



XRA^y

A White Paper

Introduction

XRA^y is a power-tool, with several micro-components architected to resolve a wide range of reconciliation issues within organizations. XRA^y is derived from years of experience with standard and inhouse systems to meet the needs of the financial and insurance industry.

XRA^y allows organizations to

- ❑ focus on the business by saving time and effort otherwise spent on tedious reconciliation processes
- ❑ reconcile information along the entire processing chain: from Sales and Production, Front-Office, Operations to Accounting and Reporting
- ❑ automate reconciliation processes by defining information sources, transformation and matching rules and reporting criteria without programming an application for each process
- ❑ consolidate existing reconciliation applications into one uniform framework
- ❑ leverage from the experience built into the solution

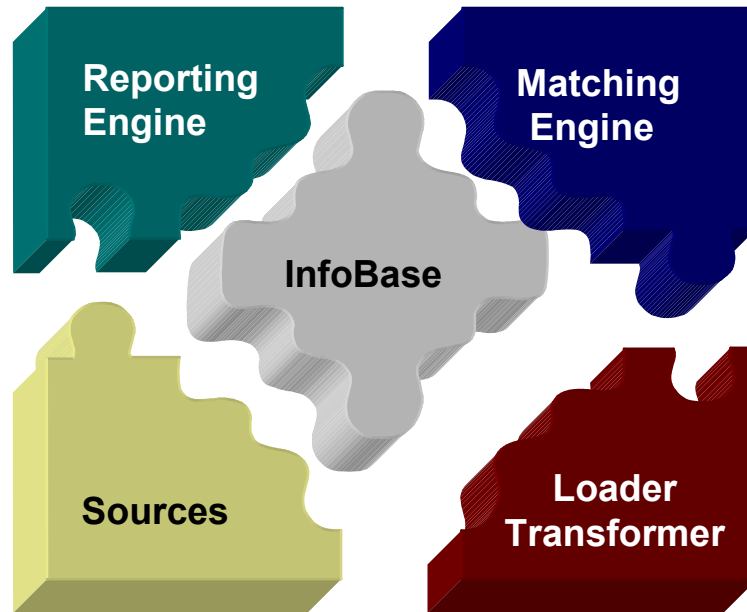
XRA^y's features include

- ❑ a normalized data model and data base for a wide variety of reconciliation needs
- ❑ a flexible load and transformation engine that supports ASCII and relational formats (XML support will be available in the next release)
- ❑ a rule editor and rule engine for the definition and execution of matching criteria between the sources of information
- ❑ a reporting engine that allows to aggregate and drill-down the results by flexible criteria

The white paper provides an overview of the features and benefits of using XRA^y for the reconciliation needs of an organization and how to include it in your reconciliation process.

XRA^y Components

The five main components of XRA^y are shown in the picture below:



