Getting Started Guide

Informatica Data Quality

(Version 8.5)

Informatica Data Quality Getting Started Guide

Version 8.5
August 2007

DISCLAIMER: Informatica Corporation provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of non-infringement, merchantability, or use for a particular purpose. The information provided in this documentation may include technical inaccuracies or typographical errors. Informatica could make improvements and/or changes in the products described in this documentation at any time without notice.

Table of Contents

List of Figures

Preface

Welcome to Informatica Data Quality, the latest-generation data quality management system from Informatica Corporation. Informatica Data Quality will empower your organization to solve its data quality problems and realize real, sustainable data quality improvements.

This guide will enable you to understand the principles of data quality and the functionality of Informatica Data Quality applications and plans. Its intended audience includes third-party software developers and systems administrators who are installing Data Quality within their IT infrastructure, business users who wish to use Data Quality and learn more about its operations, and PowerCenter users who may access Data Quality tools when working with the Informatica Data Quality Integration plug-in.

About This Book

The material in this guide is available in printed form from Informatica Corporation.

Document Conventions

This guide uses the following formatting conventions:

If you see…	It means…
italicized text	The word or set of words are especially emphasized.
boldfaced text	Emphasized subjects.
`italicized monospaced text`	This is the variable name for a value you enter as part of an operating system command. This is generic text that should be replaced with user-supplied values.
Note:	The following paragraph provides additional facts.
Tip:	The following paragraph provides suggested uses.
Warning:	The following paragraph notes situations where you can overwrite or corrupt data, unless you follow the specified procedure.
`monospaced text`	This is a code example.
`bold monospaced text`	This is an operating system command you enter from a prompt to run a task.

Other Informatica Resources

In addition to the product manuals, Informatica provides these other resources:

- Informatica Customer Portal
- Informatica web site
- Informatica Knowledge Base
- Informatica Global Customer Support

Visiting Informatica Customer Portal

As an Informatica customer, you can access the Informatica Customer Portal site at http://my.informatica.com. The site contains product information, user group information, newsletters, access to the Informatica customer support case management system (ATLAS), the Informatica Knowledge Base, Informatica Documentation Center, and access to the Informatica user community.

Visiting the Informatica Web Site

You can access the Informatica corporate web site at http://www.informatica.com. The site contains information about Informatica, its background, upcoming events, and sales offices. You will also find product and partner information. The services area of the site includes important information about technical support, training and education, and implementation services.

Visiting the Informatica Knowledge Base

As an Informatica customer, you can access the Informatica Knowledge Base at http://my.informatica.com. Use the Knowledge Base to search for documented solutions to known technical issues about Informatica products. You can also find answers to frequently asked questions, technical white papers, and technical tips.

Obtaining Customer Support

There are many ways to access Informatica Global Customer Support. You can contact a Customer Support Center through telephone, email, or the WebSupport Service.

Use the following email addresses to contact Informatica Global Customer Support:

- support@informatica.com for technical inquiries
- support_admin@informatica.com for general customer service requests

WebSupport requires a user name and password. You can request a user name and password at http://my.informatica.com.

Use the following telephone numbers to contact Informatica Global Customer Support:

North America / South America	Europe / Middle East / Africa	Asia / Australia
Informatica Corporation Headquarters 100 Cardinal Way Redwood City, California 94063 United States	**Informatica Software Ltd**. 6 Waltham Park Waltham Road, White Waltham Maidenhead, Berkshire SL6 3TN United Kingdom	**Informatica Business Solutions Pvt. Ltd.** Diamond District Tower B, 3rd Floor 150 Airport Road Bangalore 560 008 India
Toll Free 877 463 2435	**Toll Free** 00 800 4632 4357	**Toll Free** Australia: 1 800 151 830 Singapore: 001 800 4632 4357
Standard Rate United States: 650 385 5800	**Standard Rate** Belgium: +32 15 281 702 France: +33 1 41 38 92 26 Germany: +49 1805 702 702 Netherlands: +31 306 022 797 United Kingdom: +44 1628 511 445	**Standard Rate** India: +91 80 4112 5738

Chapter 1

Getting Started with Informatica Data Quality

This chapter contains information about the following topics:

Informatica Data Quality Product Suite

Welcome to Informatica Data Quality 8.5.

Informatica Data Quality is a suite of applications and components that you can integrate with Informatica PowerCenter to deliver enterprise-strength data quality capability in a wide range of scenarios.

The core components are:

- **Data Quality Workbench.** Use to design, test, and deploy data quality processes, called *plans*. Workbench allows you to test and execute plans as needed, enabling rapid data investigation and testing of data quality methodologies. You can also deploy plans, as well as associated data and reference files, to other Data Quality machines. Plans are stored in a Data Quality repository.

  Workbench provides access to fifty database-based, file-based, and algorithmic data quality components that you can use to build plans.
- **Data Quality Server.** Use to enable plan and file sharing and to run plans in a networked environment. Data Quality Server supports networking through service domains and communicates with Workbench over TCP/IP. Data Quality Server allows multiple users to collaborate on data projects, speeding up the development and implementation of data quality solutions.

You can install the following components alongside Workbench and Server.

- **Integration Plug-In**. Informatica plug-in enabling PowerCenter to run data quality plans for standardization, cleansing, and matching operations. The Integration plug-in is included in the Informatica Data Quality install fileset.
- **Free Reference Data**. Text-based dictionaries of common business and customer terms.
- **Subscription-Based Reference Data**. Databases, sourced from third parties, of deliverable postal addresses in a country or region.
- **Pre-Built Data Quality Plans**. Data quality plans built by Informatica. to perform out-of-the-box cleansing, standardization, and matching operations. Informatica provides free demonstration plans. You can purchase pre-built plans for commercial use.
- **Association Plug-In**. Informatica plug-in enabling PowerCenter to identify matching data records from multiple Integration transformations and associate these records together for data consolidation purposes.
- **Consolidation Plug-In**. Informatica plug-in enabling PowerCenter to compare the linked records sent as output from an Association transformation and to create a single master record from these records.

For information on installing and configuring these components, see the *Informatica Data Quality Installation Guide*.

Workbench, Server, and Integration

The core Informatica Data Quality applications are Workbench, Server, and the Data Quality Integration.

Data Quality Workbench is a user interface application in which users can design, test, and deploy data quality processes (or **plans**). Workbench allows users to test and execute plans on an ad-hoc basis. Users can also deploy plans, and associated data and reference files, to other Data Quality machines for further plan design and testing, or for deployment in project or live scenarios. Plans are stored in a Data Quality repository.

An important feature of Informatica Data Quality is its ease of use. Workbench enables users to derive optimal levels of value and usefulness from their data with minimum technical know-how, so that business users and IT professionals alike can develop and run plans.

