QuidProQuo

An Online Application Mapping International Actors to Their Relationships and Transactions

Entiro Systems Ltd.

White Paper

By Gerald Perkins

Gerald.Perkins@entiro.com
Contents

Section 1 .................................................................................................................. 3
Executive Summary ................................................................................................. 3
Section 2 .................................................................................................................. 4
Overview ................................................................................................................ 4
Section 3 .................................................................................................................. 6
Software Components ............................................................................................. 6
  Actor ....................................................................................................................... 6
  Relationship ......................................................................................................... 6
  Transaction ......................................................................................................... 7
  Event ..................................................................................................................... 8
Section 4 .................................................................................................................. 9
Implementation ....................................................................................................... 9
Section 5 .................................................................................................................. 10
Application Functionality ...................................................................................... 10
  User ..................................................................................................................... 10
  Data Entry .......................................................................................................... 10
  Query / Reporting ............................................................................................... 10
  Blog ..................................................................................................................... 11
Section 6 .................................................................................................................. 12
Data Loading .......................................................................................................... 12
Section 7 .................................................................................................................. 13
Proposed Deployment .............................................................................................. 13
  Complete Site Prototype .................................................................................... 13
  Alternative Media Involvement ......................................................................... 13
  Site Release ....................................................................................................... 13
Section 8 .................................................................................................................. 14
Conclusions ............................................................................................................ 14
Executive Summary

An obvious component of life is the trading of items between parties where an obligation is created when there is a disparity of value between the traded items. The party which has not been fully compensated in the trade expects at some point to have the trade made whole. Often a transaction involves an upfront payment which may go unpaid for a considerable period of time. Eventually, a balance is expected – quid pro quo.

The application described in this paper is based on a software framework which contains the necessary components to create a history of transactions which will assist in the understanding of the relationship between transaction partners. An analysis of the history and its components will assist in the understanding of historic transactions and the relationships involved.

A simple example:

Bill Smith is named a director of ABC Corporation (a dated relationship is created). Bill receives an annual fee as a director (a dated transaction exists). Bill is also on the board of XYZ Corporation (dated relationship). Tom Brown is also on the board of XYZ Corporation (dated relationship). Bill receives news from the latest ABC Corp board meeting that things don’t look good for the next quarter. Tom Brown sells ABC Corp short, making a significant profit (a transaction). Enough data exists to enable an analysis of these actions that indicates Tom and Bill may have a relationship. Researchers have a lead.

An additional element which can be useful for a more complete analysis is the event. While not obviously attached to specific relationships or transactions an event may trigger either of these actions.

The previous example demonstrates one of the many analyses that QuidProQuo can provide. The more data contained in the QuidProQuo database, the more extensive and far reaching analyses can be.
Overview

There are two fundamental requirements for the QuidProQuo web application. The first is to provide an interface for the entry of data which will enable the description of actor–relationship–transaction connections. The second, which is the purpose of QuidProQuo is application software to parse these connections, drawing a more complete picture than an individual can parse given the vast wealth of data.

Figure 1

General concepts employed by the QuidProQuo application software includes:

- Transactions will exist to create a balance between actors.
- A relationship always exists between actors party to a transaction.
- Typically, but not always, the transaction will have cash value.
• Actors can be individuals or organizations.

• An actor typically has relationships with a large number of other actors, eg an individual can be both a member of a church (organizational actor), married to another actor, etc.

• The time span of a relationship can be of any duration.

• A transaction occurs at a point in time, it has no duration.

• An event has a time span, though typically brief in comparison to a relationship. Examples of events are wars, floods, an interest rate hike, etc.

• Actors, relationships, transactions and events all have a required attribution component which links to data verifying the authenticity of the element. This authenticity component may be a news article, Wikipedia entry, PDF document, etc.
Software Components

The real world components of the QuidProQuo application are reflected in the software objects created to provide a valid and rigorous implementation. These objects are described in this section. As expected, the primary software objects are Actor, Relationship, Transaction and Event. Other objects, such as the Approval object, User object, etc. used for administration and management of site access are not described.

Actor

Object Relationships:
- Relationship – connection to another Actor
- Transaction – connection to another Actor
- Approval – object which describes validation of Actor attributes
- Attribution – object containing supporting information regarding this Actor
- Attributes – object containing a list of attributes to further describe this Actor

Attributes:
- uniqueID: unique ID of this actor
- name: actor name, must be unique (e.g. 'BC Govt')
- fullName: full name of the actor (e.g. 'Government of British Columbia')
- type: type of actor, looked up from restricted set of types
- typeDescription: description of the actor type, from lookup
- description: text description of the actor
- startDate: when the actor first comes into existence
- endDate: when the actor no longer exists
- gender: gender if Actor type = 'PERSON'
- akaID: if this is an alias or pseudonym for another actor, that actors uniqueID
- approval: approval object
- approvalLevel: approval level
- approvalType: description of the approval level
- attribution: attribution object for this actor
- attributes: attributes object for this actor

Relationship

Object Relationships:
- Actor – this object connects two Actors. In order to facilitate meaning to a description of this relationship, one Actor is defined as parent, the other as child.
- Approval – object which describes validation of Actor attributes
- Attribution – object containing supporting information regarding this Actor
Attributes:
- uniqueID: unique ID of this relationship
- parent: the actor key of that actor who provides value to the relationship
- parentName: the name of the parent (from class Actor)
- child: Actor key of that Actor who receives value from the relationship
- childName: the name of the child (from class Actor)
- startDate: date that the relationship was started
- endDate: date that the relationship ends (NULL if relationship ongoing)
- type: relationship type from the parent point of view
- typeDescription: description of the relationship type from the parent point of view
- strength: 0 – 100. Parent is Catholic Church & child is Pope Francis, value is 100
- approval: Approval object
- approvalType: description of the approval level
- attribution: description object for this actor

