Chapter 27. March 2009

Welcome to the March 2009 edition of IBM InfoSphere Information Server Developer's Notebook. This month we answer the question;

> How do I perform application and/or server consolidation or migration using Information Server? Having one database application, I've acquired another which is similar, and wish to resolve dependencies and duplicate data-
>
> *Excellent question! This edition of (IBM) InfoSphere Information Server Developer's Notebook (IISDN) is the second in a series where we examine the Information Server components of Information Analyzer, Business Glossary, Business Glossary AnyWhere, and Metadata Workbench.*
>
> The Information Server components listed above offer numerous point and then high level areas of functionality. Perhaps the coolest area of functionality is end to end data lineage; How, When, Where and What was done to the piece of data I'm looking at-

Software versions

All of these solutions were *developed and tested* on (IBM) InfoSphere Information Server (IIS) version 8.1, using the Microsoft Windows XP/SP2 platform to support IIS client programs, and a RedHat Linux Advanced Server 4 (RHEL 4) FixPak U6 32 bit SMP server (Linux kernel version 2.6.9-67.EL-smp) to support the IIS server side components.

IBM InfoSphere Information Server allows for a single, consistent, and accurate view of data across the full width of the corporate enterprise, be it relational or non-relational, staged or live data. As a reminder, the IBM InfoSphere Information Server product contains the following major components;

> WebSphere Business Glossary Anywhere™, WebSphere Information Analyzer™, WebSphere Information Services Director™, WebSphere DataStage™, WebSphere QualityStage™, WebSphere Metadata Server and Metabridges™, WebSphere Metadata Workbench™, WebSphere Federation Server™, Classic Federation™, Event Publisher™, Replication Server™, InfoSphere Data Architect™, DataMirror Transformation Server™, and others.

Obviously, IBM InfoSphere Information Server is a large and capable product, addressing many strategic needs across the enterprise, and supporting different roles and responsibilities.

27.1 Terms and core concepts

In the most recent edition of (IBM) InfoSphere Information Server Developer’s Notebook (IISDN: February/2009), we began activities with the Information Analyzer component to Information Server. In that past edition of IISDN, we specifically discussed;

- The sample tables and data used in these examples.
- Information Analyzer setup and configuration.
- Information Analyzer use;
  - Column analysis.
  - Primary key analysis, single and multi-column.
  - Foreign key analysis, single and multiple column.

In this current edition of IISDN (March/2009), we complete the last two of five core activities within Information Analyzer; namely, (cross) domain analysis, and base line analysis. Other activities involving Information Analyzer, including observed metadata sharing with other components to Information Server follow in a future edition of IISDN. We also introduce Information Analyzer pre-built reports.

(Cross) domain analysis

Perhaps the best understood use case for (cross) domain analysis using Information Analyzer is;

> You have data in one or more columns that are redundant elsewhere in your organization. This commonly occurs when you have one business application, and acquire another of similar purpose. Using a customer record, for example, it doesn’t matter what the customer key number is (it doesn’t matter what any primary key value is defined as). The (cross) domain analysis capability of Information Analyzer will find the like (overlapping) data sets; Texas Instruments, as a company name, is the same regardless.
>
> Not certain if the customer name is Texas Instruments, Texas Instrument, or Texas Instruments USA? To solve that problem, we can combine the operations of the Information Analyzer and the QualityStage stage components of Information Server, to standardize corporate names and also perform (cross) domain analysis.
>
> Think that’s too hard? It is most likely customer will have an address component to its record, which can lead you to the identifier of, in this case, customer name. Once you find one address within a single or set of databases, the remainder come pretty easily.

What do you do once you find redundant data? Well, what do you want to do-

In the case of two similar business applications, with their now discovered and mapped redundant data, does it make sense to merge one application into another? Does it cost justify? If not, perhaps you are satisfied creating the ability for merged (integrated) reporting. Perhaps a new customer record creation in one system should automatically trigger the creation of a new (stub) record in the other application, with the ability to query back. Information Server supports all of these program capabilities.

Base line analysis

Perhaps the best understood use case for base line analysis is an EDI (electronic data interchange) application. For example;

You receive a data feed from an external source which you don't control; the data source may change format, completeness, basically change in any way imaginable. This external data feed and how you consume it, may be buried deep within some other legacy application. When or if that external data feed changes, you are likely to receive an error which is entirely not helpful, or easy to track down.

When you place Information Analyzer base line analysis adjacent to any external data feed (or internal data feed for that matter), you receive automatic notification of these types of changes, and in a format that is useful. You can even tracks changes that progress slowly over months and possibly years.

Note: True story-

While there are hundreds if not thousands of companies that sell enriched data to subscribers, there is at least one that you hear in the news everyday. They sell stock index data and other corporate analysis reporting worldwide. They are to stock and business data, what McDonald's is to hamburgers.

With 1000's of systems and feeds, this company has regular internal outages and fire drills each day, day over day, for years and years. One day, they discovered that all of the data they sent to every subscriber in a given country had been wrong for a period of 4 and 1/2 weeks; its was prior years data, and marked as such, when it was to be current year data. Neither the data provider or numerous consumers had discovered the error.

Makes you wonder what the net impact of that error truly was.

27.2 Complete the following examples

This edition of (IBM) InfoSphere Developer’s Notebook (IISDN) continues from the examples used in the prior month. You must complete the examples in that document, before proceeding here.

Cross domain analysis

1. Log on to the Information Server Console, graphical program.

   Log on to the Console, and connect to the Information Analyzer Project where we had previously completed the examples from last month.

2. Choose the menu item for (cross) domain analysis.

   From the Main Menu, select Investigate -> Cross Domain Analysis. Example as shown in Figure 27-1.

