DESIGN OF A GENERIC PERFORMANCE MEASUREMENT REPOSITORY IN INDUSTRY

Alain Abran **Edgardo Palza**

École de Technologie Supérieure - ETS
1100 Notre-Dame Ouest, H3C 1K3 Montréal Québec , Canada,
aabran@ele.etsmtl.ca edgardo.palza-vargas.1@ens.etsmtl.ca

ABSTRACT

Understanding, predicting, and controlling performance is a continuous challenge, and static measurement systems are inadequate in dynamic and rapidly changing business environments. In this paper, we propose a generic, flexible and integrated Measurement System Repository to handle continuously changing business conditions, and we report our experience in its design and development at Ericsson Research Canada.

This Performance Measurement Repository has been developed based on the concept of a data warehouse environment. Reporting features are based on the definition of queries to On Line Analytical Process (OLAP) cubes. OLAP cubes are created as materialized views of the measurement data, and the user functionalities are implemented as analytical drill-down/roll-up capabilities and as Indicator and Trend Analysis capabilities.

Keywords: software measures, metrics, measurement process, measurement repository

1. Introduction

Organizational performance measurement systems are designed and developed to improve understanding, planning and control of the productivity, effectiveness, quality and timeliness of projects and products. They must be based on shared views of the organization and include a Performance Measurement Repository which organizes and stores historical measurement data to be used for trend analysis and for monitoring both products and process improvement projects.

Today, organizations are competing in complex and dynamic environments. Frequent reengineering of business units and product lines often renders a significant amount of the critical historical data stored in predominantly static measurement systems obsolete. In today's ever-changing technical and business environments, there is a critical need for the redesign of measurement system repositories to give them the flexibility to be reconfigured continuously, while at the same time preserving the value of historical data initially organized according to outdated organizational structures.

This paper reports on the design and development of a generic and flexible Performance Measurement Repository to support a dynamic measurement system which, in turn, is capable of supporting Ericsson Research Canada's business information needs in a changing and dynamic environment.

It was expected that further criteria and constraints would be met for a better fit to the Ericsson context:

- Design of a coherent and consistent model of enterprise performance evaluation,
- An integrated and generic multidimensional measurement platform,
- Application of the integrated Performance Measurement System to every unit, department and project,
- Individual and team Performance measures aligned with organizational goals,
- Ability to permit managers to extract value from the vast amounts of data and information in the organization,
- Improvements to the quality of the software engineering measures themselves.

Section 2 presents the data model of the repository, section 3 the architecture, section 4 the design, and section 5 the operational features. Observations are presented in section 6.

2. Repository Data Model

Software engineering measurement definitions are still in a state of flux, and, compared to more mature measurement domains such as accounting, are still immature. A software engineering measurement repository must take this immaturity into account. Specifically, it cannot be taken for granted that software measures will remain stable over time, and so, to accommodate any such evolution in measurement definitions, a well thought out structure is required for the data model itself [1]. We have designed a structure which includes meta-objects to represent key measurement concepts the instances of which are subject to change over time: entity, relationship, attributes and series.

The design of the data model for this measurement repository is, of course, critical. The key concepts and definitions adopted in this model were strongly influenced by ISO standard 15939 [2]. Now, to meet the constraints of a dynamic business environment, the Performance Measurement Repository must be built on a generic database which is highly flexible. This means that the database structure must not presuppose any particular measures or any relationship between them, but rather treat the measures themselves as data. This also means that the Repository must support both base and derived measures, as defined in ISO 15939 [2]. This requires that the definitions of the measures, and of their relationships, be stored in the repository in a metadata entity [3]. The metadata entity can then provide the flexibility required by the ever-changing needs of the organization.

The class diagram designed for this measurement repository is illustrated in Figure 1. It contains the set of classes listed in Table 1, together with their general description and a specific reference to a detailed description in Appendix A.

