Chapter 31.

August 2009

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

> In prior editions of this document, you discussed data discovery using the Information Analyzer (IA) component to Information Server; you covered the 5 (then) core functions of IA. Can you go farther, and show some of the power user type functions to IA?
>
> *Excellent question! Yes, those 2 prior editions of this document served as a primer to Information Analyzer. Not only were there many more functions to cover then, there have been many more functions added in 2 new product releases since that time.*
>
> In addition to the technical features detailed below, we also strive to deliver common business challenges in business terms; reviewing the tasks and challenges we often face in data discovery and data stewardship.

Software versions

All of these solutions were *developed and tested* on (IBM) InfoSphere Information Server (IIS) version 8.1.1, using the Microsoft Windows XP/SP3 platform to support IIS client programs, and a RedHat Linux Advanced Server 5 (RHAS 5) 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™, InfoSphere 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.

31.1 Terms and core concepts

As mentioned above, the February/2009 and March/2009 editions of IBM InfoSphere Information Server Developers Notebook (IISDN) detailed the (then 5) core areas of functionality inside the Information Analyzer (IA) component to IBM InfoSphere Information Server (IIS). The (then) 5 core areas of functionality include;

- Column Analysis
- Primary key Analysis
- Foreign key Analysis
- Cross domain Analysis
- Baseline Analysis

Having previously covered the 5 areas above, we still left a lot of functionality uncovered, and then releases 8.1.1 and 8.1.2 of Information Server added more functionality still. In this and the next edition of IISDN, we seek to remedy that.

In the prior editions of this document covering Information Analyzer, we used the demonstration database from IBM Informix IDS, with a few changes. That sample database is displayed in Figure 31-1. From that database, we copied some of those tables and obfuscated their identity; (his their table names, columns names, and column data types.) And we caused errors in some of the data.

Working from this text, we also now describe how to replicate these conditions for your own testing and skills development.

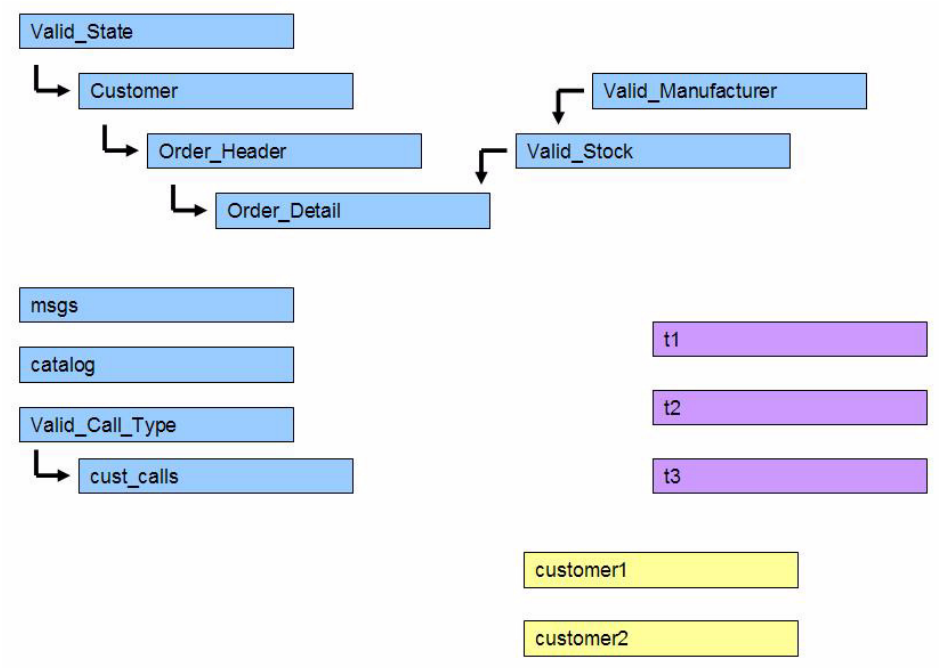


Figure 31-1 Sample tables used in this edition of IISDN.

Looking at the relationship between State and Customer

After identifying all Primary Keys columns, we naturally progress to looking for shared keys; foreign key values. In the examination of Valid_State to all other tables in the Group_1 schema, we saw the following;

- Under the Investigate -> Foreign Key Analysis, we highlight all tables in the Group_1 schema, and select, Run Foreign Key Analysis.

  (This assumes you have previously Run Column Analysis, and Selected a Primary Key for at least Valid_State.)

- We choose Valid_State to be the Base Table, that table having the Primary Key we wish to pair. And we select all other tables as potential matches. Example as shown in Figure 31-2.

  And we select Next.

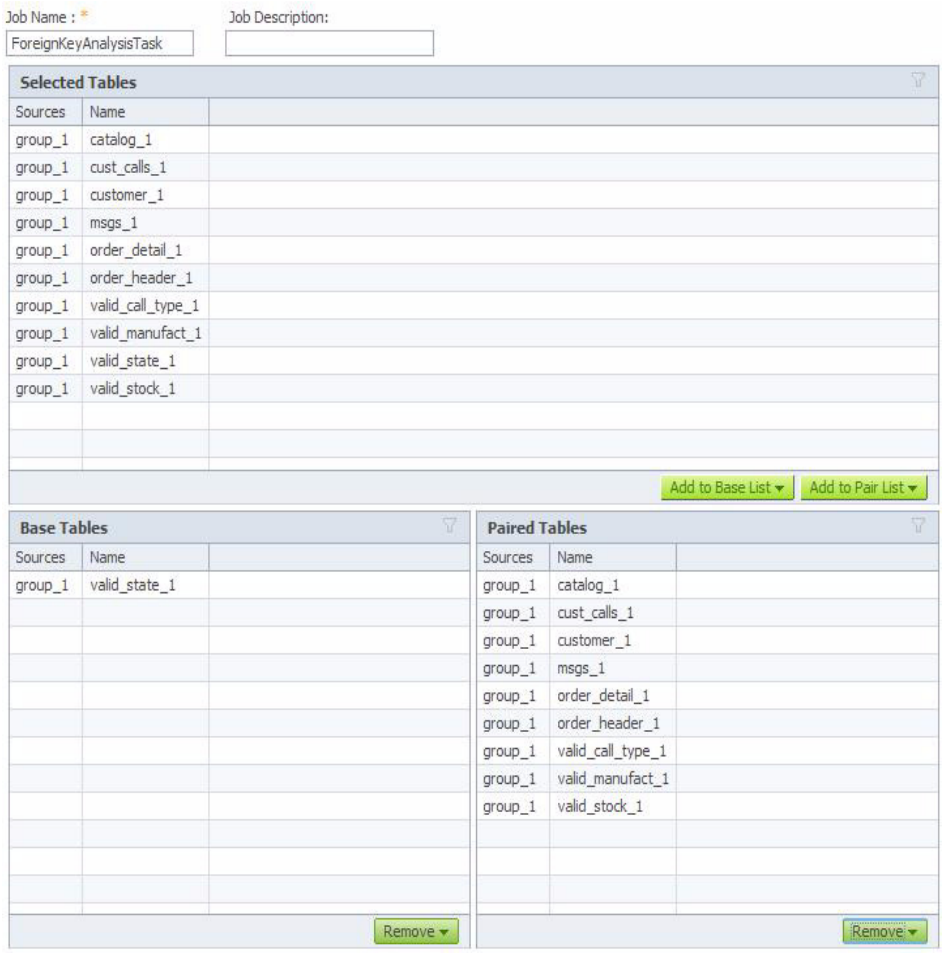


Figure 31-2 Looking for a Foreign Key to Valid_State.st_abbr, Step 1.

– Because of the work we did with Column Analysis, Information Analyzer already knows of all potential matches to Valid_State.st_abbr, which there is only one. Example as shown in Figure 31-3.

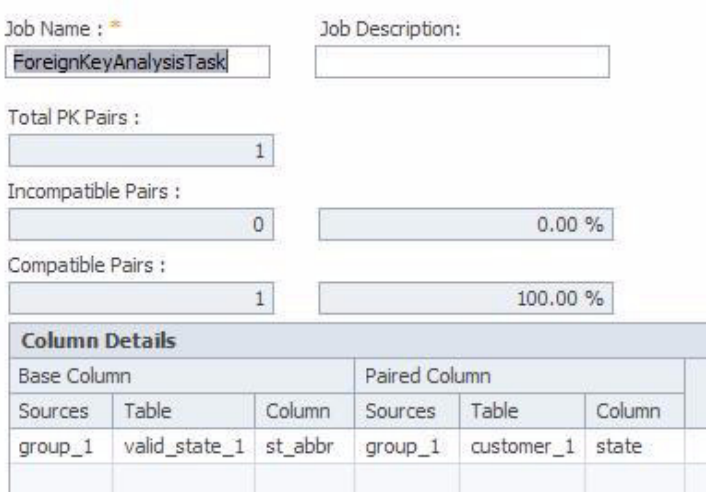


Figure 31-3 Looking for Foreign Key to Valid_State.st_abbr, Step 2.

– After a Finish -> Finish and Close, we monitor our named Job to complete, and then select, Open Foreign Key Analysis for the Valid_State table.

  On doing so, we receive the following notification, as shown in Figure 31-4.

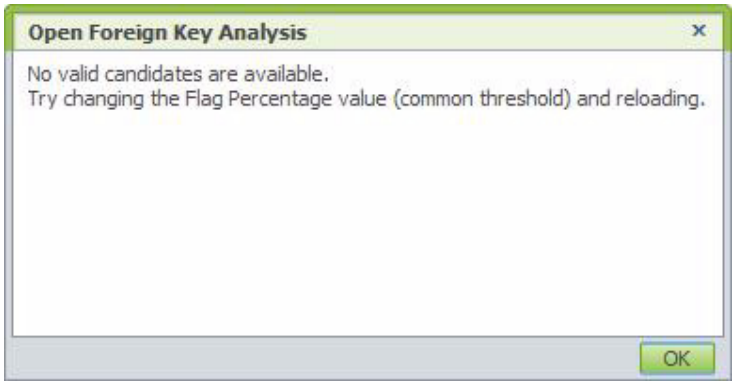


Figure 31-4 Looking for Foreign Key to Valid_State, Step 3.

– We don't know why there is no match quite yet, so we Un-Check, 'Show only valid candidates', and Click Reload. Example as shown in Figure 31-5.