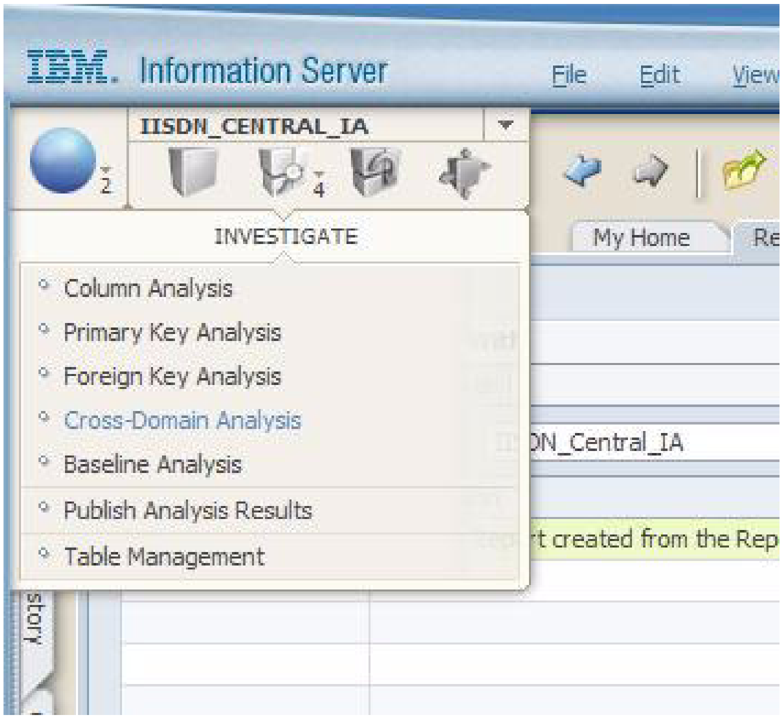


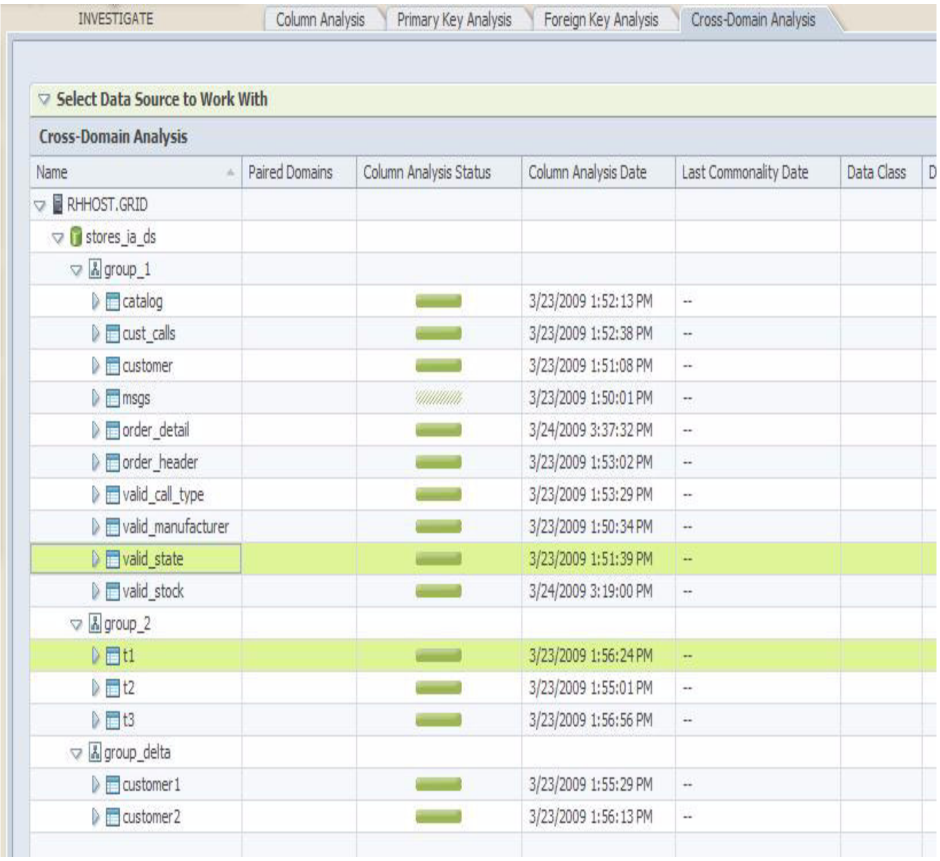
Figure 27-1   Main Menu -> Investigate -> Cross Domain Analysis menu item.

3. Select the tables, schemas (groups of tables), or larger objects to work with.

The above action produces the display as shown in Figure 27-2. By its nature, (cross) domain analysis compares at least two objects. If we are working at the table level, we must select at least two tables.

Our example offers three schemas, and the Group_2 schema was specifically created with tables that are copies from the Group_1 schema. These table and column names, and column data types, were obfuscated to make (cross) domain analysis by a human, more difficult.

To get you started, we will tell you that the Group_2.T1 table is a copy of the Group_1.Valid_State table. Select both tables via a Control-Click, and then select the menu item in the upper right region of the display entitled, Run Cross-Domain Analysis.



Cross-Domain Analysis

Name	Paired Domains	Column Analysis Status	Column Analysis Date	Last Commonality Date	Data Class	D
RHHOST.GRID						
stores_ia_ds						
group_1						
catalog		🟩	3/23/2009 1:52:13 PM	--		
cust_calls		🟩	3/23/2009 1:52:38 PM	--		
customer		🟩	3/23/2009 1:51:08 PM	--		
msgs		▨	3/23/2009 1:50:01 PM	--		
order_detail		🟩	3/24/2009 3:37:32 PM	--		
order_header		🟩	3/23/2009 1:53:02 PM	--		
valid_call_type		🟩	3/23/2009 1:53:29 PM	--		
valid_manufacturer		🟩	3/23/2009 1:50:34 PM	--		
valid_state		🟩	3/23/2009 1:51:39 PM	--		
valid_stock		🟩	3/24/2009 3:19:00 PM	--		
group_2						
t1		🟩	3/23/2009 1:56:24 PM	--		
t2		🟩	3/23/2009 1:55:01 PM	--		
t3		🟩	3/23/2009 1:56:56 PM	--		
group_delta						
customer1		🟩	3/23/2009 1:55:29 PM	--		
customer2		🟩	3/23/2009 1:56:13 PM	--		

Figure 27-2 Cross domain analysis, Select a Data Source.

4. Select columns to analyze.

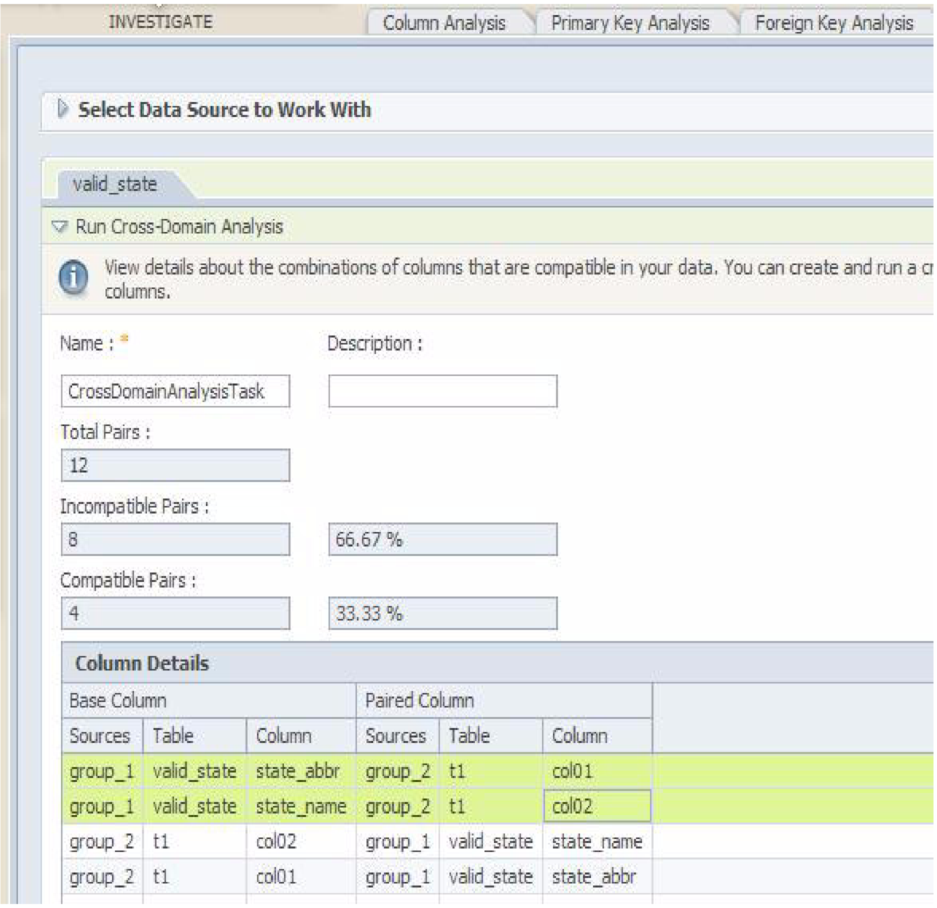
From the results of column analysis, done earlier, Information Analyzer knows there are two columns that are candidates for a (cross) domain match;