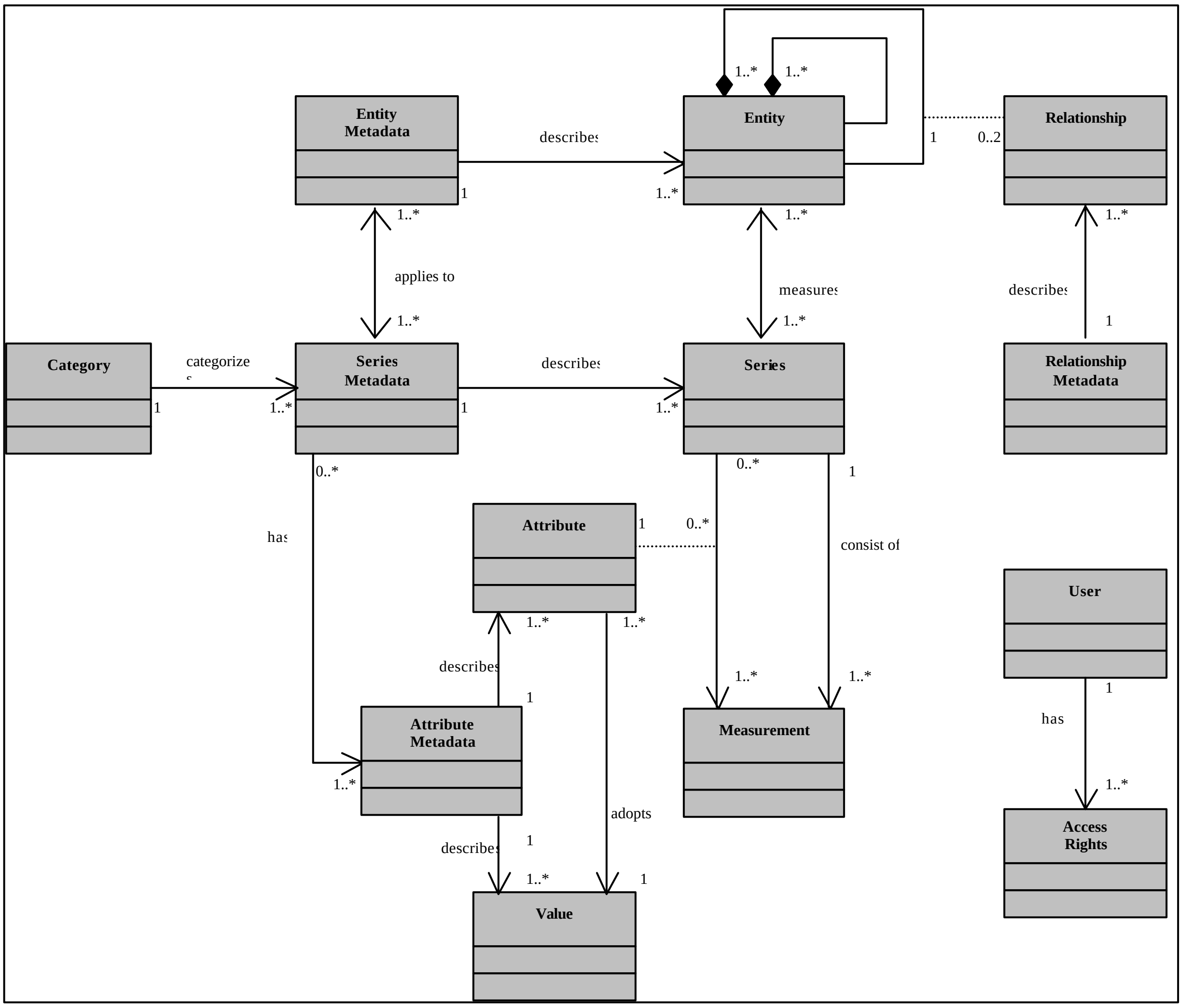

Figure 1: Class diagram of the Measurement Repository

The set of relationships among (measurement) entities is defined and stored as another object type in the repository in order to support both hierarchical and multidimensional views of data. This is made possible by taking advantage of the OLAP (On Line Analytical Process) services, such a drill-down/roll-up, for the measurements associated with a lower/upper level entity, and from an aggregated value to its atomic components.

