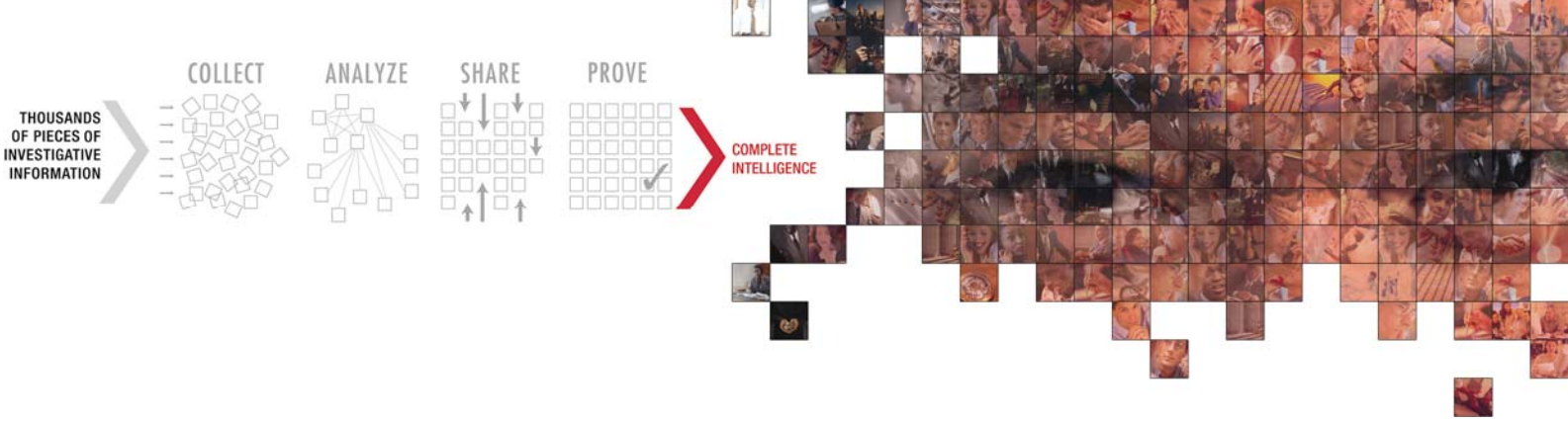


XANALYS Link Explorer

White Paper

Xanalys®

Let nothing escape you

0804-WP-XLE01

Copyright and Trademarks

XANALYS Link Explorer White Paper

August 2004

Part number: 0804-WP-XLE01

Xanalys Limited
Market Court
20-24 Church Street
Altrincham, Cheshire

WA14 4DW
United Kingdom

telephone +44 161 941 7792
fax +44 161 332 8285

www.xanalys.com

Table of Contents

This paper focuses on XANALYS Link Explorer software and its underlying database visualization and interrogation software components.

1 Introduction

This section gives an overview of the paper, and describes what XANALYS Link Explorer is.

Overview of the Paper

This paper first describes XANALYS Link Explorer from the user's point of view, and then gives a description of the technological challenges posed by the design and how Xanalys has solved them. The discussion of technological challenges and solutions, which makes up the bulk of this paper, is organized into sections, with related challenges grouped together.

What Is XANALYS Link Explorer?

XANALYS Link Explorer is an analysis tool used in any domain where analysts have to search through complex webs of interconnected data.

In XANALYS Link Explorer, analysts construct queries about their data by using a drag-and-drop icon-based Query Editor, and then choose how they want to see the results: as a report consisting of a set of tables, or in a chart where data appears as icons connected by lines. XANALYS Link Explorer has algorithms that automatically lay out the charts in ways that allow the human eye to easily see underlying patterns; and users can override any of XANALYS Link Explorer's layout choices. For example, an analyst could apply a special icon to men in their 30s who live in a certain part of town, or highlight links between known gang members.

XANALYS Link Explorer works quickly even with large databases, allowing analysts to expand or narrow their queries over and over again until the analysts find the information they're looking for. Analysts can examine the same query in a variety of ways—perhaps once as a report, once as a chart of connections, and again as a timeline. All three views could be on the screen at once, and a change in one chart is automatically propagated to the others. And because certain sorts of queries may be useful in many sorts of settings, XANALYS Link Explorer allows the user to save and reuse them.

XANALYS Link Explorer can handle data from many sources. Users can connect directly to a variety of ODBC databases, or they can import data into XANALYS Link Explorer's own internal database. Once in XANALYS Link Explorer, all data is interpreted as a set of objects and links. Users can define how XANALYS Link Explorer translates the data from the original database into its object/link structure. Once users have created this definition (called a template), XANALYS Link Explorer quickly and automatically translates all queries and manipulations back and forth as the analyst creates queries and modifies data.

2 Using XANALYS Link Explorer

This section describes XANALYS Link Explorer from the user's point of view.

XANALYS Link Explorer has two types of user. The first is the *analyst*, who uses XANALYS Link Explorer to create queries and analyze data. The other user is the *data modeler*, who uses XANALYS Link Explorer to create the set of rules by which the data is organized.

This section is divided into two parts, one for each type of user.

Using XANALYS Link Explorer: The Analyst's Experience

An analyst uses XANALYS Link Explorer to find specific information from a set of data, or to uncover hidden patterns. In a typical session, an analyst opens XANALYS Link Explorer, connects to an existing database using a template that specifies how the source data is translated into objects and links, and then engages in the Query➲Chart➲Analysis loop.

The Query➲Chart➲Analysis loop is the heart of the analyst's work. It is an iterative process in which the analyst creates a query (which might represent a hypothesis), looks at one or more charts of the results, analyzes them, and then expands or narrows the query, and so on.

For example, an analyst studying a sequence of events might create a query to see all the events connected to the movements of a person in the preceding 24 hours. The analyst could view the results as a free-form graphical chart, a textual table, or a timeline. The analyst might then see that one of the events was of particular interest, and could then refine the query to include all other people connected to that event.

The rest of this section discusses each of these stages in turn:

- opening a database
- querying
- charting
- analyzing

Opening a Database

The user begins by opening the database that they want to work with. They must also specify which *workfile* or *template workfile* contains the rules that determine how that database is to be interpreted. The rules are called *class definitions*.