Workbench provides access to fifty database-based, file-based, and algorithmic data quality components with which users build their plans. These components are presented as visual icons that can be dragged-and-dropped in the workspace and easily configured through context menus and property sheets.

Data Quality Server is an application that allows users to share plans and files and to deploy plans in a networked environment. Data Quality Server supports networking through service domains that allow Server-Workbench communication over TCP/IP.

Data Quality Server facilitates cross-team co-operation on data projects and enables increased efficiencies in the development and implementation of data quality solutions. It allows Data Quality users to configure server and client Workbench machines in a manner that maps to the architecture and resources of the organization. Plan designers can work together to share resources and avail of server processing power when defining plans, and the resultant plans can then be passed simply and safely to other domains in e.g. testing and production environments.

A Data Quality repository on the service domain grants network access to data quality plans.

- Data Quality permits plans to be deployed in **run-time** (batch/scheduled) processes on Windows and on several Unix platforms.

Data Quality Integration is a plug-in for Informatica PowerCenter. It allows PowerCenter users to connect to a Data Quality repository and to pass data quality plan instructions into a transformation. When PowerCenter runs a workflow containing the transformation, it sends the plan instructions to the Data Quality engine for execution and retrieves the data quality results back into the workflow.

Note: Data Quality Workbench and the Data Quality Integration are also components in the PowerCenter Data Cleanse and Match product offering.

Data Quality Engine and Data Quality Repository

Both Workbench and Server install with a Data Quality engine and a Data Quality repository. Users cannot create or edit plans with Server, although users can run a plan to any Data Quality engine independently of Workbench by runtime commands or from PowerCenter.

Users can apply parameter files, which modify plan operations, to runtime commands when running data quality plans to a Data Quality engine.

Every instance of Workbench has a local repository. Server installs a domain repository that stores and grants network access to data quality plans.

With Data Quality, users can deploy plans in runtime (batch or scheduled) processes on Windows and on Unix platforms.

Principles of Data Quality

In many data projects, data quality considerations are limited to checking that the metadata characteristics of source and target datasets are compatible. Although this is an important aspect of data quality, it is only one part of the picture. Data quality is concerned not only with the structure of the dataset, but with the usefulness and value of the information it contains, record by record and field by field.

Many organizations are unaware of the quantity of poor-quality data in their systems. Some organizations assume that their data is of adequate quality, although they have conducted no metrical or statistical analysis to support the assumption. Others know that their performance is hampered by poor-quality data, but they cannot measure the problem. If yours is a data-dependent organization, it is vital that you determine the quality of your data and take steps to improve quality where necessary. These two actions — data analysis and data enhancement — are the backbone of the data quality management process implemented through Informatica Data Quality.

The Informatica Data Quality software suite is at the leading edge of data quality management systems. It handles data from many industries, including product data, financial data, inventory data, and customer data. Its powerful analysis tools that let you evaluate your data according to key quality criteria and to report results clearly and graphically to business stakeholders and regulatory compliance officials. With Data Quality Workbench, you can write your own business rules with no knowledge of low-level coding. You can use Workbench to run data quality plans, to configure them for batch or scheduled execution, and to deploy them to third-party applications for real-time execution.

Note: Informatica also provides a Data Quality Integration plug-in for PowerCenter. This plug-in enables PowerCenter users to add data quality plan instructions to a PowerCenter transformation and to run the plan to the Data Quality engine from a PowerCenter session.

Data Quality Plans

The processes that you'll configure to perform data analysis and data enhancement operations on your data are called plans.

In Data Quality, a plan is a self-contained set of data analysis or data enhancement processes. A plan is composed of one or more of the following types of component:

- **Data sources** provide the input data for the plan.
- **Data sinks** collect the data output from the plan.
- **Operational components** perform the data analysis or data enhancement actions on the data they receive.

A plan must have at least one data source and data sink, and it can have any number of operational components. Some types of plan — for example, a plan writing data from one file or database to another — may not contain any operational components.

Six Measures of Data Quality

It is over-simple to speak of data that is “bad” or “wrong.” Even data of compromised quality is valuable, and the quality and value of your data can almost always be improved through an effective quality management system.

The quality of the data records in your datasets can be described according to six key criteria, and an effective quality management system will allow you to assess the quality of your data in areas such as these:

- **Completeness.** Concerned with missing data, that is, with fields in your dataset that have been left empty or whose default values have been left unchanged. (For example, a date field whose default setting of 01/01/1900 has not been edited.)
- **Conformity.** Concerned with data values of a similar type that have been entered in a confusing or unusable manner, e.g. numerical data that includes or omits a comma separator ($1,000 versus $1000).
- **Consistency.** Concerned with the occurrence of disparate types of data record in a dataset created for a single data type, e.g. the combination of personal and business information in a dataset intended for business data only.
- **Integrity.** Concerned with the recognition of meaningful associations between records in a dataset. For example, a dataset may contain records for two or more individuals in a household but provide no means for the organization to recognize or use this information.
- **Duplication.** Concerned with data records that duplicate one another’s information, that is, with identifying redundant records in the data set.
- **Accuracy.** Concerned with the general accuracy of the data in a dataset. It is typically verified by comparing the dataset with a reliable reference source, for example, a dictionary file containing product reference data.

(The list above is not absolute; the characteristics above are sometimes described with other terminology, such as redundancy or timeliness.)

The **accuracy** factor differs from the other five factors in the following respect: whereas (for example) a pair of duplicate records may be visible to the naked eye, it can be very difficult to tell simply by “eye-balling” if a given data record is inaccurate. Data Quality’s capabilities include sophisticated tools for identifying and resolving cases of data inaccuracy.

The data quality issues above relate not simply to poor-quality data, but also to data whose value is not being maximized. For example, duplicate householder information may not require any modification per se — but it may indicate potentially profitable or cost-saving relationships among customers or product lines that your organization is not exploiting.

Every organization’s data needs are different, and the prevalence and relative priority of data quality issues will differ from one organization and one project to the next. These six characteristics represent a critically useful method for measuring data quality and resolving the issues that prevent your organization from maximizing the potential of its data.

A Modular Approach to Data Quality

Informatica Data Quality's operational components provide a range of data analysis and enhancement functionality that span all aspects of a data project.

Data Quality organizes its plans and projects in a modular manner. By default, a new data quality project in Workbench will contain sub-folders for four modules: profiling, standardization, matching, and consolidation.

