rittmanmead

DELIVERED INTELLIGENCE

What to Know Before Selecting OWB

Stewart Bryson, Technical Director, Rittman Mead America

ODTUG Kaleidoscope 2009, Monterey, June 2009

T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

rittmanmead
DELIVERED INTELLIGENCE

Who Am I?

- Oracle BI/DW Architect and Delivery Specialist
- Technical Director of North America for Rittman Mead
  - Oracle BI/DW Project Delivery Specialists
  - New to the US Market
- 10+ years with Oracle Database, OWB, BI Stack
- Presenter at various conferences and user groups
- Developer of Transcend and Evolve Frameworks
  - Object and File Management
  - ETL Tool Plugin (Including OWB)
  - Part of Rittman Mead Rapid Deployment Framework

rittmanmead
DELIVERED INTELLIGENCE

Rittman Mead Consulting

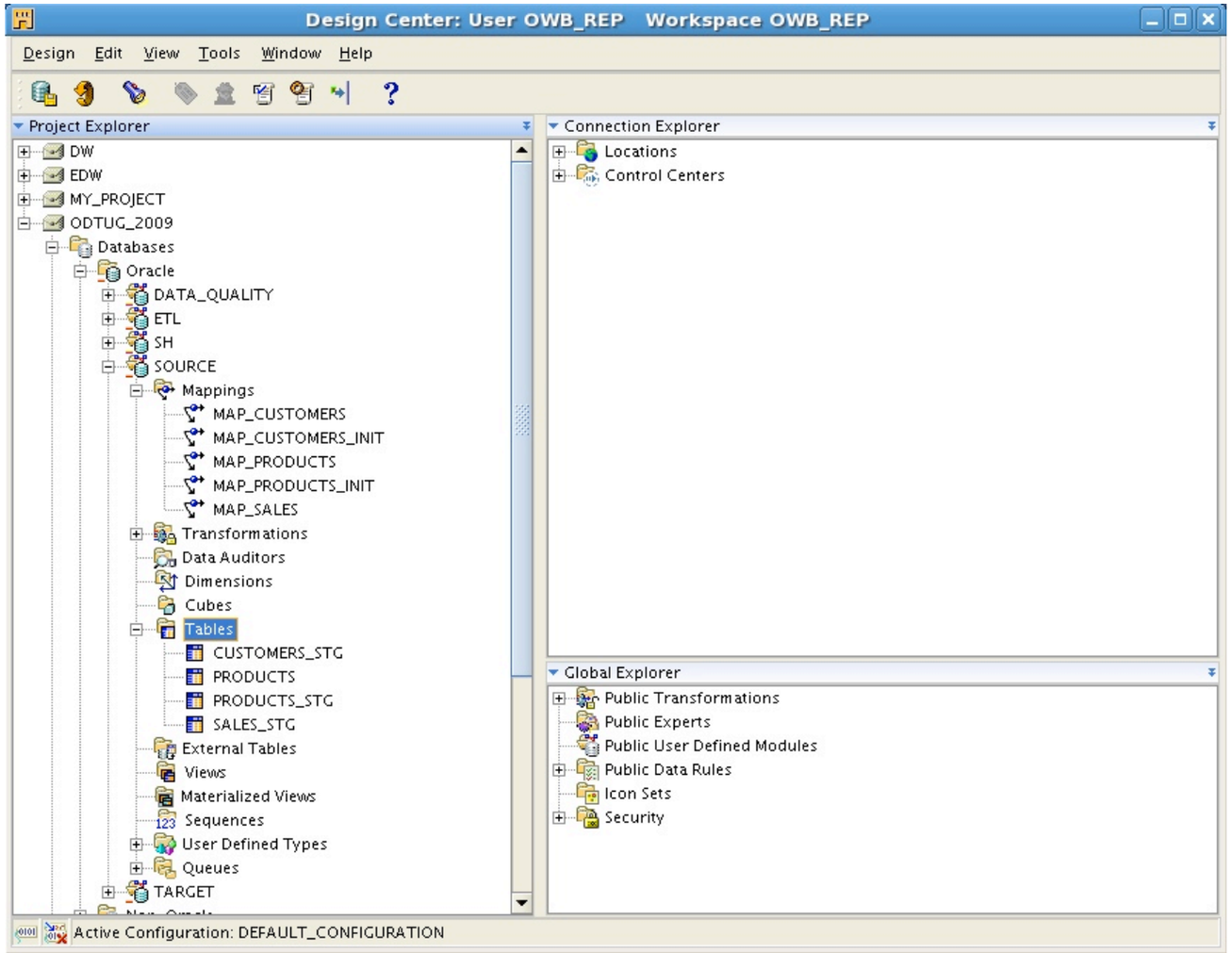
- Oracle BI&DW Project Specialists providing consulting, training and support
- Clients in the UK, USA, Europe, Middle-East
  - Recently Opened for Business in the USA
- Voted UKOUG BI Partner of the Year 2008

T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

rittmanmead
DELIVERED INTELLIGENCE

What is OWB?

- Full life-cycle ETL tool integrated with Oracle Database
- In-Database ETL
  - Basic Mappings and Process Flows
  - Basic Metadata management
- Enterprise ETL
  - Dimensions
  - Cube Loading
  - Pluggable Mappings
  - Advanced Metadata management
- Additional Add-ons
  - Data Profiling, Data Correction and Data Auditing
  - Accessing ERP Applications



T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

rittmanmead
DELIVERED INTELLIGENCE

Mappings: Loading Data from Sources to Targets

- Generate PL/SQL for most mappings
  - Set-based (Default)
  - Row-based
- Variety of Source Operators supported
  - Tables and Views
  - Table Functions
  - Flat Files (External Tables preferred)
- Multiple Target Operators supported in one mapping
  - Target Load Order
  - Splitters
- Best Practice: One target table per mapping
  - Use Staging Tables to hold records between mappings
  - Error or Discard tables are exceptions to the rule

rittmanmead
DELIVERED INTELLIGENCE

Is OWB Simply a PL/SQL Generator?

- Mappings generate PL/SQL Packages when deployed
  - Multiple internal calls for auditing and registration
  - Declarations for Constants and Input Parameters
  - Pre- and Post-Mapping processes
  - Code necessary for set-based and row-based deployment
- Set-Based Generation
  - Mappings are not merely PL/SQL FOR or FORALL loops
  - Execute as single statements
  - Generated code is optimized with Oracle features
- Row-Based Generation when Set-Based code is not possible

rittmanmead
DELIVERED INTELLIGENCE

OWB is Really a SQL Generator

Generation Results

Generation style: Intermediate | Operating mode: [PL/SQL] SET_BASED | Aspect: Loading | Explain | Statistics | SQL Tuning | View Tuning

Script | Message

```
INSERT
/*+ APPEND PARALLEL("PROMOTIONS_FACT") */
INTO
  "DIM"."PROMOTIONS_FACT"
  ("PROMO_ID",
  "PROMO_NAME",
  "PROMO_SUBCATEGORY",
  "PROMO_BEGIN_DATE",
  "PROMO_END_DATE",
  "DURATION_DAYS")
  (SELECT
  "AGGREGATOR"."PROMO_ID$1" "PROMO_ID",
  "AGGREGATOR"."PROMO_NAME$1" "PROMO_NAME",
  "AGGREGATOR"."PROMO_SUBCATEGORY$1" "PROMO_SUBCATEGORY",
  "AGGREGATOR"."PROMO_BEGIN_DATE$1" "PROMO_BEGIN_DATE",
  "AGGREGATOR"."PROMO_END_DATE$1" "PROMO_END_DATE",
  "AGGREGATOR"."DURATION_DAYS$1" "DURATION_DAYS"
FROM
  (SELECT
  "PROMOTIONS_VW"."PROMO_ID"/* AGGREGATOR.OUTGRP1.PROMO_ID */ "PROMO_ID$1",
  "PROMOTIONS_VW"."PROMO_NAME"/* AGGREGATOR.OUTGRP1.PROMO_NAME */ "PROMO_NAME$1",
  "PROMOTIONS_VW"."PROMO_SUBCATEGORY"/* AGGREGATOR.OUTGRP1.PROMO_SUBCATEGORY */ "PROMO_SUBCATEGORY$1",
  "PROMOTIONS_VW"."PROMO_BEGIN_DATE"/* AGGREGATOR.OUTGRP1.PROMO_BEGIN_DATE */ "PROMO_BEGIN_DATE$1",
  "PROMOTIONS_VW"."PROMO_END_DATE"/* AGGREGATOR.OUTGRP1.PROMO_END_DATE */ "PROMO_END_DATE$1",
  COUNT("PROMOTIONS_VW"."PROMO_ID")/* AGGREGATOR.OUTGRP1.DURATION_DAYS */ "DURATION_DAYS$1"
FROM
  "STAGE"."PROMOTIONS_VW"  "PROMOTIONS_VW"
  WHERE
  ( "PROMOTIONS_VW"."WEEKDAY_FLAG" = 'Y' )
GROUP BY
"PROMOTIONS_VW"."PROMO_NAME", "PROMOTIONS_VW"."PROMO_SUBCATEGORY",
"PROMOTIONS_VW"."PROMO_BEGIN_DATE", "PROMOTIONS_VW"."PROMO_END_DATE","PROMOTIONS_VW"."PROMO_ID"/* AGGREGATOR */) "AGGREGATOR"
  )
;
```

rittmanmead
DELIVERED INTELLIGENCE

Transformations in Set-Based Mappings

- Executed via the SELECT statement
- Joiner Operators
  - Inner
  - Outer (Right, Left, Full)
- Set Operators
  - Union, Union All
  - Minus
- Expression Operators
  - Majority of transformation logic
  - Commonly implemented by CASE statements
- Pivot, Unpivot Operators
- Aggregator Operators

Generation Results

Generatio... Intermediate Operating mode: [PL/SQL] SET... Aspect: Incoming Explain Statistics SQL Tuning View Tu