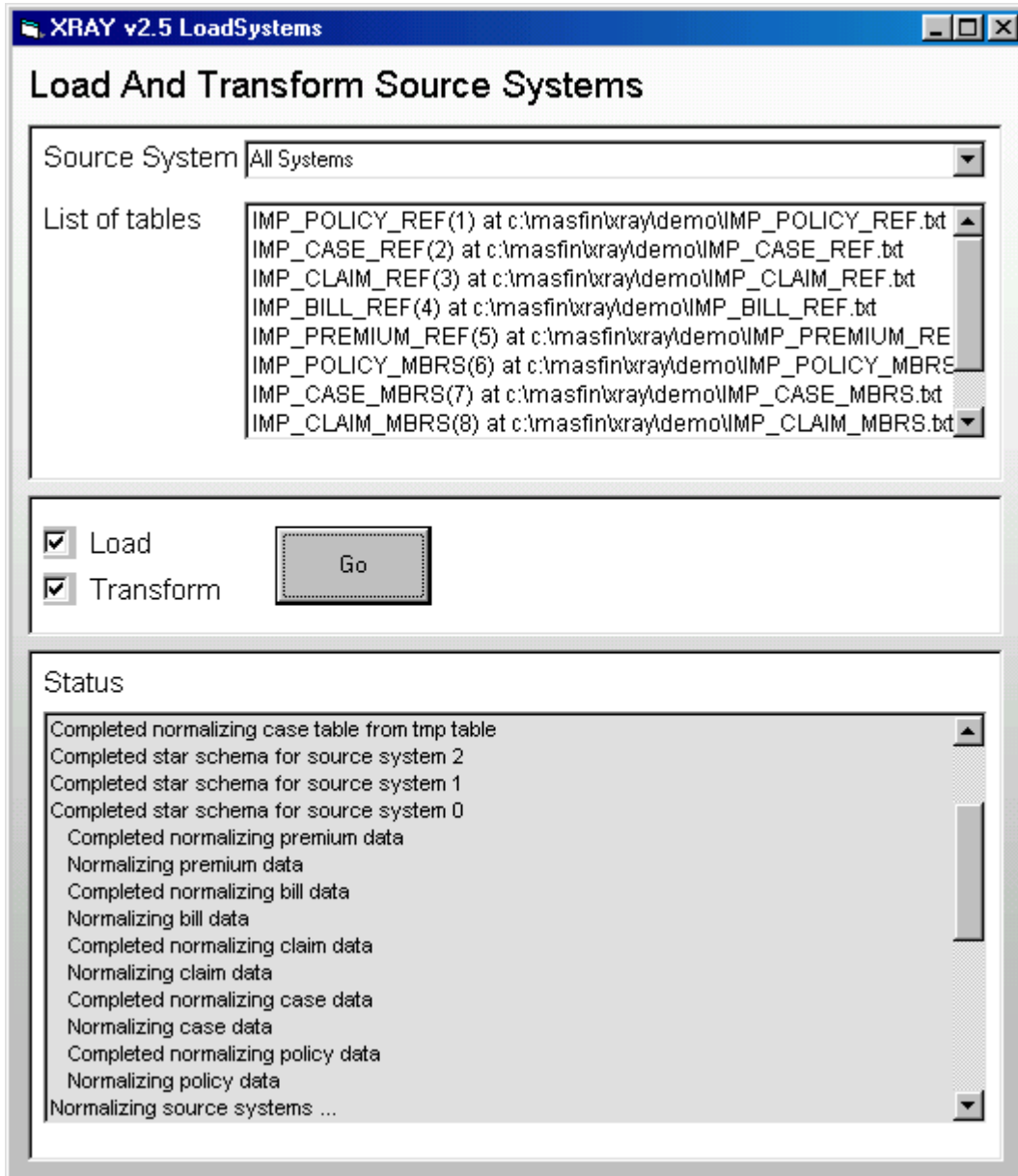
The system allows to define information **sources** that deliver the data for the reconciliation process. These sources can be in the form of ASCII files or exports from relational databases or a mixture of both. The information encompasses the name, type and contents description of each source.

At the core of XRA^y is the **InfoBase**. The InfoBase consists of a generic industry dependant data model that will cover most of the reconciliation needs, enhancements will rarely have to be done.. The data model follows the Star schema approach and distinguishes between two categories:

- event or transaction information such as insurance premium payments, claims filed, etc.
- descriptive information such as customers, location, agent, etc.