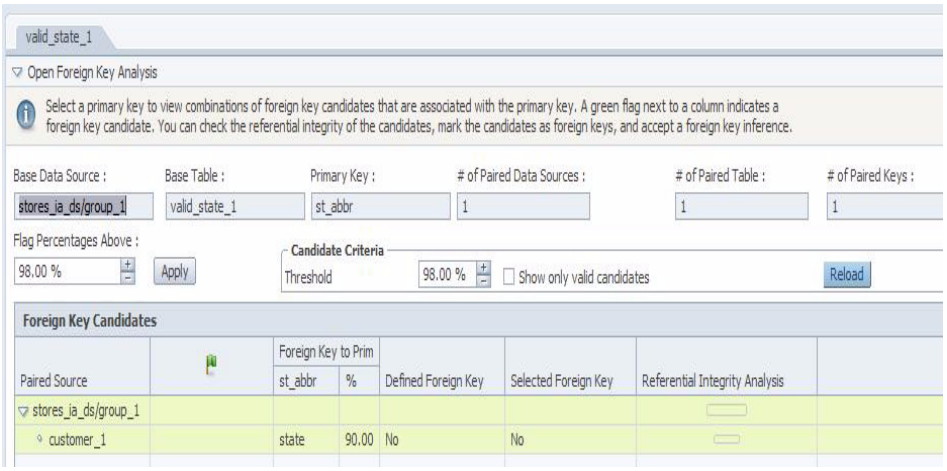


Figure 31-5 Looking for Foreign Key to Valid_State, Step 4.

– If we then highlight the single possible match in the Customer table, and Click, View Details, our condition becomes clear. Example as show in Figure 31-6.

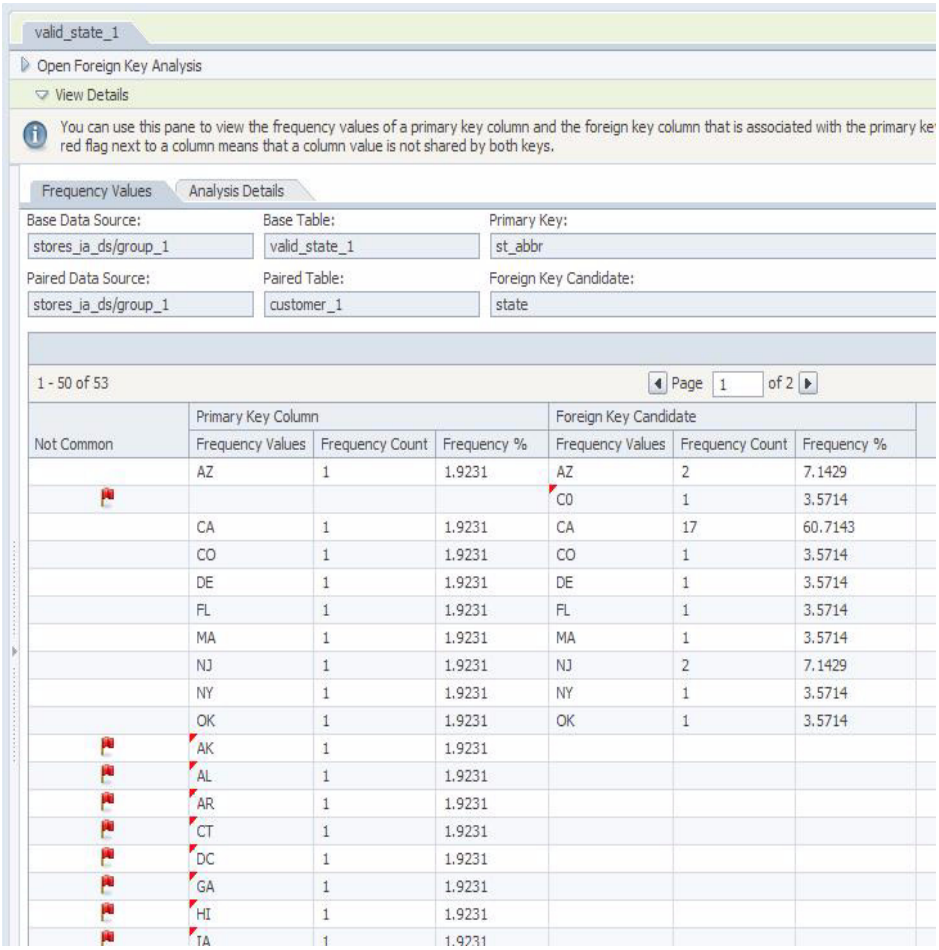


Figure 31-6 Looking for Foreign Key to Valid_State, Step 5.

– In Figure 31-6 above, our problem becomes clear.

  • We have 50 or more States in the Valid_State table, that's normal. But, this sample database, this list of Customers is for a regional supplier; they only have Customers in 5 or so States.

    And, one of the Customer States is not in the Valid_State table; one of the Customer States is spelled, "C (zero)", not "CO" for Colorado as you would expect.

Note: The above happens all of the time. With Referential Integrity being a standard database server feature, and commonly used; How did it happen that we have a State entry which is invalid?

Easy, almost every database server vendor supports Referential Integrity between 2 Tables in the same database, not across 2 Tables in Separate or Federated databases.

- The only way you could have previously known that the Customer tables had a "CO" Customer and a "C (zero)" would be to read every line in that table (and be able to distinguish zero's from Oh's, ones from L's; that's not a business we want to be in. Here, with Foreign Key Analysis, we let the software do that work for us.

  We could still accept the Foreign Key relationship between Valid_State and Customer, or we could begin to use a feature within Information Analyzer called, Virtual Tables.

  Before we get to Virtual Tables, we are going to create a Violations Table for Customer.

Violations Tables

Above we detected that the are bad records in the Customer table; with an invalid State Abbreviation, that Customer will not accurately display on any operational or decision support reports we create.

Further, if you look at an Investigate -> Column Analysis (highlight Customer.Phone) -> Open Column Analysis -> View Details -> Format TAB, you see also that we have phone numbers in 3 distinct formats. That would be bad if later we try to match on phone number.

Here we are going to record these bad values inside tables. Complete the following;

1. Make a Violations Table for Customer.Phone.
   a. From the Menu Bar, Select, Investigate -> Column Analysis -> (highlight Customer.State) -> Open Column Analysis -> View Details.
   b. All of these 2 character State abbreviations have a Format of 'AA'; meaning a 2 alpha characters. One of these State abbreviations is displayed as A9; meaning, a letter following by a number, "C (zero)".
   c. Move to the Domain and Completeness TAB.

In the Frequency Distribution table, Highlight the "C (zero)" entry, and change its Status to, Invalid.

You should see a new entry created in the Invalid Values table.

d. Click Save, then Click, Reference Tables -> New Reference Table.

We called our Table, "Z_INVALID_CUSTOMER_STATE_ABBR", Check Invalid, and Click, Preview.

You should see the invalid States values in the display to the right.\

And Click Save.

2. Repeat steps Step1,a-d above, and make a new Violations Table for Customer.Phone. Call the Violations Table, Z_INVALID_CUSTOMER_PHONE".

Valid Phone values are those similar to, "(303)555-1212".

Note: Good or bad, Violations Tables are currently created and maintained in the IADB database; the same database that houses all of this analysis data.

If we wish to Join source data with these values later, we probably wish for these tables to be hosted in the same source database we are reading from (performing analysis on). That, or we need to grant read access to the IADB database.

There is currently no menu driven means to migrate these tables. You have to use Oracle, MS/SQLServer or DB2/UDB skills, or use the DataStage component to Information Server. (We prefer using DataStage, since its allows us to not have to write SQL, or get our hands dirty.)

The Violations Table has a very simple data model; its has the Table name you gave it above, a single Column named, "distinctvalue", and it has the column data type of the column you sourced it from.

In the examples that follow, we assume you have copied these Violations Tables to your source database.

Information Analyzer Virtual Tables (8.1.1)

New in release 8.1.1 of Information Analyzer (IA) is a feature called Virtual Tables. A Virtual Table is somewhat like a SQL View, it can contain a subset of Columns from a single Table, as well as a subset of rows. *Currently Virtual Tables can not Join 2 or more source Tables.*

In the example above, we were analyzing the Group1.Customer Table, and found that it contained bad data from 2 Columns. We wish now to create a 'clean' version of that Table, with only good records.

Complete the following;

3. Create a new Virtual Table containing only good records from the Group_1.Customer Table.

   a. From the Menu Bar, Select, Investigate -> Column Analysis -> (highlight Group_1.Customer), and Select, Create Virtual Table.

      Manage the display to equal that as displayed in Figure 31-7 and Figure 31-8.

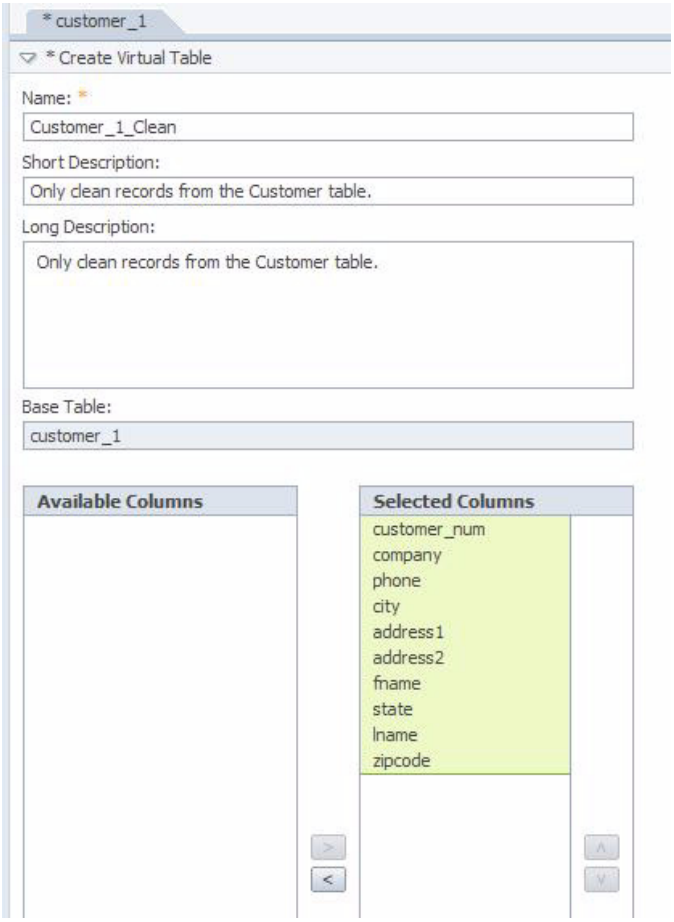


Figure 31-7 Left side of display.