Script Message

```
1  SELECT
2    TO_CHAR( "ESP_CUSTOMER_T"."CUST_NB" )/* TO_CHAR_CUST_NB.OUTGRP1.CUSTOMER_ID */ "CUSTOMER_ID",
3    case
4  when LOWER( 'Incremental' ) in ('h','hist','historical','full')
5  then TO_DATE('01/01/1993','mm/dd/yyyy')
6  else "ESP_CUSTOMER_T"."CUST_UPDATE_DT"
7  end/* GET_EFFECTIVE_DATE.OUTGRP1.EFFECTIVE_DATE */ "EFFECTIVE_DATE",
8    TRIM( "ESP_CUSTOMER_T"."CUST_NM" )/* TRIM_CUST_NM.OUTGRP1.CUSTOMER_NAME */ "CUSTOMER_NAME",
9    "ESP_CUSTOMER_T"."CUST_COMPANY_NB" "CUST_COMPANY_NB",
10   "ESP_CUSTOMER_T"."CUST_ACCT_NB" "CUST_ACCT_NB",
11   "ESP_CUSTOMER_T"."CUST_ACCT_SUFFIX_TX" "CUST_ACCT_SUFFIX_TX",
12   "ESP_CUSTOMER_T"."CUST_SALESFORCE_ID" "CUST_SALESFORCE_ID",
13   "ESP_CUSTOMER_T"."CUST_SALESPERSON_CD" "CUST_SALESPERSON_CD",
14   "ESP_CUSTOMER_T"."CUST_SALESPNT_NB" "CUST_SALESPNT_NB",
15   "ESP_CUSTOMER_T"."CUST_INDUSTRY_CD" "CUST_INDUSTRY_CD",
16   "ESP_CUSTOMER_T"."CUST_ACCT_TP" "CUST_ACCT_TP",
17   "ESP_CUSTOMER_T"."CUST_CONTACT_NM" "CUST_CONTACT_NM",
18   "ESP_CUSTOMER_T"."CUST_CONTACT_TITLE_TX" "CUST_CONTACT_TITLE_TX",
19   "ESP_CUSTOMER_T"."CUST_ADDR1_TX" "CUST_ADDR1_TX",
20   "ESP_CUSTOMER_T"."CUST_ADDR2_TX" "CUST_ADDR2_TX",
21   "ESP_CUSTOMER_T"."CUST_CITY_NM" "CUST_CITY_NM",
22   NULL ,
23   "ESP_CUSTOMER_T"."CUST_STATE_CD" "CUST_STATE_CD",
24   "ESP_CUSTOMER_T"."CUST_ZIP_CD" "CUST_ZIP_CD",
25   "ESP_CUSTOMER_T"."CUST_PHONE_NB" "CUST_PHONE_NB",
26   "ESP_CUSTOMER_T"."CUST_PHONE_EXT_NB" "CUST_PHONE_EXT_NB",
27   "ESP_CUSTOMER_T"."CUST_FAX_NB" "CUST_FAX_NB",
28   "ESP_CUSTOMER_T"."CUST_COMPANY_NM" "CUST_COMPANY_NM",
29   "ESP_CUSTOMER_T"."CUST_COMPANY_PARENT_NB" "CUST_COMPANY_PARENT_NB",
30   "ESP_CUSTOMER_T"."CUST_COMPANY_PARENT_NM" "CUST_COMPANY_PARENT_NM",
31   "ESP_CUSTOMER_T"."CUST_SLA_FG" "CUST_SLA_FG",
32   "ESP_CUSTOMER_T"."CUST_INDUSTRY_NM" "CUST_INDUSTRY_NM",
33   "ESP_CUSTOMER_T"."CUST_INDUSTRY_DIVISION_CD" "CUST_INDUSTRY_DIVISION_CD",
34   "ESP_CUSTOMER_T"."CUST_INDUSTRY_DIVISION_NM" "CUST_INDUSTRY_DIVISION_NM"
35 FROM
36   "STAGING"."ESP_CUSTOMER_T"@"DWD1@STAGING_LOCATION"  "ESP_CUSTOMER_T"
```

rittmanmead
DELIVERED INTELLIGENCE

PL/SQL Transformation Operators

- Custom PL/SQL imported into OWB
  - Functions are the natural choice as they return a value
  - Procedures with OUT parameters allow multiple return values
- Can allow organizations to ease into OWB
- Available in Set-Based or Row-Based Mappings
- Implemented as standalone program units or in a package
  - Changing Package Specs cannot be reimported gracefully
  - Modified Specs are identified as new functions or procedures
- Hide complex transformations
  - Major benefit of OWB is it's self-documenting
  - Impact Analysis cannot proceed past a PL/SQL Transformation Operator
- Best Practice: use as a last resort

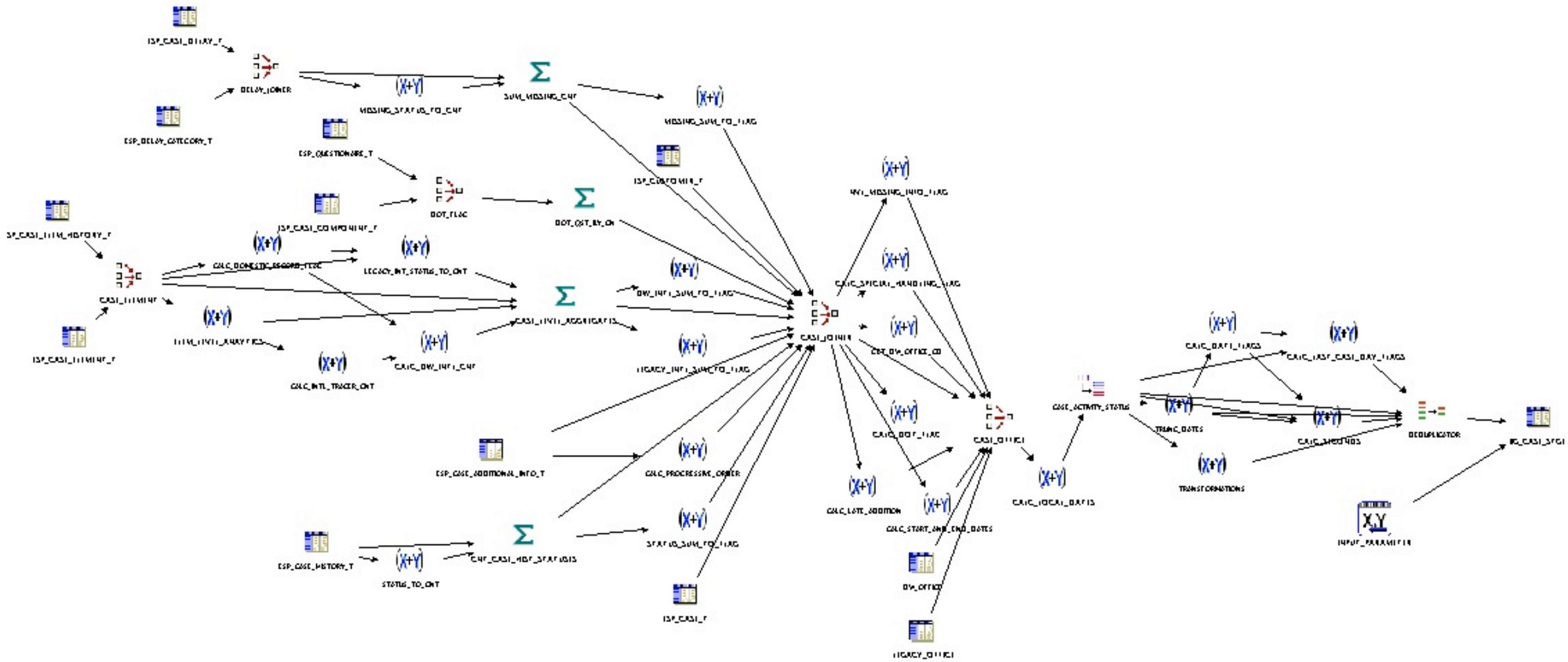
rittmanmead
DELIVERED INTELLIGENCE

OWB Paradigm: Set-Based Processing

- Common to load Data Warehouses in row-by-row fashion
  - PL/SQL based processes often use lots of Cursors and FORALL's
  - Standard ETL tools load in least common denominator fashion
- OWB stresses Set-Based loads
  - Generates Oracle-specific DML for better performance
  - Mapping deployments warn when row-by-row is required
- In OWB, a mapping that loads millions of rows from multiple source tables to multiple target tables with unlimited transformations is usually done with a single statement