Class / Association Name	Description	References in Appendix A
Entity Metadata	Instances of this class store all common data associated with a given entity type, i.e. Unit, Product, Project, Version.	Details in A.1.
Entity	Instances of this class Entity store all data associated with a specific Unit, Product, Project, or any other entity type defined by entity metadata.	Details in A.2
Relationship	This associative class is used to model arbitrary relationships between two entities. The nature of the relationship is given by the relationship metadata.	Details in A.3
Relationship Metadata	Instances of this class store the nature of the relationship.	Details in A.4
Series	Instances of this class are a chronologically ordered collection of measurements representing the value of a measure over time.	Details in A.5
Measures	This n:n relationship links a specific series to the objects or objects being measured by it.	Details in A.6
Measurement	Each instance of this class captures the value of a measurement, as well as the date on which it was taken.	Details in A.7
Series Metadata	This class describes the measures being captured by the series.	Details in A.8
Attribute	This associative class qualifies the measurements according to different attributes. For example, of the 5 TR's (Trouble Reports) recorded on October 7, 2002, three could be of severity "A", one of severity "B" and one of severity "C".	Details in A.9
Attribute Metadata	This class describes the attributes that classify the measurement in a series.	Details in A.10
Value	Instances of this class store the admissible values for a given attribute.	Details in A.11
Category	Describes the various categories, i.e. Quality, Progress, Cost, into which measures are categorized.	Details in A.12
Applies To	This n:n relationship defines the applicable set of measures for each object type.	Details in A.13
User	This class captures the user ids of those authorized to access the repository. Access to this table is restricted to the database administrator.	Details in A.14
Access Rights	Indicates the specific object and relationship instances to which a given user has access and what he or she can do with them, i.e. Create, Change, Delete. Access to this table is restricted to the database administrator.	Details in A.15

Table 1: List of Measurement classes/Associations

3. Repository Software Architecture

This section presents the architecture selected for the deployment of the functionalities of this Performance Measurement Repository (Figure 2). The measurement repository feeds an Analytical engine which then provides support to four functional capabilities:

- Management Indicators and Trends,
- Analytic and drill-down capabilities, based on responsibility levels,
- Analytic drill-down capabilities based on process,
- Administration and quality control.

Each of the key functional capabilities is described below.

- **The Indicator and Trend Analysis capabilities** present the information needed by Senior Management based on predefined reports and charts navigable in a Web-page style. A home page will display the names of the various reports available using an indented structure from which it will be possible to jump into the selected report. Use of these capabilities must be completely self-evident, so that no special training or knowledge is required on the part of the user.
- **The Analytic and drill-down/drill-up capabilities** are designed to support Middle Managers and Operations Development personnel with dynamic reports, Excel export capabilities and drill-down/drill-up functionality similar to that provided in on-line analytical applications. The power and flexibility of this interface will necessarily make it more complex to operate, and the user will require some very basic training to operate it.
- **The administration and quality control** interface allows the designated administrator to define new measures, grant privileges and audit the quality and timeliness of the data entered into the system.

- **The analytical engine (OLAP technology)** provides the capability to compute derived measures and aggregate them across multiple dimensions.

Of course, the Measurement Repository itself provides permanent storage for the measurements taken and the metadata necessary to administer them. The repository design also provides for two modes of collecting measurement data:

- **Automated**, via an Extraction, Transformation and Loading (ETL) tool for high volume/high periodicity measure ments, such as those collected through the time reporting system or the trouble reporting system.
- **Manual**, via a Web-page style data entry form, for low frequency/low volume data such as turnover rates and hourly rates.

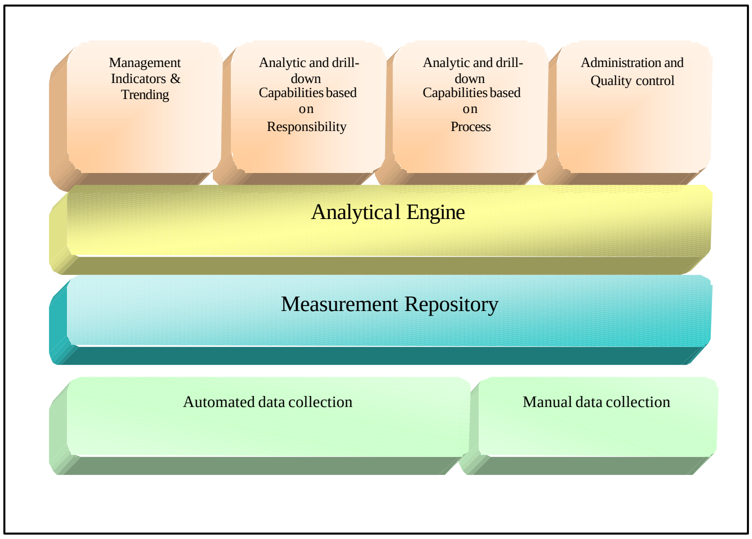

Figure 2: Performance Measurement System architecture

4. Repository Design - Overview

The On Line Analytical Process (OLAP) services play an important role in the Performance Measurement System Repository. In particular, they pull together data from multiple sources in the organization and store that data in a form convenient for further analysis and decision support [4]. These services permit the creating, querying and maintaining of OLAP cubes, which are materialized views of information, that is, a way of precomputing data summaries so that requests can be answered quickly [5]. Analytic and drill-down facilities provide users with the possibility of analyzing data at different levels of granularity. To provide the multidimensional feature, the OLAP pivoting cubes approach was selected for the dynamical display and rearrangement of multiple dimensions of data.

