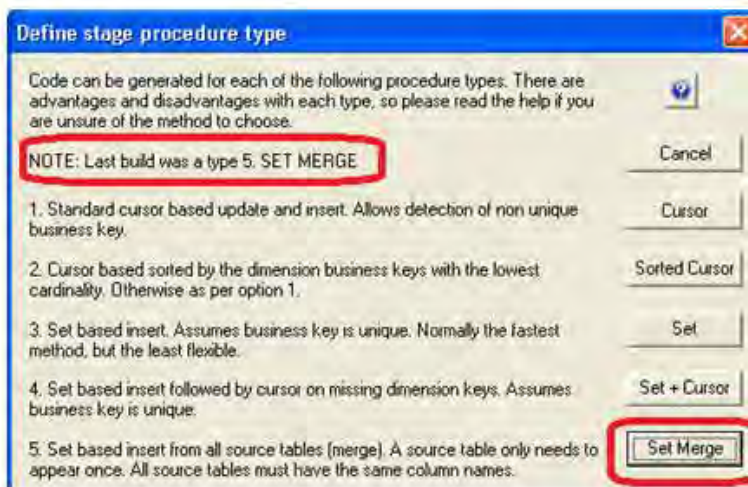
20. Double click the table name in the Browser pane again. When the pop-up window opens, click Rebuild.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

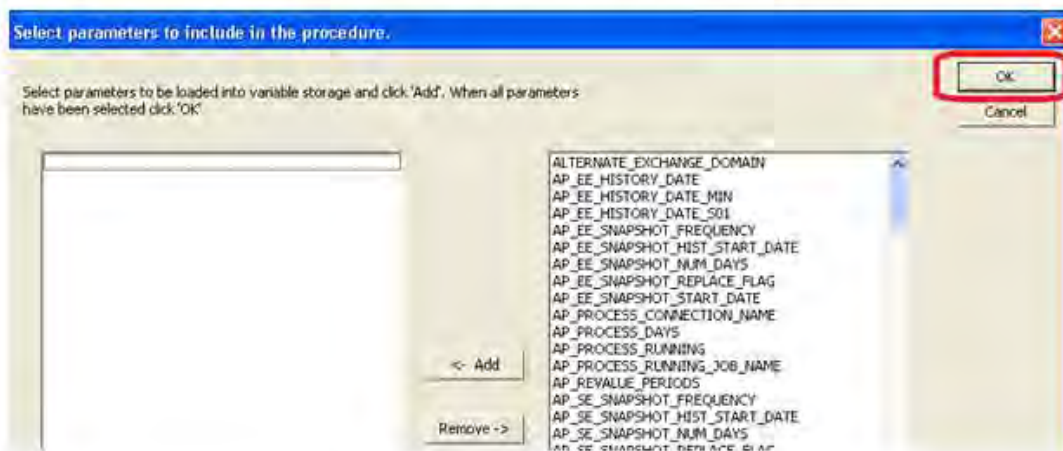
21. Use the default build type from the last build.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

22. Since no changes are being made to parameters, click OK to move forward.



Any new parameters to be picked up would be added here.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

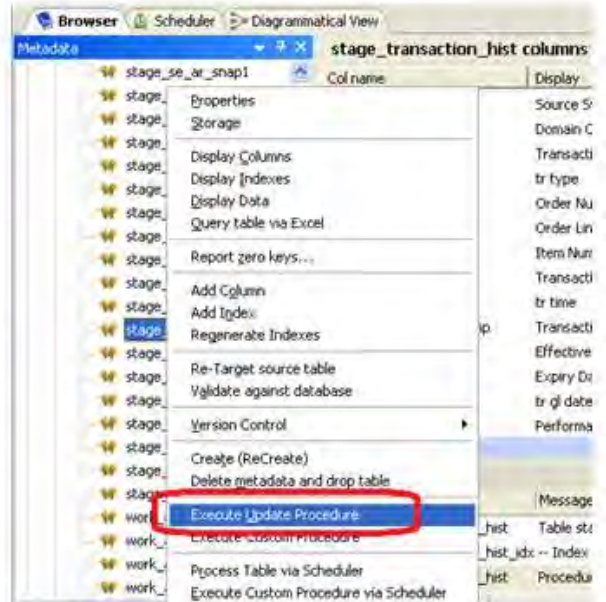
23. Since no changes are being made to table joins, click Ok through all the windows.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

24. To test the load of the table, right click on the table name and choose Execute Update Procedure to populate the table. Or run Process Table via Scheduler.

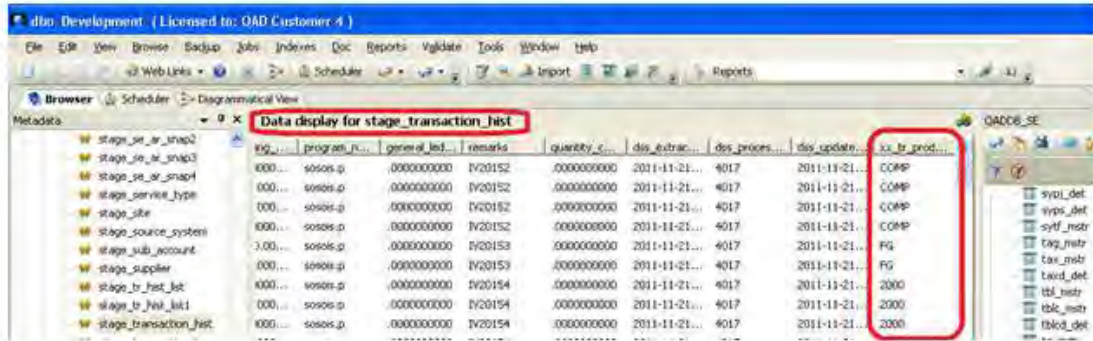


Running Process Table via Scheduler ensures resources are freed up to be run. If you run Execute Update Procedure for a very large table, it will put the DWD into Not Responding mode. For small tables it works great to Execute Update Procedure, but for large tables, always use Process Table via Scheduler.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

25. Validate the data load. Right click on stage_transaction_hist in the Browser pane and Display data. Confirm new column populated.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

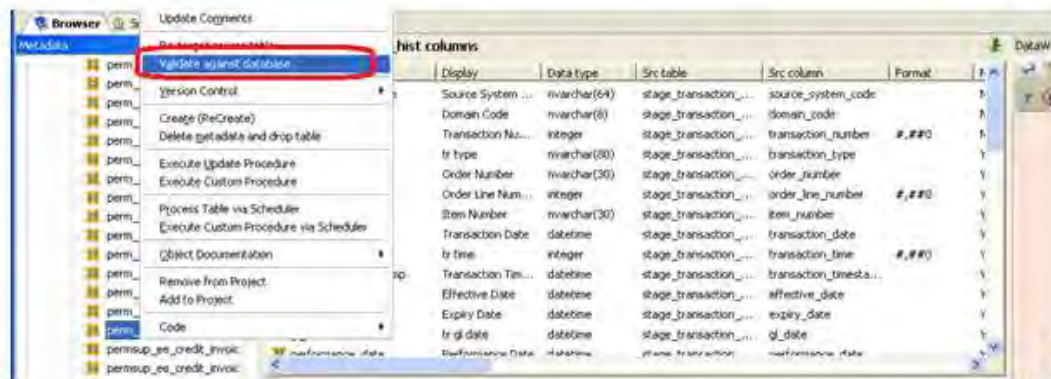
26. Add the custom column to the perm_transaction_hist table by drag and drop using the same methodology as last two tables. ***Do NOT proceed to CreateRecreate the table. See next step.***

| Col name | Display | Data type | Src table | Src column | Format |
|-----------------------------|---------------------|----------------|-----------------------|-------------------------|--------|
| subcontract_unit_cost | Subcontract Uni... | numeric(38,10) | stage_transaction_... | subcontract_unit_cost | #,##0 |
| overhead_unit_cost | Overhead Unit ... | numeric(38,10) | stage_transaction_... | overhead_unit_cost | #,##0 |
| net_price | Net Price | numeric(38,10) | stage_transaction_... | net_price | #,##0 |
| covered_amount | Covered Amount | numeric(38,10) | stage_transaction_... | covered_amount | #,##0 |
| beginning_quantity_on_h... | beginning quanti... | numeric(38,10) | stage_transaction_... | beginning_quantity... | #,##0 |
| tr loc beg | | numeric(38,10) | stage_transaction_... | beginning_quantity... | |
| program_name | Program Name | nvarchar(30) | stage_transaction_... | program_name | |
| general_ledger_amount | General Ledger ... | numeric(38,10) | stage_transaction_... | general_ledger_am... | #,##0 |
| remarks | Remarks | nvarchar(10) | stage_transaction_... | remarks | |
| quantity_consignment_adj... | quantity consig... | numeric(38,10) | stage_transaction_... | quantity_consignment... | #,##0 |
| dss_create_time | dss_create_time | datetime | | dss_create_time | |
| dss_update_time | dss_update_time | datetime | | dss_update_time | |
| xx_tr_prod_line | Product Line | nvarchar(30) | stage_transaction_... | xx_tr_prod_line | |