- Valid_State.State_Abbr -> T1.Col01
- Valid_State.State_Name -> T2.Col02

The reverse (but equal) condition of table T1 relating to Valid_State is also displayed. *There is no functional difference to choosing Valid_State over table T1 in this context.*

Highlight both choices of Valid_State.State_Abbr -> T1.Col01 and Valid_State.State_Name -> T1.Col02 using a Control-Click.

Then select, Submit -> Submit and Close, in the bottom right portion of the display. Example as displayed in Figure 27-3.



View details about the combinations of columns that are compatible in your data. You can create and run a cr
columns.

Name : * CrossDomainAnalysisTask

Description :

Total Pairs : 12

Incompatible Pairs : 8 66.67 %

Compatible Pairs : 4 33.33 %

Column Details

Base Column			Paired Column		
Sources	Table	Column	Sources	Table	Column
group_1	valid_state	state_abbr	group_2	t1	col01
group_1	valid_state	state_name	group_2	t1	col02
group_2	t1	col02	group_1	valid_state	state_name
group_2	t1	col01	group_1	valid_state	state_abbr

Figure 27-3 Select columns to analyze.

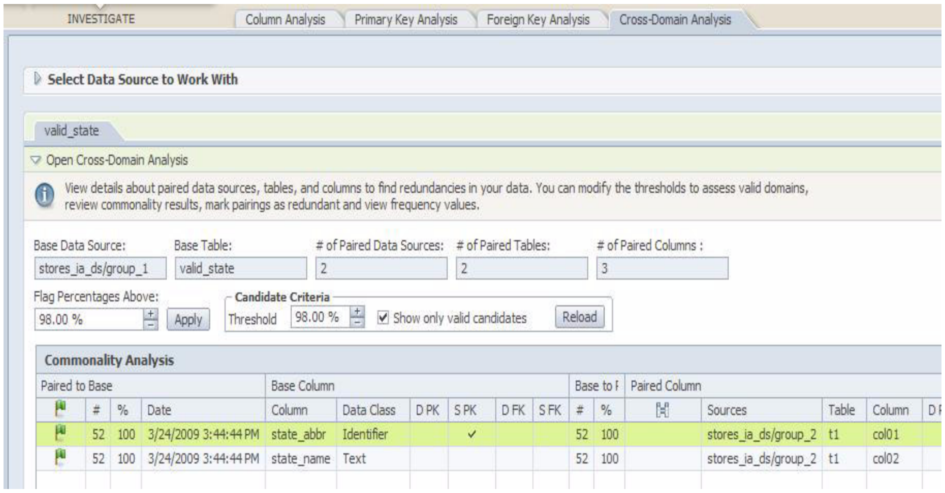
This (Job), as all others, is sent to the background for parallel and concurrent operation with the graphical program you are now using. These (Jobs) may also be scheduled for later execution via the same user interface. The prior edition of this IISDN document detailed how to monitor (Jobs) submitted to the background.

For now and for simplicity, you can detect that this (Job) has completed when you can select the menu item entitled, Open Column-Domain Analysis, for the table entitled Valid_State.

5. Examine and approve/disprove of the results of analysis.

   Figure 27-4 displays the results of the Open Cross-Domain Analysis menu item, and the results of our analysis.

Because there are two potential matches, the steps that follow must be done twice, once for each column.



View details about paired data sources, tables, and columns to find redundancies in your data. You can modify the thresholds to assess valid domains, review commonality results, mark pairings as redundant and view frequency values.

Base Data Source: stores_ia_ds/group_1
Base Table: valid_state
# of Paired Data Sources: 2
# of Paired Tables: 2
# of Paired Columns : 3
Flag Percentages Above: 98.00 %
Candidate Criteria
Threshold 98.00 %
☑ Show only valid candidates

Commonality Analysis

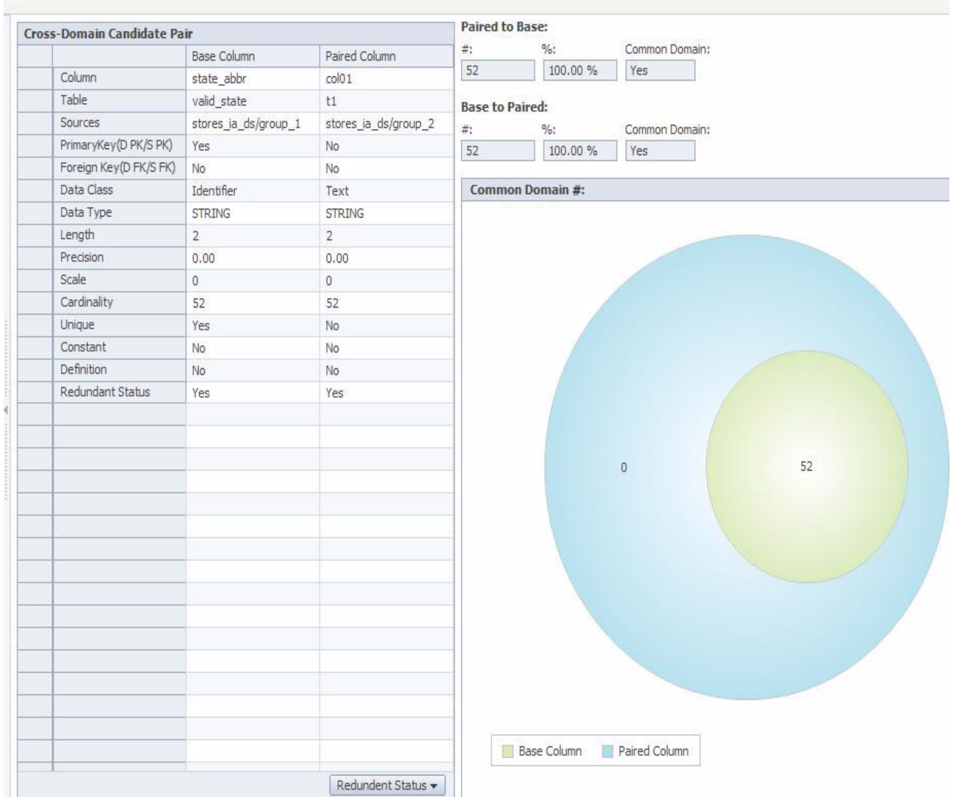
Paired to Base				Base Column						Base to F		Paired Column				
	#	%	Date	Column	Data Class	D PK	S PK	D FK	S FK	#	%		Sources	Table	Column	D F
	52	100	3/24/2009 3:44:44 PM	state_abbr	Identifier		✓			52	100		stores_ia_ds/group_2	t1	col01	
	52	100	3/24/2009 3:44:44 PM	state_name	Text					52	100		stores_ia_ds/group_2	t1	col02	

Figure 27-4 Results of (cross) domain analysis.