To provide data collection, as well as communication and a Performance Measurement education facility, a portal approach was selected for the Performance Measurement Repository, which provides a dual perspective of overall corporate performance information and individual performance information [6]. Figure 3 illustrates the component interaction between the Performance Measurement Repository and the Portal Server.

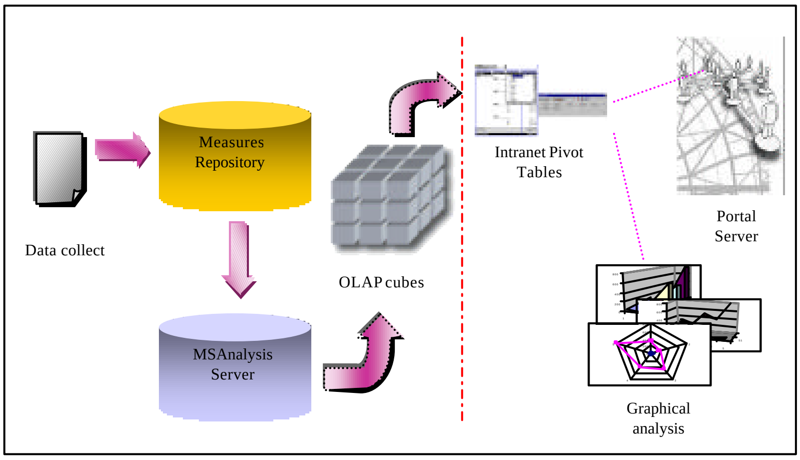

Figure 3: Repository Design overview

Of course, a security mechanism has to be designed and implemented in the Repository to prevent the accidental modification or destruction of data, as well as for audit mechanisms and for quality assurance purposes.

The repository must facilitate the mapping of the information needs of the organization to the indicators proposed to satisfy those Information Needs [7], [8]. The business indicators initially proposed were:

- Delivery according to commitments: Are we delivering according to our promises?
- Effectiveness & Efficiency: Are we delivering the right products at the right time at the promised cost?
- Financial: What is the cost of our operations? Are expenditures growing or declining? Are we meeting the goals of the efficiency program?
- Quality: Are our products satisfying our customers? Does our quality provide support for TL 9000 certification? [9], [10].
- Sustainability: What are we doing to sustain our growth? Are we improving our processes? Are we able to retain and develop our employees?
- Strategic goals: Can we monitor the specific goals set for the month/quarter/year?

In the design of this measurement repository, this will be referred to as the Indicator and Trend Analysis capabilities which provide managers with a comprehensive view of the organization.

5. Operational features

5.1 OLAP services

OLAP multidimensional capabilities are used for defining several components of the Measurement Repository, such as: Entities, Entity Metadata, Aggregations, Series, Series Metadata, Measurement, Measures, Attributes, Categories, Associations. The system architecture of the repository will only store base measurements. Derived measurements will be handled by the "Analytical Engine" (i.e. MS-Analysis Services). The Measurement Repository is capable of handling the following types of scales: Nominal, Ordinal, Interval and Ratio, according to ISO 15939 [2].

The Multidimensional Measurement Repository is constructed in a collection of multidimensional data cubes (i.e. OLAP cubes) containing the aggregation data on which multidimensional measurement analysis is based. Aggregations are precalculated summaries of measurement data which improve the efficiency and response time of user queries. OLAP multidimensional capabilities are used for defining several components of the Measurement Repository, such as: Entities, Entity Metadata, Aggregations, Series, Series Metadata, Measurement, Measures, Attributes, Categories, Associations. The dimensions of the OLAP cubes have hierarchies which specify aggregation levels, i.e. granularity for viewing data. OLAP technology provides for the graphical representation of multidimensional measures of the Measurement Repository. It provides an important function for determining why certain trends or patterns are occurring. The user is able to invoke sequences of OLAP operations (slice and dice, drill-down, roll-up and pivot) interactively starting from a cube.