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

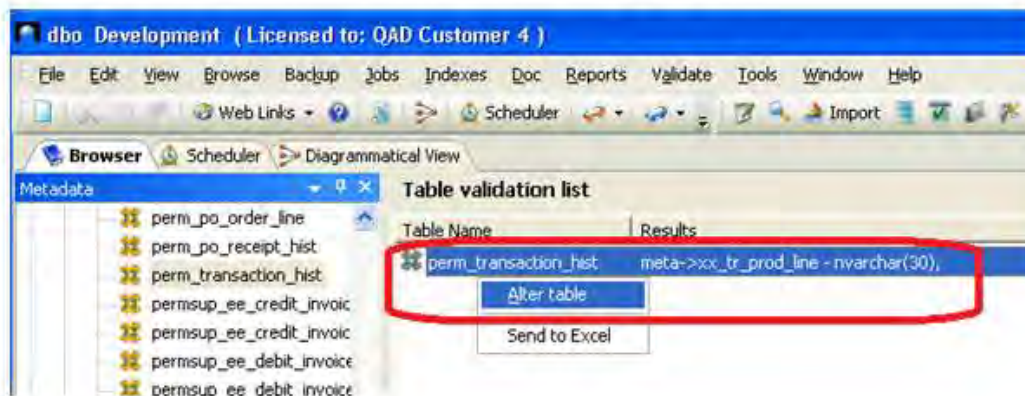
27. Because this is a perm_table, caution must be used when adding an extra column. Right click on the perm_table in the Browser pane and click Validate against database.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

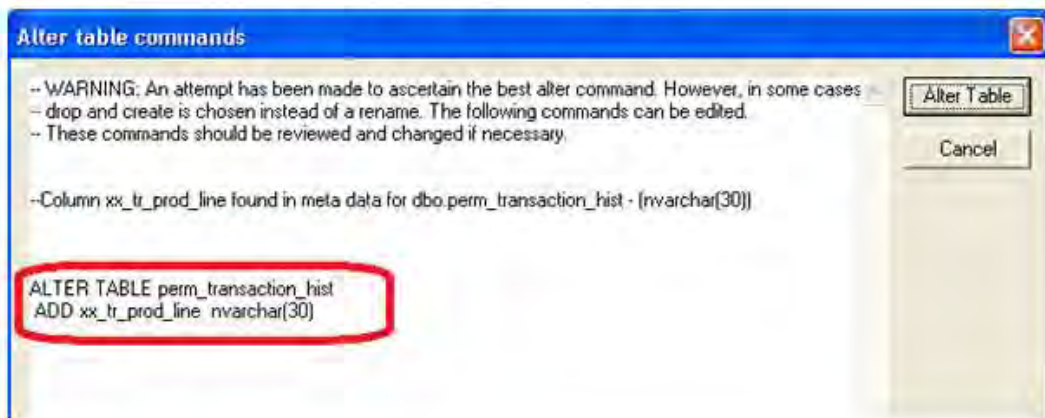
28. Table should indicate the new columns in the metadata. Right click on that message to pull up an Alter table option.



Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

29. Some an alter table script will appear. Make sure it is only adding to the existing table, not dropping any columns.



As an alternate

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

30. Some an alter table script will appear. Make sure it is only adding to the existing table, not dropping any columns.



Dim_ tables add a drop column script that is very dangerous.

Adding a Column to a Table – Drag and Drop Methodology

Adding a Column to a Table – Drag and Drop Methodology

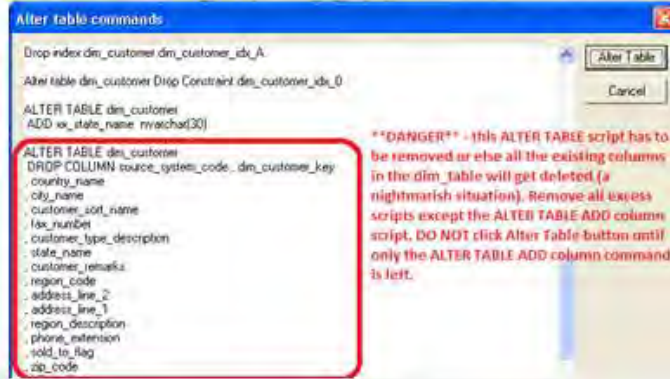
31. Populating a perm_ table's new column will require either a special update script or a repopulate historically.

This is beyond the scope of this Level I class.

Adding a Column to a Table

Adding a Column to a Table

- Adding columns to dim_, perm_ and fact_ tables. Dim_ tables are particularly dangerous. Aside from adding a column, the Alter table script also has DROP column commands that have to be removed.



Any of the three tables that data is to be permanently stored in should use the Validate database option to alter the table.

Adding a Column to a Table

Adding a Column to a Table

- In most instances, the simplest way to add a column is to drag and drop the column
- Difference between the data stored as meta-data and architecture in SQL Server
- Include explanation of how to deal with columns that exist in some environments and not others

Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

- As procedures are updated with any changes, the DWD keeps track of the different versions of the procedure which can be compared against the other versions of that procedure.
- For the purpose of this example, we'll make a minor change by adding a column transformation to column `quantity_returned` in table `stage_om_invoice`

An example of a procedure change can also be shown by looking at `stage_transaction_hist` from the prior exercise, presuming that the class took the step of compiling the existing procedure before rebuilding it.

Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

- To make a change to the quantity_returned column, find the stage_om_invoice table in the browser window and click on it once. All columns in the table should appear in the target pane. Find the quantity_returned column.

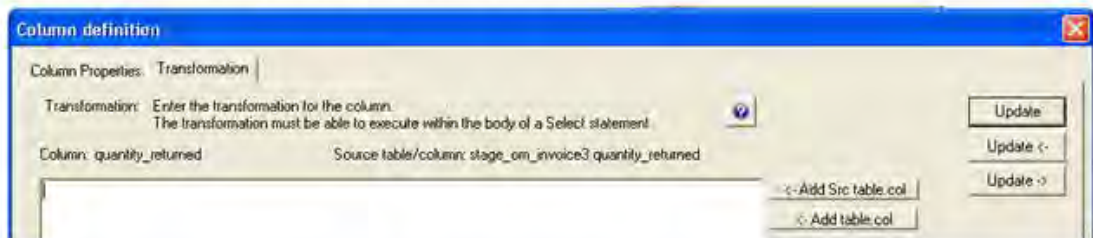
| | | | | | | |
|-------------------------|----------------------|-------------------|----------------|-------------------|----------------------|-----|
| stage_om_booking | quantity_shipped | Quantity Shipped | numeric(38,10) | stage_om_invoice3 | quantity_shipped | ### |
| stage_om_invoice | quantity_returned | Quantity Retur... | numeric(38,10) | stage_om_invoice3 | quantity_returned | ### |
| stage_om_invoice_header | unit_of_measure_code | Unit Of Measur... | nvarchar(30) | stage_om_invoice3 | unit_of_measure_c... | |

- Double click the quantity_returned column to open the properties window and click the Transformation tab.

Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

- Double click the quantity_returned column to open the Column definitions window and click the Transformation tab.



- Click the Add Src table.col button and then add ISNULL(,0) logic to it. Be sure to click Update to save.

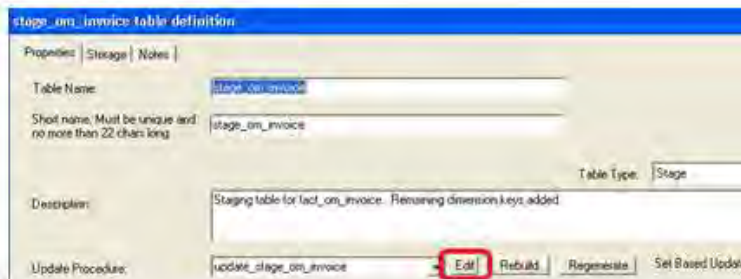
Column: quantity_returned Source table/column: stage_om_invoice3.quantity_returned
 ISNULL(stage_om_invoice3.quantity_returned, 0)

Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

With changes to a column, proceed to creating a new version of the procedure.