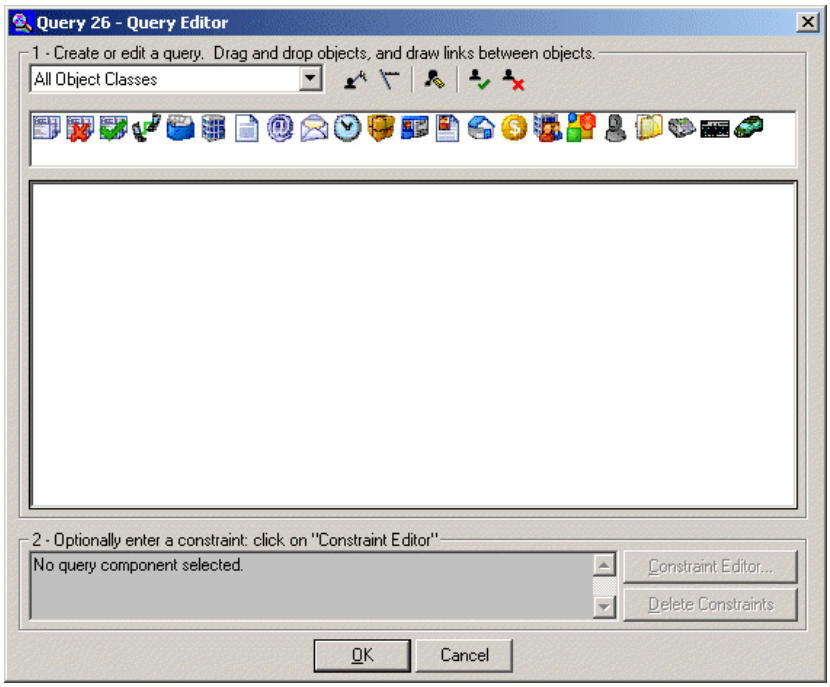
The workfile or template workfile contains not only these class definitions, but also a library of predefined queries, styles, data entry forms, and other elements that the creator of the template feels might be of use.

The database may be in XANALYS Link Explorer's internal format, or in any number of standard database formats connected via ODBC. The structure of the database does not matter, so long as someone has created a XANALYS Link Explorer template defining how XANALYS Link Explorer should interpret the data. (This is the job of the data modeler, who is discussed in Using XANALYS Link Explorer: The Data Modeler's Experience.)

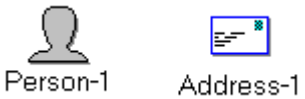
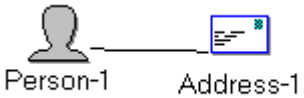
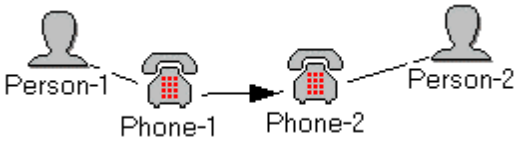
Creating a Query

Having opened a database, the analyst is likely to start by executing a query. Analysts are likely to start with one of the queries built in to the workfile, but could also create their own.

XANALYS Link Explorer analysts create the queries using a graphical Query Editor that enables you to drag and drop icons and lines. Here's a picture of the Query Editor:

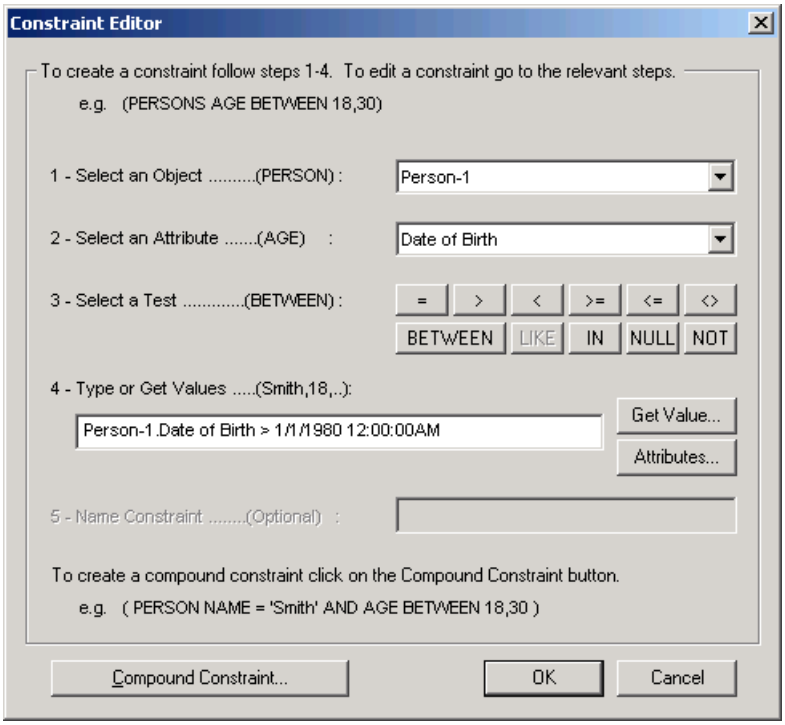


Here are some examples of queries built with the Query Editor, and what they mean:

Query meaning	Appearance in Query Editor
Show all people	
Show all people and all addresses	
Show all people all links between people and addresses	
Show all phone calls between people	

The examples above show queries as they relate to whole classes of objects, such as *all* people or *all* people linked to *all* addresses. Analysts can also create use *constraints* to narrow down a query, for example to look only for men born after 1980 linked to addresses in a certain city.

The Constraint Editor looks like this:



By applying constraints to the classes in the **Query Editor**, analysts can build up complex queries. XANALYS Link Explorer allows the user to save both queries and constraints so they can be used again.

After creating queries, the analysts typically look at the query results in charts, as described in the next section.

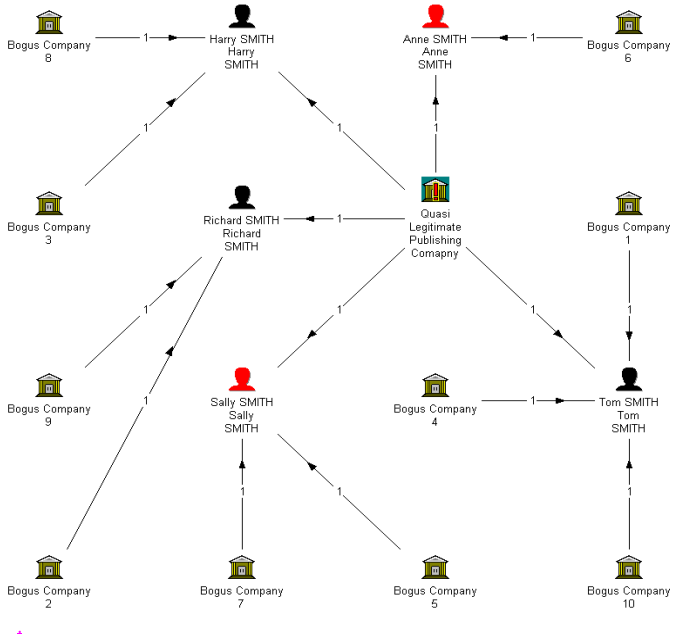
Charting the Results of a Query

Having created a query, XANALYS Link Explorer analysts can display the results in any number of charts or reports. Analysts can open many charts at once, and XANALYS Link Explorer automatically propagates changes from one chart to all of the other charts. For example, a user might have opened two views of a chart of people and events: one a simple link chart showing which people are connected to which events; and the other an event chart, which displays the same information, but with the events laid out in order on a timeline. If the analyst adds an event to one of these charts, XANALYS Link Explorer will automatically add the same event to the other chart.