5.2 Multidimensional data measure construction

The Multidimensional Measurement Repository is based on the concept of the star schema, which represents a multidimensional model consisting of a central fact table and several dimension tables. Fact tables contain records which represent measures (facts) to be analyzed. Every fact table references multiple dimension tables, each of which represents a dimension of interest, like a Unit, a Project, a Product, etc.

5.3 Measurement management & trends

OLAP technology provides for graphical representation of multidimensional measures of the Measurement Repository. It provides an important function for determining why certain trends or patterns are occurring. The user is able to invoke sequences of OLAP operations (slice and dice, drill-down, roll-up and pivot) interactively starting from a cube. The implementation at Ericsson includes visualization facilities for a specific multidimensional measurement cube with MS-Analysis Services.

5.4 Measure verification process

Data collected must be reliable and secure, and validation of the individual data is important for the integrity of the Repository. We expect, therefore, that the measures themselves will need to be revised, at periodic intervals.

6. Conclusion and next steps

In this paper, an approach was proposed for the design and development of an integrated, generic and flexible Measurement Repository. The Measurement Repository constructed is based on a multidimensional Measurement Meta-model concept. We proposed an object-oriented model to support the several measurement concepts of the Ericsson information requirements. The class diagram, design, the architecture and the design of this Measurement Repository were presented.

Data collection for the Measurement Repository is currently still predominently manual. A Web interface is being developed for collecting data manually from managers in the Ericsson Intranet environment. Data is collected directly to OLAP dimension tables. An interface for client/server databases and legacy systems is still being investigated.

Finally, in a dynamically changing business environment, the measures themselves will need to be revisited [11]:

- When new processes are added to Ericsson's process framework;
- When processes are revised and new products or process measures are needed;
- When finer granularity of data is required;
- When greater visibility into the process is required;
- When measures themselves are retired.

Acknowledgments
Repository Measurement was supported by the Enterprise Performance Unit of Ericsson Research Canada. We would like to express our particular thanks to Mr. Eduardo Miranda for his leadership and to Ahmed Bedhiaf for his valuable collaboration and participation in the realization of this project. The opinions expressed in this report are solely those of the authors and not that of Ericsson Research Canada

REFERENCES

1. Harrison, W., *A Universal Metrics Repository Project*. in International Internet & Software Quality Week Conferences, 2001.

2. ISO, ISO/IEC 15939: 2002 *Information Technology - Software Engineering - Software Measurement Process*., International Organization for Standardization, Geneva, 2002.

3. Walkerden, F., *A Design for a Software Metrics Repository*, Centre for Advanced Empirical Software Research, CAESAR Technical Report no. 96/07, University of New South Whales, Sydney (Australia), 1996.

4. Ralph Kimball, *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. John Wiley & Sons, 2002.

5. Olap Train, *Microsoft(r) SQL Server(tm) 2000 Analysis Services Step by Step*, Microsoft Press, 2000.

6. Robert Ferguson, *Special Edition Using Microsoft SharePoint Portal Server,* R.F., ed.. Québec (Canada), 2002.

7. McGarry, J., *Practical software measurement : objective information for decision maker,* Addison-Wesley, Boston, xvii, 2002, p.p 277.

8. Drake, J., *User Functional Description for the Software Metric Management Information System II,* in Army Software Metrics Newsletter, Alexandria, VA, Spring 1996.

9. Quality Excellence for Suppliers of Telecommunications (QUEST) Forum, *TL 9000 Quality System Measurements - Handbook, Release 3.0,* QUEST Forum editor, 2001.

10. Quality Excellence for Suppliers of Telecommunications Forum, *TL 9000 Quality System Requirements - Handbook, Release 3.0*, QUEST Forum editor, 2001.

11. Software Engineering Institute, *CMMI for Software Engineering (CMMI-SW, V1.1), Staged Representation,* Carnegie Mellon University, Software Engineering Institute: Pittsburgh, 2002.