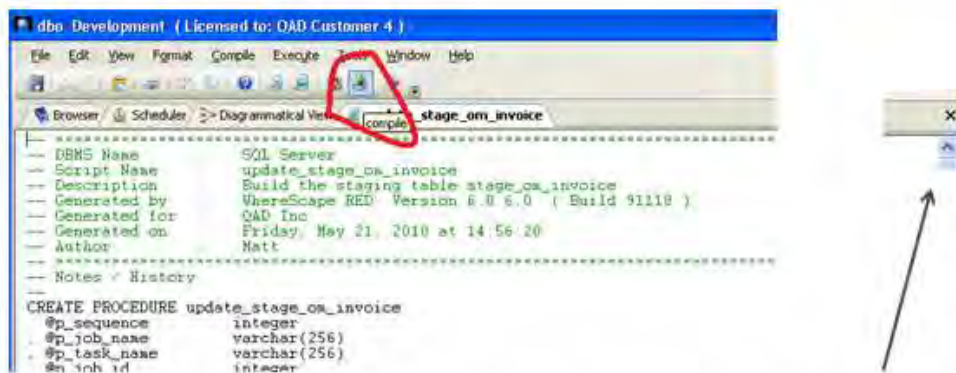
1. First save a version of the existing procedure. In the browser pane, double click on the table name to see properties, then click on the Edit button to open the procedure.



Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

2. With the existing original procedure open, click on the compile button to save a version of it.



3. Close the procedure window using the x in the far right corner of the window.

Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

4. Depending on the extent of the changes to be made, you can either Rebuild or Regenerate the procedure. This was a simple column change with no new data sources added that would require parameter additions or table joins, so just Regenerate. Double click on the table name again to open it. Click the Regenerate button.



Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

- To now do a procedure comparison among version, double click on the table name again to get to the Edit button and open the procedure again. Go to Tools and click View procedure or template.



Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

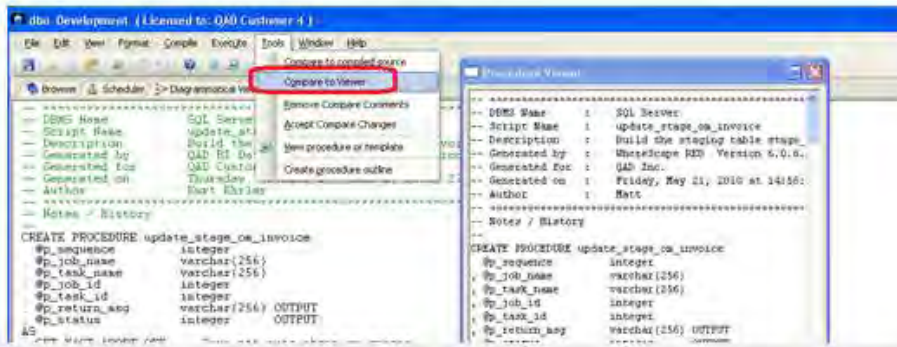
6. A view window will appear. Click the Version drop down menu and pick which version you want to compare against. The most recent version is the current version of the procedure so pick one older than that.



Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

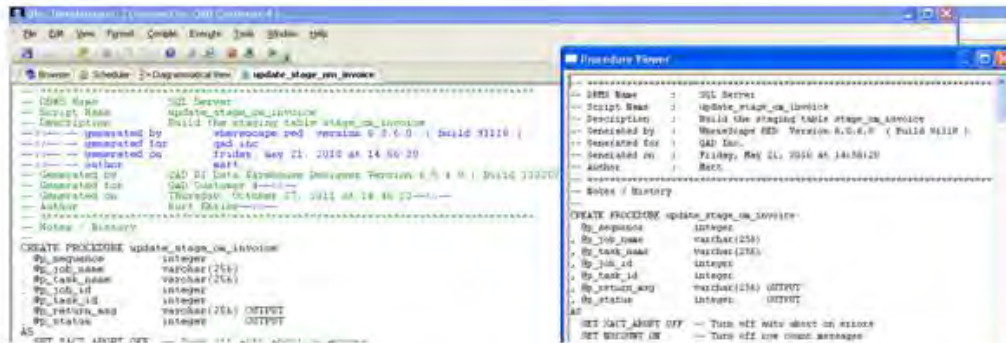
- The selected version of the procedure should now appear in the Procedure View window. Leave that window open and go to the Tools menu again and select Compare to viewer.



Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

- Any differences between the current version of the procedure and what is in the viewer will be displayed in the procedure window (not the Viewer) in blue.



Changing Procedures and Finding Differences Between Procedure Versions

Changing Procedures and Finding Differences Between Procedure Versions

- **Final Notes:** Do not try to recompile procedure code with blue notation, it won't work.
- You should almost never modify a procedure directly in a procedure window anyway unless it's already modified or custom.
- This tool can be particularly useful if it needs to be determined if anybody has made changes to the code before, or to figure out how to change things back to how they were.

Columns that Exist in Some Versions of the ERP in a Table and Not in Others

Columns that Exist in Some Versions of the ERP in a Table and Not in Others

- Sometimes a load table will have a column in one version of the ERP and not in another (think SE vs EE, but this can happen across different versions of EE also). It is possible to handle this situation with a single load table.
- A stored procedure will check the data dictionary load_syscolumns to see if the column exists. If the column does exist, the procedure sets a parameter value to <table_name.column_name>. If it doesn't exist, it gets set to a static value. The load table then references the parameter.

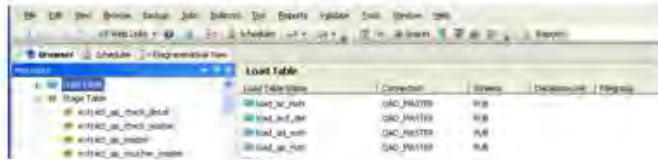
Examples of this are tr_qty_cn_adj and ih_invoicetype.

Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

If we don't already pull in data from a specific table in the ERP, it can be added quite simply using a drag and drop methodology.

1. In the browser pane, double click the Load Table option so that load tables are displayed in the target pane. This is done because a new load table is desired.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

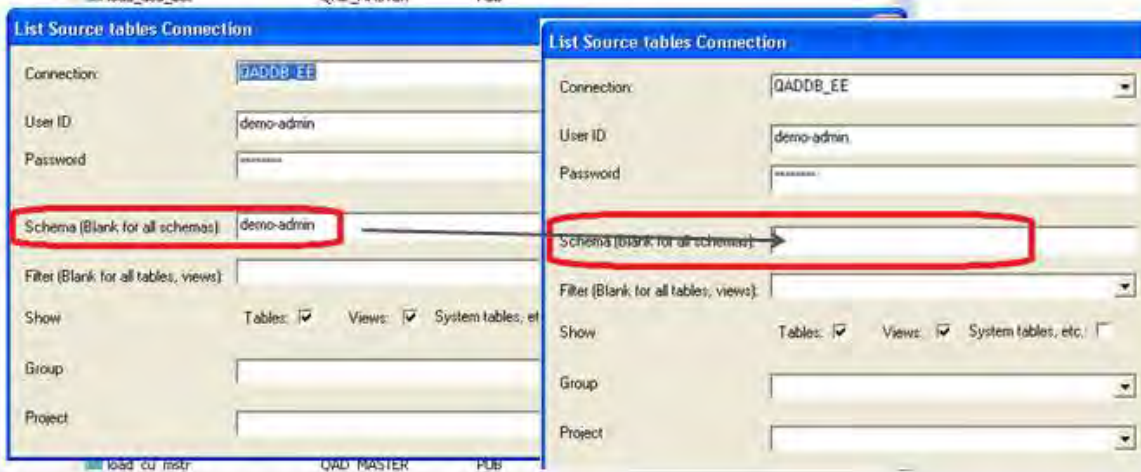
2. On the right hand side of the DWD, click on the yellow change connection icon to choose which data source you want to reference.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

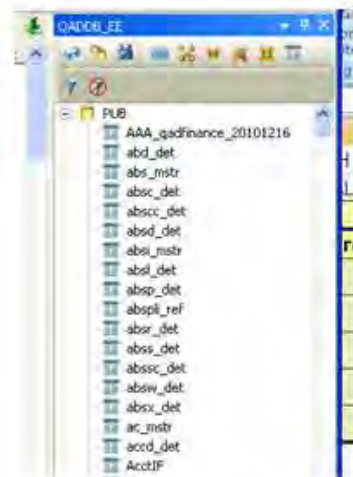
- When picking which data source connection to use, be sure to also remove the Schema for the Connection window.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

4. When the +PUB option appears in the source pane, open it and find the table you want to add in the list.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

5. Some important things to note about the Source pane. There are a number of useful things that can be done in this pane.
 - a) Display data from the table. Right click on the table and select Display Data. Results display in the target pane.