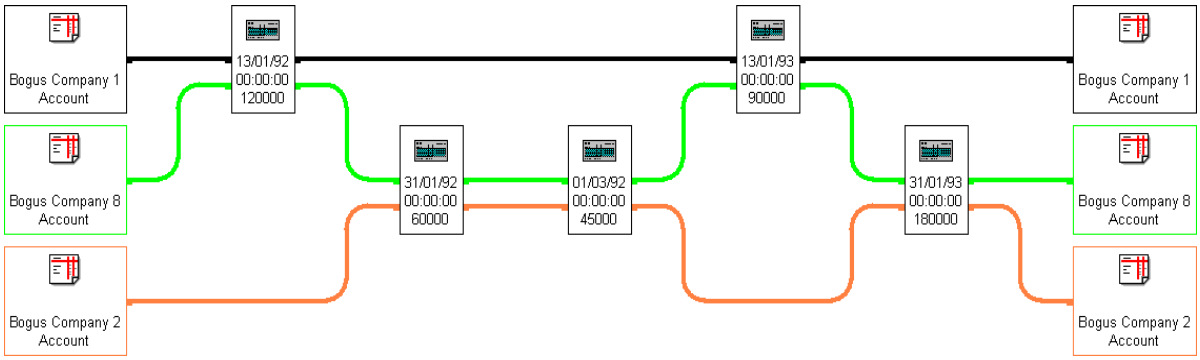
Analysts may also take advantage of this feature by having two views of the *same* chart, one miniaturized so they can see the whole chart at once, and the other enlarged to focus on one particular area. If analysts select an object in one chart, XANALYS Link Explorer automatically selects the same object in the other chart.

XANALYS Link Explorer offers analysts a wide variety of chart types:

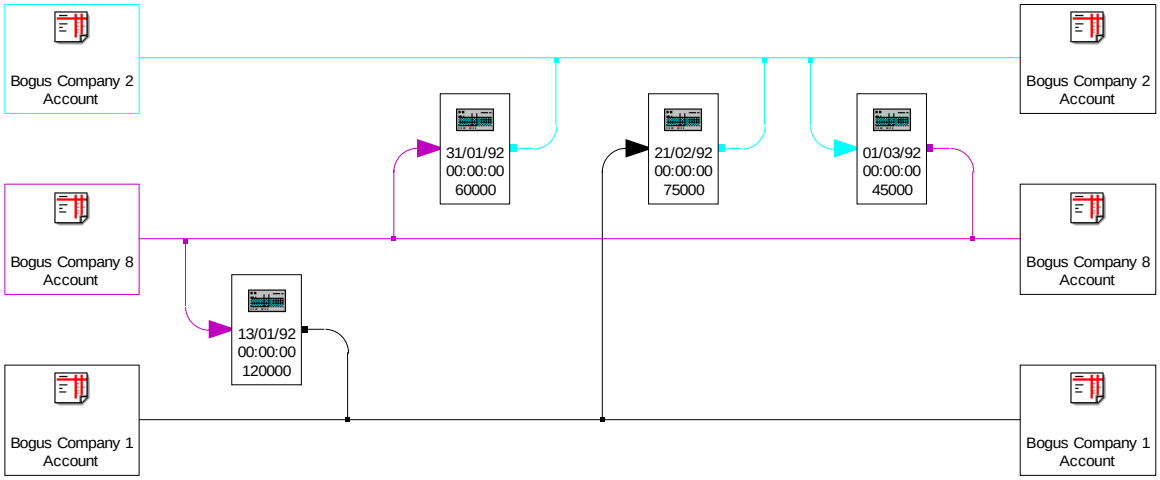
Link chart. A link chart shows how objects are linked together. XANALYS Link Explorer organizes the chart for topological clarity rather than semantic meaning. Users can move items in the chart to their liking.



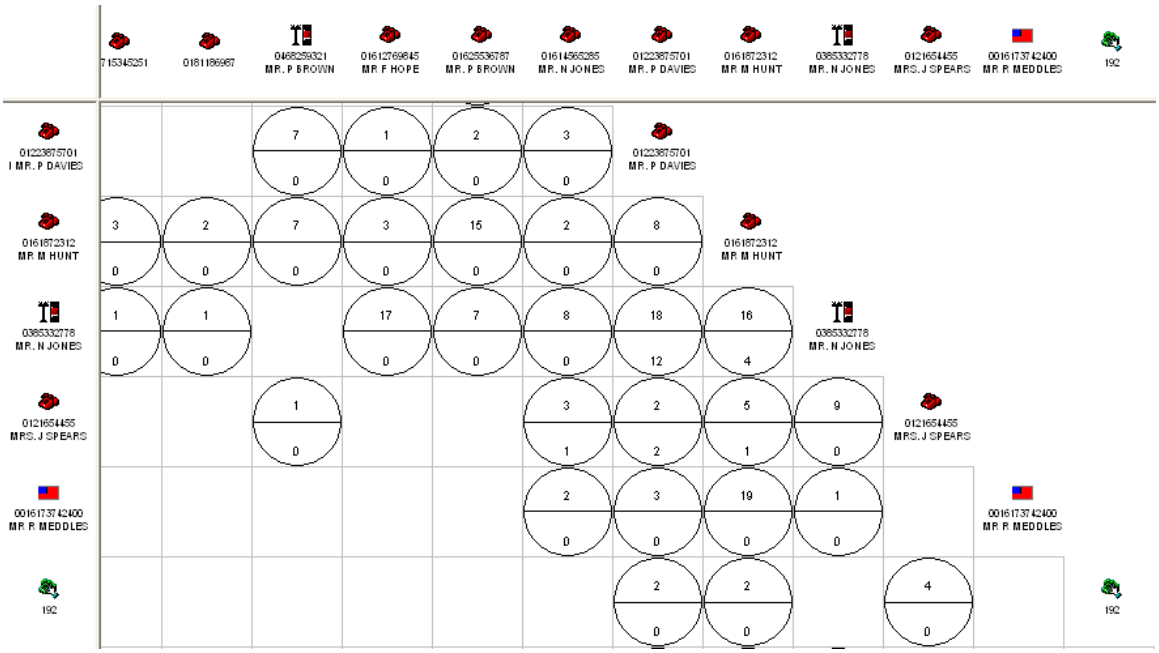
Event chart. An event chart also shows how objects are related, but organizes them in chronological order. Event charts make a distinction between events (which must have a time associated with them) and other objects, which are called actors. In an event chart, XANALYS Link Explorer places the event objects in order in the middle of the chart, and the actors at the left side of the chart (and repeated on the right side). By following the lines from the actors, you can see which actors are involved in which events. In this example, the actors are all accounts and the events are all transactions. You can see which accounts are involved in which transactions.