Highlight the first selection entitled, State_Abbr, and then select the menu item entitled, View Details, in the bottom right portion of the display. This action produces the display as shown in Figure 27-5.

This screen is for information purposes only; no changes are made here. From the graphical display (the two encased circles), we can see that the column values of T1.Col01 are wholly contained in the column values for Valid_State.State_Abbr; a complete overlap, a complete cross domain overlap.

Select the menu item entitled Close, in the bottom right area of the display.



Cross-Domain Candidate Pair

	Base Column	Paired Column
Column	state_abbr	col01
Table	valid_state	t1
Sources	stores_ia_ds/group_1	stores_ia_ds/group_2
PrimaryKey(D PK/S PK)	Yes	No
Foreign Key(D FK/S FK)	No	No
Data Class	Identifier	Text
Data Type	STRING	STRING
Length	2	2
Precision	0.00	0.00
Scale	0	0
Cardinality	52	52
Unique	Yes	No
Constant	No	No
Definition	No	No
Redundant Status	Yes	Yes

Paired to Base:

#: 52 %: 100.00 % Common Domain: Yes

Base to Paired:

#: 52 %: 100.00 % Common Domain: Yes

Figure 27-5 Graphical results of analysis.

6. Mark this relationship as reviewed, and as a result of our discovery, as being a redundant column data set.

   (Not displayed.) Select the menu item entitled, Redundant Status -> Mark Redundant Status.

7. Repeat steps 5 and 6 above, for the second (cross) domain analysis related to Valid_State.State_Name.

8. Your minimum exercise for (cross) domain analysis is now complete.

   Time permitting, discover which tables in Schema_1 are copies of the tables, Schema_1.T2, and Schema_2.T3.

Examine an Information Analyzer pre-built report

As an extra exercise, examine one of the pre-built analysis reports that comes with Information Analyzer. In this section we will run one report of the dozens that come with the product.

9. Run the Report sub-menu.

   From the Main Menu, select, Home -> Reports.

   Example as shown in Figure 27-6.

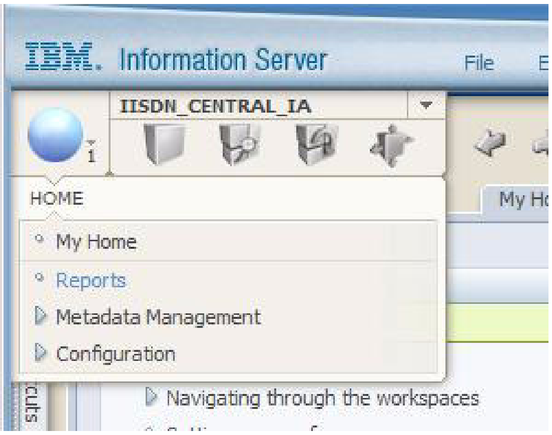
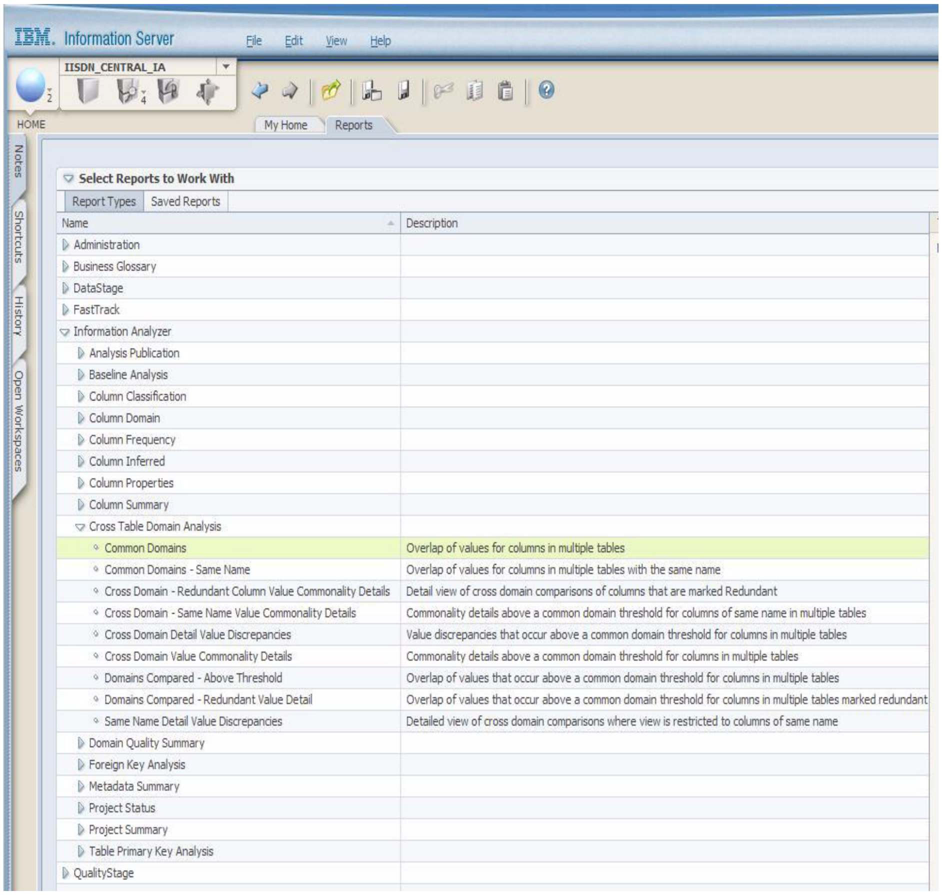


Figure 27-6   Report sub menu, Home -> Reports.

10. Navigate the Report sub menu.

    As displayed in Figure 27-7, navigate to Reports -> Report Types -> Information Analyzer -> Cross Table Domain Analysis -> Common Domains.