The screenshot shows a QAD BI interface with a data table and a context menu. The table is titled "Data from source table abs_mstr" and has the following columns: abs_shipfrom, abs_id, abs_par_id, abs_shipto, abs_type, abs_status, abs_timezone, and abs_ship_d. The context menu is open over the table, showing options: Display table columns, Display Data (highlighted), Doctor table, Row count, Query with SQL Admin, and Filter.

| abs_shipfrom | abs_id | abs_par_id | abs_shipto | abs_type | abs_status | abs_timezone | abs_ship_d |
|--------------|-------------|--------------|------------|----------|------------|--------------|------------|
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 10-100 | 6SH12131... | 5SH121310... | 10C1003 | s | | | 2010-12-15 |
| 11-100 | 6SH13121... | 5SH131210... | 10C1009 | s | | | 2010-12-15 |
| 11-100 | 6SH13121... | 5SH131210... | 10C1009 | s | | | 2010-12-15 |
| 11-100 | 6SH13121... | 5SH131210... | 10C1009 | s | | | 2010-12-15 |
| 11-100 | 6SH13121... | 5SH131210... | 10C1009 | s | | | 2010-12-15 |

Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

5. Some important things to note about the Source pane. There are a number of useful things that can be done in this pane.
 - b) Get a record count from the table being referenced. Right click on the table name and choose Row count. Result displayed in target.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

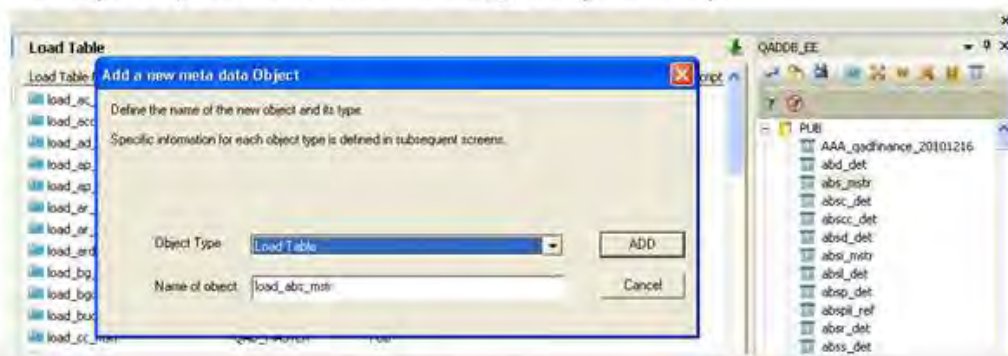
5. Some important things to note about the Source pane. There are a number of useful things that can be done in this pane.
 - c) Launch a SQL Editor window that can run queries against Progress or SQL Server databases.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

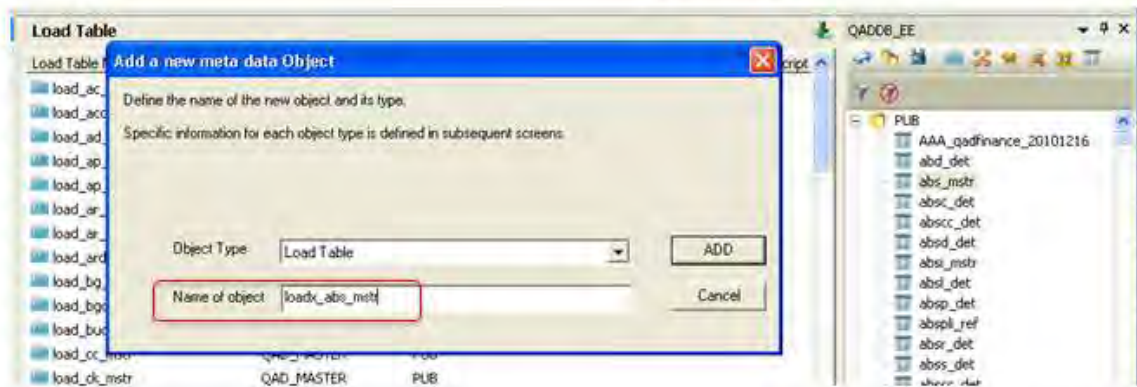
6. To create a new load table from the list of Progress table, left click on the table name in the Source pane and drag it into the Target pane (while all the load tables are displayed in the Target pane).



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

7. Change the name of the object to conform to QAD customization standards if this is a custom object. `load_abs_mstr` should instead be `loadx_abs_mstr`.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

- Put in a description and make sure that the connection is changed to QAD_MASTER.

The screenshot shows a dialog box titled "loadx_abis_mstr table definition" with the following fields and values:

| Field | Value |
|--|-----------------|
| Load Table Name: | loadx_abis_mstr |
| Short name. Must be unique and no more than 22 chars long: | loadx_abis_mstr |
| Description: | |
| Connection: | QADDB_EE |
| Load Type: | Native ODBC |
| Database Link: | |
| Script Name: | |
| Pre-load Action: | Truncate |

Annotations in the image:

- A red arrow points from the text "Be sure to put in a description for what the table is and what it will be used for." to the empty Description field.
- A red arrow points from the text "Be sure to change the connection to QAD_MASTER" to the Connection dropdown menu.

Adding a Table from QAD’s ERP Progress Database

Adding a Table from QAD’s ERP Progress Database

9. Create and populate the table.

| Col name | Display | Data type | Src table | Src column |
|----------------|----------------|-------------|-----------|----------------|
| abs_shipfrom | abs shipfrom | varchar(80) | abs_mstr | abs_shipfrom |
| abs_id | abs id | varchar(80) | abs_mstr | abs_id |
| abs_par_id | abs par id | varchar(80) | abs_mstr | abs_par_id |
| abs_shipto | abs shipto | varchar(80) | abs_mstr | abs_shipto |
| abs_type | abs type | varchar(30) | abs_mstr | abs_type |
| abs_status | abs status | varchar(80) | abs_mstr | abs_status |
| abs_timezone | abs timezone | varchar(30) | abs_mstr | abs_timezone |
| abs_st | | | | date |
| abs_s | | | | time |
| abs_o | | | | date |
| abs_a | | | | me |
| abs_o | | | | date |
| abs_a | | | | me |
| abs_a | | | | date |
| abs_a | | | | me |
| abs_apr_userid | abs apr userid | varchar(80) | abs_mstr | abs_apr_userid |

Create database table

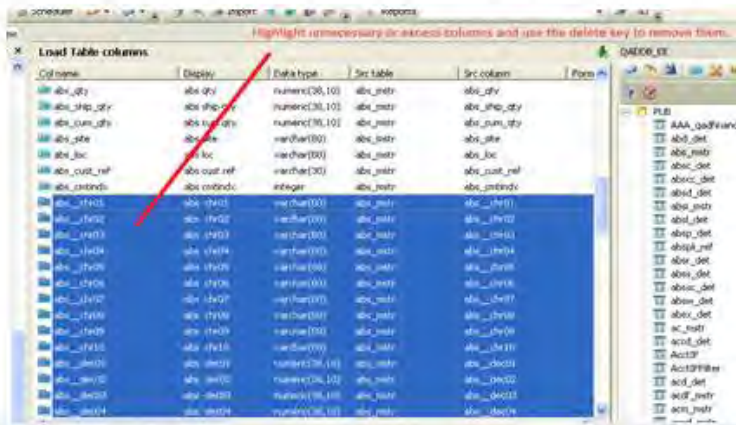
Load Table loadx_abs_mstr has been defined

Buttons: Create and Load, Create, Finish

Adding a Table from QAD’s ERP Progress Database

Adding a Table from QAD’s ERP Progress Database

10. Delete any columns that are unnecessary. In most cases, we try to keep all columns, but some tables have dozens of columns unimportant for the warehouse.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

1. Add additional necessary columns. The elements of these can be copied from other existing load tables. Be sure to assign the right data types and that the columns are last in the list.

- extract_timestamp
- extract_row_id
- process_batch_id
- source_system_code

| id | id | numeric(28,10) | id | id |
|--------------------|-------------------|----------------|--------------------|--------------------|
| extract_timestamp | extract_timestamp | datetime | extract_timestamp | extract_timestamp |
| extract_row_id | extract_row_id | nvarchar(20) | extract_row_id | extract_row_id |
| process_batch_id | process_batch_id | numeric(18) | process_batch_id | process_batch_id |
| source_system_code | source_system ... | nvarchar(64) | source_system_code | source_system_code |

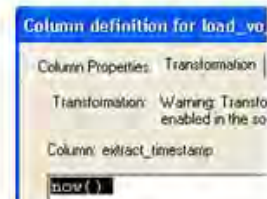
- Be sure to also add the proper transformations for each (see next slide).

Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

12. Assign transformations to the columns in question.

- extract_timestamp -> now()
- extract_row_id -> rowid
- process_batch_id ->
\$PJOB_COM_PROCESS_BATCH_ID\$
- source_system_code ->
'\$PCOM_PROCESS_CONNECTION_NAME\$'
- (be careful copy ' ' s from powerpoint..... They need to be retyped)

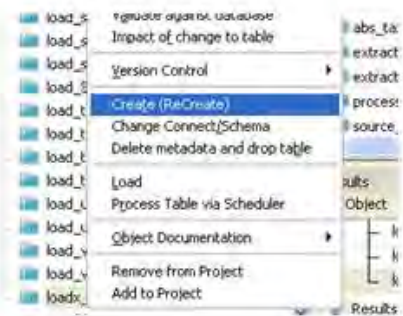


Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

13. Create (ReCreate) the table. Right click on the table name and choose Create (ReCreate).

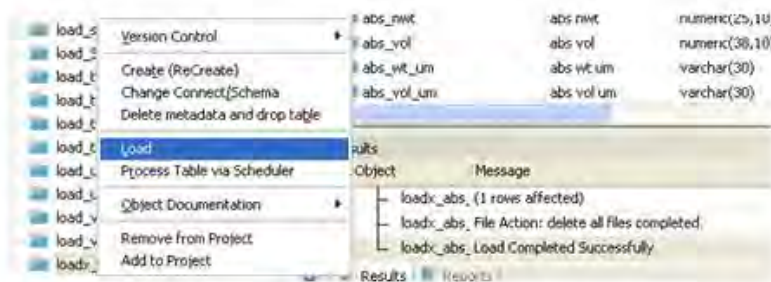
14. Click the Recreate button.



Adding a Table from QAD's ERP Progress Database

Adding a Table from QAD's ERP Progress Database

15. Load the table to confirm that the load script works .

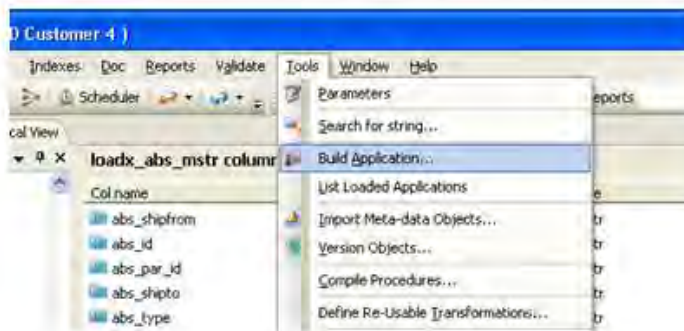


Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

Single patches and complete application suites can be created from the DWD.

1. In the DWD, with the Browser Tab open, go to Tools, Build Application.



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

2. Change the Output Directory to a location you can find later. Name the file in the Application Identifier box. Give it a version. Add a longer Application Name and Description if you desire.

Define and build an Application Distribution

Define an Application distribution | Objects to Add or Replace | Objects to Delete | **Change directory to a memorable location.**

This process builds the files necessary to allow the distribution of a QAD warehouse solution. These files are read and processed by the Setup Administrator utility. Specify an identifier, version and name for the application, and select all the tables and ancillary procedures etc. required.

Output Directory: E:\patches\ Browse

Application Identifier: New table Version: 1

Application Name: New ABS load table

Application Description: Patch with the new ABS load table to meet new customer requirements.

Pre application load SQL (the following optional SQL statement will be issued before the application load commences.)

Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

- Click on the Objects to Add or Replace tab. In the right hand pane, find the items to be added to the patch, in this case the loadx_abs_mstr table and click the Add <- button.



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

4. When finished, click OK button to complete patch. The patch will be generated by creating five .wst files. These can be zipped up and mailed or FTPed elsewhere for implementation into another system.



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

Before continuing on to the install part, four important things.

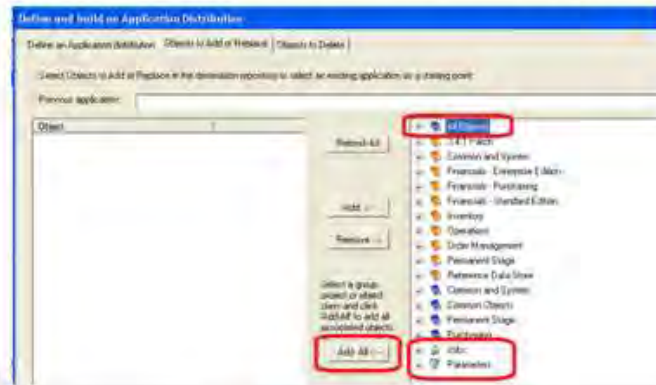
- a) If the five .wst files are zipped up, when the file is unzipped, if the name of the directory is too long (over 12 characters), the Administrator tool may fail to load the file.
- b) For the purpose of demonstrating installing the modules, since we've created a patch from the loadx_abs_mstr table, delete the table from the data warehouse so that the patch will re-install it. Find the table in the Browser tab, right click on it and choose Delete metadata and drop table.

Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

Before continuing on to the install part, four important things.

- c) If the entire application is to be copied, do not forget to also copy the Parameters and the Jobs. Parameters and Jobs are not picked up by the Add All <- button for All Objects.



At one customer site, they believed they had a complete extract of their modules by clicking on All Objects and clicking the Add All <- button. They did not capture their parameters or jobs, so later when their system crashed, they thought they were ready to simply re-install the application suite. Unfortunately with no parameters or jobs, the extract ended up being useless.

Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

Before continuing on to the install part, four important things.

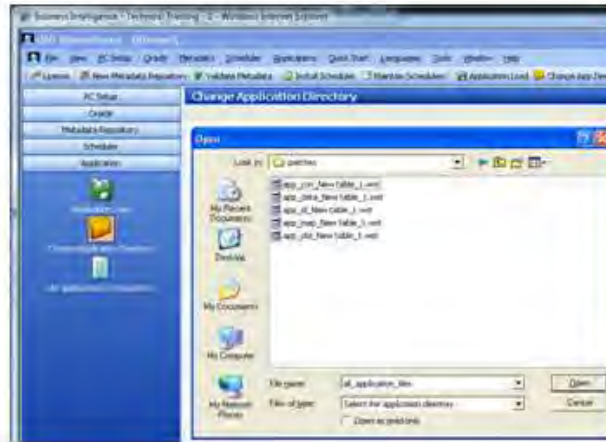
- d) It is advisable if possible, to create a separate database with it's own ODBC connection and Meta-data install that patches will be installed into for comparison purposes before proceeding with the patch install. This will allow comparisons between environments to occur prior to the actual patch installation which will help to address customizations and provide other insights into what is about to be changed.

This hasn't yet been done in class, but if there is time, it would be a good exercise. It shouldn't take very long and would ensure that students review materials from day one.

Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

5. To install the patch that was created, presuming that the file is now on the server that it needs to be, open the Administrator tool, go to the Applications tab, change to the directory where the files are.



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

6. With the new directory open, find the file to be loaded. Right click on the file and click on Install Application .



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

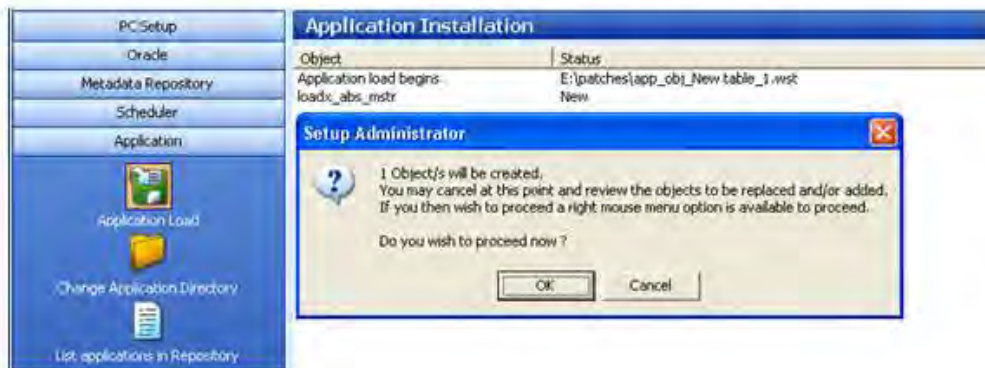
7. Pick the warehouse database in the Application Install Logon window in the Odbc Connect drop down box. Enter User Name and Password if required.