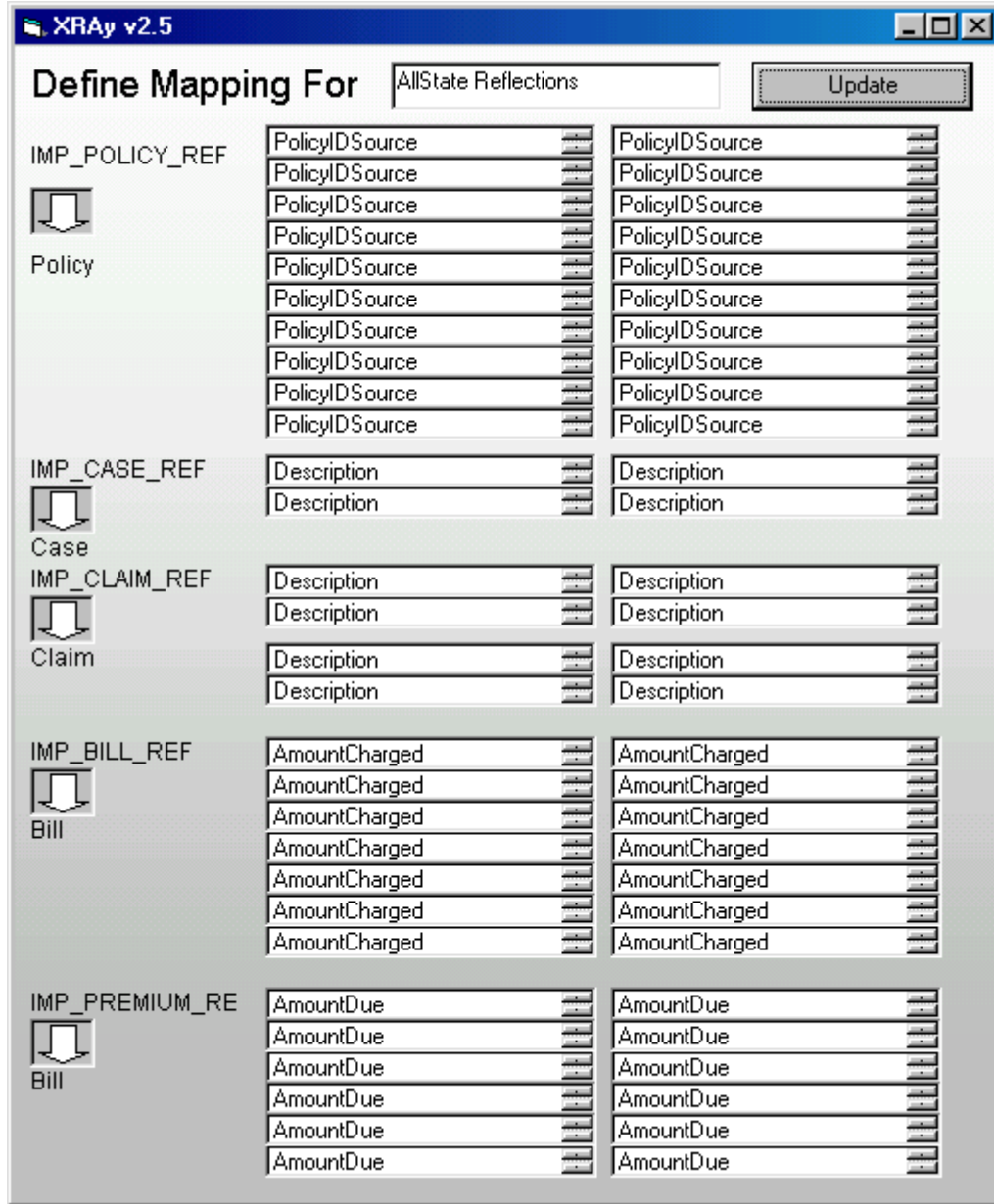
Information from these two categories define the matching process. Events or transactions represent the core information that will be aggregated and investigated along the available description categories

A simple example is shown as follows:



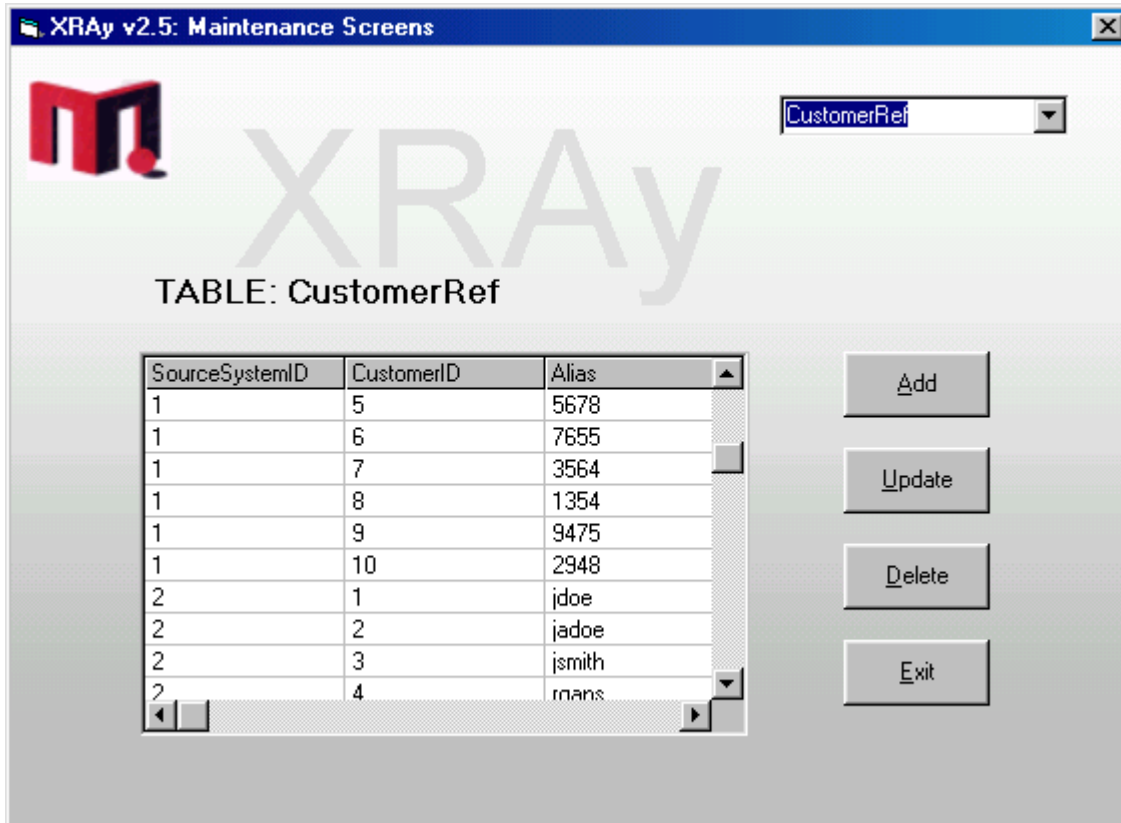
The **Loader/Transformer** allows the definition of transformation rules to map the information between the source systems and the InfoBase. In this process, XRAY^y supports both direct field-to-field transformation, multiple field-to-one-field transformation as well as transformation functions. Based upon the definitions the Loader runs the import of the information from the source into the InfoBase and provides a load status protocol. The imports can be integrated into a scheduler to support automation.