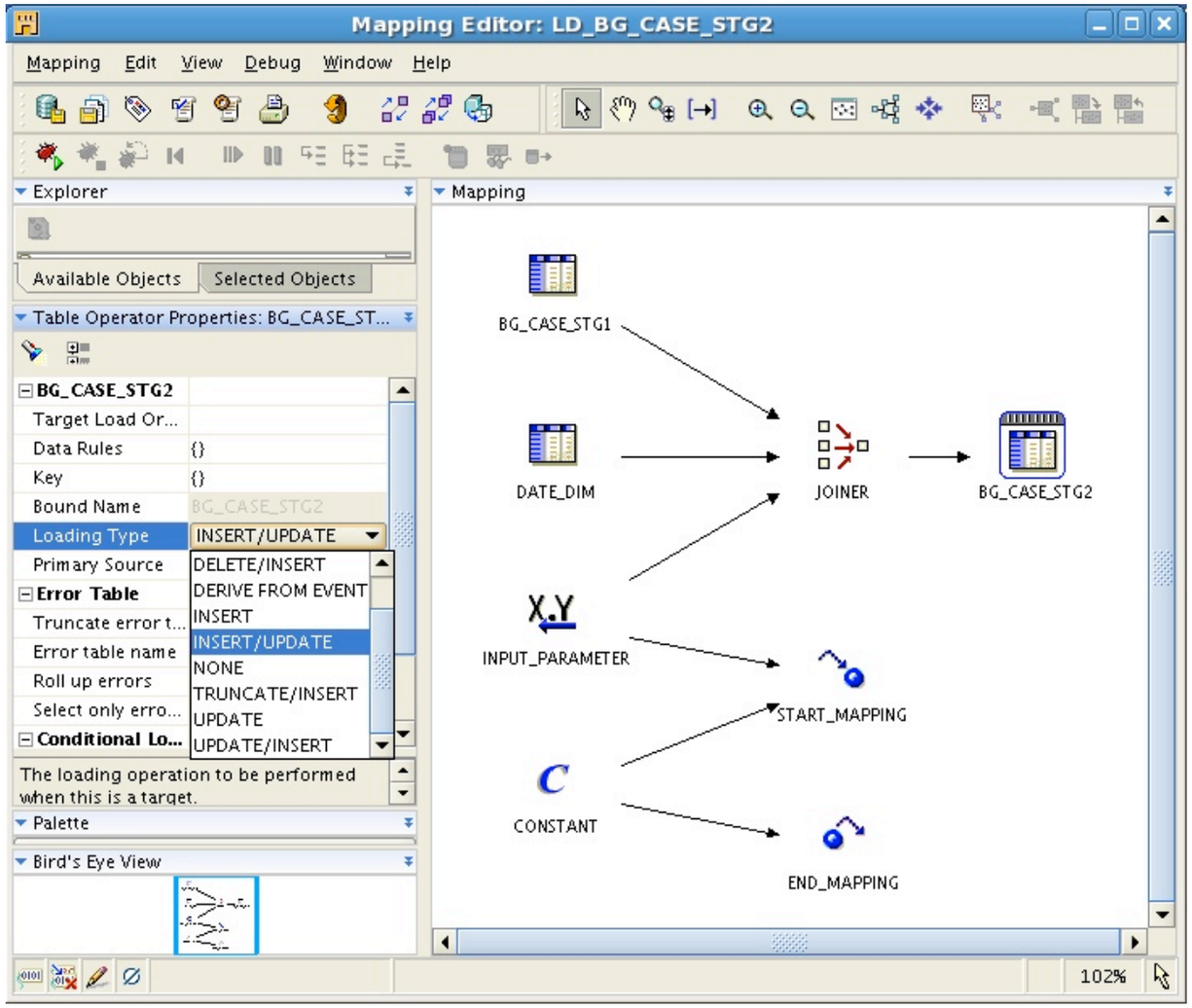
rittmanmead
DELIVERED INTELLIGENCE

One Mapping... One Statement!



MERGE Statement

- Update/Insert or Insert/Update Target Types
  - Matching, Updating and Inserting for each column
  - A Delete portion can also be configured
- Regular Update Target Load Types
  - MERGE without the Insert portion
  - Outperforms a standard Update
- Merge Optimization with Expression Operators
  - Moves the Expression out of the Select
  - Places expression in Insert or Update portion



rittmanmead
DELIVERED INTELLIGENCE

Multi-Table INSERT

- Uses a single statement to load multiple target tables
- Useful in Reject scenarios
- Generated from Splitter Operators
  - Whenever “Optimized Code” radio button is checked
  - Otherwise, multiple Insert statements are generated
- Each Outgroup is configured with a Split Condititon
- REMAINING_ROWS Outgroup is optional

```
1300 INSERT
1301 /*+ APPEND PARALLEL("BG_CASE_FACT") APPEND PARALLEL("BG_CASE_FACT_REJ") */
1302 ALL
1303   WHEN "SPLIT_COND_ALIAS" = 'N'/* REJECT_SPLITTER.OUTGRP1 */ THEN
1304     INTO
1305       "TARGET"."BG_CASE_FACT"
1306       ("CUSTOMER_KEY",
1307       "CUST_OFFICE_KEY",
1308       "LEGACY_CUST_OFFICE_KEY",
1309       "BG_CASE_KEY",
1310       "CASE_ADDED_DATE_KEY",
1311       "CASE_CLOSED_DATE_KEY",
1312       "AUDIT_KEY",
1313       "SOURCE_SYSTEM",
1314       "SOURCE_CASE_ID",
1315       "ELAPSED_SECONDS",
1316       "BUSINESS_SECONDS_EST",
1317       "BUSINESS_SECONDS_LOCAL",
1318       "LEGACY_BUSINESS_SECONDS")
1319     VALUES
1320       ("CUSTOMER_KEY$2",
1321       "OFFICE_KEY$2",
1322       "LEGACY_OFFICE_KEY$2",
1323       "BG_CASE_KEY$2",
1324       "ADDED_DATE_KEY$6",
1325       "CLOSED_DATE_KEY$6",
1326       "AUDIT_KEY$7",
1327       "SOURCE_SYSTEM$6",
1328       "SOURCE_CASE_ID$6",
1329       "ELAPSED_SECONDS$6",
1330       "BUSINESS_SECONDS_EST$6",
1331       "BUSINESS_SECONDS_LOCAL$6",
1332       "LEGACY_BUSINESS_SECONDS$6")
1333
1334   ELSE
1335     INTO
1336       "SOURCE"."BG_CASE_FACT_REJ"
1337       ("CUSTOMER_KEY",
1338       "CUST_OFFICE_KEY"
```

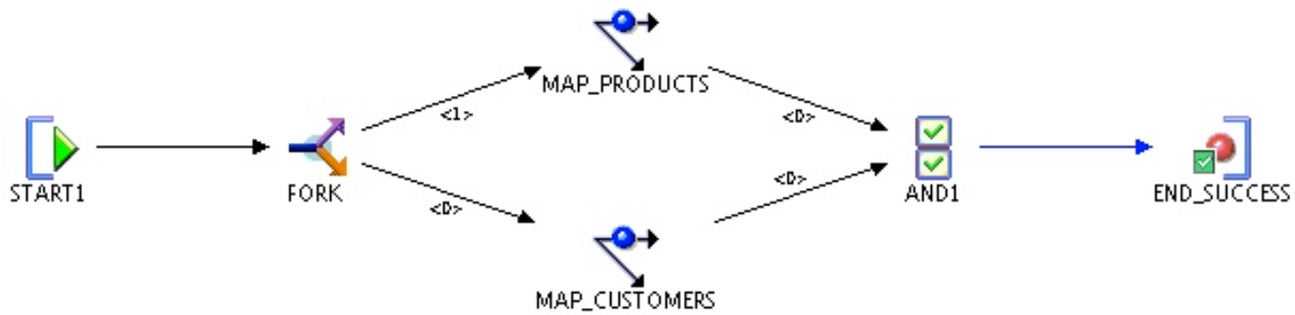
rittmanmead
DELIVERED INTELLIGENCE

Process Flows: Orchestrating Mappings

- Implemented using Oracle Workflow
- Use Process Flows to design high-level and low-level groupings
  - Mappings
  - PL/SQL Procedures or Functions
  - SQL-Plus scripts
  - External processes
  - Notifications processes (Email)
  - Other Process Flows
- Each Process Flow has a local scope
  - Parameters
  - Variables
  - Error handling

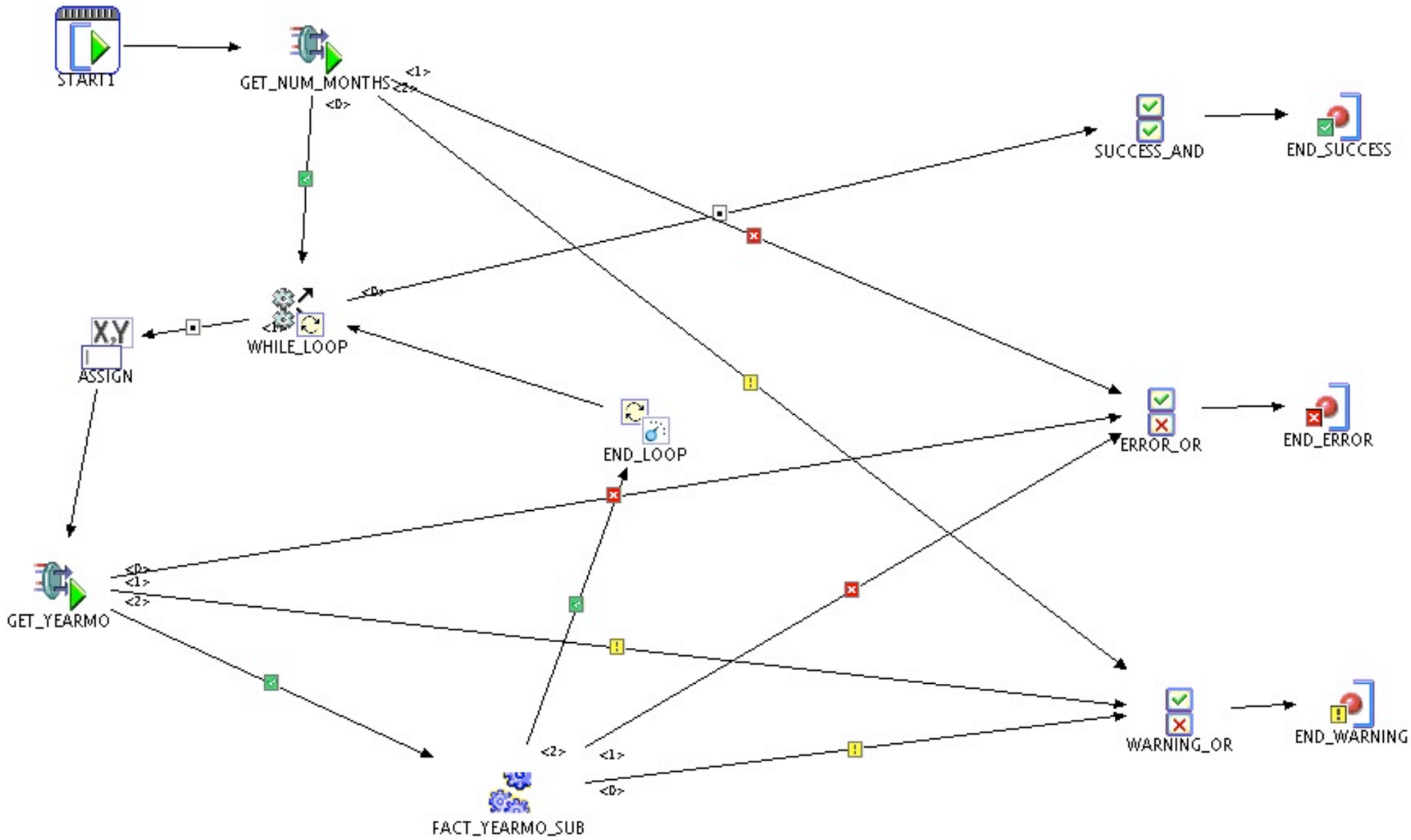
Control Structures

- Fork Operators allow for concurrent processing
  - Supports all Process Flow Activities
  - AND Activity brings processes back together
- WHILE and FOR LOOP's allow scripting-like functionality
- Parameters and Variables supported for value substitution