1. **Data profiling** performs a SWOT analysis on your data — identifying its strengths and weaknesses and the opportunities and threats it may present for your project and enterprise. Key reporting tools in the profiling stage are **scorecards**, which present clear statistical and graphical information on the quality of your data and which can trigger alarm messages when statistical thresholds are met or breached.
2. **Data standardization** will both standardize the form of your data and validate and enhance the data by comparing values to a range of filters and reference dictionaries. Standardization operations can involve elementary "noise removal" or multiple bespoke business rules; in every case, Data Quality's graphical components make defining the necessary operations simple enough for all user levels.
3. **Data matching** identifies equivalent, duplicate, or related data records within a dataset (files or database tables) or between datasets. Correspondingly it can identify inaccurate data by comparing data with reference dictionaries. Matching plans are also used to prepare for the consolidation of enhanced datasets.
    - Data Quality employs powerful matching algorithms that, like all Data Quality functionality, operate "under the hood" of the software, so that the user can define and tune matching processes quickly and easily on the desktop. Matching results can be weighted within the matching process so that the resultant data outputs can reflect business priorities.
4. **Data consolidation** is often the final step in a data quality cycle: it enables the enterprise to create, for example, the new datasets that will be used in a commercial environments, or to create the master dataset for a data migration project.

Note: You can build a data quality project in the default modular sequence, as described here, although you have complete flexibility to design your project in any manner inside or outside a modular environment.

Note also that folders and plans are arranged in the Project Manager alphabetically. The folder or plan order beneath in the Project Manager does not necessarily indicate the order in which plans should be executed. You can apply your own naming conventions to projects and plans to enhance the clarity of your projects in Project Manager.

Data Quality's flexibility of use means that the enterprise can define a custom-tailored data quality process that focuses on one or all of these modules. Taken together, analysis, standardization, matching, and consolidation can represent a *data quality life-cycle* that delivers an end-to-end data quality methodology for the enterprise and that can also slot into the planning of other data projects and ongoing data quality maintenance.

Chapter 2

Working with Plans in Data Quality

This chapter contains the following sections:

Overview

This chapter will help you learn more about Data Quality Workbench and the operations of data quality plans.

Although the Workbench user interface is straightforward to use, the data quality processes, or *plans*, built in Workbench can be as simple or complex as you decide. Also, a key characteristic of a data quality plan is the inter-dependency of the configured elements within it.

Bear in mind that the goal of a data quality plan is not always to achieve the highest data quality readings — that is, to find zero duplicates, or deliver a 100% data accuracy score. At the end of a data quality project, these may be realistic objectives. However, the goals of most analysis plans are twofold: to achieve the most faithful representation of the state of the dataset, and to highlight the outstanding data quality characteristics of interest to the business.

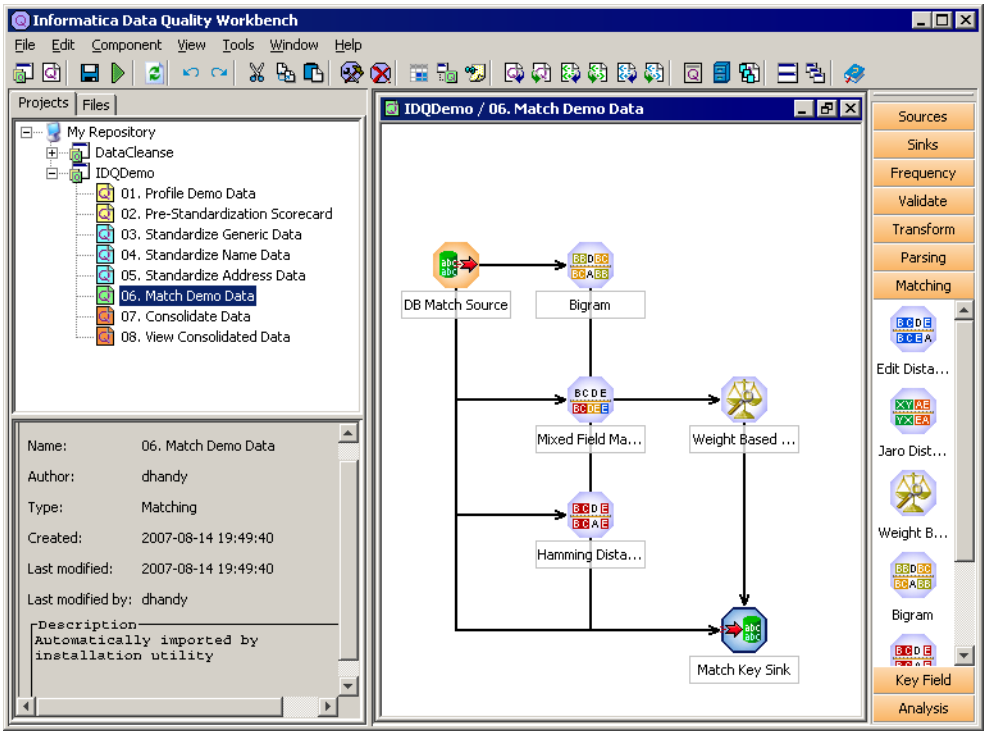
While a plan designer may tune a plan so that it captures the data quality characteristics more accurately, it is also possible to tune a plan's settings so that the data quality results appear to improve beyond the reality of the data. The ability to tune a plan properly comes with training and experience in Data Quality Workbench.

Note: The full range of functionality available in Data Quality Workbench is described in the Informatica Data Quality User Guide.

Data Quality Workbench

By default, Informatica Data Quality installs to C:\Program Files\Informatica Data Quality and adds a program group to the Windows Start menu. Within this program group is the executable shortcut to Workbench. To start Workbench, select this shortcut (e.g. Start > Programs > Informatica Data Quality > Informatica Data Quality Workbench).

Figure 2-1. Informatica Data Quality Workbench User Interface



Name: 06. Match Demo Data
Author: dhandy
Type: Matching
Created: 2007-08-14 19:49:40
Last modified: 2007-08-14 19:49:40
Last modified by: dhandy

Description
Automatically imported by installation utility

The Workbench user interface shows tabs to the left for the Project Manager and File Manager and a workspace to the right in which you'll design plans. The fifty data components that you can use in a plan are shown on a dockable panel on the right-hand side.

There are three basic types of component:

Data Sources, which define the data inputs for the plan. Sources can connect to files or database tables.

Operational Components, which perform data analysis or data transformation operations on the data.

Many operational components make use of reference dictionaries when analyzing or transforming data; dictionaries are explained below.

Data Sinks, which define the data outputs that will be written to file or to the database.

A plan must have at least one data source and data sink. A typical plan will also contain multiple operational components.

The component palette displays the components available as octagonal icons that can be added to the plan workspace by mouse-click. The icons are according to the type of operation the component performs, under headings such as Analysis, Parsing, and Matching.

Data sources, operational components, and sinks act as links in a chain to pass data from start to finish within a plan: a component is not active within a plan unless it has been configured to accept inputs from and provide outputs to other components in this chain. Active components are connected to one another by arrowed lines, showing the approximate flow of data along the chain.