Select Reports to Work With

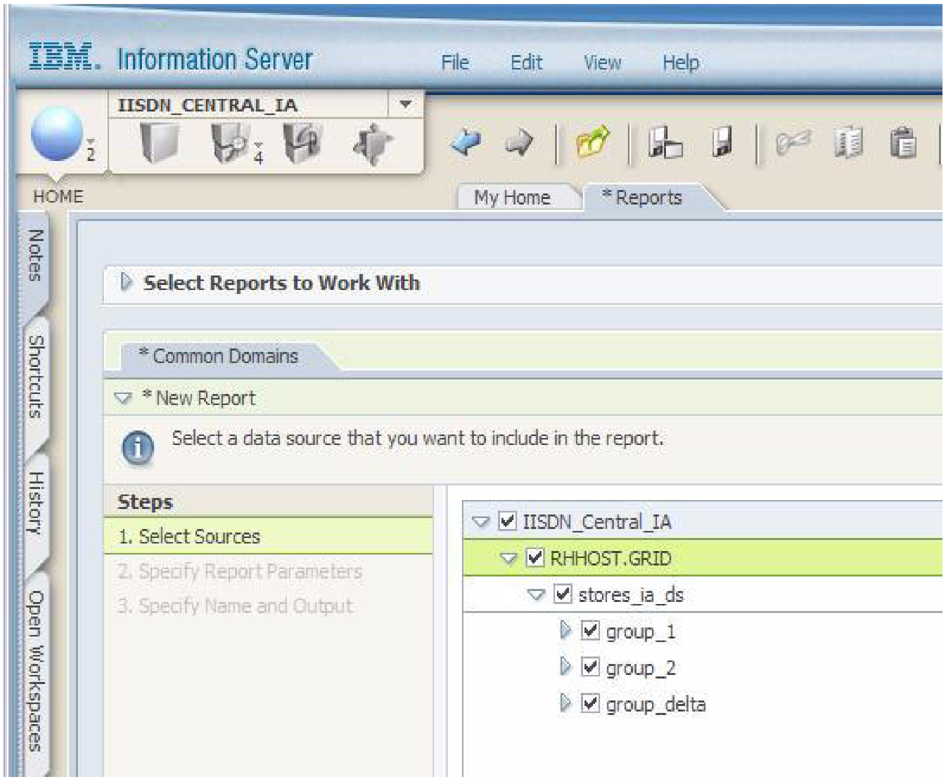
Name	Description
Administration	
Business Glossary	
DataStage	
FastTrack	
Information Analyzer	
Analysis Publication	
Baseline Analysis	
Column Classification	
Column Domain	
Column Frequency	
Column Inferred	
Column Properties	
Column Summary	
Cross Table Domain Analysis	
Common Domains	Overlap of values for columns in multiple tables
Common Domains - Same Name	Overlap of values for columns in multiple tables with the same name
Cross Domain - Redundant Column Value Commonality Details	Detail view of cross domain comparisons of columns that are marked Redundant
Cross Domain - Same Name Value Commonality Details	Commonality details above a common domain threshold for columns of same name in multiple tables
Cross Domain Detail Value Discrepancies	Value discrepancies that occur above a common domain threshold for columns in multiple tables
Cross Domain Value Commonality Details	Commonality details above a common domain threshold for columns in multiple tables
Domains Compared - Above Threshold	Overlap of values that occur above a common domain threshold for columns in multiple tables
Domains Compared - Redundant Value Detail	Overlap of values that occur above a common domain threshold for columns in multiple tables marked redundant
Same Name Detail Value Discrepancies	Detailed view of cross domain comparisons where view is restricted to columns of same name
Domain Quality Summary	
Foreign Key Analysis	
Metadata Summary	
Project Status	
Project Summary	
Table Primary Key Analysis	
QualityStage	

Figure 27-7 Selecting a report related to (cross) domain analysis.

11. Then in the upper right area of the display, select the menu item entitled, New Report. This action produces the display as shown in Figure 27-8.

    In Figure 27-8, we chose to report on all tables in our system, the entire Data Source.

    Click Next to proceed.



Select Reports to Work With

* Common Domains

* New Report

Select a data source that you want to include in the report.

Steps
1. Select Sources
2. Specify Report Parameters
3. Specify Name and Output

☑ IISDN_Central_IA
☑ RHHOST.GRID
☑ stores_ia_ds
☑ group_1
☑ group_2
☑ group_delta

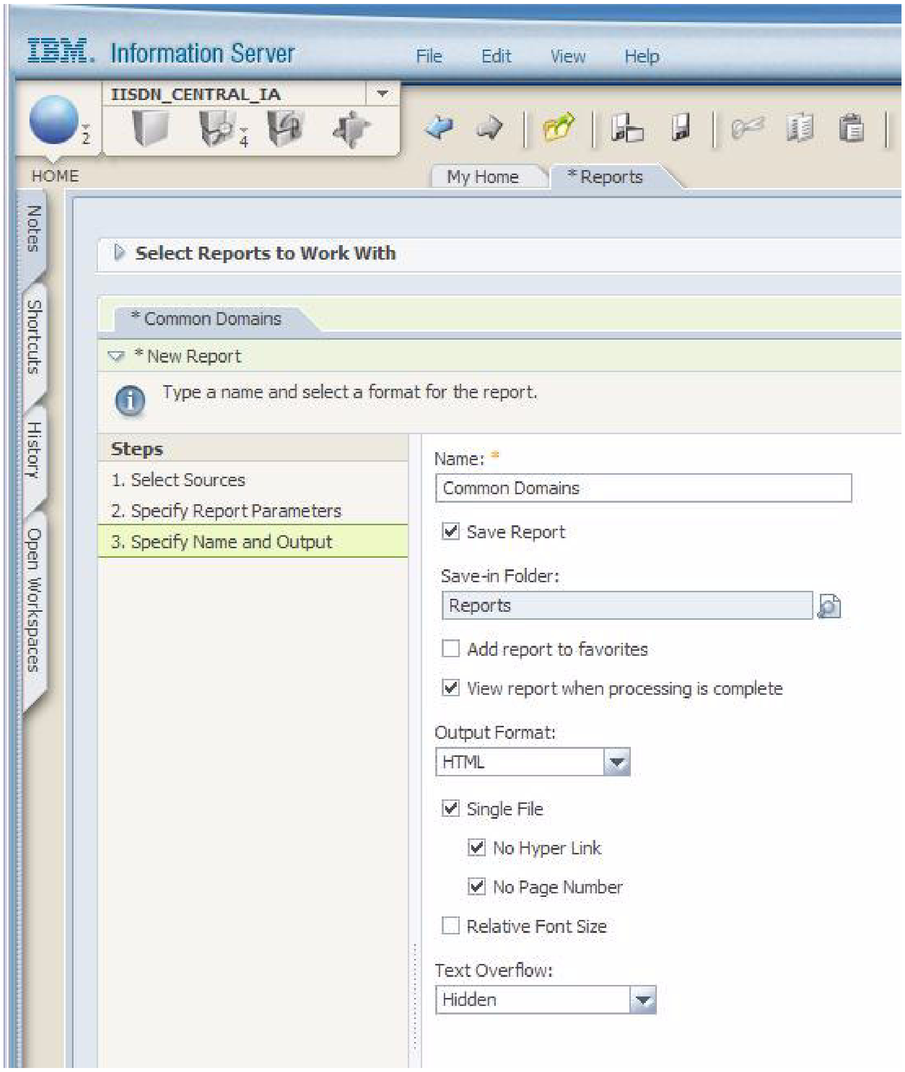
Figure 27-8   Selecting a Data Source for a given report.