rittmanmead
DELIVERED INTELLIGENCE

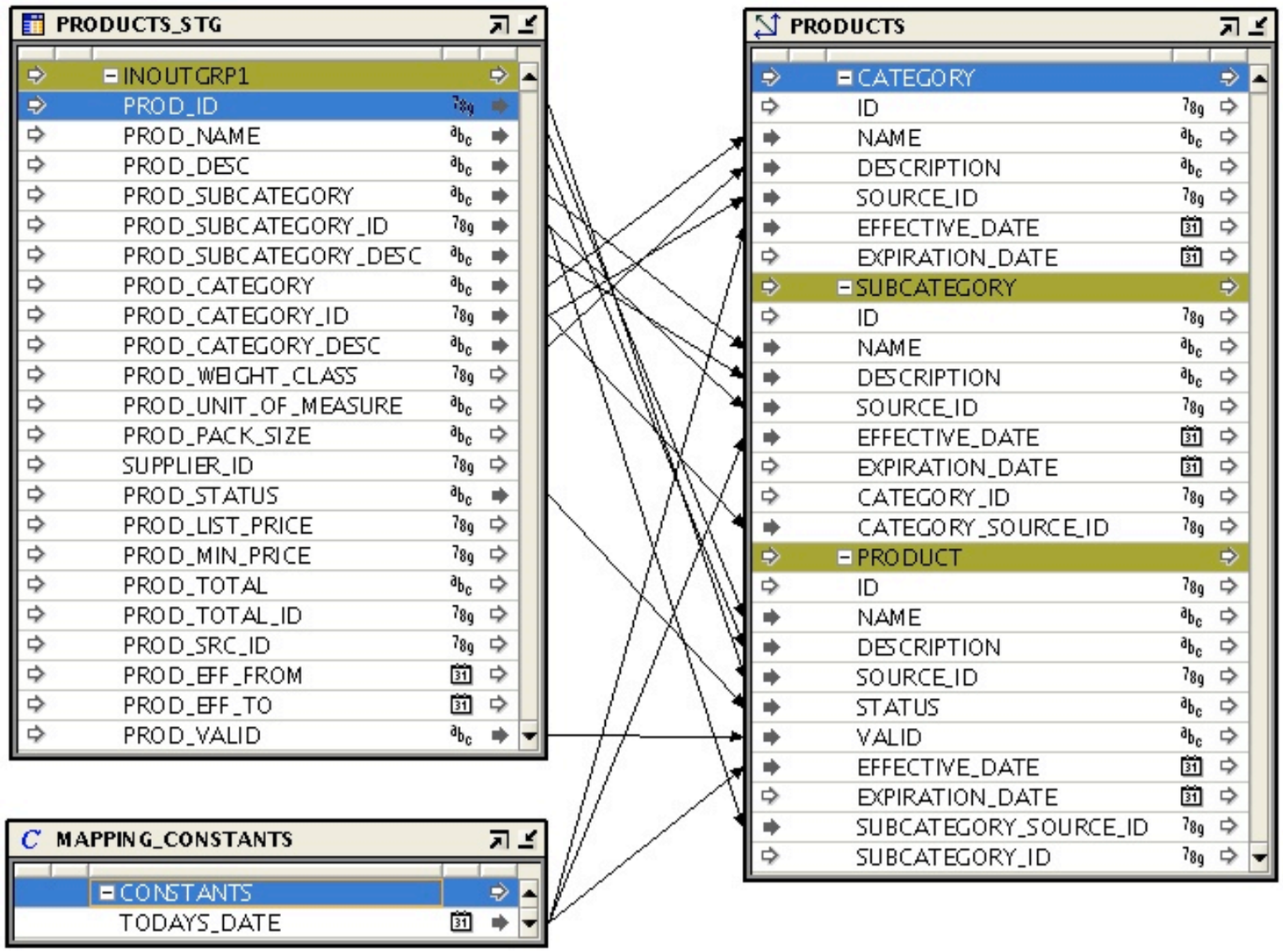
Process Flow Example: While Loop with Error Handling



rittmanmead
DELIVERED INTELLIGENCE

OWB ETL Accelerators: Dimension and Cube Operators

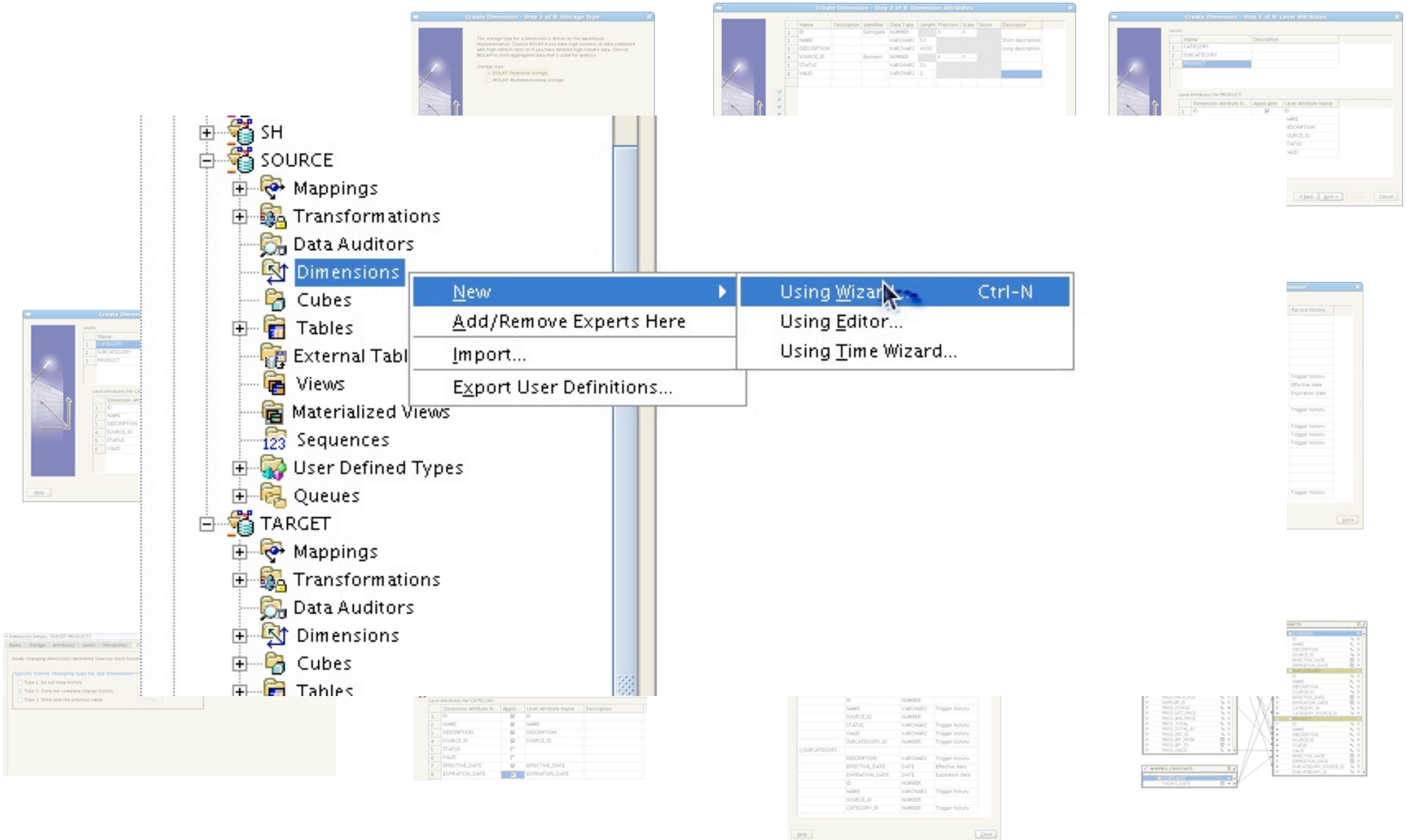
- Integrate with Database Dimensions
  - Define levels
  - Define level attributes
  - Define hierarchies across levels
- Generate Metadata for Report Frameworks
- Provide a mechanism for facilitating a Star Schema load