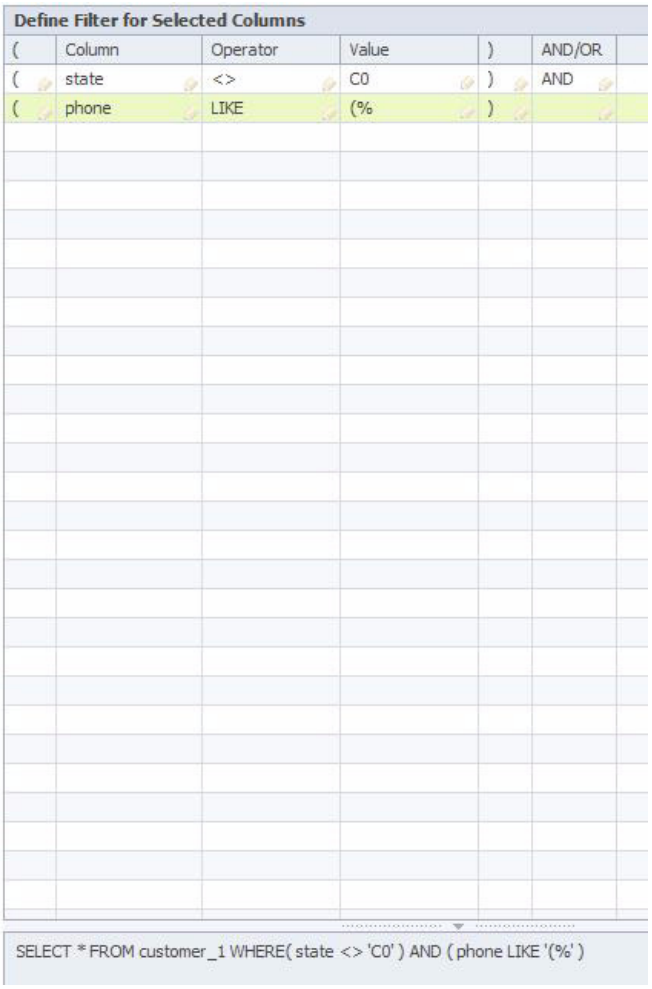


Figure 31-8 Right side of display.

If you wish to get the most powerful definition, you have to use the Free Form Editor to achieve what is displayed in Figure 31-9.

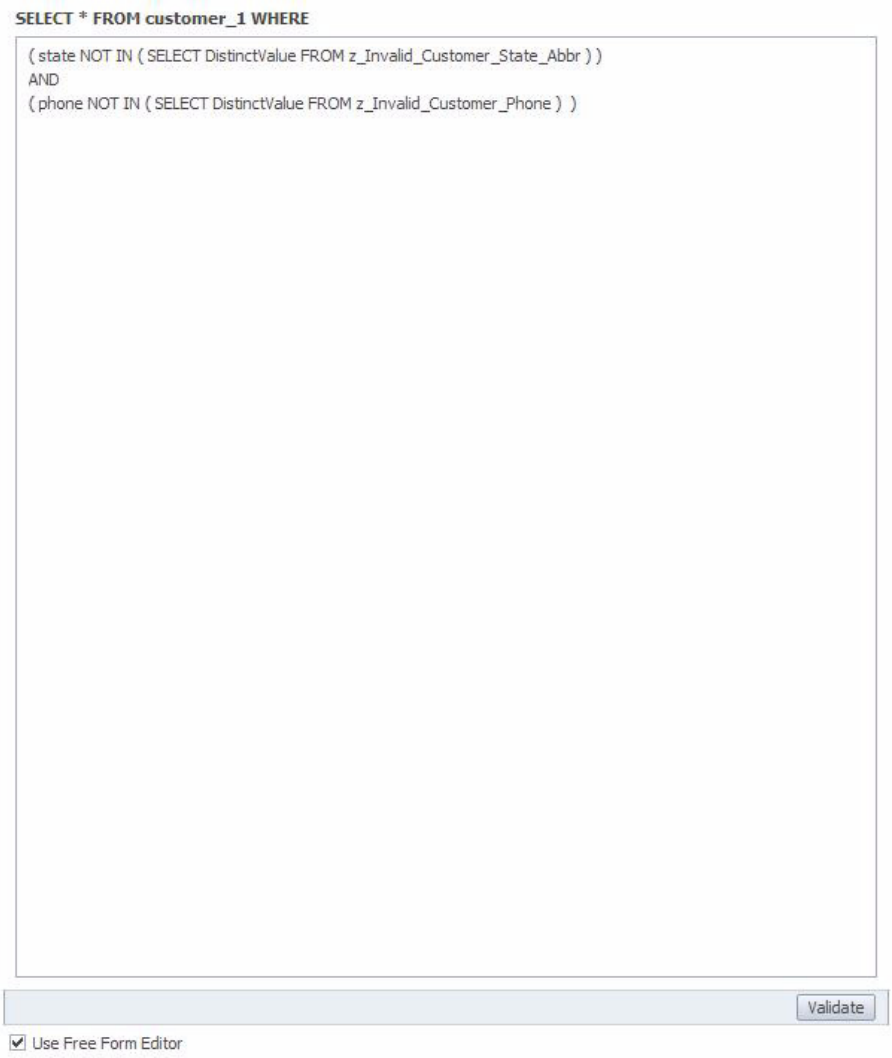


Figure 31-9 Most advanced solution to the above; reads data dynamically.

b. Click Save -> Save and Close.

   To Check your work, Run a full Column Analysis on your newly create, Virtual Table. Recheck the Foreign Key Analysis, that's what kicked this whole thing off.

Note: You don't have to create a Virtual Table in order to analyze a subset of a Table. You can apply a WHERE clause directly to a Table via,

From the Menu Bar, Select, Overview -> Project Properties -> Analysis Settings TAB -> Data Source View -> (highlight Table) -> Modify -> Where clause TAB.

You can also overview all WHERE clauses to Virtual Tables here.

Using Virtual Columns

Virtual Tables, listed above, are a new feature with version 8.1.1 of Information Analyzer. We use Virtual Tables in a variety of means. Above we used them to separate good data from bad, allowing us to perform analysis to the point of knowing we have only good data.

Virtual Columns have been a feature of Information Analyzer for some time. Multi-column Primary Keys and Foreign Keys automatically get a Virtual Column created on them, so that we may analyze the properties of this concatenated key; uniqueness, domain and other properties.

In the 2 prior editions of this document, and when we performed BaseLine Analysis, we reviewed a table that lost 1 column and gained 3 more. Example as displayed in Figure 31-10.

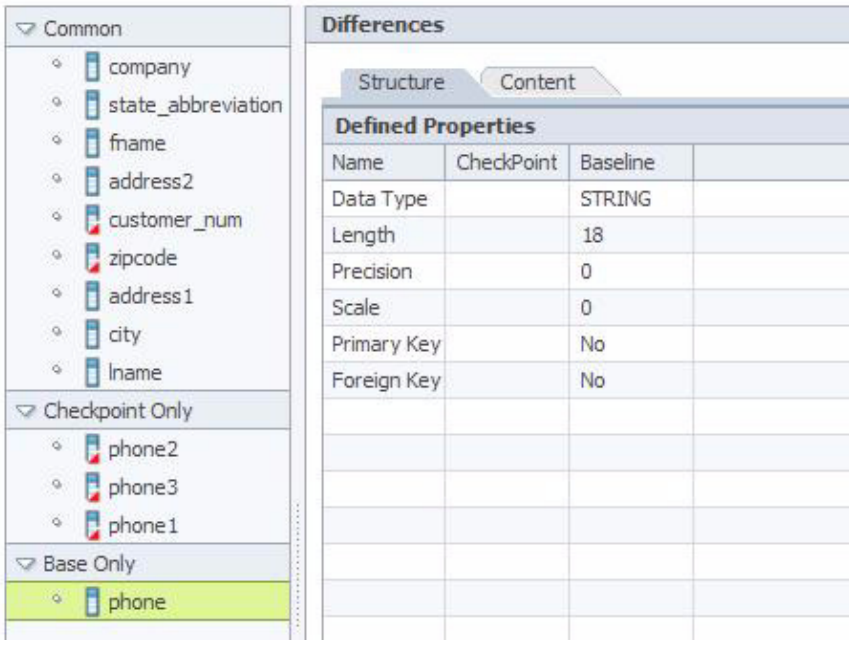


Figure 31-10 Cross Domain Analysis, table lost 1 and gained 3 columns.

In Figure 31-10, Cross Domain Analysis tells us that the Phone column was split into 3 columns; phone1, phone 2, and phone3. (We also need the Column Analysis report to allow us to see column content.) Someone, it appears, has over normalized the Phone number column into Area Code, Exchange, and so on.

If, however, we wish to analyze the now 3 columns with the remainder of our phone numbers which are still kept in 1 column, we will need to concatenate these 23 new columns, by using a Virtual Column.

Note: This is but one use case for Virtual Columns. Basically anytime you need to analyze multiple columns (their concatenated result), you use Virtual Columns.

Complete the following;

4. Create a new Virtual Column, containing the formatted values from Phone1, Phone2 and Phone3.

a. Return to the Column Analysis Menu, highlight the Group_Delta.Customer_d table, Click Manage Virtual Columns, and then Click, Create New.

b. Manage your display to equal that as shown in Figure 31-11, a Click Save.