Appendix A

A.1 Entity Metadata

Attributes	Description
Entity Type	Primary key
Description	Description (i.e. Project, Unit, Product, etc.) corresponding to an entity of type Entity Type
Security Key	Value used to grant access to a particular user or group of users

Table A.1: Entity Types table

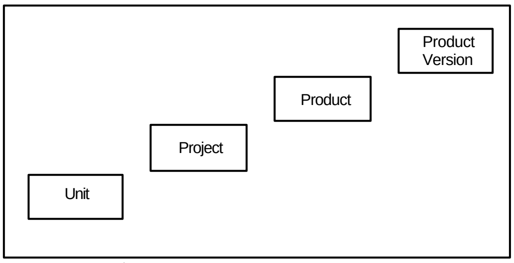

Figure A.1: Entity Types

A.2 Entity

Attributes	Description
Entity id	Primary key
Entity Type	Link to entity metadata. Implements the relationship "describes"
Entity Name	Name of the entity being measured
Entity Parent	Link to the parent entity. Implements the "aggregates" relationship
Security Key	Value used to grant access to a particular user or group of users

Table A.2: Entity instances table

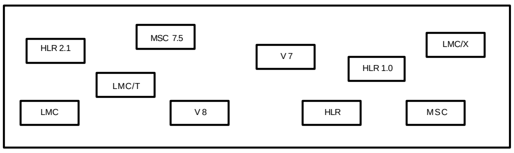

Figure A.2.1: Entity instances

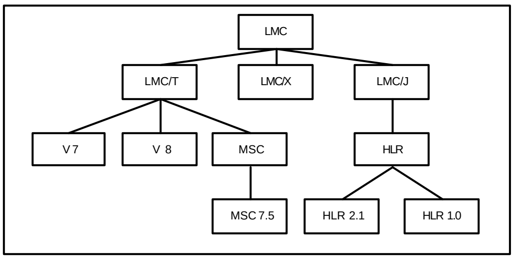

Figure A.2.2: Hierarchy among entity instances supported by the aggregates relationship

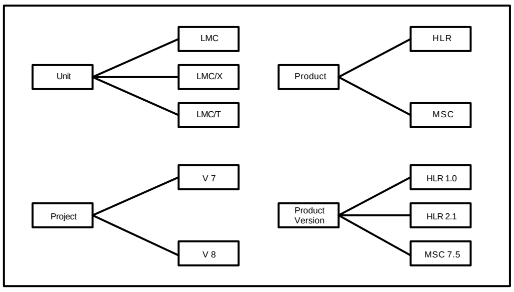

Figure A2.3: Relationship between entity types and entity instances

A.3 Relationship

Attributes	Description
Siblings id	Primary key
Entity Name X	Link to one of the entities participating in the relation
Entity Name Y	Link to one of the entities participating in the relation
Relationship Type	Points to the table describing the nature of the relation. Implements the relationship "describes"
Security Key	Value used to grant access to a particular user or group of users

Table A.3: Relationship between siblings

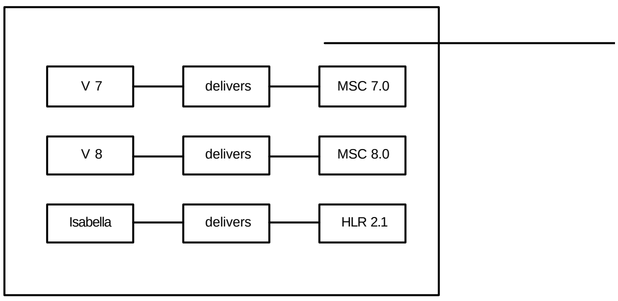

Figure A3: Relationship between siblings

A.4 Relationship Metadata

Attributes	Description
Relationship id	Primary key
Description	Defines the type of relationship
Security Key	Value used to grant access to a particular user or group of users

Table A.4: Relationship between siblings

A.5 Series

Attributes	Description
Series id	Primary key
Series Type	Defines the measure
Security Key	Value used to grant access to a particular user or group of users

Table A5: Series table

A.6 Measures

Attributes	Description
Series id	Link to series. Primary key
Entity name	Link to entity being measured
Security Key	Value used to grant access to a particular user or group of users

Table A6: measures relationship

A.7 Measurement

Attributes	Description
Measurement id	Primary key
Series id	Identifies the series to which the measurement belongs; implements the relationship "consists of"
Time Stamp	Date at which the measurement was made
Value	Value of the measurement supports ordinal, interval and ratio scales; in the case of nominal scales the value is 1
Date of First Entry	Date when the measurement was first entered into the repository
Date of Last Entry	Date when the measurement was last entered, i.e. modified
Version Number	Number of times the value was changed
Security Key	Value used to grant access to a particular user or group of users