T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

rittmanmead
DELIVERED INTELLIGENCE

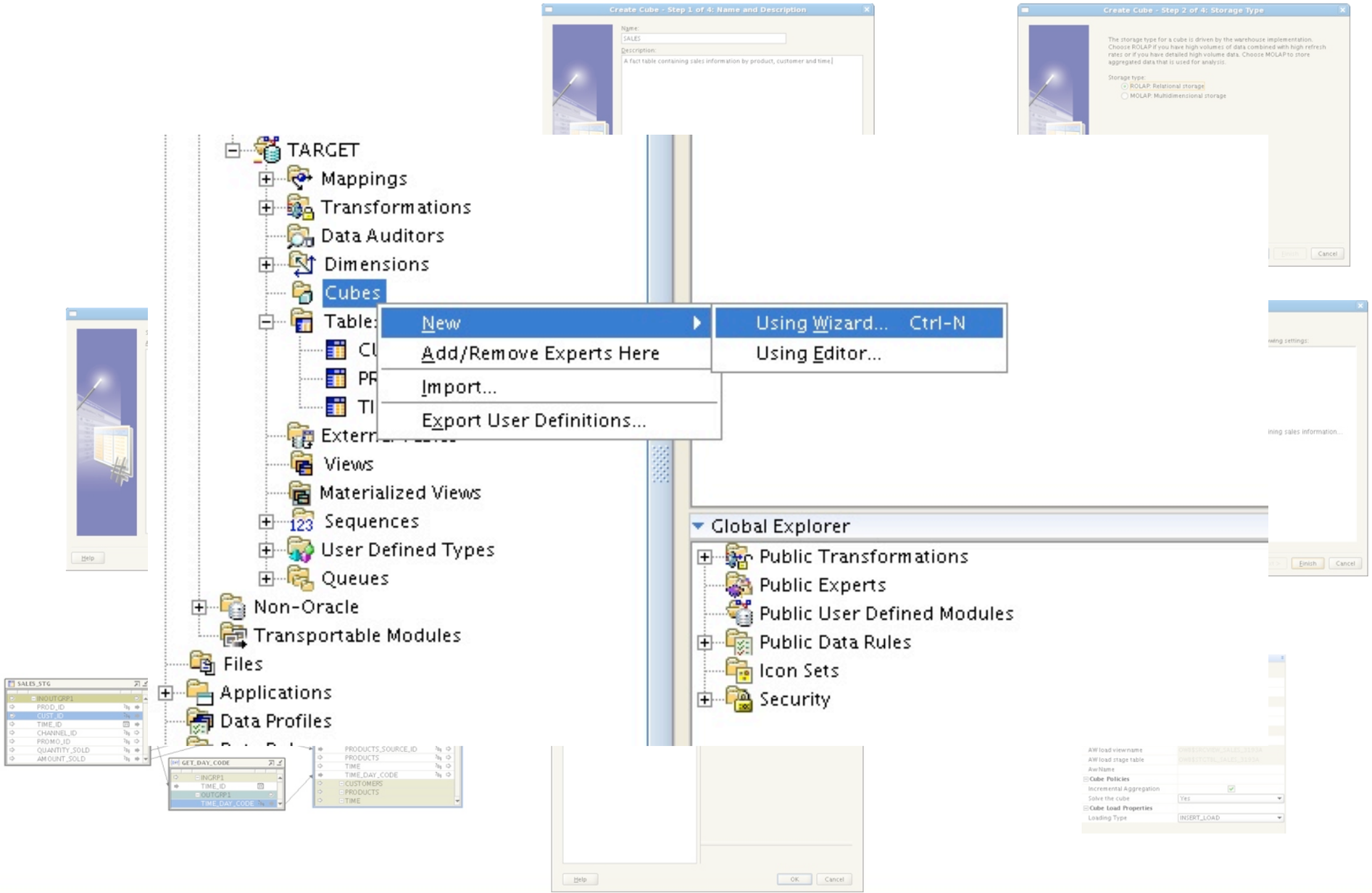
Creating an SCD2 Dimension: Example



T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

rittmanmead
DELIVERED INTELLIGENCE

Loading Fact Tables with Cube Operators: Example



T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

OWB ETL Operators: Too Focused on Levels?

- Level-based SCD's
  - Surrogate Keys at every level
  - Effective and Expiration Dates for every Level
- Inclusion of "Solved" Dimension rows
- Dimension load logic is grouped in levels

Dimension Key	Category Key	Category	Sub-Category Key	Sub-Cateogry	Product Key	Product
-310	-310	Peripherals and Accessories				
-342	-310	Peripherals and Accessories	-342	Monitors		

rittmanmead
DELIVERED INTELLIGENCE

Solved Dimensions versus Shrunken Dimensions

- Shrunken Dimensions are Dimension Tables with lower level attributes removed
- Shrunken Dimensions versus Solved Dimensions
  - Solved Dimensions require conditional logic
    - Negative Surrogate Keys
    - NULL Attributes
  - Shrunken dimensions require multiple versions of the same Dimension Table
    - Most BI Reporting and Dashboard products are aggregate aware
- Cube Operators don't provide the ability to load Aggregates using Solved Dimensions
- Oracle Query Rewrite is better than any of these alternatives
  - Database Optimizer handles Aggregate Navigation
  - True Aggregates can be added and dropped like indexes

Slowly Changing Dimensions (SCD's)

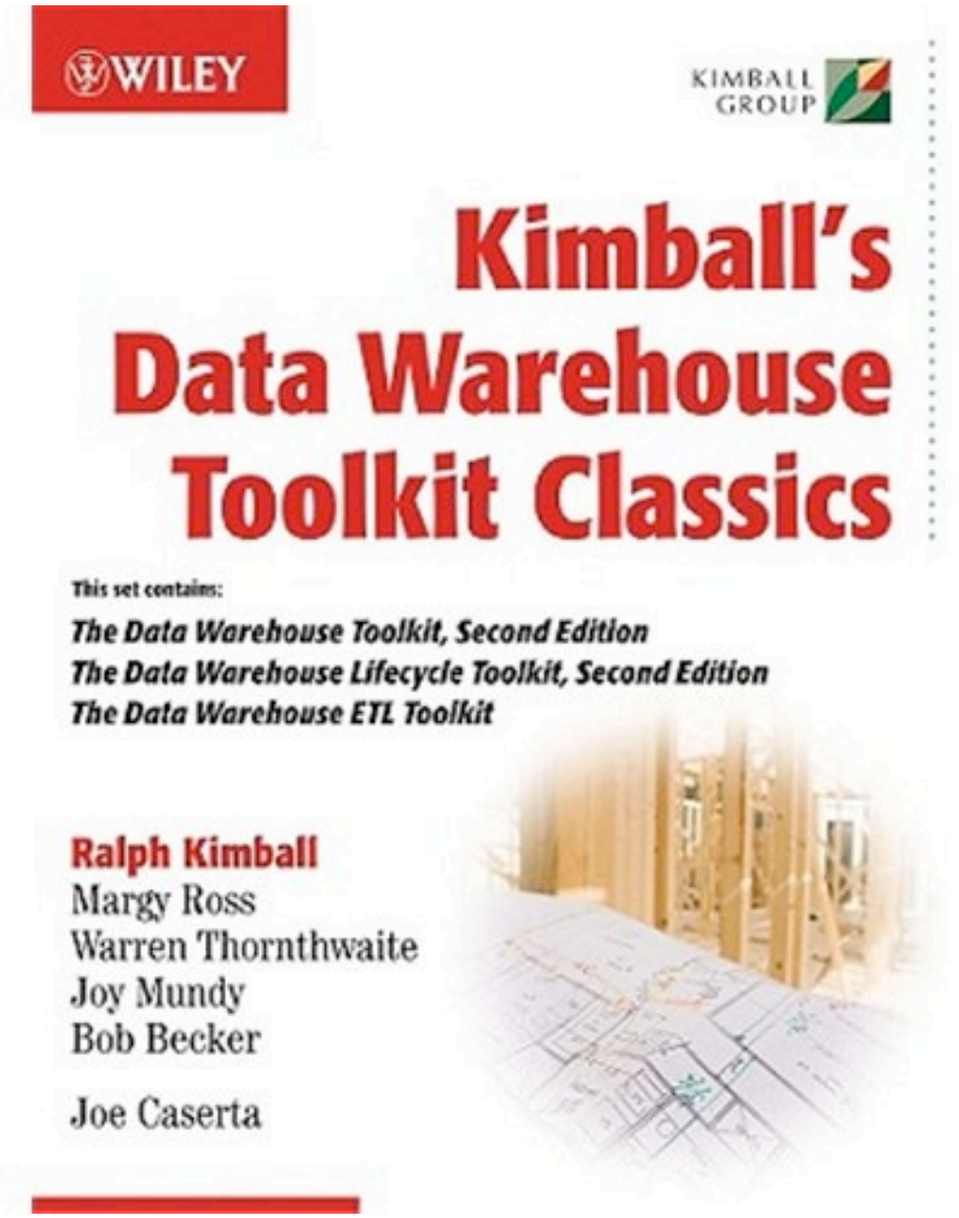
- Type 1: Overwriting
  - History is ignored
  - Only the most current value is recorded
- Type 2: Inserting a new row
  - Full history is retained
  - The usual standard in data warehouses
- Type 3: Updating a different attribute in the same row
  - Partial history is retained
  - The last two values are recorded

Surrogate Key	Business Key	Attribute
201	14	17" LCD w/built-in HDTV Tuner

Surrogate Key	Business Key	Attribute
201	14	22" LCD w/built-in HDTV Tuner

OWB: Non-Standard SCD Handling

- SCD Types (1, 2, and 3) were defined by Ralph Kimball
  - *The Data Warehouse Toolkit,* John Wiley & Sons, 1996
  - *The Data Warehouse Lifecycle Toolkit,* John Wiley & Sons, 1998
  - *The Data Warehouse ETL Toolkit,* John Wiley & Sons, 2004
- Kimball never discusses Level-based SCD handling
  - Surrogate Keys only exist for the Dimension
  - Attribute changes are only recorded for the Dimension
  - Effective Dates for the Dimension only

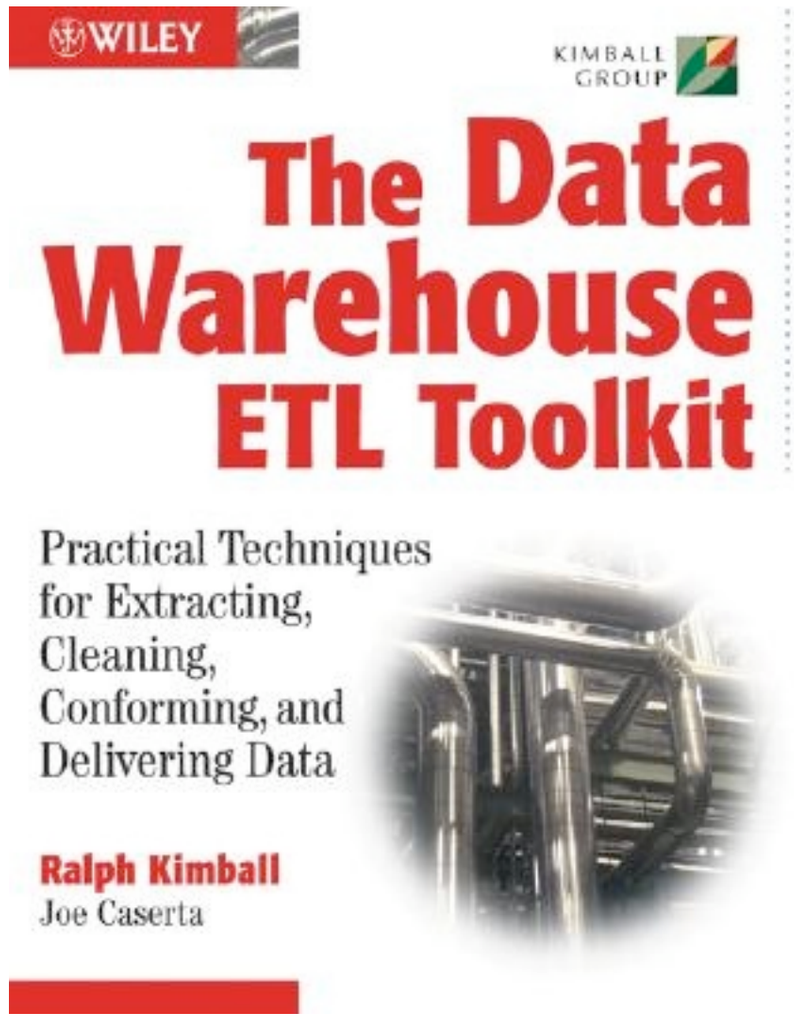


rittmanmead
DELIVERED INTELLIGENCE

Defining an SCD Type for the Entire Dimension?