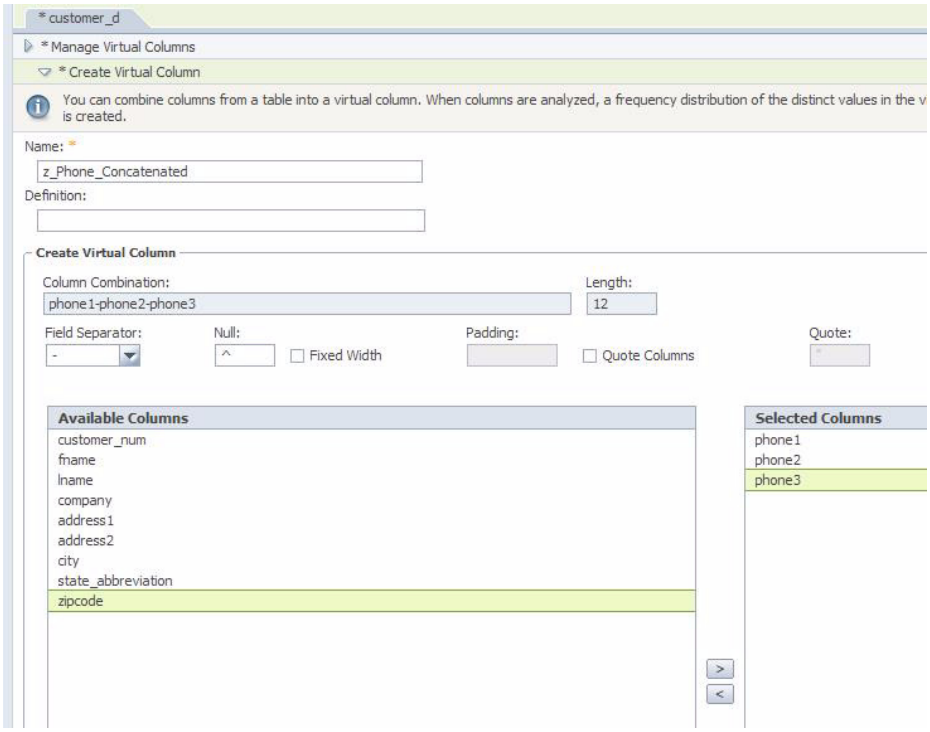


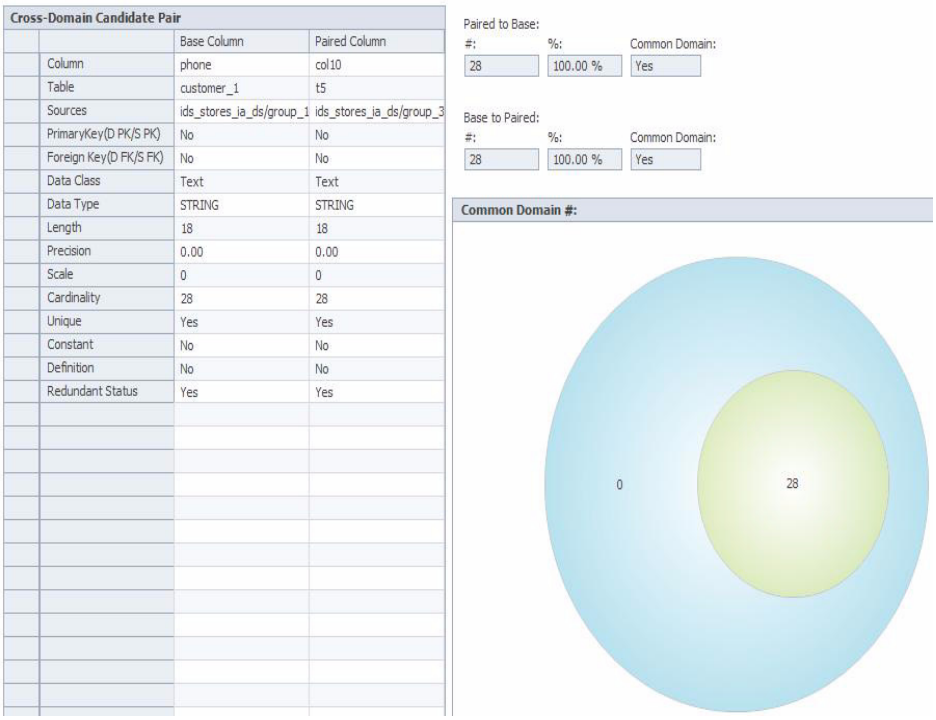
Figure 31-11 Creating a Virtual Column on Phone.

c. Perform Column Analysis on this new column.

d. Perform a Cross Domain Analysis of this new Virtual column into Customer.

   Since we really didn't perform strong formatting on our new column, it appears as, "303-555-1212", we will only match phone numbers in that exact format. Data of this format appears in our Group_1.Customer_1.Phone column.

   Example as shown in Figure 31-12.



Cross-Domain Candidate Pair

	Base Column	Paired Column
Column	phone	col10
Table	customer_1	t5
Sources	ids_stores_ia_ds/group_1	ids_stores_ia_ds/group_3
PrimaryKey(D PK/S PK)	No	No
Foreign Key(D FK/S FK)	No	No
Data Class	Text	Text
Data Type	STRING	STRING
Length	18	18
Precision	0.00	0.00
Scale	0	0
Cardinality	28	28
Unique	Yes	Yes
Constant	No	No
Definition	No	No
Redundant Status	Yes	Yes

Figure 31-12 Cross Domain Analysis, Virtual Column to Column.

Reporting

In the prior editions of this document centered on the Information Analyzer component to IBM InfoSphere Information Server, we detailed how to create report; more specifically, we detailed the Cross Domain Analysis Report.

Example as shown in Figure 31-13, and Figure 31-14 displaying the Valid_State and Group_2.t1 tables.

Cross-Table Domain Analysis

Common Domains

Project :	IISDN_Proj_IA
Report Name :	My Cross Domain Report
Report Generated :	2009-12-18 09:02:40
Time Zone :	UTC -07:00
User :	iisadmin iisadmin
Comments :	My Cross Domain Report
Host Name :	RHHOST.GRID
Data Store :	stores_ia_ds
Data Store Alias :	
Database Name :	group_2
Database Alias :	
Table Name :	t1

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
t1	valid_state	col01	state_abbr	52	52	52	100.000%	100.000%
		col02	state_name	52	52	52	100.000%	100.000%

Figure 31-13 Cross Domain Analysis Report, page 1.

Table Notes

Group_2.t1 table incorrectly loaded twice.

Type :	Action	Status :	Opened	Created :	2009-12-17	Modified :	2009-12-17
Creator :	iisadmin						

Group_1.t1.col01

Every key value is found twice.

This table appears to have been loaded twice incorrectly.

Figure 31-14 Cross Domain Analysis Report, page 2.

Other reports we like include;

– Column Domain Values, examples as shown in Figure 31-15 and Figure 31-16.

Column Domain Values

Domain Analysis Detail

Project :	IISDN_Proj_IA
Report Name :	Domain Analysis Detail 20091218T161443Z
Report Generated :	2009-12-18 09:07:12
Time Zone :	UTC -07:00
User :	iisadmin iisadmin
Comments :	
Host Name :	RHHOST.GRID
Data Store :	stores_ia_ds
Data Store Alias :	
Database Name :	group_1
Database Alias :	
Table Name :	customer
Table Alias :	
Column Name :	state_abbr
Column Alias :	
Data Class :	Text
Sort Order :	Descending by Count of Value
Excluded Indicator(s) :	

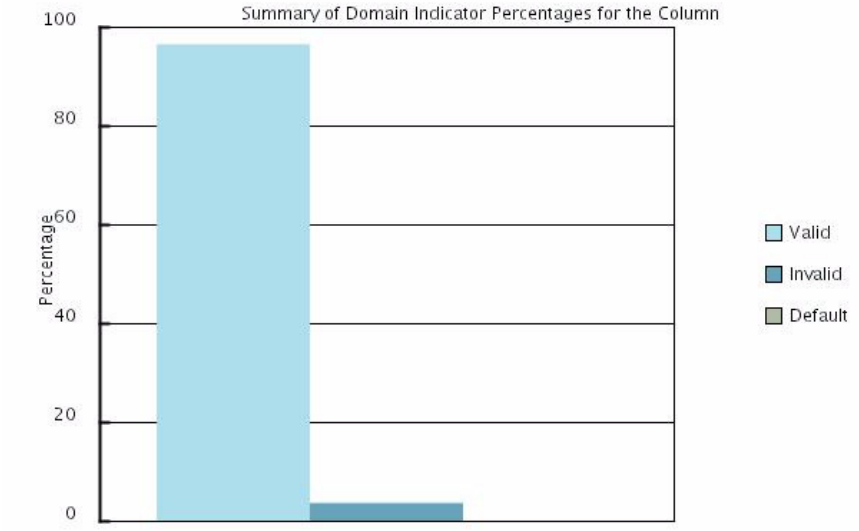
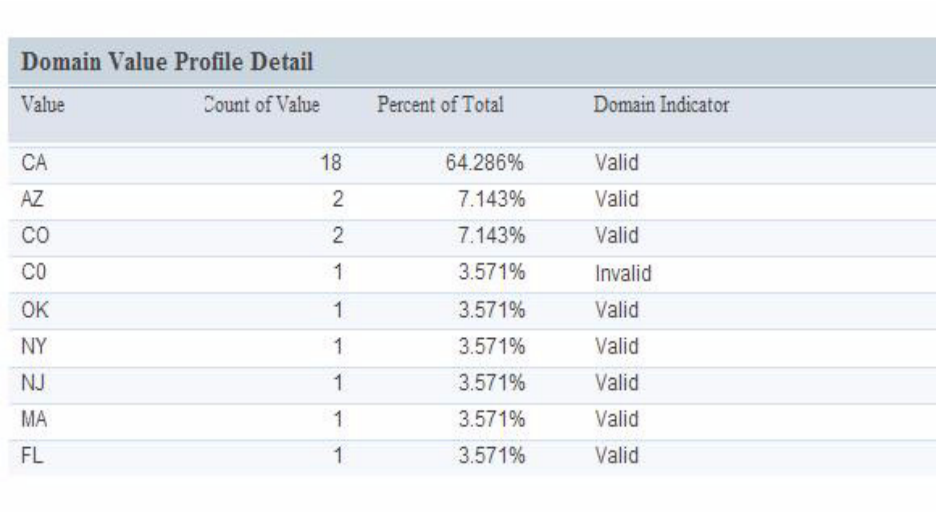


Figure 31-15 Common Domain Values Report, page 1.



Domain Value Profile Detail

Value	Count of Value	Percent of Total	Domain Indicator
CA	18	64.286%	Valid
AZ	2	7.143%	Valid
CO	2	7.143%	Valid
C0	1	3.571%	Invalid
OK	1	3.571%	Valid
NY	1	3.571%	Valid
NJ	1	3.571%	Valid
MA	1	3.571%	Valid
FL	1	3.571%	Valid

Figure 31-16 Common Domain Values Report, page 2.

– And then also the Format Violations Report, displayed in Figure 31-17.