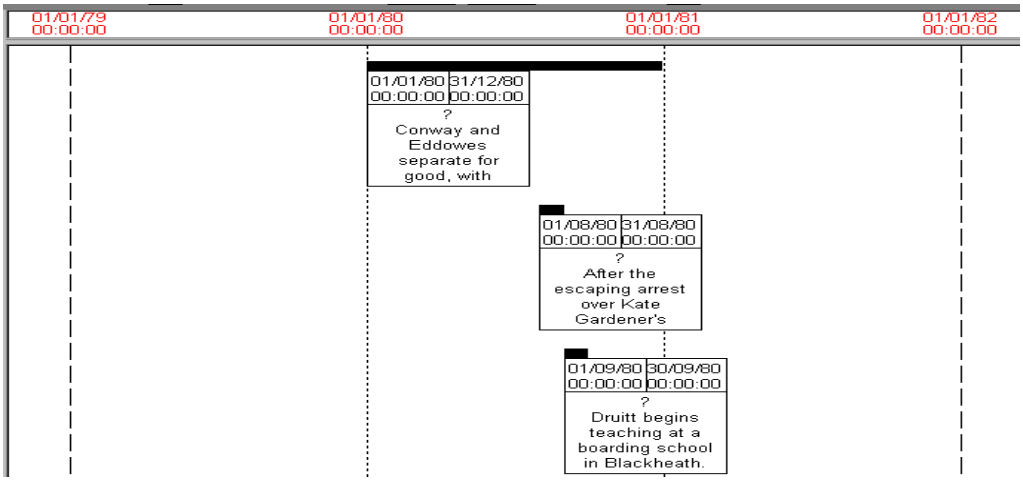
Transaction chart. The transaction chart is similar to the event chart, but it shows not only the order or events, but the flow between the objects. In this example, you can see the same data as in the event chart example, but with flow indicated by the arrows.



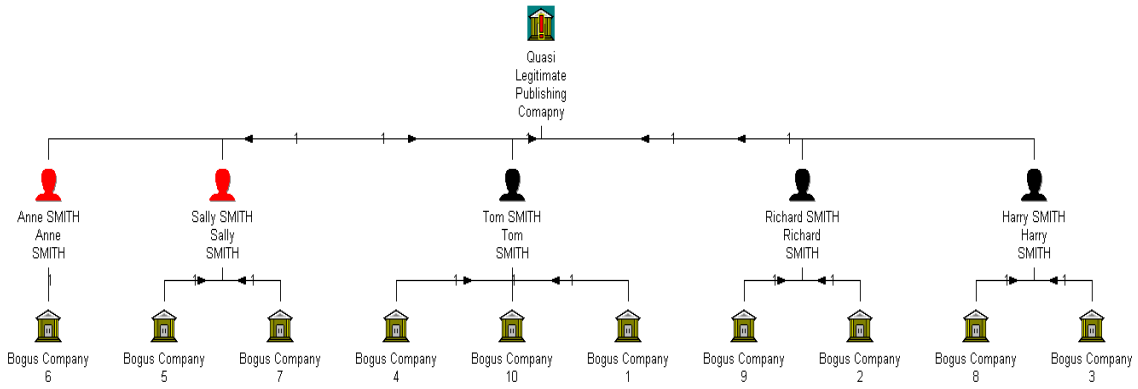
Association Matrix chart. Association Matrix charts enable you to view the link relationships between two objects. Using a grid layout, Association Matrix charts display XANALYS Link Explorer queries in a format that clearly show the number of links between two objects. Association Matrix charts can be customized in various ways and use the same default styles as link charts.



Activity timeline. In activity timelines, the focus is on the duration and overlap of events.



Hierarchy chart. Hierarchy charts emphasize the relationships between parent, child, and sibling objects. XANALYS Link Explorer allows the analysts to choose what the root object is and what sort of layout they want.



Report. Reports contain sets of tables to show query results in textual form. Analysts can control most aspects of the reports, including which columns to show, how the text is formatted, and how wide columns are.

Person-Person (9)

Person_ID1	Person_ID2	Label	Category	Eval_Code
00000018	000001	Accountant	Employee	Unspecified
000001	000007	sister	relative	Unspecified
000002	000001	married	relative	Unspecified
000003	000001	brother	relative	Unspecified
000004	000001	inlaw	relative	Unspecified
000004	000007	married	relative	Unspecified
000005	000001	supplier	supplier	Unspecified
000005	000016	fiancees	fiancees	Unspecified
000006	000001	supplier	supplier	Unspecified

Person (14)

Person_ID	Category	Label	First_Names	Date_of_Birth
00000018	Accountant	F Johnson	Frank	
000001	drugs : dealer	W Stevens	William	7/7/72 12:00:00 A
000002	relative : wife	R Stevens	Raquel	11/9/75 12:00:00
000003	relative : brother	J Stevens	Joseph	3/23/68 12:00:00
000004	relative : inlaw	R Edwards	Robert	1/7/76 12:00:00 A
000005	drugs : pusher	N Robertson	Nigel	5/3/75 12:00:00 A
000006	drugs : pusher	D Jones	David	9/23/73 12:00:00
000007	relative : sister	S Edwards	Susan	6/9/73 12:00:00 A
000009	drugs : user	B Inkson	Brian	8/9/64 12:00:00 A
000010	drugs : user	S Stoner	Susan	6/8/72 12:00:00 A
000011	drugs : user	I Davidson	Ian	1/9/74 12:00:00 A
000012	drugs : user	J Evans	Jane	12/9/72 12:00:00
000016	drugs : user	D Richards	Diane	6/12/64 12:00:00
000017		unknown witne		

Analyzing the Results a Query

Once analysts have a chart showing query results, they can use a number of XANALYS Link Explorer tools to make sense of the results:

- In link charts, analysts can rearrange the objects by dragging them, or by changing their spacing.
- In all charts, analysts can define rules controlling the appearance of objects and links, including labels, colors, line widths, and icons. In a chart containing a lot of people, the analysts could assign a different icon to people who had a prison record. In a simple chart, analysts could make the labels long and comprehensive, while making those in a more complex chart much shorter.
- In all charts, analysts can add annotations. Annotations can provide titles for the chart, explanations, or keys to symbols.
- In link and hierarchy charts, which can often contain intricate webs of links, the analysts can select two objects and ask to see how they are connected.
- In all charts, analysts can select objects by various criteria, including what constraints they meet, or how many links they have to other objects.