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

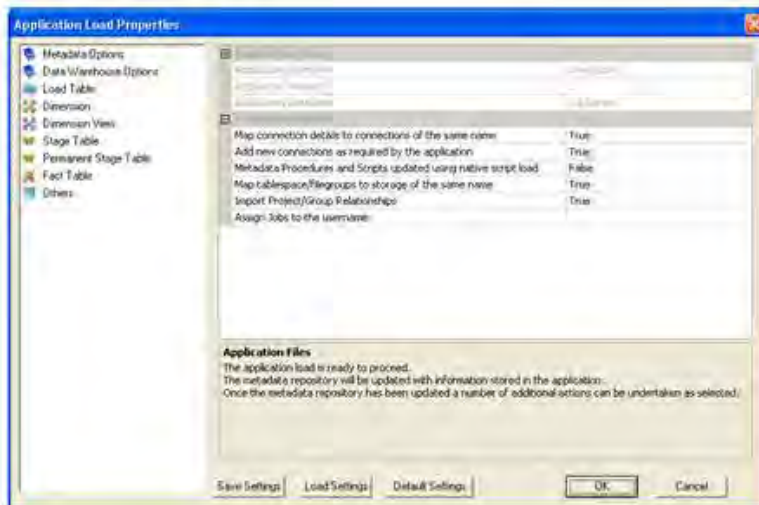
8. An Application Installation window displays what tables will be created or replaced. The option to continue is provided here, in case the patch will overwrite something that shouldn't be.



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

- 9. An additional window provides the option of backing out of the install or changing how patches are treated.



Creating a Patch/Installing a Patch

Creating a Patch/Installing a Patch

10. Patch is installed. Results are posted in the Application Installation window.



11. In the DWD, the table should now appear if the Browser pane is refreshed.

Environment Comparisons

Environment Comparisons

- If multiple data warehouse databases exist in an environment where they can both be reached by ODBC, it's possible to compare the two environments to see what is different between them.

Environment Comparisons

Environment Comparisons

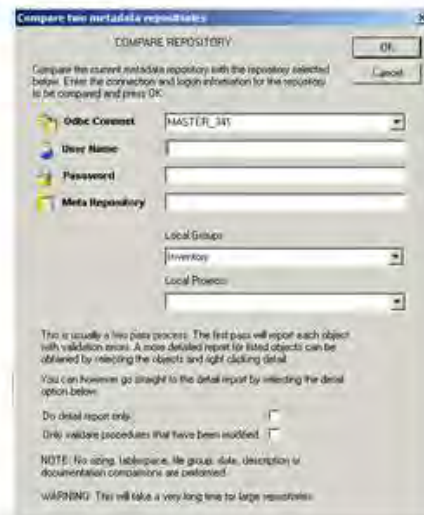
1. With the Browser pane open, click on the Validate menu, then the Compare Meta-data Repository to another... option.



Environment Comparisons

Environment Comparisons

- When the Compare two metadata repositories window opens, choose Odbc Connect drop down and pick the database to compare to. Narrow down the comparison to just local groups and projects if a faster narrower result is desired.



Environment Comparisons

Environment Comparisons

3. The comparison process takes quite a long time, because every procedure is being compared between the two environments. Limiting the query to individual Groups and Projects help speed up the process. Sometimes comparison need to be made originating from each database.

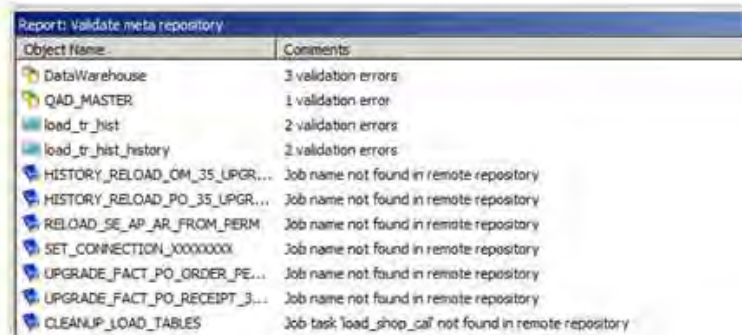


If there are tables or procedures that are in one database and not another, for the comparison process to work, the database that has the table or procedure needs to be the originator of the comparison. In instances where there are databases that just hold patches, it's best to do the comparison from the patch database against the bigger database. If the comparison were to be done the other direction from the bigger database against the patch database, every single item in the bigger database that didn't exist in the patch database would show up as missing or different.

Environment Comparisons

Environment Comparisons

- Any procedures or tables or jobs that have differences will be displayed, including the number of differences.



| Object Name | Comments |
|------------------------------|---|
| DataWarehouse | 3 validation errors |
| QAD_MASTER | 1 validation error |
| load_tr_hist | 2 validation errors |
| load_tr_hist_history | 2 validation errors |
| HISTORY_RELOAD_OM_35_LPGR... | Job name not found in remote repository |
| HISTORY_RELOAD_PO_35_LPGR... | Job name not found in remote repository |
| RELOAD_SE_AP_AR_FROM_PERM | Job name not found in remote repository |
| SET_CONNECTION_XXXXXXXX | Job name not found in remote repository |
| UPGRADE_FACT_PO_ORDER_PE... | Job name not found in remote repository |
| UPGRADE_FACT_PO_RECEIPT_3... | Job name not found in remote repository |
| CLEANUP_LOAD_TABLES | Job task 'load_shop_cal' not found in remote repository |

Environment Comparisons

Environment Comparisons

- To see details of the differences that are found, right click on an individual object that is listed as different, and choose the Details option.

| Report: Validate meta repository | |
|----------------------------------|---|
| Object Name | Comments |
| DataWarehouse | 3 validation errors |
| QAD_MASTER | 1 validation error |
| load_tr_hist | 2 validation errors |
| load_t | 2 validation errors |
| HISTO | Job name not found in remote repository |
| HISTORY_RELOAD_PO_35_UPGR... | Job name not found in remote repository |

- The details of the differences should now appear below the object.

| | |
|----------------------|---|
| QAD_MASTER | 1 validation error |
| load_tr_hist | 2 validation errors |
| load_tr_hist | Load column 'tr_qty_cn_adj' transform model doesn't validate(\$pload_tr_hist_tr_qty_cn_adj\$, \$PTR_QTY_CN_AD.\$) |
| load_tr_hist | Load column 'tr_qty_cn_adj' transform code doesn't validate(\$pload_tr_hist_tr_qty_cn_adj\$, \$PTR_QTY_CN_AD.\$) |
| load_tr_hist_history | 2 validation errors |



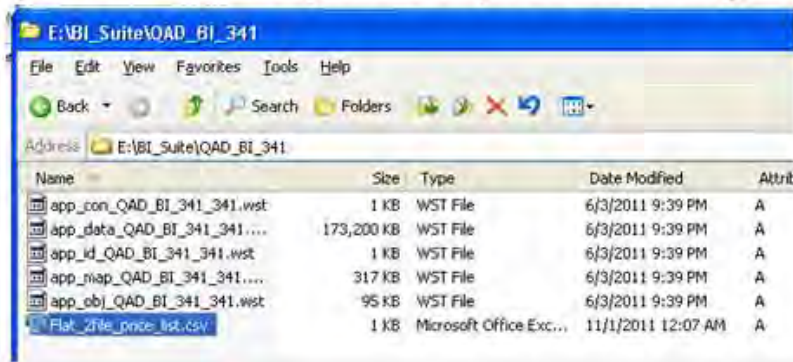
There is also the option to Excel all the results.

Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

Data in flat file like csv files can be accessed just like a database would be. In the following example, a file with unicode characters in it is used which requires a couple extra steps for mapping.

1. Put the Flat_file_unicode_price_list.txt file in an appropriate directory for easy referencing.



Flat_file_unicode_price_list.txt should be provided to the students via email to be retrieved into their training environment so they can use it for this part of the class.

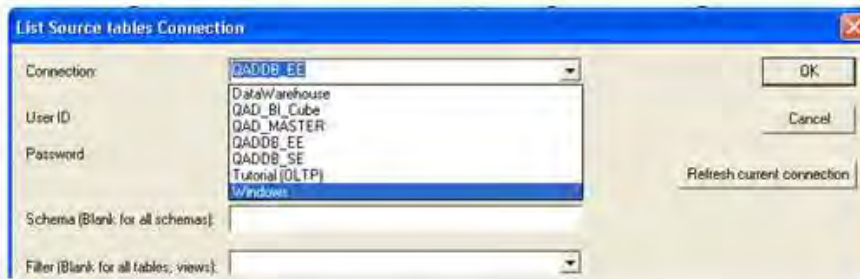
Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

- In the Source pane, click the change the connection button to bring up the source connection window.