Column Frequency

Format Violations

Project :	IISDN_Proj_IA
Report Name :	My Format Violations
Report Generated :	2009-12-17 16:00:20
Time Zone :	UTC -07:00
User :	iisadmin iisadmin
Comments :	Format Violations, Group_1.Customer.Phone
Host Name :	RHHOST.GRID
Data Store :	stores_ia_ds
Data Store Alias :	
Database :	group_1
Database Alias :	
Table :	customer
Table Alias :	
Column :	phone
Column Alias :	

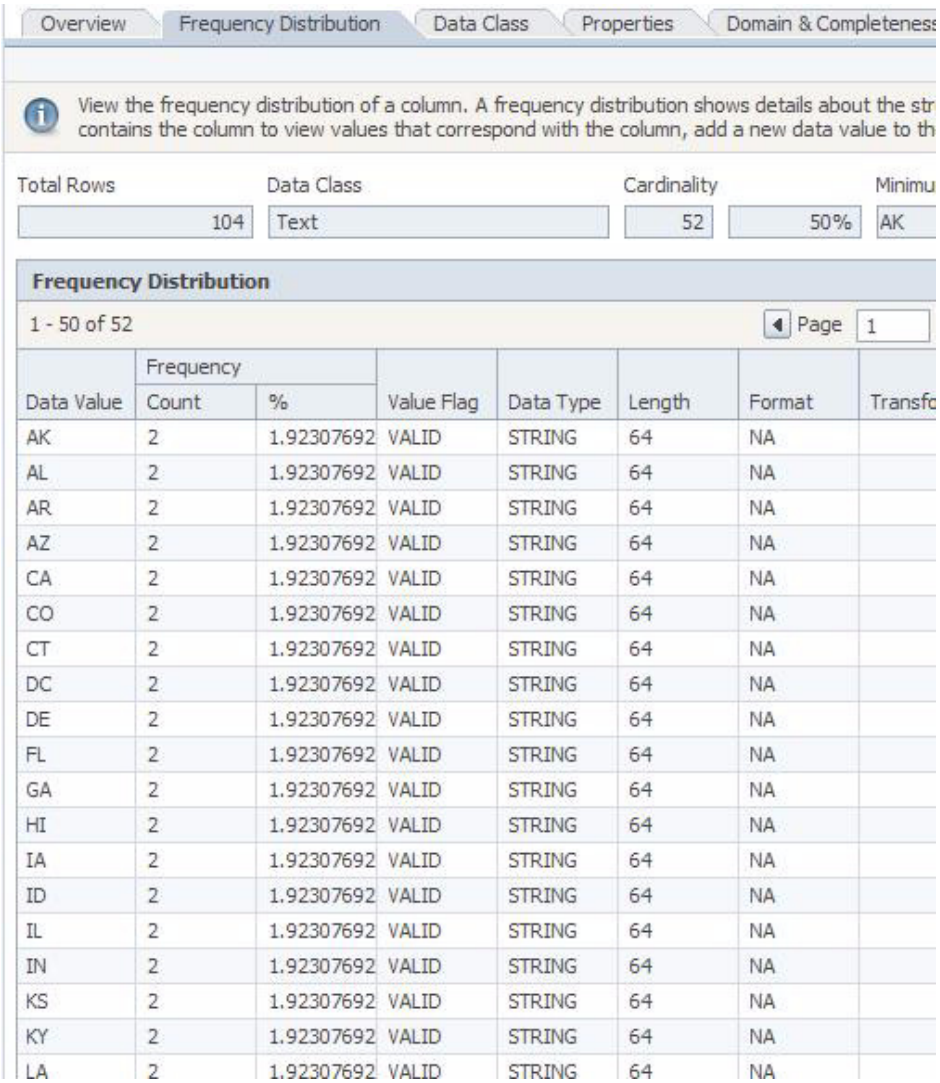
Column Level Details

Status	Format	Count	Total Rows %	Example Values
Conform	(999)999-9999	18	64.29	(408)277-7245
				(415)887-7235
				(415)886-6677
				(415)822-1289
				(415)776-3249
Violation	999-9999	7	25.00	232-4159
				944-5691
				823-4239
				663-6079
				533-1817
Violation	999-999-9999	3	10.71	303-936-7731
				719-395-7511
				719-395-0054

Figure 31-17 Format Violations Report.

Sharing analysis data with ETL developers

Through our analysis of these tables and in this example, we discovered that the Group_2.t1 table was incorrectly loaded twice. Example as shown in Figure 31-18.



Data Value	Frequency Count	Frequency %	Value Flag	Data Type	Length	Format	Transfo
AK	2	1.92307692	VALID	STRING	64	NA	
AL	2	1.92307692	VALID	STRING	64	NA	
AR	2	1.92307692	VALID	STRING	64	NA	
AZ	2	1.92307692	VALID	STRING	64	NA	
CA	2	1.92307692	VALID	STRING	64	NA	
CO	2	1.92307692	VALID	STRING	64	NA	
CT	2	1.92307692	VALID	STRING	64	NA	
DC	2	1.92307692	VALID	STRING	64	NA	
DE	2	1.92307692	VALID	STRING	64	NA	
FL	2	1.92307692	VALID	STRING	64	NA	
GA	2	1.92307692	VALID	STRING	64	NA	
HI	2	1.92307692	VALID	STRING	64	NA	
IA	2	1.92307692	VALID	STRING	64	NA	
ID	2	1.92307692	VALID	STRING	64	NA	
IL	2	1.92307692	VALID	STRING	64	NA	
IN	2	1.92307692	VALID	STRING	64	NA	
KS	2	1.92307692	VALID	STRING	64	NA	
KY	2	1.92307692	VALID	STRING	64	NA	
LA	2	1.92307692	VALID	STRING	64	NA	

Figure 31-18 Open Column Analysis -> View Details -> Frequency Distribution.

As a result of this discovery, we wish to send a note to the ETL developer, to either correct or handle this anomaly. We can send and receive notes in many

areas. Generally this button is available in the upper right region of the display. Example from Column analysis displayed in Figure 31-19 and Figure 31-20.

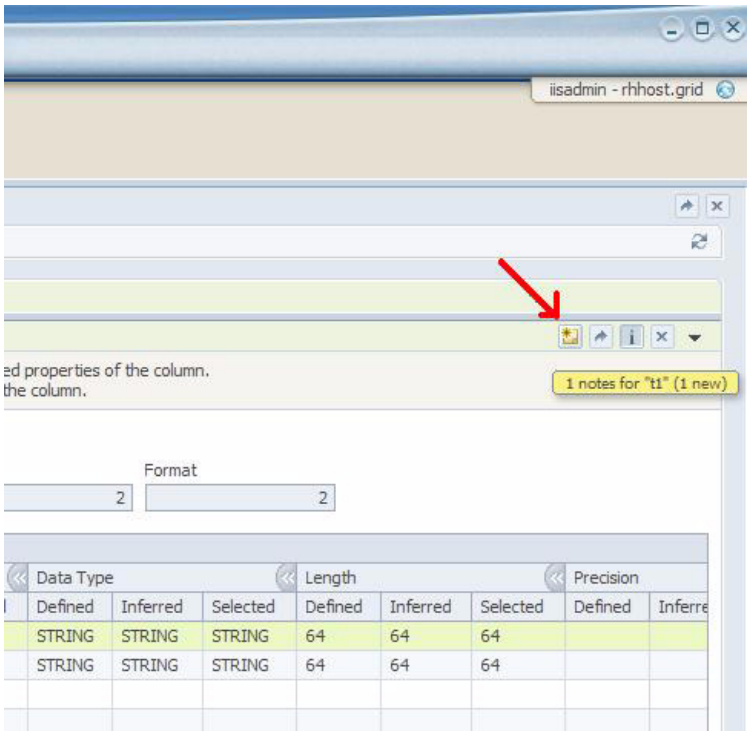


Figure 31-19 Sending and receiving Notes, in this case, to ETL developer., page 1.

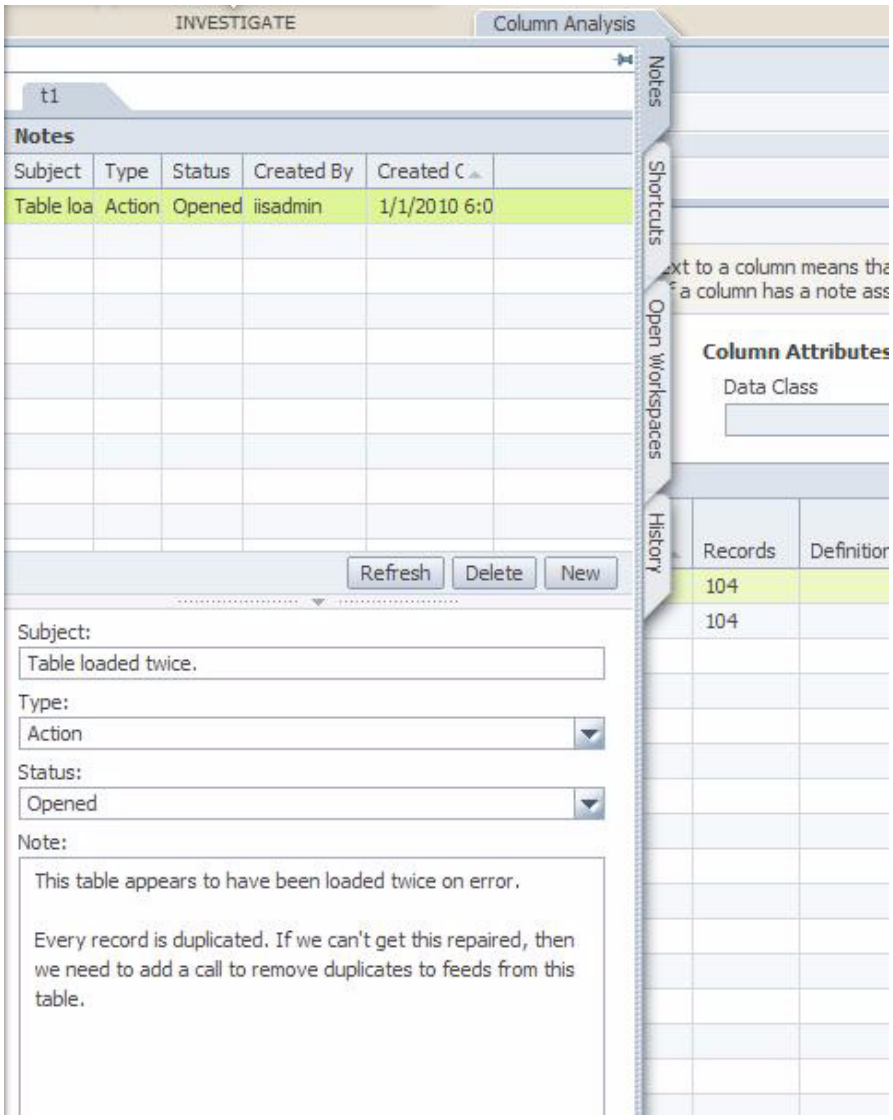


Figure 31-20 Sending and receiving Notes, in this case, to ETL developer., page 2.