Continuing the Query➲Chart➲Analysis loop

Once the analysts have finished analyzing the chart, they can create a different chart of the same query, or they can create another query. If the new query is quite different from the current one, they could just start afresh. But often the new query will be strongly related to the current one — finding extra links between objects in the chart or finding new links between a specific object in the chart and other objects not yet in the chart. XANALYS Link Explorer has tools to facilitate all these cases, and once the analysts create a new query, they continue by creating charts of the query, and analyzing the results.

Entering and Modifying Data

XANALYS Link Explorer offers a number of tools that allow users to import data, to create new objects and links, and to enter and modify attribute data:

- **Text importer.** To enter consistently formatted, delimited text data, users can invoke the text importer wizard.
- **Flexible Data Importer.** To import more complex data, users can take advantage of the Flexible Data Importer.
- **Data Entry Forms.** To enter data using the keyboard, users can use customizable data entry forms.
- **QuickChart Toolbox.** To create new objects and links by direct manipulation, users can drag objects from the QuickChart Toolbox to any chart, and create new links by drawing lines between objects.
- **Object cards.** To enter or modify object attributes, users can open object cards and change information there.

Using XANALYS Link Explorer: The Data Modeler's Experience

This section discusses how the data modeler experiences XANALYS Link Explorer.

Why Is There a Data Modeler?

For simplicity, XANALYS Link Explorer represents all data as a set of objects and links. However, the underlying databases may use a much more complex data structure. It is the job of the data modeler to define a set of object classes based on the underlying databases. These sets of class definitions are stored in a template workfile. In simple terms, it is the job of the data modeler to create template workfiles.

What Does Creating a Template Workfile Involve?

A template workfile consists of a set of class definitions plus an optional library of tools for the analyst to use, including predefined queries, charts, styles, and data entry forms.

But the heart of the template workfile is the set of class definitions. The data modeler decides how many classes the workfile should have. Each class represents a specific type of object. A typical template for investigation might, for example, contain the following classes: person, address, vehicle, event, phone, account, and organization, plus classes representing all the possible links between those classes. If a modeler was creating a template for military intelligence, they might want classes to represent weapons, bases, and satellites. Similarly for business, financial and market intelligence.

For each class, the data modeler must specify how many attributes it has, and for each attribute, the modeler must define what type it is, and where in the underlying database the information comes from. In some cases, the data for a single attribute could come from many tables, or may have to be computed from more than one underlying attribute.

For example, a Person object in XANALYS Link Explorer may have attributes for first name, last name, age, and phone number. The names may come directly from the Person table, the age may be calculated from the date of birth in that same table, and the phone number from a separate table keyed to the Person table.

To create these class definitions, the data modeler uses the Class Editor tool in XANALYS Link Designer.

Basic Options for the Data Modeler

Because data may be stored in any number of types of databases and structured in any number of ways, XANALYS Link Explorer offers a variety of options to the data modeler to create template workfiles:

- If the data already exists in a compatible database, the data modeler can create a template using XANALYS Link Explorer's *integration mode*. In this mode, XANALYS Link Explorer analyzes the structure of the existing database and either automatically creates a compatible template, or guides the data modeler to do so.
- If the data exists in an incompatible database, the data modeler can create a template using XANALYS Link Explorer's design mode, then export the data to a text file and import that text file into XANALYS Link Explorer's internal database.

- If the data does not yet exist but the users want to enter the data directly into XANALYS Link Explorer, the data modeler creates a template using XANALYS Link Explorer's design mode, then creates data input forms that an analyst can use to enter the data.

3 Technology: Data Modelling

This section describes several technological issues related to data modeling in XANALYS Link Explorer.

Creating a Data Modeling Method that Works in Many Domains

The issue. A visual analysis tool is required to work in any domain.

The solution. XANALYS Link Explorer adopts a simple, unified model for representing all data: the entity-relation model. Entity-relation modeling is recognized to be a general and powerful analysis.[1] In this model, all data is represented as objects and links. An object might be a person, an address, a vehicle, a server, or a document on the web. Links represent connections between objects. For example, a model might have a person object, an address object, and a link between the person and the address to indicate the person lives or works at the address.

Objects and links have attributes. For example, a person object might have attributes for the person's first and last names, and date of birth. The person-address link might have an attribute that describes whether it is a home or business link.

Each object and link belongs to a class, and it is the job of an expert user called a *data modeler* to define these classes. The class definitions determine what attributes each type of object and link has, and also how those attributes relate to the underlying database.

This model gives the data modeler a high degree of flexibility. Using the example of people and addresses, one modeler might represent them as people objects, address objects, and people-address links; another might represent them only as people objects, with the address as an attribute of the person.

Translating between Databases and XANALYS Link Explorer's Data Model

The issue. No matter how XANALYS Link Explorer itself represents data, most data is stored in databases and XANALYS Link Explorer needs a way to translate the data from the tables in a database into XANALYS Link Explorer's objects and links.

The solution. XANALYS Link Explorer implements the entity-relation model using an object-oriented approach. In XANALYS Link Explorer, a particular entity-relation model is called a *template*, entities in a template are called *classes,* and the relations between classes are called *links.* A template contains a set of definitions for classes plus a set of definitions for links.

To analyze a particular database, a user first opens the template that defines the classes for that database. The template is created by a data modeler using XANALYS Link Designer's **Class Editor** tool.

A class definition in XANALYS Link Explorer is a functional description that maps the attributes of the class into the database. XANALYS Link Explorer takes those functional descriptions, translates them into database queries appropriate to the particular database being used, and maps the results into the corresponding objects and attributes. A XANALYS Link Explorer class definition includes a database query (expressed in XANALYS Link Explorer's own query

language, called Object Query Language) that defines where the information is coming from, and information about how that information, once retrieved, maps onto the attributes of the XANALYS Link Explorer class.