12. Continue configuring the report.

    In Figure 27-9, we skipped Step-2 of the report creation; there were no options there that we needed for this example.

    Navigate the display in Figure 27-9 to equal what is shown. Be certain to Check, Save Report.

    Click Finish when done.



Select Reports to Work With

* Common Domains

* New Report

Type a name and select a format for the report.

Steps

1. Select Sources
2. Specify Report Parameters
3. Specify Name and Output

Name: * Common Domains

☑ Save Report

Save-in Folder: Reports

☐ Add report to favorites

☑ View report when processing is complete

Output Format: HTML

☑ Single File

☑ No Hyper Link

☑ No Page Number

☐ Relative Font Size

Text Overflow: Hidden

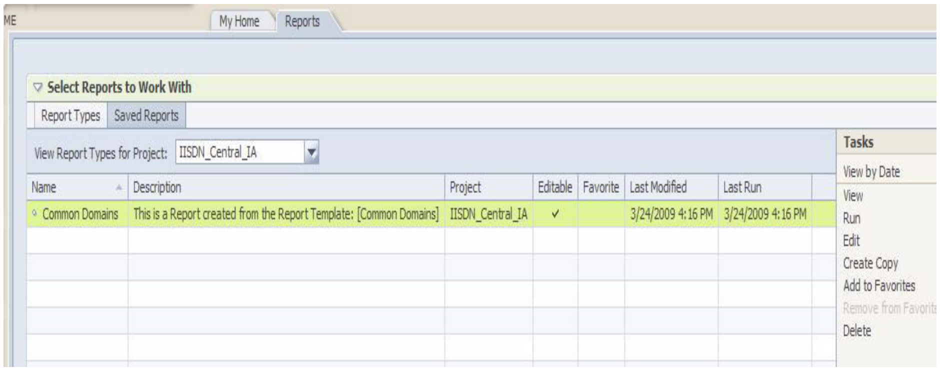
Figure 27-9 Step 3 of 3 to create a report, be certain to Check, Save Report.

13. Running and Viewing the saved Report from above.

Move from the TAB entitled Report Types, to the TAB entitled Saved Reports.

Highlight the Report you just created, and select Run. Later select View.

Example as shown in Figure 27-10.



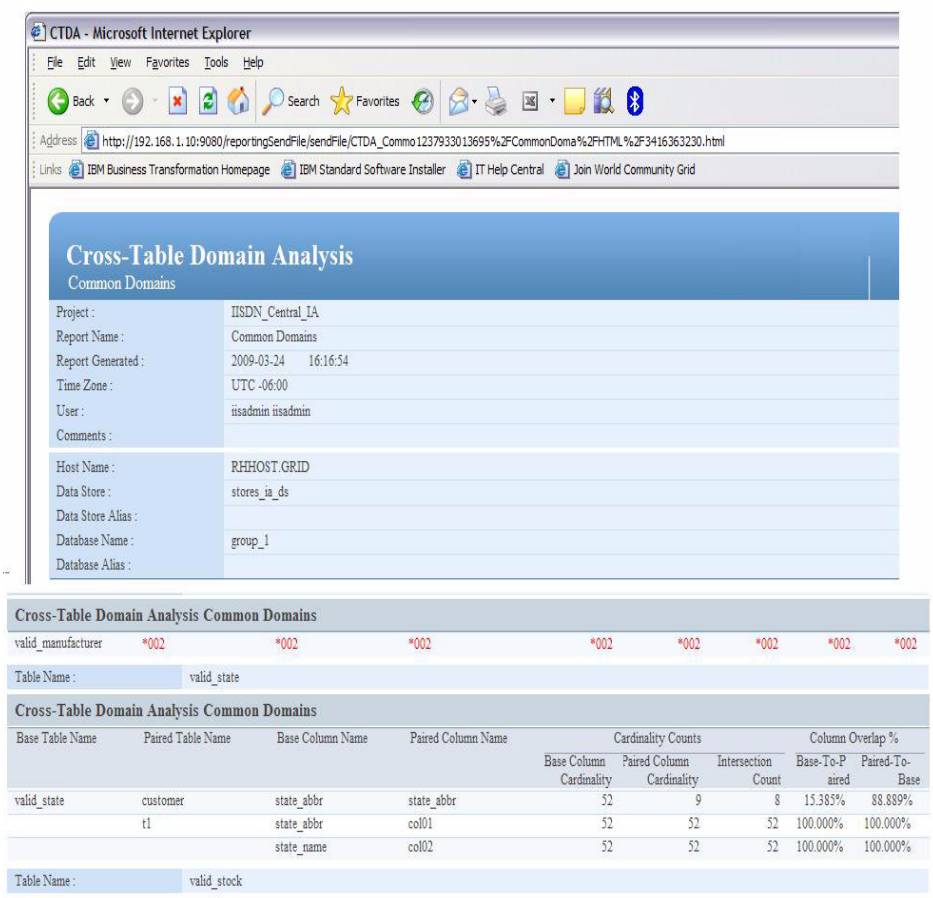
View Report Types for Project: IISDN_Central_IA

Name	Description	Project	Editable	Favorite	Last Modified	Last Run
Common Domains	This is a Report created from the Report Template: [Common Domains]	IISDN_Central_IA	✓		3/24/2009 4:16 PM	3/24/2009 4:16 PM

Figure 27-10 Running and Viewing a Saved Report.

A newly run HTML style Report appears in the default Web browser, as shown in Figure 27-11.

Note: Notice also in Figure 27-11 that you see display of Foreign Key table relationships. By their definition, Foreign Keys share data with their Primary Key counterparts, causing a cross domain match.


CTDA - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Address http://192.168.1.10:9080/reportingSendFile/sendFile/CTDA_Commo1237933013695%2FCommonDoma%2FHTML%2F3416363230.html
Links IBM Business Transformation Homepage IBM Standard Software Installer IT Help Central Join World Community Grid

Cross-Table Domain Analysis

Common Domains

Project :	IISDN_Central_IA
Report Name :	Common Domains
Report Generated :	2009-03-24 16:16:54
Time Zone :	UTC -06:00
User :	iisadmin iisadmin
Comments :	

Host Name :	RHHOST.GRID
Data Store :	stores_ia_ds
Data Store Alias :	
Database Name :	group_1
Database Alias :	

Cross-Table Domain Analysis Common Domains

valid_manufacturer	*002	*002	*002	*002	*002	*002	*002	*002

Table Name : valid_state

Cross-Table Domain Analysis Common Domains

Base Table Name	Paired Table Name	Base Column Name	Paired Column Name	Cardinality Counts			Column Overlap %	
				Base Column Cardinality	Paired Column Cardinality	Intersection Count	Base-To-Paired	Paired-To-Base
valid_state	customer	state_abbr	state_abbr	52	9	8	15.385%	88.889%
	t1	state_abbr	col01	52	52	52	100.000%	100.000%
		state_name	col02	52	52	52	100.000%	100.000%