Copying Plans in the Data Quality Repository

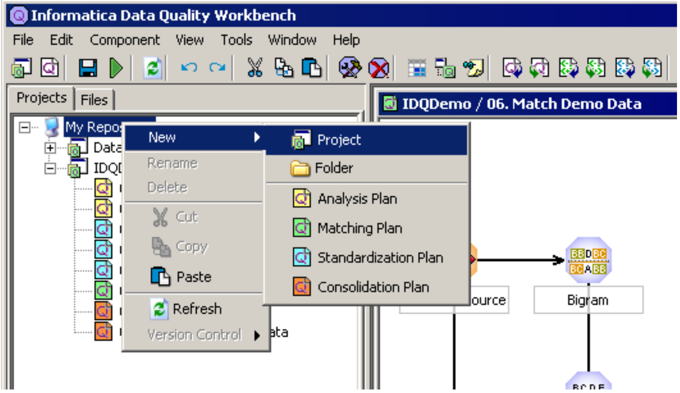
If you want to explore the functionality of a data quality plan, it is good practice to copy the plan to a new location in the Data Quality repository so that the original plan cannot be damaged. This section describes how to copy the installed plans within the repository for testing/learning purposes, and steps you can take to modify them in Data Quality Workbench.

The main steps are as follows:

- Create a new project in Data Quality Workbench
- Copy or import more plans to this project
- Rename the project and imported plan(s)
- Change the data source associated with the plan

Create a new project in Data Quality Workbench by right-clicking My Repository under the Projects tab and selecting New > Project from the context menu.

Figure 2-2. Creating a New Project in Data Quality Workbench



You can rename or delete the four folders default-created beneath this project.

Copy or import one or more plans to this project. You can make a copy of a plan within the Project Manager by highlighting the plan name and typing Ctrl+C. You can then paste the plan to the project you have just created by right-clicking the project name and selecting Paste from the context menu.

You can also import a plan file in PLN form or XML format from the file system. This may be appropriate if you have received pre-built plans from Informatica, in which case backup copies of the plans may be installed to your file system.

- For more information on importing plans, see the Informatica Data Quality User Guide.

Rename the project and imported plan(s). You should give names to your new projects and plans that clearly distinguish them from the installed data quality project.

To rename a project or plan, right-click it in the project structure and select Rename from the context menu. The name is highlighted and a flashing cursor indicates that it is editable: replace the old name with a new name and click elsewhere onscreen to commit your changes.

Change the data source associated with the plan. This enables you to define a flat file as input for the plan, so that you do not need to provide real-time inputs each time the plan is run. This is the most complex of these steps; it is described in detail below.

Configuring Components

There are many types of component, and their configuration dialogs are organized in many different ways. However, most configuration dialog boxes display tabs that provide options for defining data inputs, setting parameters for data analysis or transformation, and defining data outputs.

To open the configuration dialog box for a component, double-click its icon on the plan workspace.

Changing the Data Source for a Plan

Note: The procedures below can be performed quickly and easily by experienced Data Quality users. However, they constitute major changes to plan structure, and they must not be performed on plans in commercial or live project scenarios without prior consultation with Informatica personnel.

Types of Data Source

Data quality plans can be defined with several types of source component, and the type of source applied to a plan depends on the purpose for which the plan will be used. For example, plans that have been designed for use in PowerCenter transformations make use of source and sink components that are realtime-processing enabled.

For testing purposes, you can replace a Realtime Source or Sink with a CSV Source or SInk that connects to a delimited flat file. (Plans that use realtime-enabled sources or sinks can only process one record at a time when run directly within Workbench.)

To replace a Realtime Source component with a file-based component:

1. Click the Realtime Source in the plan workspace and press Delete.
2. Click a CSV Source in the component palette and add it to the workspace. If the plan already contains a CSV Source or a CSV Match Source, skip this step.

   Tip: Place the CSV Source in the workspace so that it above or to the left of the other components in the plan.
3. Right-click the CSV Source icon and select Configure from the context menu that appears.
4. When the source configuration dialog box opens, click the Select button and navigate to a delimited file.
5. Review the other options in the configuration dialog: confirm the field delimiter for the file values, and indicate if the first line of the file contains header information.

   Note: make sure that the Enable Realtime option is cleared.
6. Click OK to add the file to the plan.

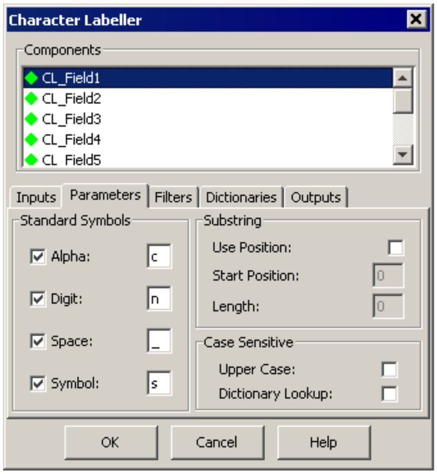
You can now run the plan using the local file as the input dataset.

Note: To run a plan, click the Run Plan button on the Workbench toolbar. For more information on running plans, see “Running Plans: Local and Remote Execution” on page 8.

Example 1: Configuring a Character Labeller

For example, the Character Labeller component, which identifies the types of characters that comprise a data string, has the following configuration dialog:

Figure 2-3. Character Labeller configuration dialog box — Parameters tab



CL_Field1
CL_Field2
CL_Field3
CL_Field4
CL_Field5

☑ Alpha: c
☑ Digit: n
☑ Space: _
☑ Symbol: s

Use Position: □
Start Position: 0
Length: 0

Upper Case: □
Dictionary Lookup: □

This settings in this dialog box can illustrate the implications of changing component settings within a plan.

The Standard Symbols group box provides settings that determine the types of character that will be labelled by this component and how they will be labelled. For example, you can pass a telephone number record through this component and define a new output field that will map its characters by type, such that the number 555-555-1000 generates a new value *nnn-nnn-nnnn*.

The manner in which a number like this, or non-numeric data, is labelled depends on the symbol settings. For example, you can check the Symbol field and enter a label for non-alphanumeric values. For example, the number 555-555-1000 may appear as *nnnxnnnxnnnn* depending on the symbol value you provide here.

If your plan is concerned simply with labelling the characters by type, then changing the symbol value to x will not meaningfully affect the results of the plan. However, if the component output is read by another component — or the plan outputs read by another plan — then such a change may have a serious impact.

Example 2: Configuring a Rule Based Analyzer

Consider for example a plan designed to identify potential zip codes as five-digit numeric strings. The plan may include a Character Labeller component that labels data characters as above and a Rule Based Analyzer component that analyzes these labels and zip codes writes conformant label patterns to the output file.

The Rule Based Analyzer lets you write business rules using If-Then-Else logic and applies these rules to the data elements you specify. It provides a wizard for users who are unfamiliar with this logic.

The Rule Based Analyzer may contain the following rule:

```
IF Input1 = nnnnn

OR Input1 = nnnnn-nnnn

THEN Output1: = Output1

ELSE Output1: = "Bad Zipcode"
```

(Output in this example refers to the original data fields that were labelled by the Character Labeller.)