This functional description also allows data modelers to create attributes by more complex means. For example, a duration attribute in a XANALYS Link Explorer template might be calculated from a start time and finish time in the underlying database tables. Or for a more complex example, a class of telephones might be derived from the union of two tables, one a list of source numbers, and the other a table of phone calls made by the source numbers to a list of destination numbers. The XANALYS Link Explorer class of phone numbers would be the union of the phones that appear in the two tables. In fact, a class can be based on any valid database query no matter how complex. No matter how data is organized in the original database. The construction of complex queries allows the data modeler to organize data in any way that works best for analysis.

Defining XANALYS Link Explorer Links from the Underlying Database

The issue. Most databases contain implicit links. How can XANALYS Link Explorer recognize them?

The solution. XANALYS Link Explorer's flexible system of mapping databases onto XANALYS Link Explorer objects and links makes it possible to extract the links implicit in many database tables. Imagine for example, a database table containing a list of phone calls, with columns for the phone that placed the call, the one that received the call, the date and time the call started, and its duration. XANALYS Link Explorer can resolve that database into a class of phones (representing the union of all phones), and a class of links (representing the calls between the phones).

Furthermore, XANALYS Link Explorer allows a dual interpretation of such a link class — either as a set of links between phone objects, or as a set of objects themselves. So analysts, for example, could view a chart of phones with lines between them making it easy to see who was calling whom; or they could show a chart of icons, with each icon representing a phone call, and then color code the icons by time of day or area code of the originating phone number.

XANALYS Link Explorer also allows a simpler kind of link, called a *direct join*. Link classes and direct joins both contain information about two objects with some kind of relationship. However, link classes contain additional information. A direct join between a person class and an address class would only represent the relationship; a link class between a person class and an address class would allow extra information about the relationship, such as whether the link represented the person's home address or business address.

Supporting Database Connectivity

The issue. Users may have their data in any number of database systems. XANALYS Link Explorer must have a general architecture to support all of them while insulating the analyst from the details of a particular DBMS (database management system).

The solution. This problem requires a particularly robust set of solutions, described below:

- XANALYS Link Explorer supports standard connectivity technologies. XANALYS Link Explorer currently supports connections to relational databases via ODBC. However, XANALYS Link Explorer's architecture could also support extensions to other data sources, such as the World Wide Web and object-oriented databases.

- XANALYS Link Explorer allows expert users, called data modelers, to create templates for any existing database. XANALYS Link Explorer's Class Editor tool has an *integration mode* that analyzes the database and allows the user to automatically create classes based on those tables, or to create more complex class definitions from calculations or from parts of tables.
- XANALYS Link Explorer provides a consistent user interface for building queries no matter which database system the user is connected to. The user interface consists of a Query Editor that uses icons and lines to represent the queries.
- XANALYS Link Explorer contains an abstract querying language that serves as an intermediary between XANALYS Link Explorer and the database. This language, called Object Query Language or OQL, preserves database independence. XANALYS Link Explorer uses OQL internally for the class definitions and to represent user queries. However, the users themselves don't need to know OQL because they never see it. The Class Editor and the Query Editor convert the user's definitions into OQL. Then, at the moment a query is executed, XANALYS Link Explorer translates the query from OQL into the query language of the underlying database. XANALYS Link Explorer handles these translations at execution time. This abstract language insulates users from the different query languages of various underlying databases, and could in the future permit simultaneous connection to databases of different types.

Dealing with Users Who Don't Have Legacy Databases

The issue. Some users may not have their data in databases yet, or their data may be in incompatible databases. XANALYS Link Explorer must give these users a way to model their templates, to store new data, and to import existing incompatible data.

The solution. XANALYS Link Explorer combines a number of different technological solutions to address this challenge:

- For creating data models not based on an existing database, XANALYS Link Explorer provides data modelers with a *design mode*. Data modelers can use the Class Editor's design mode to create a template with all the class definitions required, and then use that template to create a new database.
- For storing data, XANALYS Link Explorer has its own internal database format. Users can create new databases using XANALYS Link Explorer's internal database format, freeing customers from the need to purchase an expensive third-party database tool. The database takes on the design required by the class definitions in the workfile that it is connected to. Similarly, if the user connects to an empty third-party database, XANALYS Link Explorer will create the tables needed by the class definitions in the workfile it is connected to.
- For importing data from incompatible data sources, XANALYS Link Explorer provides a powerful set of importing tools. Users with simple delimited data can use a wizard to import the data; users with more complex data can use XANALYS Link Explorer's Flexible Data Importer. These tools provide a bridge for users who have databases that cannot be connected to XANALYS Link Explorer.

- For entering data by hand, XANALYS Link Explorer includes a number of easy-to-use data entry tools. These tools include customizable data entry forms, and a drag-and-drop system for adding new objects and links directly to charts.

4 Technology: Queries

This section describes the technological issues related to allowing the XANALYS Link Explorer analyst to extract and filter information from databases.

Creating a Query Editor

The issue. The core job of the analyst is to find meaningful information within large pools of data. When analysts use computers, they must extract and filter information from a database, creating and refining queries all day long. The goal for XANALYS Link Explorer, then, was to create a **Query Editor** that the user would find powerful and easy to use, and that would insulate the user from the varying query languages of the underlying databases.

The solution. XANALYS Link Explorer has a graphical drag-and-drop **Query Editor** that allows users to create queries of any level of complexity using icons to represent the classes in the current template and lines to represent links. The **Query Editor** also allows the user to place constraints on the classes based on the values of attributes. Users could, for example, ask to see only those Person objects whose Sex attribute is equal to "Male" and whose Date_Of_Birth attribute is greater than "31 December 1980". The **Constraint Editor** consists of a structured form that allows naive users to construct their query piece-by-piece using pick lists, and expert users to type the constraint by hand. In either case, the **Constraint Editor** makes sure the constraint is correct.

Users can save, reuse, and modify both queries and constraints.

Users can also generate queries indirectly from XANALYS Link Explorer's analysis tools, such as the **Explore** commands. The user issues these commands while working in a specific chart, and the commands automatically alter the query upon which the chart is based.

No matter how the query is defined, XANALYS Link Explorer stores it in its own query language, called Object Query Language or OQL.

Executing the User's Query

The issue. XANALYS Link Explorer must communicate the user's query to the database using the database's own query language, and then translate the results from database tables into objects and links. The details of the translation depend on what type of database the user is connected to. The twin goals of the query execution module are: (1) to insulate the user from the implementation details of the underlying databases; and (2) to make it easy to support many kinds of data sources, from new DBMS products to object-oriented databases and the World Wide Web.

The solution. XANALYS Link Explorer has a query execution module that handles all the work from the moment the user executes a query (whether from the Query Editor or from an analysis command), to the time the answer is received back from the database and translated into objects and links.