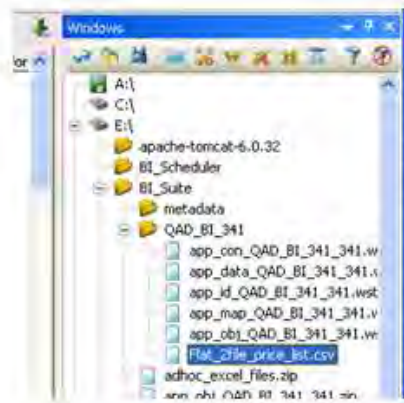
- Choose a Connection of Windows



Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

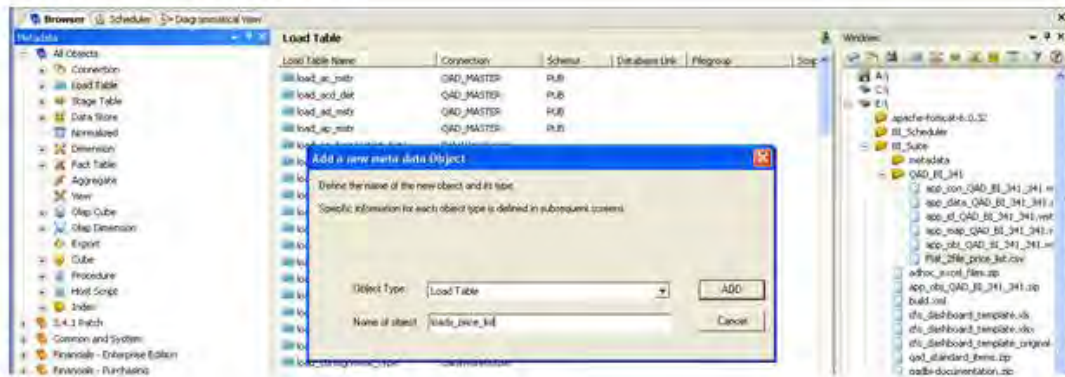
4. In the Source pane, there should now be the various drives of the computer just added. Find the file to be loaded.



Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

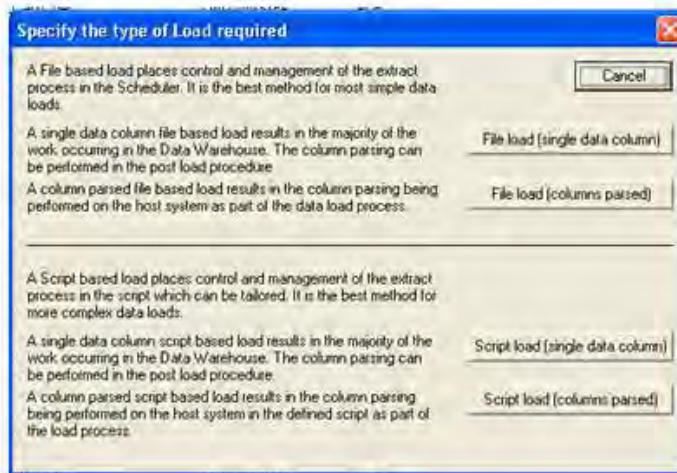
- In order to create a new load table from the flat file, first double click on the Load table folder in the Browser pane to get all the load tables to appear in the Target pane. Then, to create the table from the flat file, drag and drop it into the Target pane. Be sure to properly name the table.



Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

- When asked to Specify the type of Load required, choose the File load (columns parsed) option (for this file).



Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

7. For the table definition, put in a useful description.

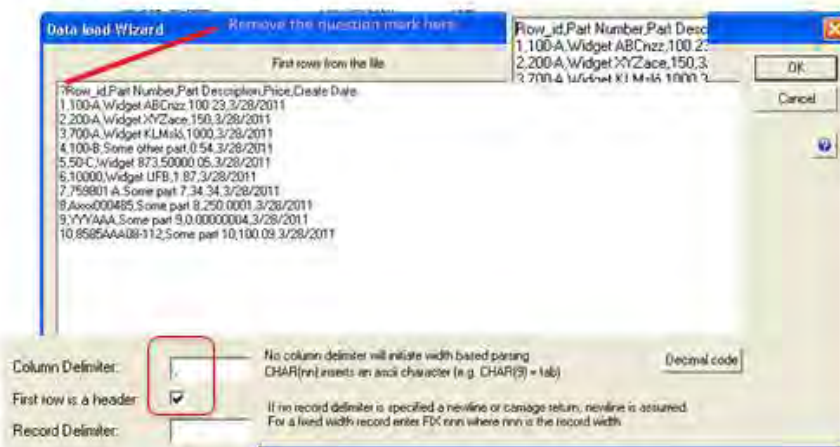
The screenshot shows a dialog box titled "loadx_price_list table definition" with the following fields and values:

| Field | Value |
|---|--|
| Load Table Name: | loadx_price_list |
| Short name. Must be unique and no more than 22 chars long | loadx_price_list |
| Description: | New price list table from the unicode flatfile |
| Connection: | Windows |
| Load Type: | File load |
| Database Link: | |
| Script Name: | |
| Pre-load Action: | Truncate |

Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

- When the Data Load definition comes up, remove the ? symbol that appears at the beginning of the data. Set Column Delimiter to , and check the First row is a header box

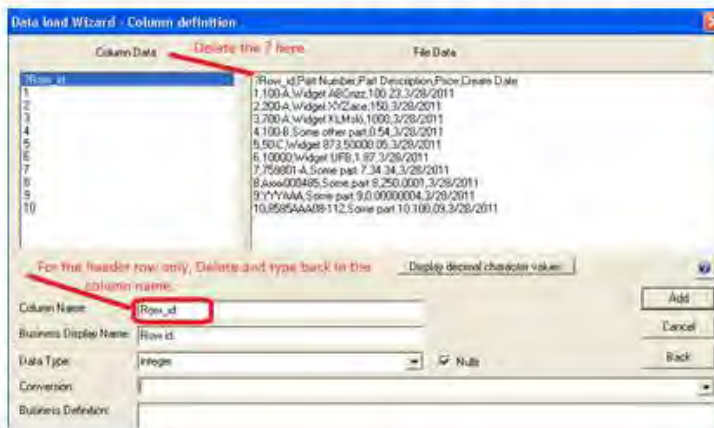


Recently in a class where Chinese characters were added to the flat file, all of the rows ended up being in double quotes for the whole row. This will cause various problems with the load. Make sure that the files the customer are presenting are not in double quotes.

Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

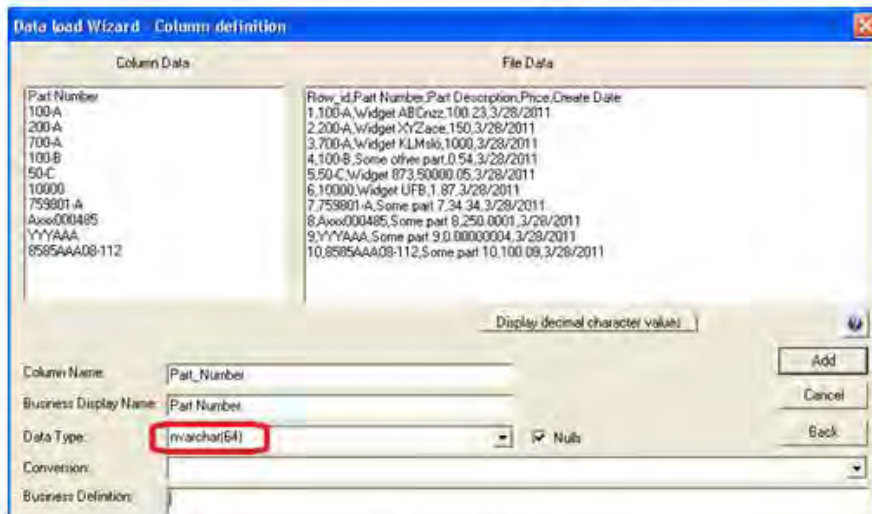
- For the header row, delete the ? in the File Data box. Also, for the Column Name, remove the Row_id and type it back in. There is an invisible character here that needs to be deleted. Click the Add button to get to the next column.



Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

10. For the next columns, any columns that are varchar, change them to nvarchar. Add business definitions for columns as appropriate.



Importing a Flat File with Foreign Characters

Importing a Flat File with Foreign Characters

- When all the columns have been added, click Yes to CREATE and LOAD the table. Note because we didn't replace the Row_id for Display name, the invisible ? appears here. It can simply be edited after the table is created.