Note: This Note and this Meta Data derived from Column Analysis is published to the Shared Meta Data Repository from the Menu Bar,

Investigate -> Publish analysis Results -> (highlights objects to publish) -> Publish Results Summary.

An example of a subset of the shared meta data shared with the ETL developer appears in Figure 31-21.

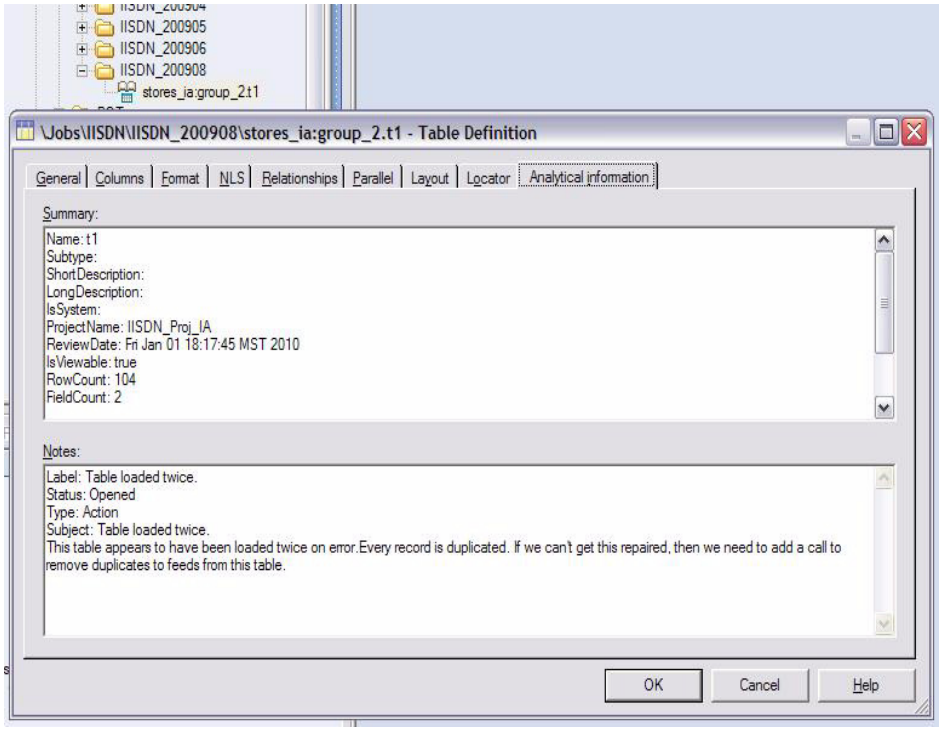


Figure 31-21 Subset of meta data shared with ETL developer.

Diagnosing Information Analyzer (internal) operations

The Information Analyzer (IA) component to IBM InfoSphere Information Server (IIS) executes a lot of native SQL statements, and as you can find yourself analyzing a wide variety of data sources, it is useful to be able to trace the SQL that is generated and then run.

The steps below detail how to set up and then view a trace of the SQL that Information Analyzer submits.

Complete the following:

5. Create a new Logging Configuration and a new Log View.
   a. Log on to the Information Server Web Console, and Select, Administration TAB -> Log Management -> Logging Components.
   b. Click the Radio Button entitled, "Connector Access", then Click, Manage Configurations.
   c. Check the ConnectorAccess.WARN control, then Click, Copy.
   d. Change all of the displayed Event Categories and their associated, Severity Levels to, "All", then Click, Save and Close.

      Example as shown in Figure 31-22.

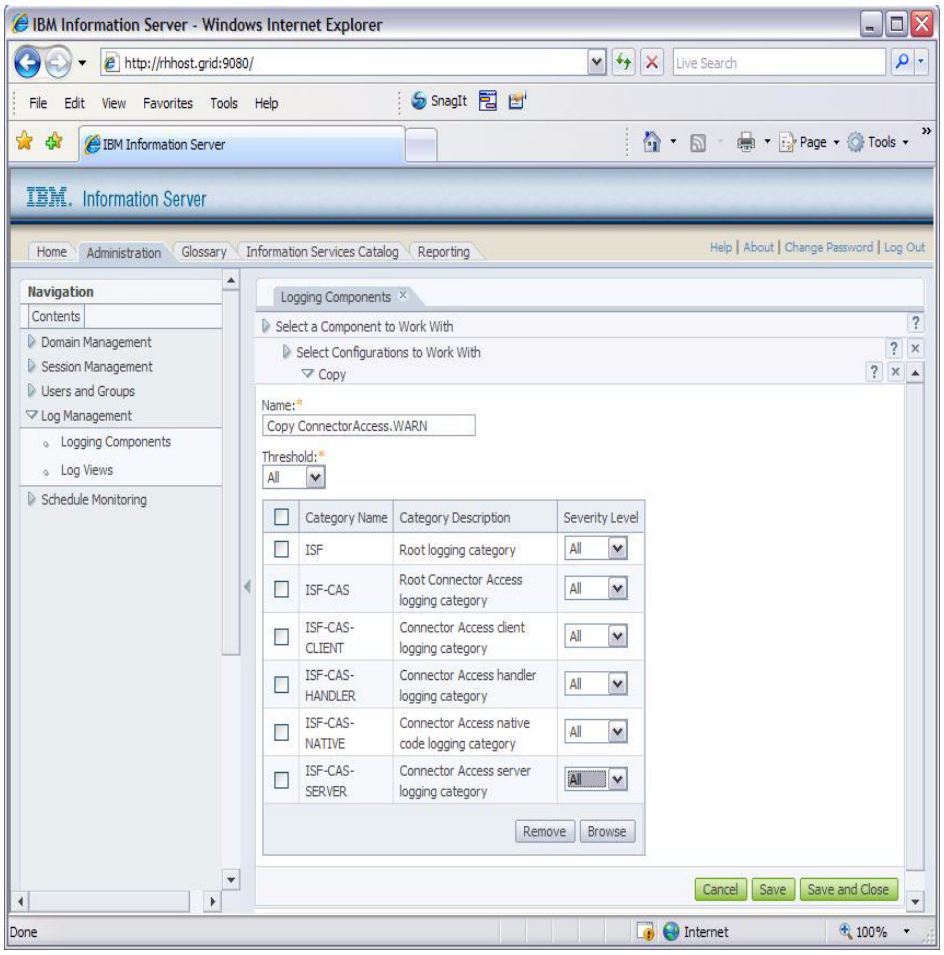


Figure 31-22 Setting Severity Level for these Logging Events.

The action above results in a new Logging Configuration. Example as shown in Figure 31-23.

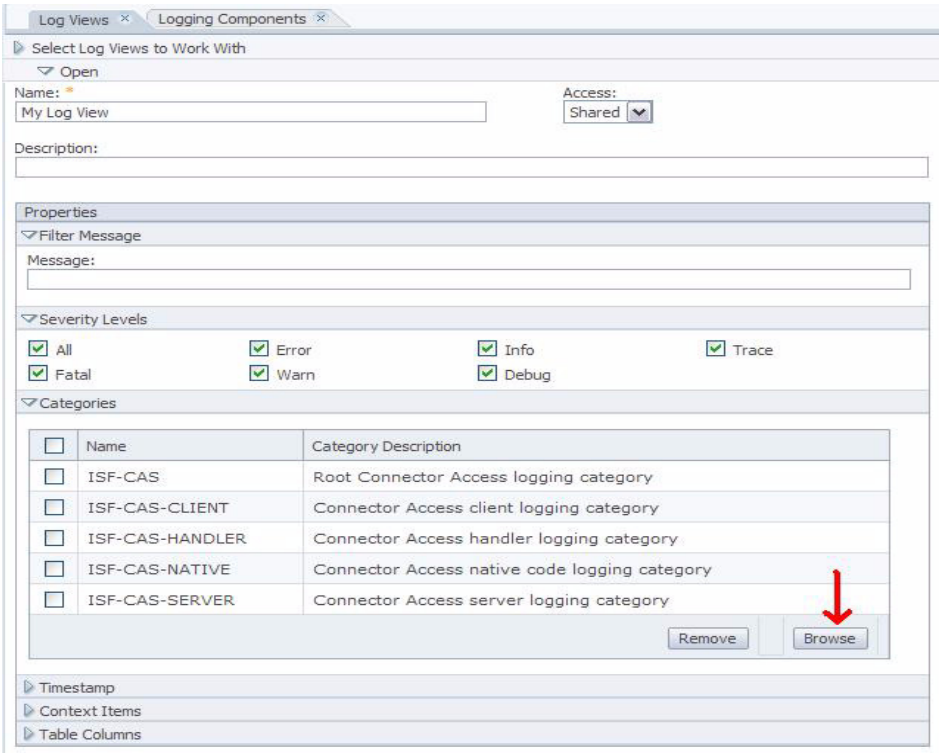


Figure 31-23 New Logging Configuration.

Note: Be certain to go back in later, and delete this Logging Configuration, setting the original to Active.

e. Now perform the activity you wish to trace.

   We chose, Open Column Analysis -> (highlight a given column) -> DEtails -> Frequency Distribution -> (highlight a given value) -> Drill Down.

   Wait 1 or 2 minutes, for the logging data to arrive.

f. View the given Log.

   Still inside the Information Server Web Console, Select, Administration TAB -> Log Management -> Log View.

   Click, New Log View.

   Give the Log View a name, then Click Browse under the Categories section of the display. Example as show in Figure 31-24.

Under "Severity Levels", Click, All.