"It is common to have a dimension containing both Type 1 and Type 2 fields. When a Type 1 field changes, the field is overwritten. When a Type 2 field changes, a new record is generated. In this case, the Type 1 change needs to be made to all copies of the record possessing the same natural key. In other words, if the ethnicity attribute of an employee profile is treated as a Type 1, if it is ever changed (perhaps to correct an original erroneous value), the ethnicity attribute must be overwritten in all the copies of that employee profile that may have been spawned by Type 2 changes."

R. Kimball & J. Caserta, *The Data Warehouse ETL Toolkit,* John Wiley & Sons, 2004, page 193



Hybrid SCD Dimensions

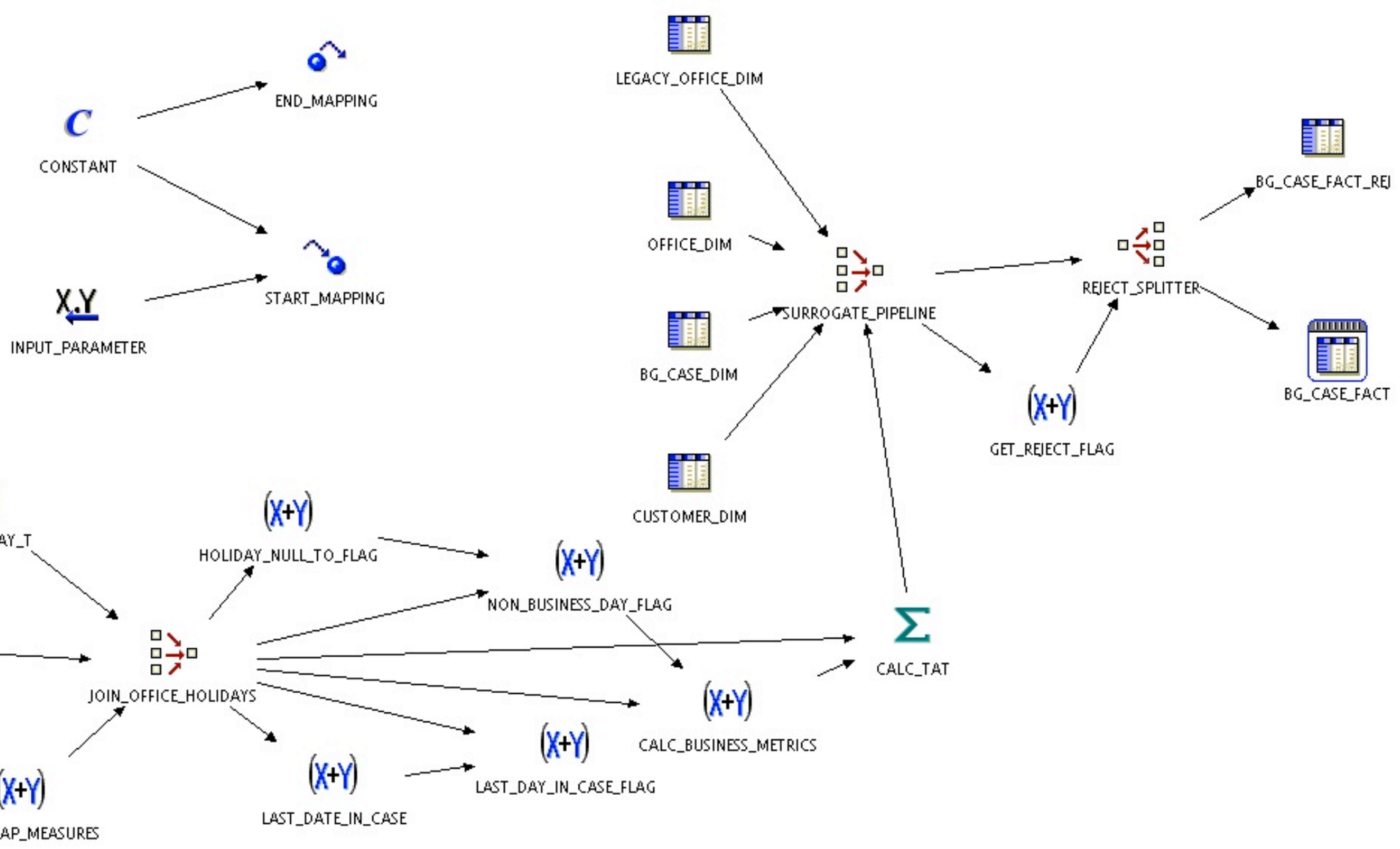
Surrogate Key	Customer Number	Customer Name	City	State	Ethnicity	Effective Date	Expiration Date
101	56743	Stewart Bryson	Nashville	TN	Unknown	Jan 1, 2004	

Surrogate Key	Customer Number	Customer Name	City	State	Ethnicity	Effective Date	Expiration Date
101	56743	Stewart Bryson	Nashville	TN	Unknown	Jan 1, 2004	Jun 1, 2006
102	56743	Stewart Bryson	Atlanta	GA	Unknown	Jun 1, 2006	

Surrogate Key	Customer Number	Customer Name	City	State	Ethnicity	Effective Date	Expiration Date
101	56743	Stewart Bryson	Nashville	TN	Caucasian	Jan 1, 2004	Jun 1, 2006
102	56743	Stewart Bryson	Atlanta	GA	Caucasian	Jun 1, 2006	

rittmanmead
DELIVERED INTELLIGENCE

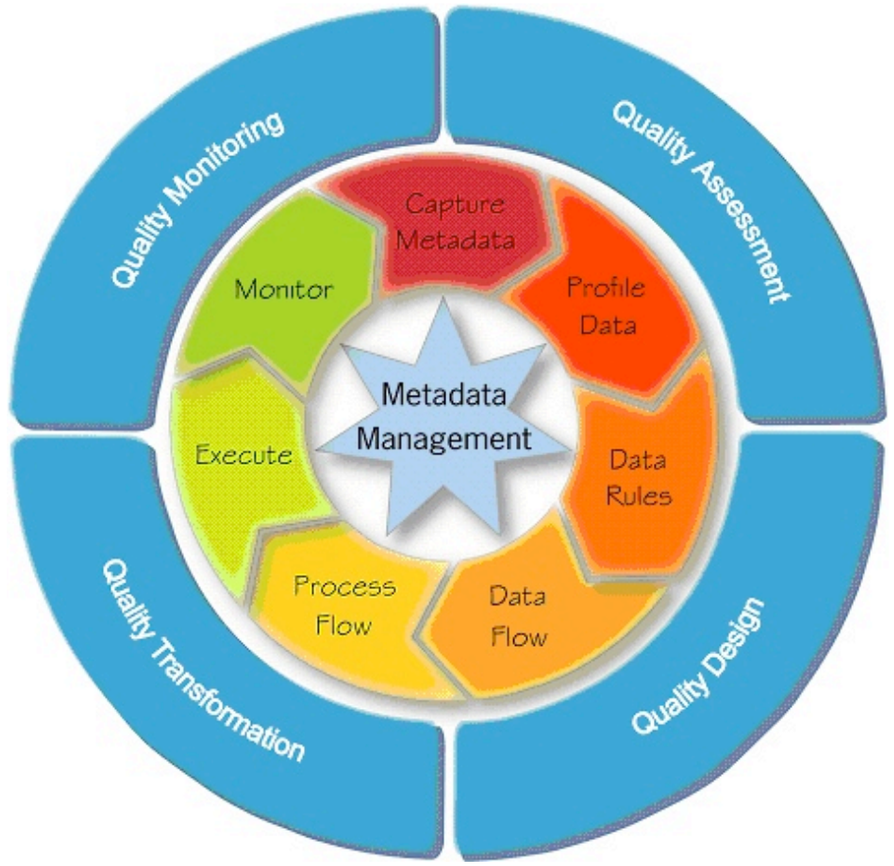
Mapping Example - Loading a Fact Table using Core Features