Here's what the query execution module does:

1. Takes the query, expressed in OQL, and converts it to the query language of the underlying database.
2. Communicates the query to the database via an appropriate API, such as ODBC.
3. Waits for the database to return the results, which are expressed in the form of tuples.
4. Converts the tuples to objects and links.

While query execution is a generic module, the components that deal with specific types of databases are implemented as plug-ins. These plug-ins convert the OQL to the database-specific query language, and then convert the results from data tables into objects and links. To support another kind of database (or any other kind of data source), Xanalys need only author another plug-in. At present, XANALYS Link Explorer has plug-ins to support a variety of relational DBMS's, including MS Access, Informix, and Oracle, plus XANALYS Link Explorer's own proprietary database format.

For the relational databases, the OQL queries are converted to SQL and communicated via Microsoft's Open Database Connectivity (ODBC) API. The databases always return the results of the queries as tables of data.

5 Technology: Charts

This section describes the technological issues related to creating charts based on queries.

Creating an Interface to Get Multiple Views of the Same Data

The issue. Having created a query, analysts may need many different ways to view the data so they can discover the underlying patterns.

The solution. XANALYS Link Explorer employs a number of technologies to help the analyst get the views they need:

- To help the user see patterns, XANALYS Link Explorer provides a variety of chart types. Some charts simply show objects and links; others, such as event and transaction charts, organize the objects according to other semantic information.
- To handle a variety of circumstances, XANALYS Link Explorer offers several automatic layouts for some chart types. When dealing with scores or hundreds of objects, the geometry of the layout can affect how the user perceives the relationships of the objects. Users can easily switch between layout formats.
- To give users full control, XANALYS Link Explorer allows them to override automatic layouts. When an automatic format does not satisfy the user, the user can create their own layout by dragging objects where they like.
- To allow users to compare views, XANALYS Link Explorer allows multiple charts to be open at once, and updates them as necessary. An analyst may want to have more than one chart of the same query open on the screen. XANALYS Link Explorer supports Microsoft's multiple document interface (MDI), treating each chart as a separate window. XANALYS Link Explorer adopts a common chart architecture and a change propagation module to ensure that changes in one chart are reflected in other related charts, and that two charts of the same

query are kept in synch—for example, if an object is selected in one chart, it will also be selected in the other.

- Using Link Chart Merging, you can combine two link charts. XANALYS Link Explorer first merges the two queries and then re-draws the link charts, using the original two charts as a template. After two charts have been combined, the original annotations, positions and styles are retained.

Implementing a Common Chart Architecture

The issue. To accommodate the user's need to display query results simultaneously in many different charts, XANALYS Link Explorer needs a way to enable the efficient sharing of common information between charts.

The solution. XANALYS Link Explorer implements a common chart architecture. To understand the chart architecture, it helps to have an understanding of the steps XANALYS Link Explorer goes through to display a particular view of a particular chart of a particular query.

1. XANALYS Link Explorer executes the query and gets the results.
2. The query provides a generic linkage description based on the link objects and direct joins.
3. A graph builder for the specific type of chart then uses the generic linkage description to generate a specialized connectivity graph. The graph builder uses semantic information from the template's class definitions and link information from the query results to filter and sort objects, and to construct alternative links between objects. There is a different graph builder for each type of chart, but they all use a common protocol to communicate with the template and query modules.
4. A layout component then takes the specialized connectivity graph and uses it to generate an optimized layout based on the user's layout choices. The layout includes geometric information associated with each graphical object.
5. The optimized layout is then drawn in the window according to how the user has set the magnification, scale, and rotation.

Much of the key work in this process is done by the graph builder and layout components. The graph builder applies meaning to the query results; the layout components enhance the visualization of that meaning.

The common architecture allows information to be shared at each stage:

- The generic linkage description is shared by all charts of a particular query, regardless of chart type.
- The specialized connectivity graph is shared by all charts of the same type of that query, regardless of layout choices.
- The optimized layout is shared by all views of a particular chart, regardless of zoom factor, scale, or rotation.

6 Technology: Data Input

This section describes the technological issues related to entering data.

Implementing a Generic Data Importer

The issue. XANALYS Link Explorer offers users several ways to enter large amounts of data: the text importer, the Flexible Data Importer, and customizable data entry forms. Though these seem like very different processes to the user, they have much in common from a technological point of view. All follow the same process of converting data to objects and links, and then translating those objects and links into a form that can be stored in the database. Also, they all need a certain amount of integrity checking and the ability to deal with exceptions, such as duplicate data.

The solution. Rather than implement the data input procedures separately, Xanalys created a Generic Data Importer component. The GDI works with the data stream from any of XANALYS Link Explorer's bulk data input sources, such as the data entry forms, text importer, or Flexible Data Importer, and then processes the data as follows:

1. The GDI uses a piece of software called the import description object to parse and extract data from the data stream and use it to create XANALYS Link Explorer objects and links.
2. The GDI then references the class definitions in the template to convert the object/link data into a form that the underlying database can understand, first in OQL, then in the database's own querying language.
3. The data is then communicated to the database via an appropriate API, such as ODBC.

The importer also automatically handles various exceptional cases such as duplication of data, type check failures, and parsing problems. The strategy for handling these exceptions is part of the import description, and reflect options set by the user.

Supporting Multi-User Data Input

The issue. When users are connected to an external database and begin to modify or add data, the system must ensure that data integrity is not harmed by allowing two users to have write access to the same piece of data.

The solution. XANALYS Link Explorer makes use of features provided by the database itself, typically by using the database's locking facility. Locking is a mechanism for providing a user with temporary exclusive access to part of a database.

During the import process, XANALYS Link Explorer can even work with the database to lock a whole class if that is necessary (and if the database supports the operation).

7 Analysis Features

This section describes the technological issues related to analyzing data once it is in a chart.

Creating an Interface to Make It Easy to Extend Queries

The issue. Analysts spend much of their time incrementally refining queries to find what they are looking for. They need methods to make this process easier.

The solution. The XANALYS Link Explorer interface provide commands to make it easy for a user to refine a query so as to get information about objects and links that are not yet part of the query results. For example, analysts could use the **Explore** commands to find extra links between objects that are already in the chart, or to find links for objects already in the chart to objects that are not yet in the chart.

In these examples, the user could simply edit the query in the Query Editor, but even though this process takes less than a minute it can disrupt an analyst's train of thought. Consider an analyst looking at a chart of people and phone calls. The analyst decides to investigate whether a specific phone call preceded a particular event. Without leaving the chart, the analyst could select the person's icon and ask to see all the events connected to that person. The analyst could add constraints to the exploration to filter events before the time of the phone call. In response, XANALYS Link Explorer modifies and executes the query, and adds the linked events to the chart automatically.