The screenshot displays the QAD BI Data Warehouse Designer interface. At the top, a 'Load Table' dialog box is open, showing a table with the following columns:

| Col name | Display | Data type | Src table | Src column | Format |
|------------------|------------------|---------------|---------------------------|------------|--------|
| Row_id | ?Row id | integer | Flat_2file_price_list.... | COL1 | |
| Part_Number | Part Number | nvarchar(64) | Flat_2file_price_list.... | COL2 | |
| Part_Description | Part Description | nvarchar(64) | Flat_2file_price_list.... | COL3 | |
| Price | Price | numeric(13,8) | Flat_2file_price_list.... | COL4 | |
| Create_Date | Create Date | datetime | Flat_2file_price_list.... | COL5 | |

Below the 'Load Table' dialog, a confirmation dialog box titled 'QAD BI Data Warehouse Designer' is displayed with the message: 'Load table loadx_price_list has been defined. Do you wish to CREATE and LOAD the table now?' with 'Yes' and 'No' buttons.

To the right, a 'Results' window shows the execution output:

```

Object      Message
-----
loadx_pric | FIRSTROW = 2
loadx_pric |
loadx_pric | 10 rows loaded
    
```

Running Jobs Stand-Alone.

Running Jobs Stand-Alone.

- If one of the jobs, such as a single historic job needs to be run stand-alone, to prevent any other jobs from launching when it is completed, set the Parameter `JOB_CHAINING_ENABLED` to N. The `submit_next_job` procedure at the end of the job list will not launch the next job if this parameter is set to N.

HISTORIC RELOAD

HISTORIC RELOAD

1. Check the scheduler Run/Fail button to make sure no jobs are running.
2. If there is a chance of the historic reload running so long that it overlaps with the scheduled DAILY_START job, put the DAILY_START job on hold.
3. Set parameter JOB_CHAINING_ENABLED to N.
4. Make sure that the QAD_MASTER is set to the right connection as well as the connection parameters. In 3.5 on up, this can be achieved by running the SET_CONNECTION_<connection_name> job.

HISTORIC RELOAD

HISTORIC RELOAD

5. Run the PERM_EXTRACT to get the data into the extract table for the module. If the data sets for the customer are very large, make a custom version of PERM_EXTRACT that will only load the relevant module tables.
6. If all the fact tables for the module are to be loaded FOR ALL data source, truncate the tables. If the data to be loaded is for one of many sources, delete by the source_system_code to be loaded for each fact table. In the event that only one fact table needs to be reloaded, make a cloned custom job from the main job and strip it down to only load the necessary fact(s).

HISTORIC RELOAD

HISTORIC RELOAD

7. Start the HIST_<module_name> job and run to completion.
8. Validate the data.
9. Set the DAILY_START job back to Daily for it's scheduled time.
10. DONE!

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

1. Search for and cleanup problem Progress tables/columns.
 - \$DLC/bin/dbtool
{full_path_to_database}/{db_name} 2>
{output_file_name}.{extension}
 - **For example:**
\$DLC/bin/dbtool /dr01/qad/2011se/db/seprod
2> seprod_dbtool.log



The dbtool portion of the class is for advanced technical users of Progress, to better explain how to track down overpacked columns that may cause the data load to hang.

This is usually handled by a customer's Progress DBA, and can be skipped over in the class if the audience is inappropriate, but it should at least be noted that this tool exists and it is how a customer's Progress DBA would go about finding overpacked columns. It should be run by the Progress DBA prior to doing the Warehouse implementation.

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

2. In the Database Tools menu, select option number 1: SQL Width & Date Scan w/Report Option

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

3. At the <connect> prompt, enter the appropriate connection type:
 - If a database broker is not running, you must select zero (0)
 - If a database broker is running, enter the number of threads to use for the scan.
 - **For example:** Enter 1 for a self-service client connection against a database with an active server process

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

4. At the <table> prompt, enter "all" to scan all tables
5. At the <area> prompt, enter "all" to scan all storage areas

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

- 8. At the <display> prompt, enter a value of 0 to 3 to specify the amount of activity logging performed, with 0 being the least verbose and 3 producing the most detailed entries.
 - **Recommendation:** Use logging level of 1, 2 or 3 to obtain the names of tables/fields with violations. 0 produces a summary count of violations only.

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

7. Open the dbtool output file in a text editor. Any database table columns having values exceeding the defined column width (MAX-WIDTH attribute) are identified with *** in the report's Error column. If no fields are shown to be in violation of the defined MAX-WIDTH, the database is ready to be used as input for QAD BI. However, if fields are flagged as having values which exceed the MAX-WIDTH, perform the following steps to rectify the problems before attempting to use this database as a source for QAD BI.

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

8. Using the table name, field name and SQLWidth values from the dbtool report, run a query to obtain the key fields for the records having the long values.
 - **For Example:**
for each code_mstr where length(code_value) > 80:
disp code_domain code fldname code_value.

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

9. Using the results of the above queries, determine if the long column values in each table are necessary and appropriate.
 - If the values are not correct and appropriate, write database commands to update the field values to comply with (be less than or equal to in length of) the SQLWidth value in the dbtool report.
 - If the values must be retained, the SQL column width (MAX-WIDTH) attribute for the field must be increased.



Using the results of the above queries, determine if the long column values in each table are necessary and appropriate.

If the values are not correct and appropriate, write database commands to update the field values to comply with (be less than or equal to in length of) the SQLWidth value in the dbtool report.

If the values must be retained, the SQL column width (MAX-WIDTH) attribute for the field must be increased.

This can be done by dumping the table's data definitions into a .df file and altering the file to include a MAX-WIDTH attribute which meets or exceeds the length of the longest value in the field.

Alternatively, run dbtool again but this time select option 2, SQL Width Scan w/Fix Option and specify an appropriate padding percentage (5 is recommended). This option updates the field to have a SQL Width (MAX-WIDTH) value of the longest data value in the field plus the padding percentage.

Pre-run dbtool Progress Database Check

Pre-run dbtool Progress Database Check

10. Run dbtool again using option 1: SQL Width & Date Scan w/Report Option. When no fields are shown to be in violation of the defined SQL Width, the database is ready to be used as input into QAD BI.

Eb2 – Pre-domain Environments Integration Into our BI Solution.

Eb2 – Pre-domain Environments Integration Into our BI Solution.

- There is a document for customizing the BI environment to work with pre-domain environments, but recent efforts have been made to instead create a schema filled with views for the BI system to reference. The schema views have faked domains and other key elements necessary for a pre-domain environment to be pulled into the BI environment. QAD's services team should have access to this solution by the 3.9 release if not sooner.

The Pieces so far- How This all Ties Together

The Pieces so far- How This all Ties Together

- SQL Server
 - Where the data warehouse resides
- Administrator tool used for configuration for all the parts of the DWD tool, including:
 - License key
 - Odbc connections
 - Metadata implementation
 - Scheduler set-up
 - Applications/patches/languages install.
- DWD
 - Main interface to the set up and loading of data

Summarization

Summarization

- Customization review
- Dbtools explanation
- Added a column to a table.
- Showed how to do procedure versioning.
- Added a table to the data warehouse, including from a flat file outside data source.
- Created and installed a patch.
- Compared database environments.
- Explanation of a historic reload.

Parking Lot Review ?



Upon completion of Level 1 Certification you will be able to do the following:

1. Demonstrate core functionality – while you will not yet be a SME knowing all the subtle nuances of the product, you will be able to utilize the product being taught and demonstrate its high level functionality.
2. Guide customers to Best Practices – this program will help you to understand what are best practices around the particular product you are studying and why they are best practices. This will enable you to speak confidently in making recommendations for business improvements.
3. Recommend an appropriate Solution – perhaps worse than never recommending one of our products is recommending a product to address an issue it was never designed to address. Following your certification, you should be able to identify appropriate opportunities for solutions.
4. Execute upon a SME Guided Project Plan - following Level 1 certification you should not just start working with EAM all by yourself. You should be able, though, to take direction and guidance from a SME and then execute upon that mentoring.

5. Begin customer facing activities – whether you are in support, pre-sales, services, or some other area of expertise, following your Level 1 Certification you will be equipped to begin working face-to-face with customers and properly represent the solution. You should begin using your education immediately! In general:

- Support Resources will be able to address T1 and T2 issues with some assistance from a SME
- Services Resources will be able to be primary resources on Implementations working alongside a Mentor
- Pre-Sales Resources will be able to provide customer-specific product demonstrations with support from a Mentor

www.qad.com

© 2013 QAD Inc