rittmanmead
DELIVERED INTELLIGENCE

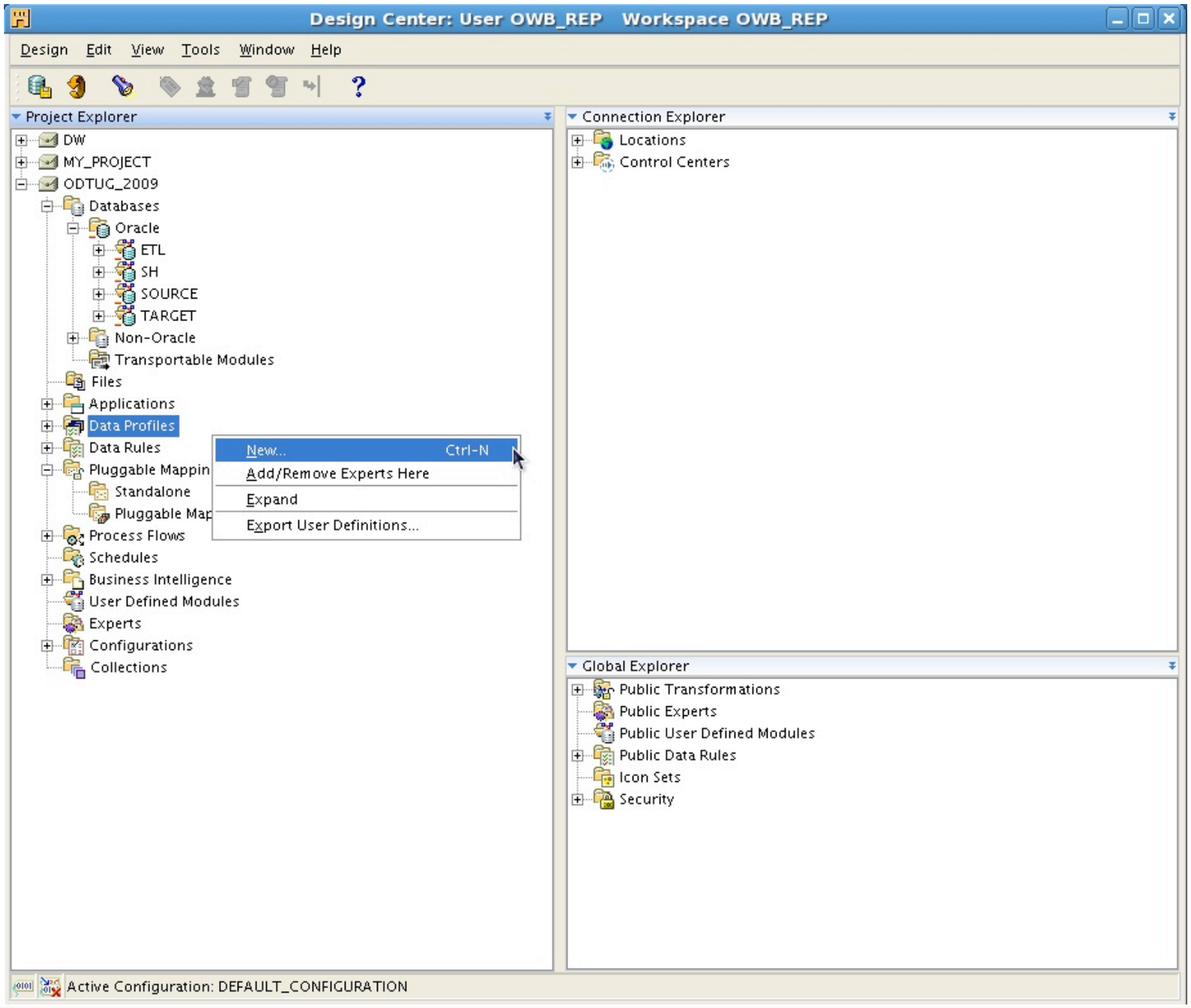
Data Quality Lifecycle: High Level

- Quality Assessment using Data Profiling
  - Performed after data is loaded
  - Diagnoses patterns, valid attributes, and dependencies
- Quality Design using Data Rules
  - Diagnose Data Profiles to determine valid values
  - Also use the Data Rule Wizard
  - Creates Mappings to guarantee data correctness
- Quality Transformation using Data Corrections
  - Transform already loaded data
  - Incorporated into load routines going forward
- Quality Monitoring using Data Auditors
  - Monitors the load routines on an ongoing basis
  - Generates metrics for reporting on data quality



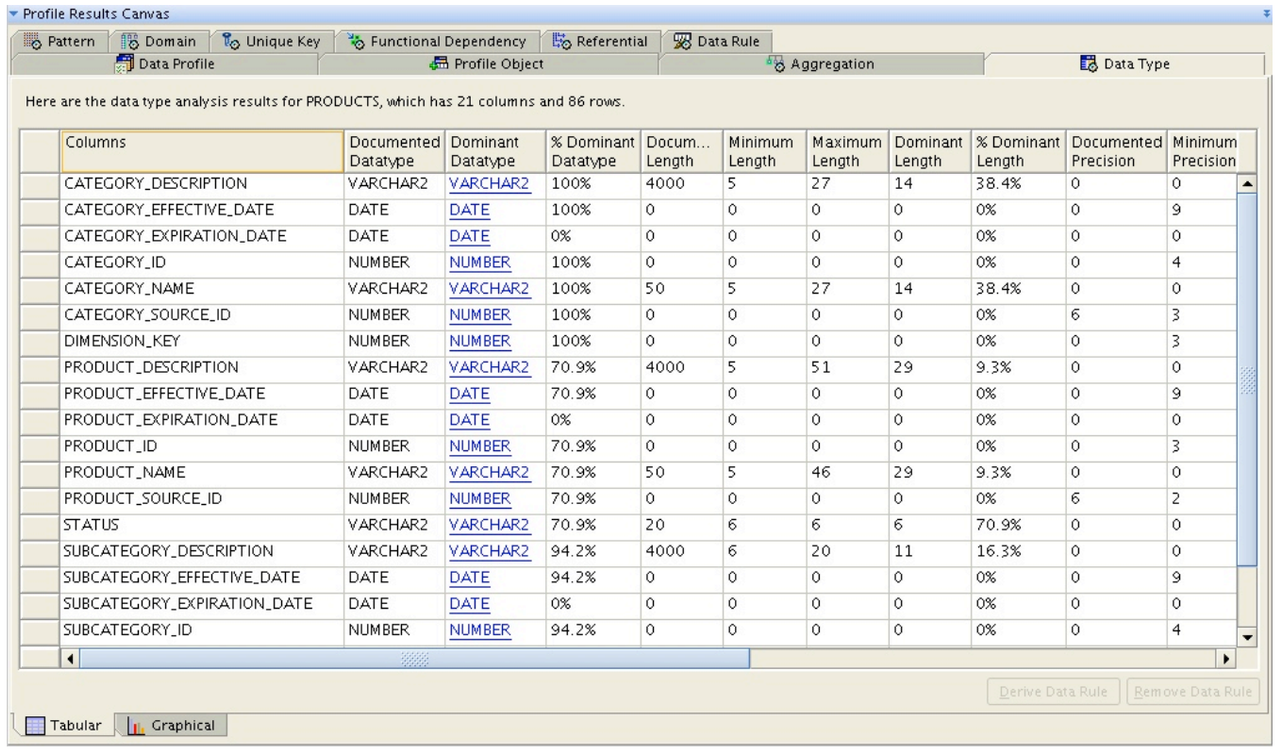
T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

Data Profiling: Example



Data Profile Screen

- Data Type Tab
  - Documented versus Actual Datatypes
- Unique Key Tab
  - Recommended Unique Constraints
- Profile Object Tab
  - View data contents inside Data Profile screen
- Domain Tab
  - Recommended set of values for each column
  - The Domain includes any value that exists more than once



Columns	Documented Datatype	Dominant Datatype	% Dominant Datatype	Docum... Length	Minimum Length	Maximum Length	Dominant Length	% Dominant Length	Documented Precision	Minimum Precision
CATEGORY_DESCRIPTION	VARCHAR2	VARCHAR2	100%	4000	5	27	14	38.4%	0	0
CATEGORY_EFFECTIVE_DATE	DATE	DATE	100%	0	0	0	0	0%	0	9
CATEGORY_EXPIRATION_DATE	DATE	DATE	0%	0	0	0	0	0%	0	0
CATEGORY_ID	NUMBER	NUMBER	100%	0	0	0	0	0%	0	4
CATEGORY_NAME	VARCHAR2	VARCHAR2	100%	50	5	27	14	38.4%	0	0
CATEGORY_SOURCE_ID	NUMBER	NUMBER	100%	0	0	0	0	0%	6	3
DIMENSION_KEY	NUMBER	NUMBER	100%	0	0	0	0	0%	0	3
PRODUCT_DESCRIPTION	VARCHAR2	VARCHAR2	70.9%	4000	5	51	29	9.3%	0	0
PRODUCT_EFFECTIVE_DATE	DATE	DATE	70.9%	0	0	0	0	0%	0	9
PRODUCT_EXPIRATION_DATE	DATE	DATE	0%	0	0	0	0	0%	0	0
PRODUCT_ID	NUMBER	NUMBER	70.9%	0	0	0	0	0%	0	3
PRODUCT_NAME	VARCHAR2	VARCHAR2	70.9%	50	5	46	29	9.3%	0	0
PRODUCT_SOURCE_ID	NUMBER	NUMBER	70.9%	0	0	0	0	0%	6	2
STATUS	VARCHAR2	VARCHAR2	70.9%	20	6	6	6	70.9%	0	0
SUBCATEGORY_DESCRIPTION	VARCHAR2	VARCHAR2	94.2%	4000	6	20	11	16.3%	0	0
SUBCATEGORY_EFFECTIVE_DATE	DATE	DATE	94.2%	0	0	0	0	0%	0	9
SUBCATEGORY_EXPIRATION_DATE	DATE	DATE	0%	0	0	0	0	0%	0	0
SUBCATEGORY_ID	NUMBER	NUMBER	94.2%	0	0	0	0	0%	0	4

Creating a Data Rule: Example

Profile Results Canvas

Pattern | Domain | Unique Key | Functional Dependency | Referential | Data Rule

Data Profile | Profile Object | Aggregation | Data Type

Here are the domain analysis results for PRODUCTS, which has 21 columns and 86 rows.