If the Character Labeller configuration was changed so that symbols were written as any character other than a hyphen — e.g. nnnnnsnnnn — and the associated rule was not changed, then the plan will not produce meaningful results.

The Rule can be changed by adding a line to this effect, i.e.

```
IF Input1 = nnnnn

OR Input1 = nnnnn-nnnn

OR Input1 = nnnnnsnnnn

THEN Output1: = Output1

ELSE Output1: = "Bad Zipcode"
```

Example 3: Configuring Matching Components

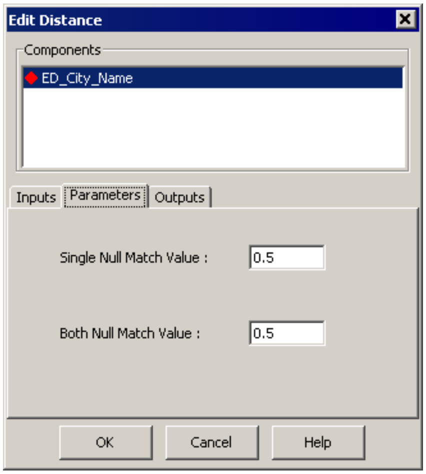
Data Quality's matching components allow you to assess the levels of similarity or difference between data values. The components work by applying algorithms to values passed a pair at a time from selected columns in the plan dataset; each pair is assigned a match score between zero (no similarity) and 1 (perfect match) indicating the similarity between them according to the parameters of the algorithm applied.

Edit Distance

For example, the Edit Distance component derives a match score for two data strings by calculating the minimum "cost" of transforming one string into another by inserting, deleting, or replacing characters. The dissimilarity between the strings **Washington** and **Washingtin** can be remedied by editing a single character, so the match score is 0.9 (the score is penalized by ten per cent as one of the ten characters must be edited).

When one (or each) input string is null, the component applies a set score that can be tuned by the user. The default scores are 0.5. You can change these scores to reflect the severity of the presence of null fields in the selected data.

Figure 2-4. Edit Distance, Parameters tab (Null Settings)



Single Null Match Value : 0.5

Both Null Match Value : 0.5

Example 4: Configuring a Weight Based Analyzer

Matching components, including the Edit Distance, can feed their output scores to the Weight Based Analyzer, a component that allows the plan to calculate an aggregate, weighted score across all matching scores for a pair of records.

Consider a personnel records dataset: in such a dataset, similar surnames are more likely to indicate matches than similar forenames. If your dataset has discrete surname and forename fields, you can increase the weight of the surname score in proportion to the increased priority of that field. (Note that other factors, such as the type of matching algorithm applied, also apply here.)

Changing the weights associated with matching outputs, and observing the changes in plan results when you execute the plan, is a good way to learn how components interact and determine the plan results.

The Role of Dictionaries

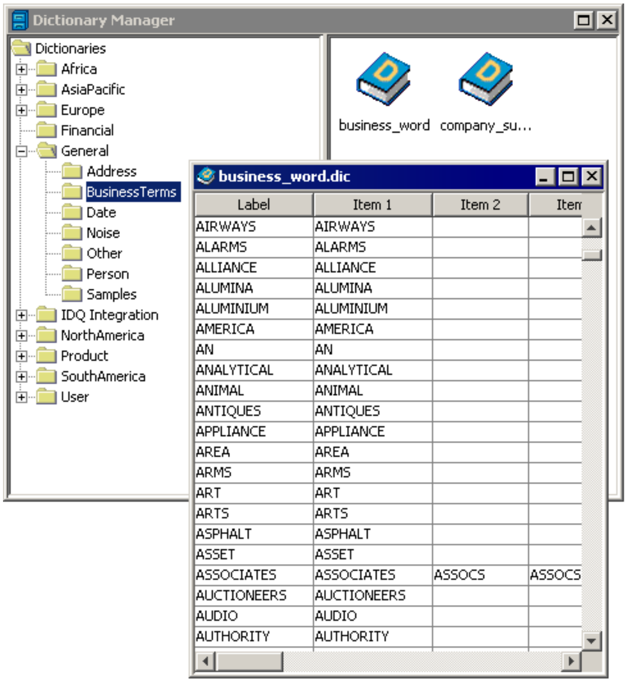
Plans can make use of reference dictionaries to identify, repair, or remove inaccurate or duplicate data values. Informatica Data Quality plans can make use of three types of reference data.

- **Standard dictionary files**. These files are installed with Informatica Data Quality and can be used by several types of component in Workbench.

  All dictionaries installed with Data Quality are text dictionaries. These are plain-text files saved in .DIC file format. They can be created and edited manually.
- **Database dictionaries**. Informatica Data Quality users with database expertise can create and specify dictionaries that are linked to database tables, and that thus can be updated dynamically when the underlying data is updated.
- **Third-party reference data**. These data files are provided by third-parties and are provided by Informatica customers as premium product options. The reference data provided by third-party vendors is typically in database format.

Standard dictionaries cover standard business data types. including zip code and postcode formats, personal salutations, common forenames and last names, and business terms. These dictionaries are accessible through the Dictionary Manager in the Workbench user interface.

Figure 2-5. Dictionary Manager and business_word Dictionary sample.



Label	Item 1	Item 2	Iten
AIRWAYS	AIRWAYS		
ALARMS	ALARMS		
ALLIANCE	ALLIANCE		
ALUMINA	ALUMINA		
ALUMINIUM	ALUMINIUM		
AMERICA	AMERICA		
AN	AN		
ANALYTICAL	ANALYTICAL		
ANIMAL	ANIMAL		
ANTIQUES	ANTIQUES		
APPLIANCE	APPLIANCE		
AREA	AREA		
ARMS	ARMS		
ART	ART		
ARTS	ARTS		
ASPHALT	ASPHALT		
ASSET	ASSET		
ASSOCIATES	ASSOCIATES	ASSOCS	ASSOCS
AUCTIONEERS	AUCTIONEERS		
AUDIO	AUDIO		
AUTHORITY	AUTHORITY		

Third-party dictionaries originate from postal address database companies that provide authoritative address validation capability to Data Quality's validation components. They are provided with premium product options from Informatica. These dictionary databases are not accessible through the Dictionary Manager.

For more information, see the Informatica Data Quality User Guide.

Note: In general, you should not change the order in which dictionaries are listed in a component's configuration dialog box. Doing so will change the order in which Data Quality applies the dictionaries to the plan data, and this will affect the plan results.