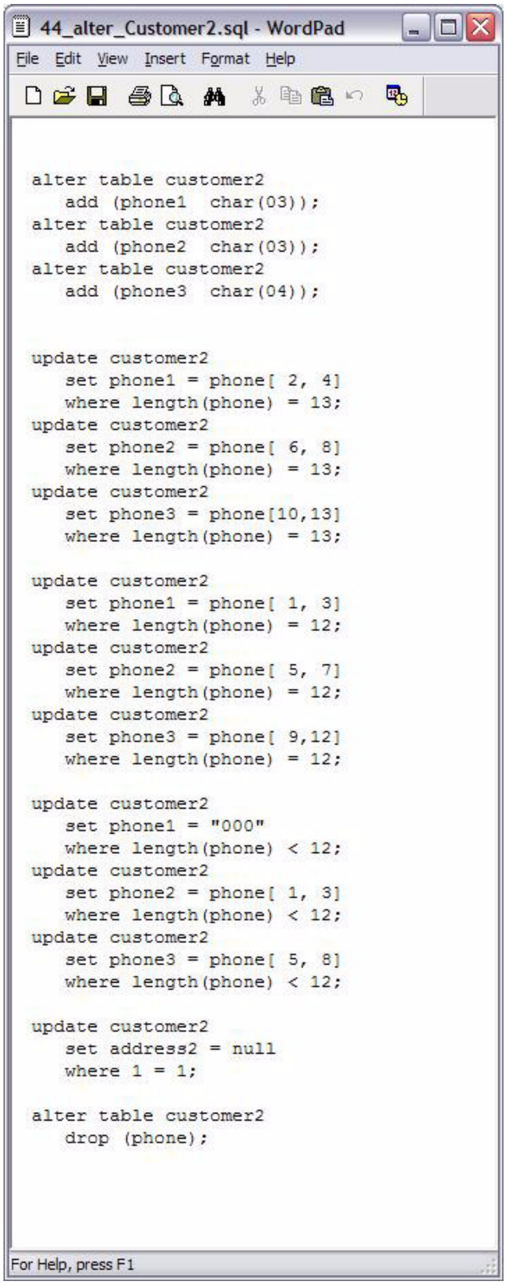
Table Name : valid_stock

Figure 27-11 (HTML style) Report output.

Perform base line analysis

To perform baseline analysis, we are going to use the Group_Delta schema, and the Customer2 table. In the real world, folks other than ourselves would be changing the completeness or format of the data we receive, and possibly also the structure of the table itself. In this sample environment, we will call to make those changes.

Figure 27-12 displays the SQL command file we will run to alter our data, and also our table structure. First we will record our current data and table structure, then run the SQL, then call to sample again; reporting observed changes.


44_alter_Customer2.sql - WordPad
File Edit View Insert Format Help
For Help, press F1

```
alter table customer2
   add (phone1  char(03));
alter table customer2
   add (phone2  char(03));
alter table customer2
   add (phone3  char(04));


update customer2
   set phone1 = phone[ 2, 4]
   where length(phone) = 13;
update customer2
   set phone2 = phone[ 6, 8]
   where length(phone) = 13;
update customer2
   set phone3 = phone[10,13]
   where length(phone) = 13;

update customer2
   set phone1 = phone[ 1, 3]
   where length(phone) = 12;
update customer2
   set phone2 = phone[ 5, 7]
   where length(phone) = 12;
update customer2
   set phone3 = phone[ 9,12]
   where length(phone) = 12;

update customer2
   set phone1 = "000"
   where length(phone) < 12;
update customer2
   set phone2 = phone[ 1, 3]
   where length(phone) < 12;
update customer2
   set phone3 = phone[ 5, 8]
   where length(phone) < 12;

update customer2
   set address2 = null
   where 1 = 1;

alter table customer2
   drop (phone);
```

Figure 27-12   SQL command script to alter our source table and data.

14. Be certain you have completed at least column analysis for table, Group_Delta.Customer2.

15. Select the base line analysis menu item.

    From the Main Menu, select, Investigate -> Baseline Analysis.

    Example as shown in Figure 27-13.

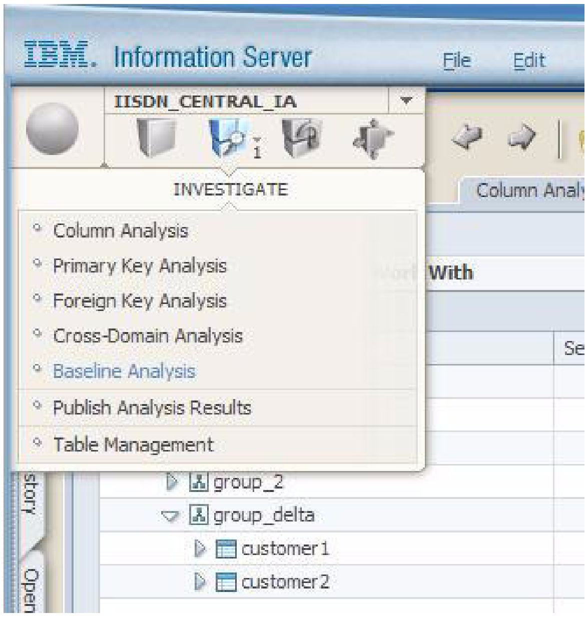


Figure 27-13 Main Menu -> Investigate -> Baseline Analysis.

16. Select the table to be analyzed.

    Highlight the table entry entitled, Group_Delta -> Customer2.

    From the menu in the upper right area of the display, then select, Set Baseline -> Current Analysis -> OK.

Note: In effect we are saying that this moment in time of our analysis is considered the 'base line'. We can later compare to a 'base line', or a 'check point'; think of a check point as an interval between base lines. And, you can set base lines or check points at the table level or higher.

17. Run the SQL command script in Figure 27-12 to affect both the data and table structure to the Group_Delta.Customer2 table.

    This SQL script adds 3 new columns, and copies data from an existing column into these new columns. Then, we drop the 1 previously existing source column. The net result is 2 new columns to our table.

Note: If your system only allows you to change data, fine. If you can change the table structure, all the better. We are running the most complete use case; dropping *and adding columns*. Adding columns we invoke additional steps.

Recall that we define the Data Store at a system level, and then also a project level. Not seeing given columns at the project level is a feature; what if those columns are sensitive- Because our use case adds columns, we have the option to make these columns available to our project users, *or not*.

18. Adjust the Data Store definition for table Group_Delta.Customer2 at the system level.

    a. As documented in the prior edition of IISDN, return to the Home menu -> Metadata Management -> Import Metadata menu item.

    Example as shown in Figure 27-14.