When XANALYS Link Explorer adds the new objects to the chart, it leaves all existing objects in place so the user doesn't lose the old context, though users can clean up the layout if they choose to do so.

Creating an Interface for Rule-Based Styles

The issue. The human eye is one of the most effective tools of the analyst, capable of recognizing subtle patterns or anomalies. XANALYS Link Explorer needs to give analysts the ability to mark meaningful distinctions in the data by using visual distinctions in the charts, so analysts can see patterns inherent in the data. Furthermore, since the distinctions drawn are often based on complex criteria, XANALYS Link Explorer needs to provide the ability to define complex criteria, to save and reuse the criteria, and to make use of a sophisticated system of defaults.

The solution. XANALYS Link Explorer offers a power set of ways to set and apply rule-based styles. Users can set styles for objects (including color, icon, shape, and label) and links (including color, width, pattern, and label). Users can set the styles for a single object, a whole class of objects, or only objects that meet certain criteria. And for all these styles, they can set the rules for a single chart, all charts of one type, or all charts. When an object satisfies more than one set of style rules, the conflict is resolved using a simple user-defined ordering of style rules and rule sets.

The ability to set styles for objects or links based on certain criteria is particularly powerful, allowing analysts studying, for example, incidents involving a population of prison inmates to assign different icons to prisoners based on gang affiliation, cell block, or age. These criteria are defined using the same Constraint Editor as is used in the Query Editor, and as with the Query Editor the constraints can be named, saved, and reused.

All style rules are stored as part of the charts, and so as the data changes, any new objects appearing in the charts are displayed using the same rules.

The charts and style rules are, in turn, stored as part of the workfile, and if the workfile is connected to another database, the analyst automatically inherits the style rules stored in the workfile.

Creating an Interface for Analysis by Selection

The issue. When viewing complex charts, analysts can sometimes have a difficult time seeing how objects are linked, or finding objects that meet particular criteria.

The solution. The XANALYS Link Explorer interface allows the user to trace the path between two selected objects by selecting all the links and objects between them, and to select objects that meet a certain criteria.

The ability to select the objects that the analyst is concerned with is particularly useful because once they are selected the analyst can perform many other operations on them, such as applying a style to them or basing a new query on the existing selection.

8 Technology: Infrastructure

This section describes the technological issues concerning the infrastructure for the rest of the software components and the user interface.

Creating a Framework for Semantic Events

The issues. XANALYS Link Explorer can be said consist of a *user interface*, composed of the menus, dialog boxes, and windows that the user sees, and a set of *operations*, which include the ability to manipulate, filter, and display data. Normally, the user interface controls the operations, but Xanalys wanted to decouple the two so that other applications could also control the operations. Such a design also provides a good framework for implementing the undo feature within XANALYS Link Explorer.

The solution. XANALYS Link Explorer supports a design called *semantic events* which is an extension of the classic Command design pattern used in object-oriented programming.[ii]

In XANALYS Link Explorer, a request for an operation is encapsulated as a first-class object called a semantic event. Other software objects can assemble an appropriate semantic event with relevant parameters, and tell the semantic event to execute itself.

The user interface can create semantic events—that's how menu commands are executed—, but so can other applications communicating with XANALYS Link Explorer via DDE. In the future, it will be possible to communicate via other interoperability technologies as well, including CORBA and OLE Automation.

In addition to knowing how to execute itself, a semantic event object can store state information for reversing itself, thus supporting an undo feature. To facilitate entire sequences of operations to be scheduled or undone, XANALYS Link Explorer can store semantic event objects in queues or history lists. It would also be easy to use these objects to log operations.

Providing a Mechanism for Change Propagation

The issue. XANALYS Link Explorer is a collection of cooperating modules. Some modules display a chart, some change data, some affect styles and selection. It is essential to maintain

consistency across all the modules, and to do so without compromising their overall generic applicability.

The solution. XANALYS Link Explorer implements a version of the "observer" object-oriented design pattern.[iii]

The mechanism inside XANALYS Link Explorer is called Change Registration and Propagation (CRP). In the CRP design, objects in one module register an interest in changes to other objects. Whenever a CRP object changes its state, it broadcasts messages to the objects that registered an interest in it. The registered objects then handle the change—possibly changing their own internal state and broadcasting more change descriptions. This network of registrations across the various modules. is called the *CRP dependency graph*.

Change propagation operates within a global loop. The loop is initiated by the execution of a semantic event. When a semantic event changes a state in a XANALYS Link Explorer object, a change description is broadcast to the registered dependent objects. The dependent objects store the change description and broadcast an invalid flag to all their registered dependent objects. Thus invalidity of objects is propagated up the CRP dependency graph.

When execution of the semantic event is finished, the CRP graph is walked, starting from user-visible objects, updating all invalid objects. An object is invalid if it has either stored change descriptions or its invalid flag is set. An object is updated when all of the objects that it depends on are updated and all of its change descriptions have been processed.

The rationale for starting from visible objects is simple: the user needs only visible objects to be updated—that is, there is no point in forcing the update of a chart that is closed or has no view associated with it. The operation of walking the dependency graph continues until all visible objects are updated.

The CRP dependency graph is a significant extension to the classical object-oriented design. It is implemented as a set of abstract mixing classes. The core protocol of CRP is inherited by concrete classes in the various modules. Each concrete class must implement its own processing of change descriptions.

The CRP is central to XANALYS Link Explorer for guaranteeing consistency of displayed information. This is the mechanism permitting the proper propagation of selection as well as propagation of change in object values therefore permitting visual analysis. This system is also responsible for the propagation of changes in queries results when queries are modified during the process of analysis by query refinement. Finally when several views are based on the same chart layout, the CRP ensures that the geometrical positions of objects directly manipulated by the users remain consistent across all views, permitting comparative analysis at the chart level.

This section describes how the technologies already implemented in XANALYS Link Explorer provide possibilities for easy-to-implement features.

Because XANALYS Link Explorer is written in Lisp, it would be possible to port it to other platforms.

---

[i] John Sowa's "Conceptual Structures: Information Processing in Mind and Machine" and Timothy Nagle, Janice Nagle, Laurie Gerholz and Peter Eklund "Conceptual Structures: current research and practice" are both good accounts of the general applicability of Entity-Relation models.

[ii] *Design Patterns: Elements of reusable object-oriented software.* Erich Gamma et al. Addison-Wesley.

[iii] Ibid.

Xanalys Limited
20-24 Church Street
Altrincham, Cheshire
WA14 4DW
United Kingdom

telephone +44 161 941 7792
fax +44 161 332 8285

www.xanalys.com

Xanalys®
Let nothing escape you.