Columns	Found Domain	% Compliant	Six-Sigma
CATEGORY_DESCRIPTION	Electronics | Hardware | Peripherals and Accessorie...	100%	7.00
CATEGORY_EFFECTIVE_DATE	01-JAN-98	100%	7.00
CATEGORY_EXPIRATION_DATE	.	0%	-6.25
CATEGORY_ID	-550 | -548 | -549 | -546 | -547	100%	7.00
CATEGORY_NAME	Hardware | Software/Other | Electronics | Peripher...	100%	7.00
CATEGORY_SOURCE_ID	202 | 201 | 204 | 203 | 205	100%	7.00
DIMENSION_KEY	.	0%	-6.25
PRODUCT_DESCRIPTION	.	0%	-6.25
PRODUCT_EFFECTIVE_DATE	01-JAN-98	70.9%	2.05
PRODUCT_EXPIRATION_DATE	.	0%	-6.25
PRODUCT_ID	.	0%	-6.25
PRODUCT_NAME	.	0%	-6.25
PRODUCT_SOURCE_ID	.	0%	-6.25
STATUS	STATUS	70.9%	2.05
SUBCATEGORY_DESCRIPTION	Printer Supplies | Monitors | Modems/Fax | Record...	94.2%	3.07
SUBCATEGORY_EFFECTIVE_DATE	01-JAN-98	94.2%	3.07
SUBCATEGORY_EXPIRATION_DATE	.	0%	-6.25
SUBCATEGORY_ID	-584 | -573 | -575 | -579 | -587 | -585 | -...	94.2%	3.07
SUBCATEGORY_NAME	Portable PCs | Camcorders | Home Audio | Monito...	94.2%	3.07
SUBCATEGORY_SOURCE_ID	2043 | 2056 | 2033 | 2011 | 2031 | 2052 | ...	94.2%	3.07
VALID	A	70.9%	2.05

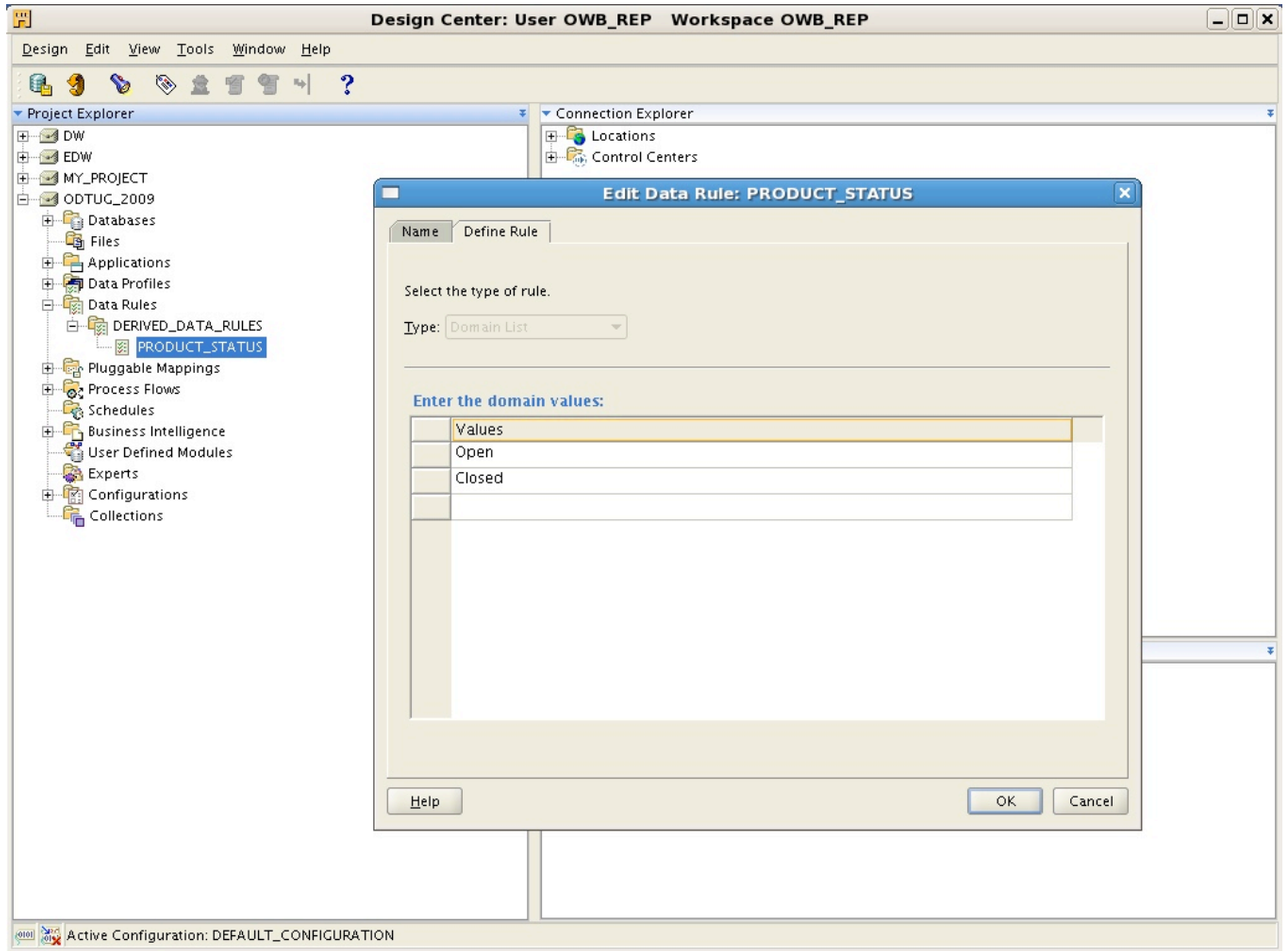
Derive Data Rule | Remove Data Rule

Tabular | Graphical

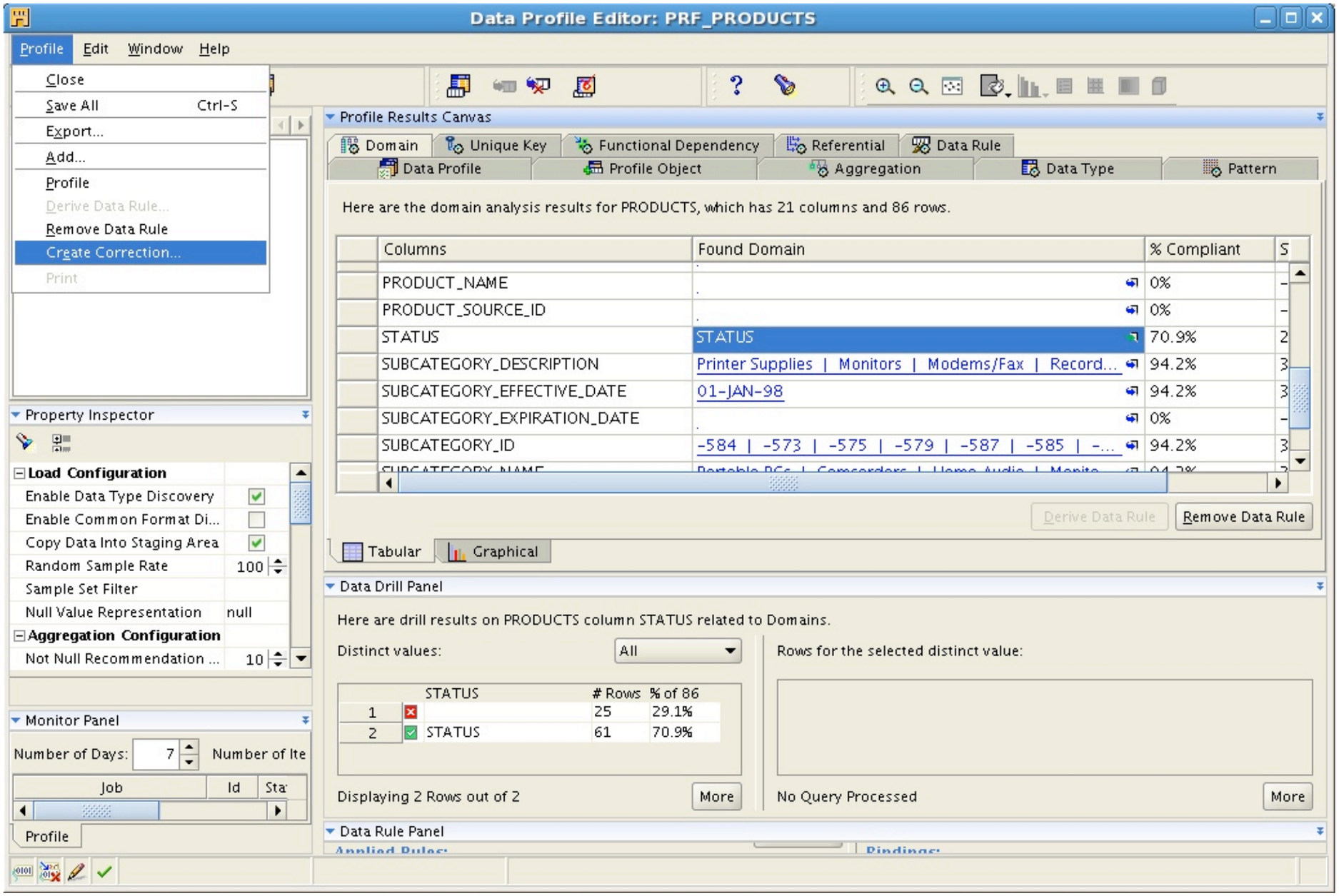
rittmanmead
DELIVERED INTELLIGENCE

Data Rules: Building Blocks of Quality

- Provide the basis for Data Corrections and Data Auditors
  - Corrections and Auditors can be built with one or more Data Rules
- Much simpler to use than Mappings or other standard ETL processes
  - Non-development resources (such as business users) can create Data Rules
- Compliant with the “Screen” concept detailed by Ralph Kimball in the *Data Warehouse ETL Toolkit.*
  - A Screen can be made up of several different Data Rules.



Building a Data Correction: Example



rittmanmead
DELIVERED INTELLIGENCE

Summary

- OWB can handle the full data warehouse lifecycle, including design, development and data quality
- The In-Database core features, such as Mappings and Process Flows, provide an excellent vehicle for deploying ETL code
- Generated load processes harness the Oracle Database providing optimal performance
- ETL Accelerators, such as Dimension and Cube Operators, are implemented in non-standard ways, and present challenges for a BI/DW implementation
- Consider coding your own Dimension and Fact loads, using basic Mappings and Process Flows
- The Data Quality option encompasses the entire lifecyle of cleaning and conforming data.

T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com

rittmanmead

DELIVERED INTELLIGENCE

What to Know Before Selecting OWB

Stewart Bryson, Technical Director, Rittman Mead America

ODTUG Kaleidoscope 2009, Monterey, June 2009

T : (888) 631-1410 E : info@rittmanmead.com W: www.rittmanmead.com