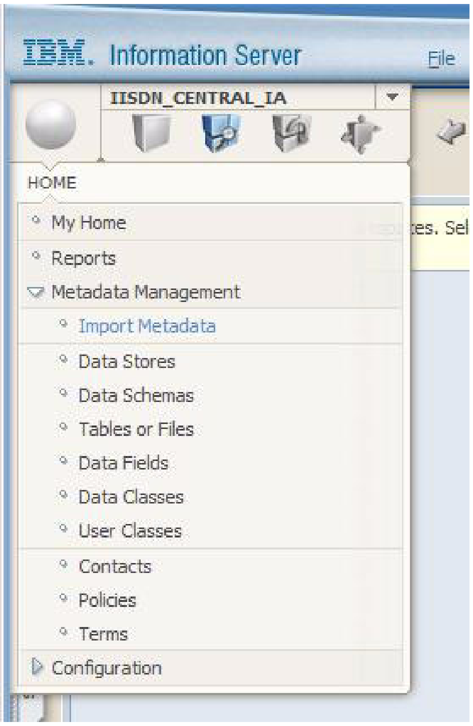


Figure 27-14 Home -> Metadata Management -> Import Metadata.

b. Update the definition for the Group_Delta.Customer2 table at the *system level.*

- Highlight the Group_Delta.Customer2 table with a single Click.
- From the menu in the upper right area of the display, select the menu item entitled, Identify Next Level.

  Table Customer2 should update its display from having 10 columns to now 12.
- With table Customer2 still highlighted, select the menu item entitled, Import.
- Close this window to ensure all changes are complete.

19. Adjust the Data Store definition for table Group_Delta.Customer2 at the *project level.*

– Select the menu item entitled, Overview -> Project Properties -> (TAB) Data Sources.

– Navigate the display to the Group_Delta -> Customer2 table.

Highlight this entry with a single Click.

Note: If you expand the table at this point, you will see that it has 10 columns with given column names and column data types.

– From the menu on the bottom right of the display, select Remove.

– From the same menu, select Add.

Navigate the display to add the Group_Delta.Customer2 table.

Note: If you expand the table at this point, you will see that the table definition lost 1 column, and added 3 others.

– Click, Save All, and then exit this display to ensure that all changes are complete.

20. Re-run column analysis for table Group_Delta.Customer2.

– From the main menu, select Investigate -> Column Analysis.

– Navigate to the Group_Delta.Customer2 table.

Note: If you expand the table at this point, you will see that 3 of this table's column have not been analyzed.

– From the menu in the upper right area of the display, select, Run Column Analysis.

– Select, Submit -> Submit and Close.

You are done when the display shows all columns for this table are analyzed. Right-click in the display and chose Refresh if you must.

21. Return to the Investigate -> Baseline Analysis menu item.

   a. From the main menu, select, Investigate -> Baseline Analysis.

   b. Navigate to the Group_Delta.Customer2 table. Highlight this table with a single Click.

   c. From the menu in the upper right area of the display, select, View Baseline Analysis.

   You are given the option to view a baseline analysis from the current state, or from the last check point. we don’t have a last check point, so select ‘(from) Current Analysis’.

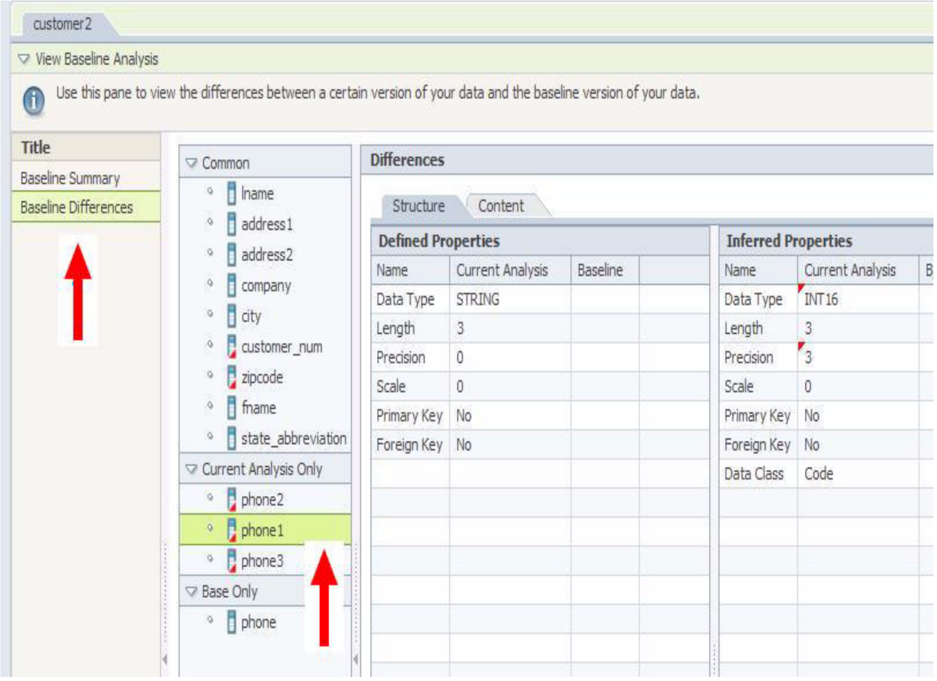
   d. Click OK.

22. Review the baseline analysis results.

In Figure 27-15, we’ve already moved to the Baseline Differences TAB of the View Baseline Analysis screen. Also, we’ve highlighted a column (Phone1) which is present in the table currently (Current Analysis), but was not in the table before (Base Only).

If you highlight a column that was in the base version of the table as well as the present version, you will see a side by side comparison of the data column type, and more.

If you return to the Baseline Summary TAB, you are presented with a summary of number of columns, number of NULL values and more.



Use this pane to view the differences between a certain version of your data and the baseline version of your data.

Defined Properties

Name	Current Analysis	Baseline
Data Type	STRING	
Length	3	
Precision	0	
Scale	0	
Primary Key	No	
Foreign Key	No	

Inferred Properties

Name	Current Analysis	B
Data Type	INT16	
Length	3	
Precision	3	
Scale	0	
Primary Key	No	
Foreign Key	No	
Data Class	Code	

Figure 27-15 View Baseline Analysis -> Baseline Differences.

27.3 In this document, we reviewed or created:

As part of a continuing edition series, we examined the (IBM) InfoSphere Information Server (IIS), Information Analyzer component. Specifically we performed the Information Analyzer component activities of; (cross) domain analysis and base line analysis. We also took a first look at the pre-built reports that Information analyzer contains, (first look to the IISDN document series at least).

Information Analyzer is a hugely productive component to Information Server. Besides the automatic team communication activities it supports, the activities of Information Analyzer are automated; normally you'd be writing tons of complicated SQL and spreadsheet analysis to do this work. And, all of the knowledge Information Analyzer gathers is automatically placed in a shared metadata repository, where it benefits the other components to Information Server, promoting higher productivity, data quality, and an audit trail of end to end data lineage.

Persons who help this month.

Walter Crockett Jr, Steve Fazio, Bill 'if he pokes my chair one more time' Shubin, James Wilson, Carrie Klapheke and Doleen Wilbur.

Additional resources:

The IBM InfoSphere Information Server, Information Analyzer component, product tutorial.

Legal statements:

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating trademarks that were owned by IBM at the time this information was published. A complete and current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

Other company, product or service names may be trademarks or service marks of others.

Special attributions:

The listed trademarks of the following companies require marking and attribution:

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Microsoft trademark guidelines

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Intel trademark information

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

Other company, product, or service names may be trademarks or service marks of others.