Chapter 3

Working with the Demonstration Plans

This chapter contains information about the following topics:

Overview

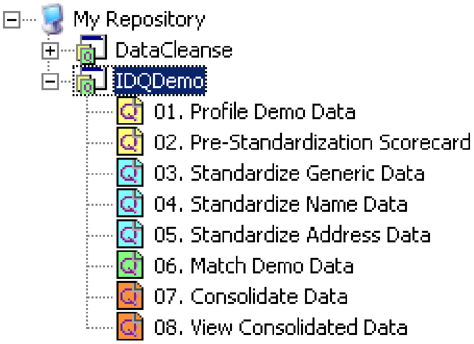
This chapter demonstrates plan building in action by following a simple data quality project from start to finish. The project operations take place in Data Quality Workbench. They do not involve any Workbench-Server functionality or any interaction with PowerCenter.

IDQDemo Plans

The Data Quality install process installs a project named IDQDemo to the Data Quality repository and also writes a copy of the plan files to the Informatica Data Quality folder structure. The high-level objective of this sample project is to profile and cleanse a dataset of business-to-business records.

Figure 3-1 shows the layout of the installed plans in Workbench.

Figure 3-1. Sample Plan List in Workbench User Interface1



My Repository
- DataCleanse
- IDQDemo
  - 01. Profile Demo Data
  - 02. Pre-Standardization Scorecard
  - 03. Standardize Generic Data
  - 04. Standardize Name Data
  - 05. Standardize Address Data
  - 06. Match Demo Data
  - 07. Consolidate Data
  - 08. View Consolidated Data

The project analyzes, cleanses, and standardizes a United States business-to-business dataset of approximately 1,200 records. This dataset is installed in the IDQDemo\Sources folder with the filename IDQDemo.csv. The dataset comprises the following columns:

- Customer Number
- Contact Name
- Company Name
- Address 1
- Address 2
- Address 3
- Address 4
- Zipcode
- ISO Country Code
- Currency
- Customer Turnover

The IDQDemo folders also include SSR report files that permits project progress to be measured at key stages.

Each plan in the project analyzes or enhances the dataset in ways that contribute to the creation of a clean dataset at the project's end. The plan explanations that follow look at the sources, sinks, and operational components used and explain how each one contributes to the end result. You can read this chapter while reviewing the plans onscreen: as you proceed through the plans, bear in mind that the output of a given component may be used by more than one plan in the project.

Plan 01: Profile Demo Data

This plan analyzes the IDQDemo.csv source data and generates a Informatica Report file as output.

The components have been configured to assess data quality in the following ways:

- The Merge component has been configured to merge four fields from IDQDemo.csv (Address1—Address4) into a single column named Merged Address. It also merges the ISO Country Code and Currency columns into a Merged CountryCode and Currency column that will be analyzed by the Token Labeller.
- The Merged Address column is used as input by the Context Parser, which applies a reference dictionary of city and town names to the merged data and writes the output to a new column named CP City or Town. Any American city names found in Merged Address are written to this new column.
- The Rule Based Analyzer has been configured to apply business rules to the Customer_Number and CP City or Town fields.

  The Test Completeness of Cust_No rule comprises a simple IF statement. If a value is present in a Customer_Number field, the rule writes Complete Customer_Number to a corresponding field in a new column named Customer Number Completeness. If not, the rule writes Incomplete Customer_Number in the relevant field.

  The Test Completeness of Town rule profiles the completeness of the CP City or Town column through a similar IF statement. An empty field in the CP City or Town column indicates that the underlying address data lacks recognizable city/town information. Conversely, any name in a CP City or Town field has already been verified by the Context Parser (see above); the rule writes such names to a new column named City_or_Town Completeness.
- The Character Labeller analyzes the conformity of the Customer_Number field. (Brief analysis of IDQDemo.csv indicates that all customer account numbers begin with the digits 159, 191, or 101: type F6 to open the Source Viewer and examine a subset of the data.)
- On the Filters tab of the component's configuration dialog box, filters have been defined to identify the account numbers that begin with 159, 191, and 101. All numbers so identified are written to a new column named Customer Number Conformity as specified on the component's Outputs tab.
- The Token Labeller applies reference dictionaries to analyze conformity in the Contact Name, Company Name, Zipcode, Currency, ISO Country Code, and Merged CountryCode and Currency columns.

  Contact and company name data are analyzed against dictionaries of name prefixes, first names, and surnames, and against dictionaries of US company names. Similarly, zip code and currency data are analyzed against dictionaries of valid zip codes and currency names respectively.

Note that a “Do Not Use” dictionary is also applied to contact name and company name data: this dictionary contains several common phrases and indicators that can flag a record as unusable.

The Token Labeller also cross-checks that the currency applied to an account is compatible with the country in which the account holder is resident. It does so by comparing the merged country code and currency entries with a dictionary that contains valid country name and currency combinations. (See the Token Labeller - Country Code and Currency Consistency instance configured in this component.)

- The Range Counter counts the number of records that fall into user-defined categories based on data from a specified column. In this case, the Range Counter is configured to count the number of accounts with debit or credit balances of varying sizes. The designer of this plan is interested in excessively high or low balances, that is, balances that may indicate bad data entries — for example, balances of minus $50,000 or lower, and balances of $100,000 and higher. Thus the plan designer defines several numeric ranges on the Parameters tab of the configuration dialog in order to identify the values across the spectrum of account turnover.
- The use of a Report Sink means that the plan will generate a Informatica report file, tabulating the quantities of records in the columns selected by the frequency component.
- To configure the Report Sink, the plan designer has added a Count component and selected, from the available inputs, the required columns.

Note: The a Report Sink component only accepts data columns that have been identified by a frequency component, such as a Range Counter or Count. Also, the Informatica report allows you to drill down through its data results to the individual records underlying the results. Informatica reports are saved in the proprietary .SSR file format.

Plan 02: Pre-Standardization Scorecard

This plan uses a similar component set to plan 01 above. However, its purpose and outputs are different.

Plan 01 and 02 both generate a data quality profile for the condition of the dataset in advance of the enhancement operations defined in later plans. However, plan 02 writes its output data via a report sink to a CSV file, whereas plan 01 presents its output in Similarity's proprietary SSR format. Moreover, plan 02 facilitates the production of a scorecard based on its output.

- A scorecard displays the results of the data quality process in a bar chart format. Scorecards are built by associating your plan output with a Microsoft Excel template pre-built by Informatica.
- Scorecarding provides precise quantification of the data quality in the dataset and enables the comparison of the quality metrics with the targets set for each area. It acts as a diagnostic tool at the start of the quality process, and at later stages enables you to evaluate your process methodology and identify any data areas that require further treatment.

The plan components are configured identically to those in plan 01, with the following exceptions:

- The Rule Based Analyzer implements four rules in this plan. As well as testing the completeness of the customer number and town data, it writes the current date to a new column and tests the validity of the customer turnover values.