Table A.7: Measurement table

A.8 Series Metadata

Attributes	Description
Series Type	Primary key
Category Type	Points to the category to which the measure belongs. Implements the relationship "categorized"
Series Description	Describes the measure captured by series. i.e. Open TR, BCWP, etc.
Security Key	Value used to grant access to a particular user or group of users

Table A.8: Series Metadata table

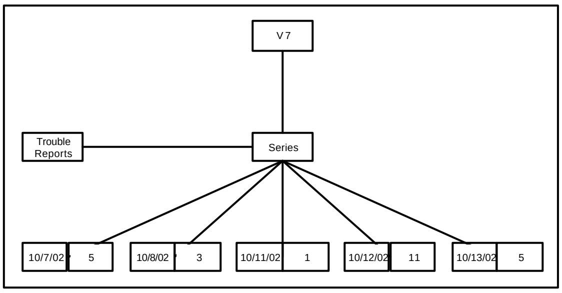

Figure A.8: Chronologically ordered measurements

A.9 Attribute

Attributes	Description
Attribute id	Primary key
Series id	Link to the series to which the value being qualified belongs
Measurement id	Points to the measurement being qualified
Value id	Points to the value the attribute adopts for a particular measurement. Implements the relationship "adopts"
Security Key	Value used to grant access to a particular user or group of users

Table A.9: Attribute table

A.10 Attribute Metadata

Attributes	Description
Attribute id	Primary key
Description	Defines the measure
Security Key	Value used to grant access to a particular user or group of users

Table A.10: Attribute Metadata table

A.11 Value

Attributes	Description
Value id	Primary key
Attribute id	Points to the table describing the nature of the relation. Implements the relationship "describes"
Security Key	Value used to grant access to a particular user or group of users

Table A.11: Value table

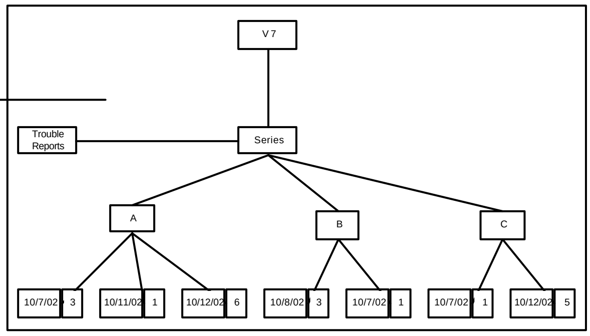

Figure A.11: Measurement qualification

A.12 Category

Attributes	Description
Category id	Primary key
Category Description	Describes the category
Security Key	Value used to grant access to a particular user or group of users

Table A.12: Category table

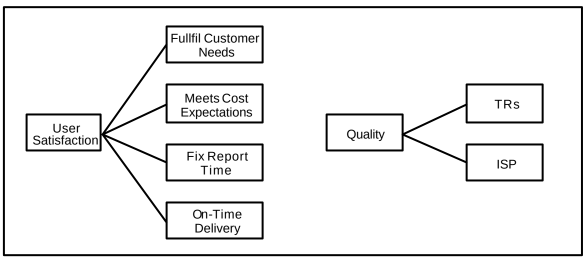

Figure A.12: Categories

A.13 Applies To

Attributes	Description
Entity id	Points to the entity metadata. Primary key
Series Type	Points to the series metadata. Primary key
Security Key	Value used to grant access to a particular user or group of users

Table A.13: Applies To relationship

A.14 User

Attributes	Description
User Number	Primary key
User Id	Identifies the user
Organization	Organization to which the user belongs
Security Key	Value used to grant access to a particular user or group of users

Table A.14: User table

A.15 Access Rights

Attributes	Description
Access id	Primary key
Table Id	Identifies the database table to which a right applies
Privileges	Permissions granted

Table A.15: Access Rights table

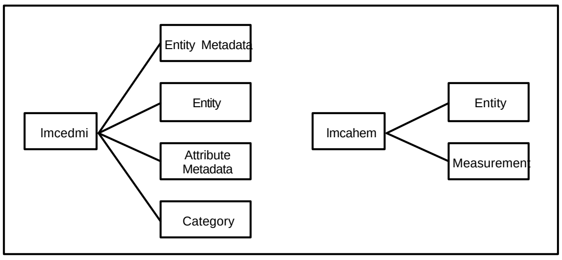

Figure A.15: Security mechanisms