Transaction

Object Relationships:
- Relationship – the Transaction object connects two Actors who are described in the associated Relationship object.
- Approval – object which describes validation of Actor attributes
- Attribution – object containing supporting information regarding this Actor
- Attributes – object containing a list of attributes to further describe this Actor

Attributes:
- uniqueID: unique ID of this transaction
- relationshipKey: foreign key to relationship table to the relationship for this transaction
- date: date of this transaction
- type: type of transaction, lookup available in LU_TransactionType table
- typeDescription: from lookup
- description: textual description of the transaction
- textValue: text string containing the value of the transaction (may not be currency based)
- ascribedValue: documented value of the transaction
- actualValue: actual, “proven” value of this transaction
- currency: currency of ‘ascribedValue’ and ‘actualValue’
- profit: profit for relationship parent for this transaction in ‘currency’
- approval: Approval object
- approvalType: description of the approval level
- attribution: Attribution object for this actor
- attributes: Attributes object for this transaction
Event

Object Relationships:
- Approval – object which describes validation of Event attributes
- Attribution – object containing supporting information regarding this Event
- Attributes – object containing a list of attributes to further describe this Event

Attributes:
- uniqueID: unique ID of this transaction
- name: event name, non unique
- type: type of event, lookup available in LU_EventType table
- typeDescription: from lookup
- description: text description of the event
- startDate: date that the event was started
- endDate: date that the event ended (may be equal to startDate)
- approval: Approval object
- approvalType: description of the approval level
- attribution: Attribution object for this actor
- attributes: Attributes object for this transaction
Implementation

QuidProQuo is a web based application developed primarily with Open Source components. It is designed to run on Apache Server, programmed in a combination of PHP and JavaScript, using jQuery as an adjunct library for JavaScript functionality. The database backend has initially been deployed using MySQL although there is no issue easily implementing another Open Source database should this become a requirement.
Section 5

Application Functionality

QuidProQuo provides the standard functionality expected from a social media type website. Specific functionality is outlined in this section.

User

While anonymous users are encouraged to access the site, registered users are both a resource and useful advocates for QuidProQuo.

- Access
  - Anonymous
  - Data entry
  - Editor
  - Administrator
- Subscriber
  - Newsletters
  - Chat access

Data Entry

Without data entry by authorized users the entry of the required volume of data which will make QuidProQuo most useful would be impossible. QuidProQuo will use a Wikipedia like model with editors specializing in specific data sets, validating updates.

Data can be inserted and updated via forms interfaces. Bulk data loads will be custom built as bulk data becomes available. An XML model will be created for those able to supply bulk data in this format.

Query / Reporting

Via a forms interface, users will be able to query the QuidProQuo database in an ad hoc fashion. Results of the queries can be displayed as:

- Tabular reports
- Timelines
- Graphs
- Interactive map
Due to the richness of the data, queries can be used to uncover:

- **Relationships between actors which are not apparent**
  - Actor X and Y both have relationships with 3 other actors but not each other
  - Actor X on more than 1 occasion has a transaction with Actor Z after an event of type ‘ABC’
  - Why do some transactions have a disproportionately high or low value compared to others within the same type of relationship?
  - Detect relationships which have little or no transaction value. Can these predict future transactions or new relationships?

- **Transactions between actors which have non-apparent linkages**
  - Actor X repeatedly has a transaction with Actor Y after an event of type ‘XYZ’

- **Event analysis**
  - Does an event of type ‘X’ regularly trigger a specific transaction?
  - Does an event of type ‘X’ occur after the initiation of a specific transaction or relationship?

After the site has been used for a relatively short period of time, more queries and connections which are not initially obvious will be explored. The spark supplied by existing reports will ignite different querying strategies which currently are not obvious.

**Blog**

Site editors are encouraged to contribute to the QuidProQuo blog with their own tips for querying, search results or analysis. This will both encourage site traffic and the addition of users with the interest and expertise to enrich the data repository.
Data Loading

While forms interfaces exist to load data it will be invaluable to the strength of QuidProQuo to acquire bulk data from existing sources. It is important to raise awareness of the site in order to make QuidProQuo visible to third parties with access to bulk actor – relationship – transaction – event data. The acquisition of actor and event data will be somewhat straight-forward compared to relationship and transaction data which will require more difficult verification.

Initially, it is essential to make QuidProQuo visible to existing bloggers and other web based information disseminators.
Section 7

Proposed Deployment

Complete Site Prototype

- Expose the QuidProQuo site to select participants.
- Encourage involvement
- Solicit data entry assistance
- ‘Advertise’ within the alternative media community

Alternative Media Involvement

This is a key element. Existing alternative media viewers and subscribers are a rich source of data. Hopefully by enlisting involvement amongst this user community, data loading and site testing can be greatly facilitated.

Site Release

Go viral with data resources growing at a great pace and QuidProQuo being used by an exponentially growing worldwide user community.
Conclusions

The deployment of a website with a rich repository of data allowing the study of relationships and transactions between influential world actors will provide an enormously useful resource to the internet community. QuidProQuo currently has the software infrastructure which can make this proposed deployment available in a time frame only restricted by the speed with which data can be loaded to the QuidProQuo data repository.

Once the volume of data reaches “critical mass” the site will be increasingly useful for researchers as well as interested general users. The site will allow the researcher to uncover connections which currently can only be discovered by serendipity or by very focused research covering a limited number of actors.

This White Paper is for informational purposes only. ENTIRO SYSTEMS LTD. MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS WHITE PAPER. Entiro cannot be responsible for errors in typography or photography.

Information in this document is subject to change without notice.