  This component effectively replaces the Range Counter component present in plan 01. Whereas the Range Counter grouped customer turnover values according to their numerical ranges, a new rule in the Rule Based Analyzer recognizes the values as valid or invalid based on whether they fall inside or outside a user-defined threshold.
- A second Token Labeller has been added to profile the two output columns created by the first Token Labeller in the plan. It profiles the Customer Number Conformity and Contact Name Conformity columns by applying reference dictionaries that explicitly label each field as valid or invalid. Like the earlier Token Labeller, it analyzes the company name field from the source dataset in order to generate valid/invalid results for each name. These profiles are generated for scorecarding purposes.

When you open the plan's CSV file output in Microsoft Excel, you'll note that the data column selected in the Count component are shown with raw values (counting the number of values in each column) and also percentages — indicating the proportion of the underlying data column that satisfies the relevant data quality criterion.

Plan 03: Standardize Generic Data

This plan cleanses the non-name and non-address data in IDQDemo.csv. In this scenario, data cleansing means (i) correcting data errors and (ii) creating a new dataset from which unwanted records have been removed – specifically, records with account turnover figures outside a desired range and records that are flagged Do Not Use.

A key aspect of cleansing the data is standardization. To facilitate the creation of a clean dataset, the source data values are standardized, so that the plan's business rules can reliably identify cleansed data and the plan's data sinks can write the cleansed data (and the exceptions) to new files. As part of this process, data records are also cleansed of extraneous punctuation, or *noise*.

The plan generates three sets of outputs:

- Database table containing the cleansed data records
- CSV file containing the cleansed data values
- CSV file containing data exceptions (i.e. source data records not included in the cleansed dataset)

The components have been configured as follows:

- The Merge component merges the four address columns in the source dataset into a single column named Merged Address. This will enable other components to identify any records whose address fields contain a Do Not Use or equivalent entry.
- The Token Parser takes the merged address output from the Merge component and applies a "Do Not Use" dictionary — that is, a dictionary composed of several terms indicating that the record should be ignored, such as Do Not Use, Bad Data, Ignore, Delete. Any records containing values found in the dictionary will be parsed to a new output column.

  Similarly, the component applies the Do Not Use dictionary to the Company Name column.

  These outputs will be used by a Rule Based Analyzer to flag exceptions later in the plan.
- The Search Replace component removes noise from data fields. You'll see that an instance of the component has been configured to remove noise from Customer Number column values, deleting non-essential characters and replacing erroneous occurrences of upper-case letters O and I with 0 (zero) and 1 respectively.

  The component performs similar operations on the Merged Address, ISO Country Code, and Company Name columns.
- A Character Labeller is configured to profile the cleansed customer number output from the Search Replace component. No filters or dictionaries are applied: the purpose of the profile is to describe the format of each customer number. As all correctly-formatted numbers have eight digits and no other characters, the plan designer can write a rule, based on the Character Labeller output, stating that valid customer numbers are exclusively of the format nnnnnnnn. (See the Rule Based Analyzer below.)

- The Word Manager in this plan has a single input — the new SR ISO Country Code column generated by the Search Replace component. The component applies a country code dictionary to the input in order to standardize the country code data, so that each country is represented by a single code. The component output is used by the Rule Based Analyzer later in the plan.
- The Context Parser derives an ISO country code for the Merged Address data values by identifying town and city names in the address data and returning the correct ISO country code for each one. It does so by applying the US Cities dictionary, which contains several thousand town and city names and their correct ISO country codes.

  The Context Parser also parses the first word from the Final Company Name column. The output is later used by the Soundex component. The Soundex output is written to the database table for use as a group key in a match plan later in the process.
- A second Context Parser uses the final_iso_country_code output from a Rule Based Analyzer to derive the correct currency for each record; another Rule Based Analyzer then uses the derived currency to create a final currency column.
- The Token Labeller applies an ISO Country Codes dictionary to confirm the accuracy of the Word Manager output (i.e. the WM ISO Country Code column). It likewise applies a dictionary to check the accuracy of the Currency column in the source data. Its outputs are country code and currency profiles that will be used (in conjunction with the Context Parser output) by two Rule Based Analyzers later in the plan.
- The plan contains three Rule Based Analyzers.

  One standardizes the country code values using outputs from the Word Manager, Context Parser, and Token Labeller.

  Another standardizes currency values based on Context Parser and Token Labeller outputs.

  The third Rule Based Analyzer flags record exceptions; these will be written to an exceptions file when the plan executes. Records are flagged as exceptions if they fail to meet one of the following criteria:

  - Customer numbers must be composed of eight digits.
  - Company name or address fields must not contain Do Not Use labels.
  - ISO country codes and currencies must be valid.
  - Customer turnover values must be numeric and fall within the range -50,000 to 100,000.
- The plan has three sink components:

  The DDL in the Before tab of the DB Sink demonstrates how to drop and create a table in the Data Quality staging area. Likewise, the DDL in the After tab demonstrates how to apply indexing in the staging area. The During tab demonstrates the configuration required to insert values into a database table. The configuration of the Rule Condition dialog demonstrates how rows can be filtered so that exceptions are not loaded into the database table.

The plan has two CSV Sink components. One contains the standardized generic data that constitutes the principal output for the plan; the other contains the records flagged as exceptions.

Plan 04: Standardize Name Data

This plan cleanses the name data in the dataset that originated with IDQDemo.csv and that has been loaded into the Data Quality staging area. The plan writes its output to a database table and a CSV file by means of by a DB Sink and CSV Sink respectively.

The main purpose of this plan is to generate a usable, cleansed list of contact names. The main steps involve (i) identifying a contact name for each account, (ii) identifying the gender of each contact name, (iii) adding a gender-appropriate prefix to each contact name, and (iv) creating a generic salutation ("Customer") for all records for which gender could not be ascertained.

- The plan uses a Search Replace component to remove noise (in this case, extraneous spaces) from the contact_name field.
- The output from the Search Replace component is used by the Token Labeller to create a profile of the name data. It does so in two ways.

  First, the Token Labeller standardizes the punctuation symbols present in the input data; that is, it reformats symbols such as & and / (found in records with multiple contact names) to the text AND/OR. This will make it easier for the Profile Standardizer to recognize multiple account names.