Matching Engine:



The **Matching Engine** contains the rules and criteria for the comparison of information in the InfoBase that was loaded from various sources. All rules are configurable and thus provide the required flexibility to change or enhance the reconciliation process quickly. The matching process produces a status report and result information that can be viewed and investigated by using the **reporting engine**. The information presented by the reporting engine can be aggregated along various description criteria so that the user is able to slice-and-dice the information for investigation or reporting purposes.

Reconciliation and Data Reference Tables

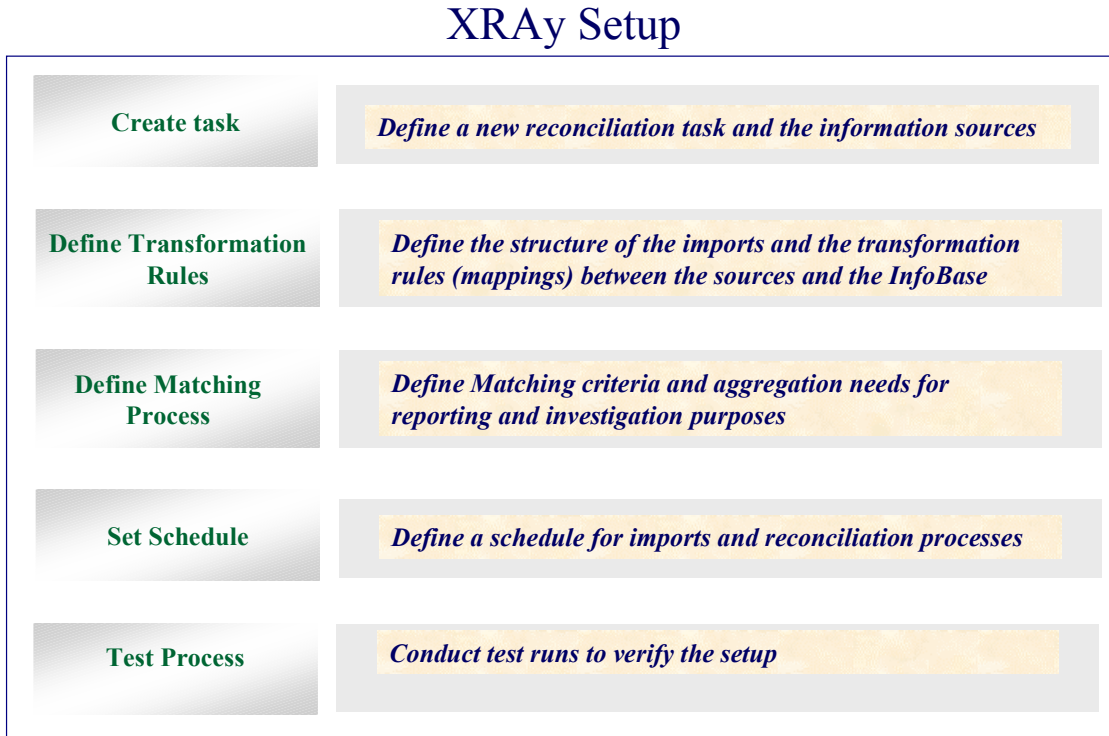


The screenshot shows the XRAY v2.5 Maintenance Screens interface. At the top left is the MASfin logo. The title bar reads "XRAY v2.5: Maintenance Screens". A dropdown menu on the right is set to "CustomerRef". The main area displays "TABLE: CustomerRef" above a data table. To the right of the table are four buttons: "Add", "Update", "Delete", and "Exit".

SourceSystemID	CustomerID	Alias
1	5	5678
1	6	7655
1	7	3564
1	8	1354
1	9	9475
1	10	2948
2	1	jdoe
2	2	jdoe
2	3	jsmith
2	4	mans

The XRA^y reconciliation process

The setup of a reconciliation task in XRA^y requires the following steps:



© Masfin Inc. 1999

After completion of the test runs the reconciliation task will be performed regularly according to the defined schedule.



How about the technology ?

The technologies for the implementation of the XRA^y power-tools was chosen to reach the following objectives:

- ❑ leverage from widely used databases and programming languages
- ❑ enable a low cost maintenance of the components
- ❑ provide a reliable platform for the reconciliation tasks
- ❑ support multiple sources and generally available source types to reduce the effort of creating the imports for XRA^y.

Based on these criteria, we built XRA^y with the following technical components:

- ❑ MS-Access, SQL-Server and Oracle as the standard database engine for the InfoBase.
- ❑ Visual Basic as the programming language
- ❑ ODBC/ADO for the communication with the underlying database
- ❑ ASCII or database to database link for importing the source data
- ❑ XML Support
- ❑ Integration with OLAP Reporting Tools and support of OLAP databases

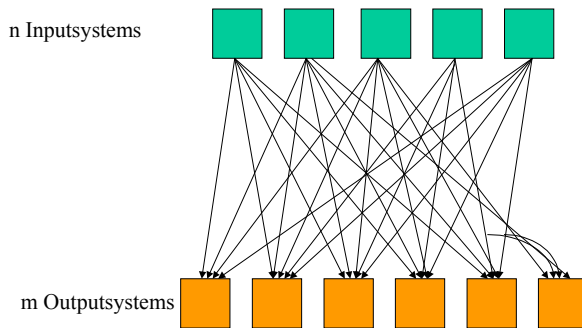
For additional info please contact us.

Email: contact@masfin.com / Tel: (201) 459-9902 / Fax: (201) 459-9904/ www.masfin.com
Address: Masfin Consulting Inc. 26 Journal Square, Ste.1100 Jersey City, NJ 07306

APPENDIX A:

BENEFITS OF NORMALIZED DATA-MODEL

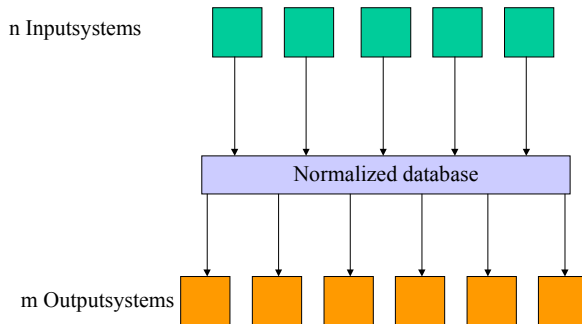
Consider the case where n input systems require m -output mechanisms which include feeds, interfaces, reports etc.



Result: $n * m$ interfaces are needed!

Here every input system needs an interface to every output system. The result is that a bank needs for n input systems and m output systems about $n * m$ interfaces. This is ineffective and cost's a lot of computer power and time. Since time is money this should be more efficient.

With the help of the normalized database approach of Masfin's X-Ray power-tool. The way from the input systems to the output systems would look like this:



Result: $n + m$ interfaces are needed!

Every input / output system needs only ONE interface The result of these approach is that a bank needs for n input systems and m output systems only $n + m$ interfaces.