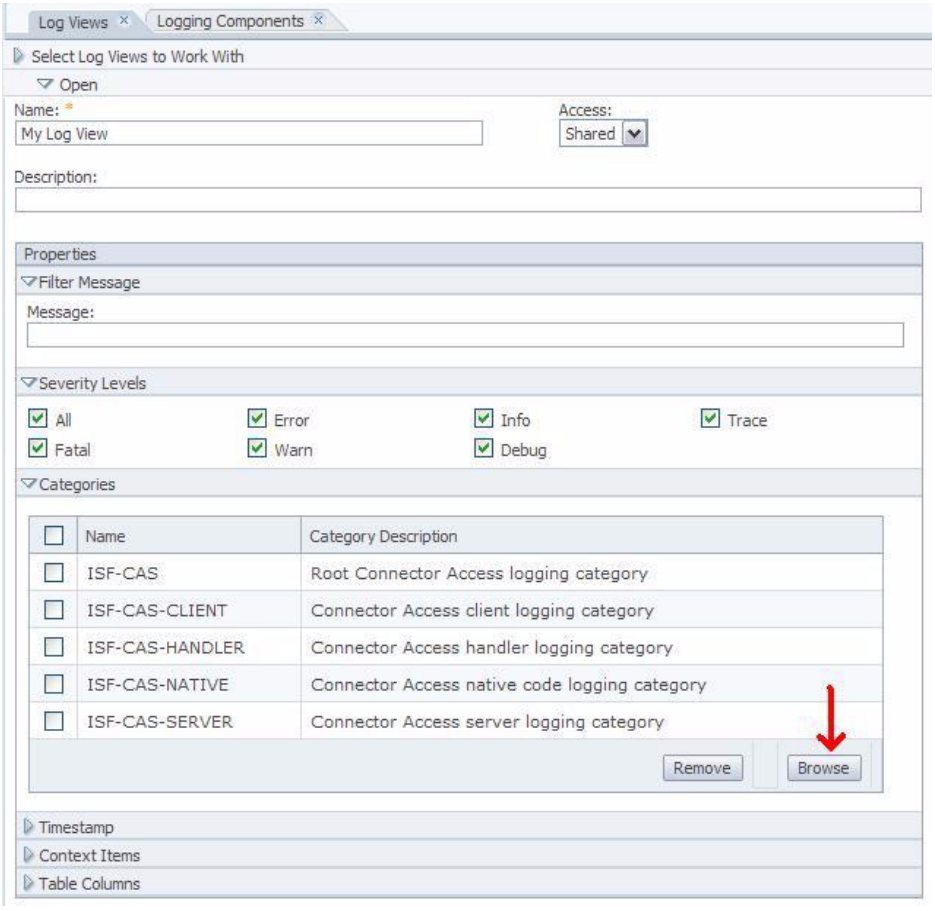


Figure 31-24 Log View, part 1.

g. In the "Browse Categories" dialog box that is produced, select all Categories entitled, "ISF-CAS" something.

   Example as shown in Figure 31-25.

   Then Click, OK.

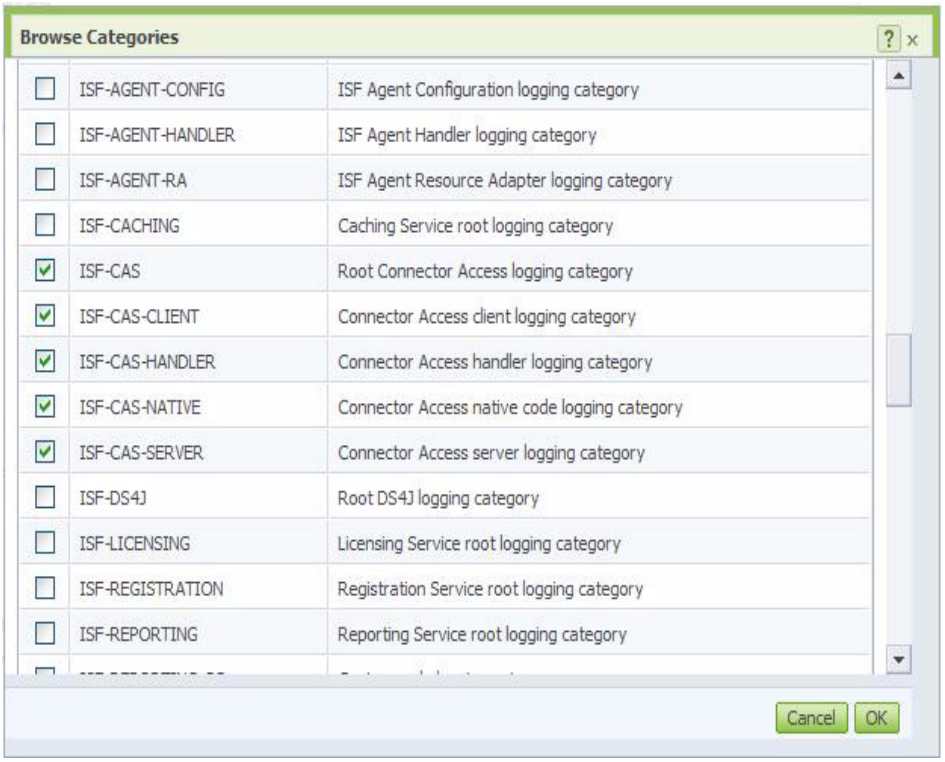


Figure 31-25 Log View, part 2.

h. And Click, Save and Close.

i. Your newly created Log View should appear in the list of available Log Views.

   Check your newly created Log View, Click, View Log -> Export Log, and then Save to File.

   Use your favorite program or line editor to view the log.

   Example as shown in Figure 31-26.

Note: Here we had introduced an error; we changed the name of the Violations Table inside the database using SQL. We should have changed it only through Information Analyzer.

Again, remember to go back and delete this Logging Configuration.

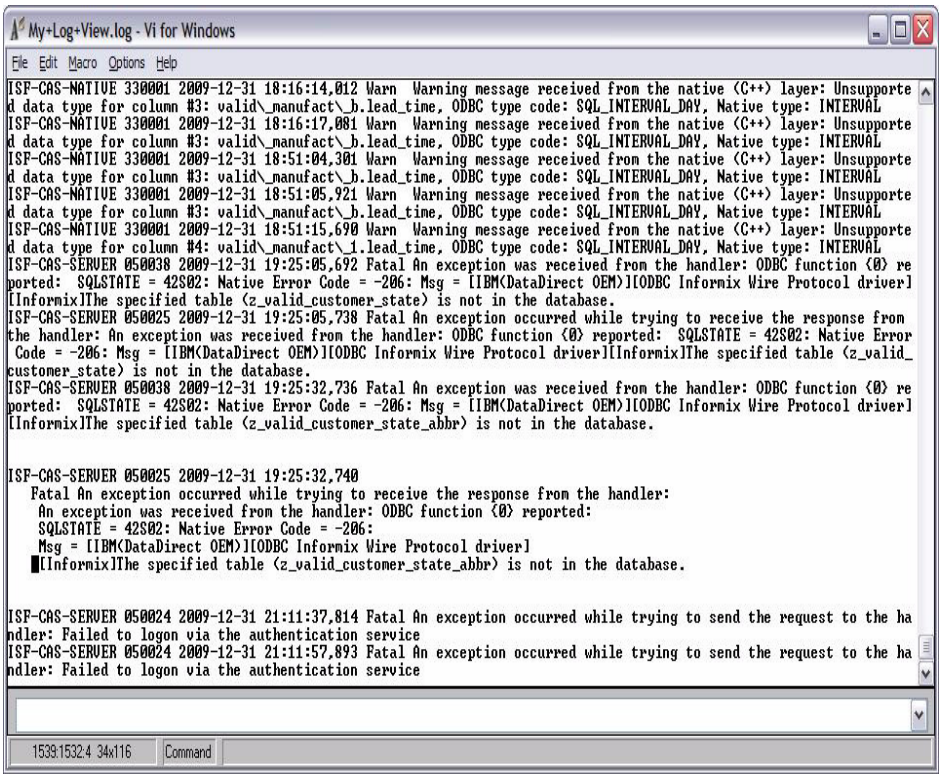


Figure 31-26 Exception from Drill Down command.

Using the new Flat file Definition Wizard (8.1.1)

Using the Information Analyzer component to IBM InfoSphere Information Server with Flat File data sources is easier in release 8.1.1 with the new Flat File Definition Wizard. After an initial one time configuration of the ODBC Text File Driver, the remainder of the task of importing Flat file definitions is easy enough to do for end users.

Complete the following;

6. One time only, define the ODBC Text File Driver Definition and add it to the Analyzer Project.
   a. Edit the /opt/IBM/InformationServer/Server/DSEngine/.odbc.ini file as displayed in Figure 31-27.

      This is the default path name to this file; your path name may vary.

      We copied the entry entitled, "[Text]", and edited it as shown below.

Don't forget to add a single new line entry to the "DSN Entries Table" at the top of the file.

For the entry entitled, Database, we enter the Directory path name we wish to give to users to place and then process ASCII Text Files from. There should be no trailing slash after this value. Most other values do not matter, as we will be able to override them in the Flat File Definition Wizard that follows.

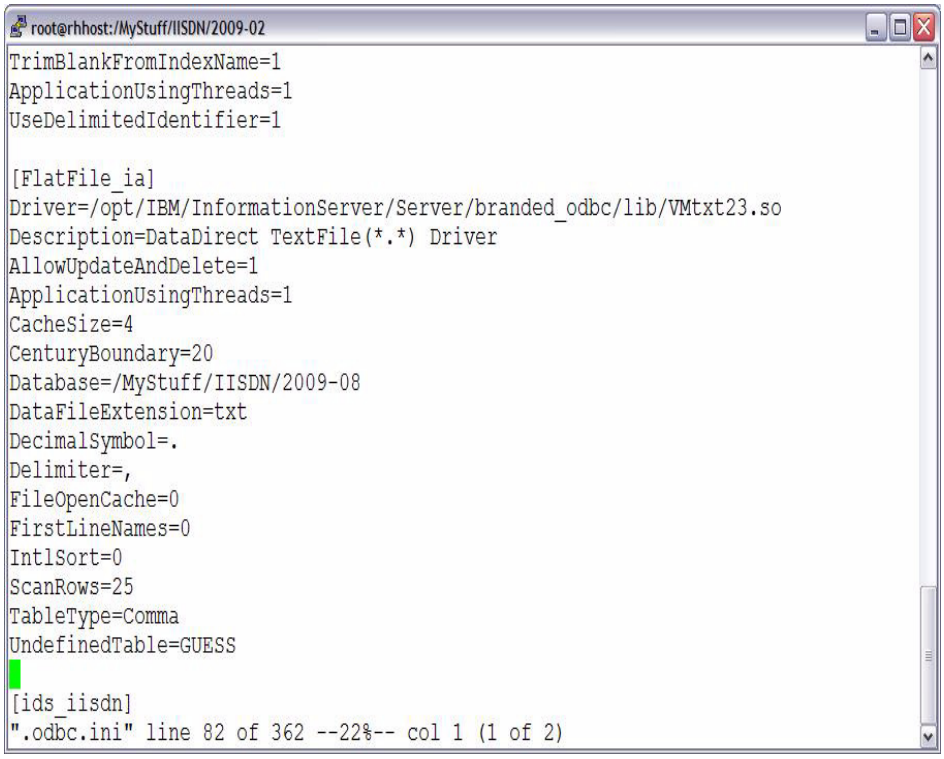


Figure 31-27 /opt/IBM/InformationServer/Server/DSEngine/.odbc.ini file.

b. Then edit the /opt/IBM/InformationServer/Server/Projects/uvodbc.config file as displayed in Figure 31-28.