  Next it identifies the tokens in the contact name column by applying a series of dictionaries to it.
- The Token Labeller output is used by the Profile Standardizer to parse first names and surnames into individual columns. Note that the Profile Standardizer allows you to disregard those profiles that do not conform to the expected or required shape of the contact name data. In this plan, the component has been configured to write recognized name prefixes, first names, and surnames to new columns. You can review the Overflow column from the Profile Standardizer by assigning it to a data sink and running the plan: you'll see that the overflow data concerns records that are flagged as unusable or for test purposes only.
- The Token Parser component is used to derive gender data from the firstname columns created in the Profile Standardizer. It applies a US Firstnames dictionary to the data in both columns and creates output columns with the correct gender appended to each name recognized by the dictionary.
- This data is in turn used by the Rule Based Analyzer to append an appropriate name prefix to the firstname columns. The IF statements for each record are shown below:

  ```
  IF (Input1 = "") AND (Input2 = "F") THEN

  Output1 := "Ms"

  ELSEIF (Input1 = "") AND (Input2 = "M") THEN

  Output1 := "Mr"

  ELSE

  Output1 := Input1

  ENDIF
  ```

- When the gender of each contact name has been identified, the Merge component combines the cleansed surname data with the appropriate name prefix in a single, new field, called Merged Salutation. Note that the plan identifies a single contact name for each record — that is, a single point of contact will be identified for each account.
- The second Rule Based Analyzer in the plan generates a final list of contacts by checking that the input columns for the merged salutation contain valid prefix and surname values. The Rule Based Analyzer creates a new cleansed salutation column and populates it with the validated merged salutation records; it also creates a column for records not validated by this rule that identifies these contacts as "Customer."
- The To Upper is the final operational component in the plan: it ensures that the final contact details columns are written in title case, e.g. Mr Smith.
- The plan writes its output to a DB Sink and CSV Sink.

Plan 05: Standardize Address Data

Whereas plan 04 cleanses name data, this plan cleanses the address data in the same dataset. The plan writes its output to three sink components: a DB Sink, CSV Sink, and Report Sink. The main purpose of this plan is to generate a usable, cleansed list of address values.

- The Merge component combines the four address columns into a single Merged address column.
- Next, a Search Replace component removes noise from the merged address data and generates a new SR Merged Address column as output.
- A Token Parser splits the zip code data from the SR Merged Address.

  Zip code values are parsed into a new TP_Zipcode from Merged Address column. The remaining address tokens are written to an overflow column, TP_Merged Address Overflow.
- A Word Manager standardizes the TP_Merged Address Overflow by applying three dictionaries to standardize address terms: for directional terms (e.g. North, South), for address suffixes (e.g. Avenue, Road), and for numerals spelled as text (e.g. Twenty One, Twenty-First).
- The Token Labeller creates a profile of the standardized merged address column (i.e. the Word Manager output). This profile will be used by the Profile Standardizer to parse address tokens to new address qualifier, building number, street, city, and state columns. Bear in mind that the parsing operations in the Profile Standardizer are user-defined.
- Next, a Word Manager standardizes the address qualifier field to convert instances of ste to suite and fl to floor. (Note that a Search Replace component could also have been used in this case, as the number of converted terms is low.)

The components that standardize the zip code data are as follows:

- A Character Labeller is used to identify the non-numerical characters in the zip code source data by labelling alphabetic characters, spaces, and symbols. All numerical data is left unchanged.
- A Search Replace component can then remove all alphabetic characters, spaces, and symbols from the zip codes. This component is also configured to remove null or placeholder data strings such as 99999. The output from this component is a partially-cleansed (i.e.all-numeric) zip code column called SR Zipcode.
- A second Character Labeller profiles the SR Zipcode column and creates another character profile for the zip code data; this profile is called Zipcode Profile2.
- The Rule Based Analyzer standardizes the zip code data and creates five-digit and zip-plus-four outputs. It applies the following four rules:
  - Four-digit zip codes found in Zipcode Profile2 are prefixed with a leading zero to create a five-digit code. (Many spreadsheet applications automatically strip a leading zero from a number, rendering e.g. zip code 02110 (Boston) as 2110.)
  - Nine-digit zip codes found in Zipcode Profile2 are split into five-digit and a zip-plus-four codes.

  - Eight-digit zip codes found in Zipcode Profile2 are prefixed with a leading zero and split into five-digit and a zip-plus-four codes.
  - Where an SR Zipcode field is empty and a five-digit code is present in the TP_Zipcode from Merged Address field, the latter value is used to populate the five-digit output field.
- The plan data is written to a DB Sink and CSV Sink. Also, note that the merged address profile is selected by the Count component for inclusion in output from a Report Sink. Note that this Report Sink is configured with a Min Count figure of 67. (The Min Count parameter specifies the minimum number of times a given value must occur in a column in order to be listed in the report output.)

Plan 06: Match Demo Data

This plan performs matching operations on the dataset and generates a HTML match report illustrating the data matches identified. The match criteria are determined by the four matching components shown below, and the matches are listed in a HTML report. The output can also be saved as a CSV file.

The DB Match Source provides the source data for the plan. Note that the DB Match Source creates two identical sets of columns from the data in the staging area, so that, for example, the Company Name field is rendered as Company Name_1 and Company Name_2.

- The Bigram component analyzes the Company Name and Address1 fields. This component looks for matches based on the presence of common consecutive characters in the two fields.
- The Hamming Distance component analyzes the zip code fields. It derives a match score for the data in a pair of fields by calculating the number of positions in which characters differ between them.
- The Mixed Field Matcher component matches several address fields against one another in order to identify common data values across different but related columns — for example, where a correct valid city name has been entered in an inappropriate field. The component generates a single score for each match identified.
- Note the several options available on the component's Parameters tab: Informatica recommends retaining the default settings on this tab.
- The Weight Based Analyzer takes the output scores from the plan's other matching components, including the Mixed Field Matcher, and produces an aggregate score for each record based on user-defined weights assigned to the scores from the other components.
- The plan output is created by the DB Match Sink. Note that you must define database connection settings for this component in the same manner as for the DB Match Source.

Plan 07: Consolidate Data

This plan consolidates the cleansed data in the Data Quality staging area. It has been configured to (i) flag the first record in each match cluster as the master record and (ii) calculate a new customer turnover value by aggregating the customer turnover values of each record in the cluster. The plan has two components: a DB Source and DB Sink.

The components are configured by working with the database tables stored in the repository:

- In the DB Source component, the SQL statement in the Before tab flags the first record in each match cluster as the master record. The During tab selects all duplicate records from the 'cleansed_Data Quality_data' table (i.e. all non-master records).

  In the DB Sink, an Update statement in the During tab calculates a 'new_turnover' value for each master record by adding the customer turnover values from the duplicate records in the cluster to the customer turnover value of the master record.

Plan 08: View Consolidated Data

This plan writes the data consolidated in the previous plan to a Informatica report. Specifically, it generates a list of the match clusters in the dataset — the duplicate records identified in plan 06 above.

The report contains a master_flag column that identifies the master record in each cluster and a new_turnover column that lists the turnover amount for each record in the cluster.

Note that the turnover figure for the master record was updated in plan 07. With the master record identified, you can archive the other records in the cluster and retain the most accurate account information.

Like its predecessor, plan 07, the plan makes use of a DB Source. However, it uses a Count component and Report Sink to facilitate report generation.

The Count component specifies a single column for the report output — the group_id column — in order to display the match clusters.

Note that the Count component has been configured to return all record matches via a Min Count setting of 2, i.e. all group_id values that occur twice or more are reported. Note also that the Max Cases parameter is set to 100, i.e. the plan is designed to retrieve the first one hundred group_ids found in the dataset.

Index