   Again, this is the default path name to this file; yours may vary.

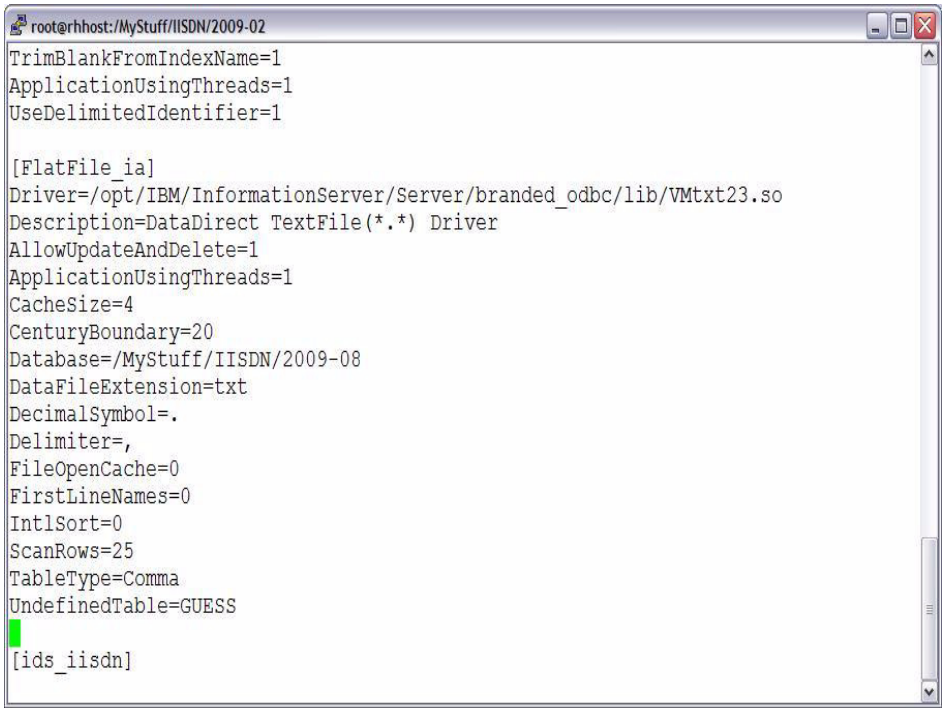


Figure 31-28 /opt/IBM/InformationServer/Server/Projects/uvodbc.config file.

7. One time only, create a new Data Store and Data Connection definition.

   Inside the Information Server Console, Select, Home -> Configuration -> Sources -> (highlight your given host), Select, New Data Store.

   After your screen matches the display in Figure 31-29, Click, Connect to test your work.

   Then Click, Save -> Save and Close.

   The above action defines a new system wide Data Store.

   Next we must use the Flat File Definition Wizard to gather meta data for any new flat files.

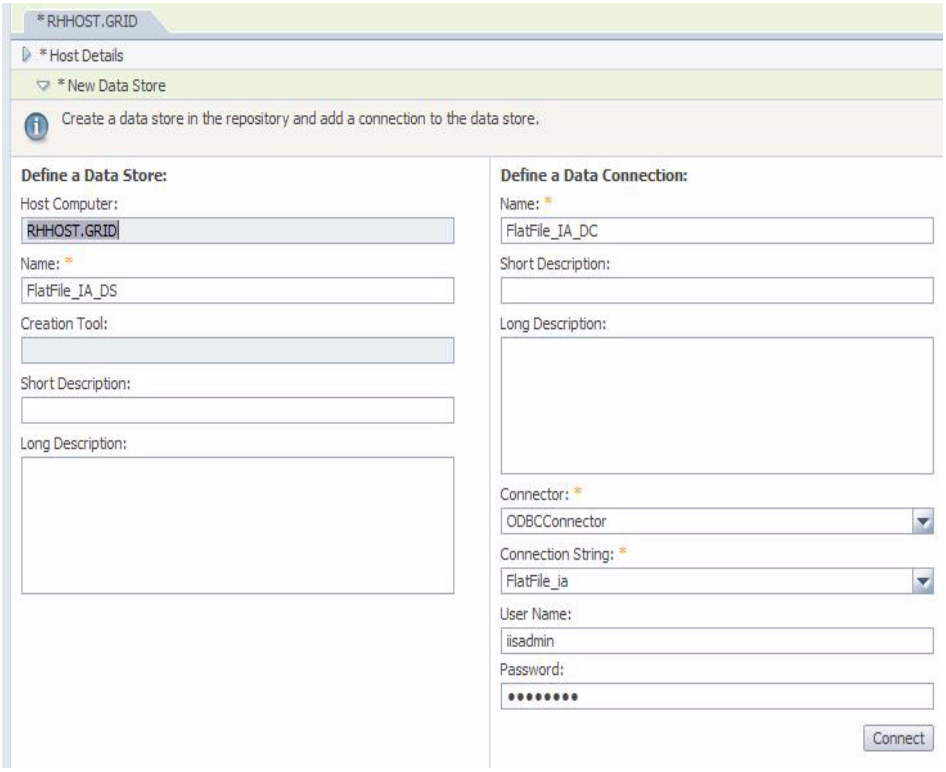


Figure 31-29 Creating a new Data Store of type Flat File.

8. Import Flat File Meta Data, and add to Project.

   This step must be done for each new Flat File.

   a. Inside the Information Server Console, Select, Home -> Metadata Management -> Import Metadata -> (highlight your newly created Data Store) -> Click, Identify Flat File.

      You will be lead through a series of 4 panes, and need to respond based on the definition of your exact file; what type of delimiters, other.

      Figure 31-30, displays the second pane, and results of the Update Preview button.

      The third pane allows you to overwrite the default output column data types and related. Experiment; there is no penalty for guessing and guessing wrong.

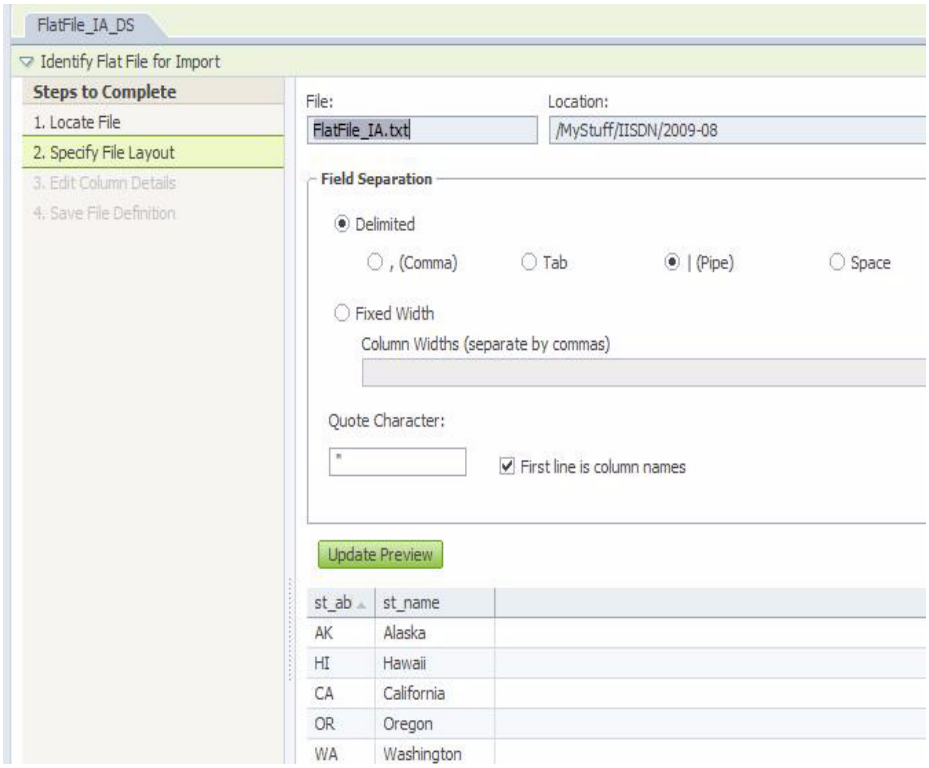


Figure 31-30 Flat file Import Wizard.

b. When your flat file definition meets your needs, Click, Finish -> Finish and Close.

   This action created a system wide definition for this Flat File, as a driver.

   Now we must import its given meta data; its given column names and types.

c. Highlight the entry you just created above. Our was called, FlatFile_IA_DS, and Click, Import Next Level.

   When this step is complete, highlight the next lowest level entry, and so on until you get to the file level.

Upon reaching the file level, Click, Import.

Note: Each of these levels gets us closer and closer to the file. You are not done until you can Click on Import, and Import the Column Definitions.

Why click on so many levels (2 or 3 in this case)? This program capability allows you to have multiple Directories or sub-groupings of files.

At this point of given flat file is defined at the system level. Now we must add this flat file definition to our list of available choices for our given Information Analyzer Project.

9. Adding a given (Table or) Flat file to our specific Information Analyzer Project.

   This step must be done for each new Flat File.

   a. Connect to your given Information Analyzer Project.
   b. Under the Main Menu, Select Overview -> Project Properties -> Data Source TAB -> Add Button (Its in the lower right).
   c. Navigate the display to find your newly added Flat File definition, and Click, OK.
   d. Click, Save All, when you are done.

      You should now be able to go under the Investigate Menu, and perform Column Analysis and all other tasks as though this Flat File with a standard (SQL) Table.

31.2 In this document, we reviewed or created:

We detailed tons of new features inside the Information Analyzer component to IBM InfoSphere Information Server, including;

- Violations Tables
- Virtual Tables and Virtual Columns
- Flat File Definition Wizard
- New Reports
- Shared meta data with ETL developers
- Custom logging for trouble shooting

Persons who help this month.

Steve Fazio, Robert Dickson.

Additional resources:

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.

