CATALYST
CONSULTING GROUP, INC.

Deliverable: 1.9 VoteCal Requirements Traceability Matrix and Gap Analysis Plan

VoteCal Statewide Voter Registration System Project

State of California, Secretary of State (SOS)

Vote Cal

October 20, 2009

Version: 2.1

Work Product Acceptance Form

Catalyst Consulting Group is pleased to present the following VoteCal Project work product/deliverable. This work product is now complete and is ready for the Secretary of State to review and approve.

Work Product: Deliverable 1.9 VoteCal Requirements Traceability and Gap Analysis Plan

SOW Reference #: Attachment 1 Statement of Work

Exhibit 2 VoteCal System Task and Deliverables

Delivery Date: October 20, 2009

Secretary of State

By: _Mary Winkley_

Date: _10/21/09_

Name: Mary Winkley (SOS Project Director)

Authors

This document was prepared by:

Contributors		
Don Westfall Catalyst Consulting Group 211 West Wacker Drive Suite 450 Chicago, IL 60606	Kurt Schwartz Catalyst Consulting Group 211 West Wacker Drive Suite 450 Chicago, IL 60606	Matthew Benton Catalyst Consulting Group 211 West Wacker Drive Suite 450 Chicago, IL 60606

Version Control

Date	Version	Document Revision Description	Revision Author
09/21/2009	0.1	Revised based on project planning phase and initial SOS feedback	Matthew Benton
09/22/2009	1.0	Submitted to SOS for Review	Matthew Benton
10/05/2009	*2.0	Revisions (*Reversioned)	Don Westfall
10/07/2009	*3.0	Submitted for Approval (*Reversioned)	Don Westfall
10/20/2009	*3.1	Updates from SOS Review (*Reversioned)	Don Westfall
10/20/2009	2.1	Submitted for Approval	Don Westfall

Table of Contents

1. Introduction

This document is Deliverable 1.9 VoteCal Requirements Traceability Matrix and Gap Analysis Plan. It has been developed to the specifications presented in Deliverable Expectation Document (DED) 1.9, VoteCal Requirements Traceability Matrix and Gap Analysis Plan and as reviewed by the Secretary of State (SOS).

Throughout the remainder of this document, the Requirements Traceability Matrix and Gap Analysis Plan will be referenced as the Requirements Management Plan (RMP), maintaining consistency with the industry standards cited in Section 1.3 of this document.

The following table presents list of some commonly used terms to provide a consistent definition of terms used throughout this document.

This Requirements Management Plan describes the processes involved in the identification of requirements for the VoteCal project and documentation of those requirements in a requirements management system that allows bidirectional traceability of the requirements and to manage changes to the requirements. The Requirements Management Plan also contains a process for obtaining commitment to the requirements and identifying inconsistencies between the requirements and the project work.

1.1 Purpose and Objectives

The Requirements Management Plan describes the process of identifying, organizing, documenting, and tracking the requirements of the VoteCal System – forward and backward – throughout the VoteCal project lifecycle.

In the context of a large-scale system implementation, such as VoteCal, the requirements management process establishes a common understanding among all of the implementation participants as to the functional, technical, operational, and administrative needs, addressed and supported during the execution of the VoteCal Project.

1.2 Scope

The Requirements Management Plan focuses on *requirements management* processes as opposed to *requirements development* processes. The VoteCal requirements have been identified and articulated in the VoteCal RFP. Thus, requirements development is not a process that is necessary for the VoteCal Project. This plan will focus on management of those requirements throughout the VoteCal Project lifecycle.

The plan also includes a discussion of the content and development of a Requirements Traceability Matrix and a discussion of how the requirements will be tracked to the functional specifications, design documentation, programming tasks, and test scenarios during all project phases. The plan also describes how the Matrix will allow for tracing the test scenarios to validated requirements during the acceptance testing phase.

1.3 Applicable Standards

The requirements management processes described in the RTGAP and implemented throughout the VoteCal project are based on the following industry standards.

- Institute of Electrical and Electronics Engineers (IEEE) Standard 1233-1998 – Guide for Developing System Requirements Specifications
- The Institute of Electrical and Electronics Engineers (IEEE) Standard 830-1998 – Recommended Practice for Software Requirements Specifications
- Capability Maturity Model Integration for Development Version 1.2 (CMMI 1.2) – Requirements Development Process Area
- Capability Maturity Model Integration for Development Version 1.2 (CMMI 1.2) – Requirements Management Process Area

The SOS has adopted the state's (previously Department of Finance's) Project Management Methodology as its standard, as was described in Section 200 of the Statewide Information Management Manual (SIMM) in March 2006 when the project was approved. The methodology also reflects industry-standard processes described in the Project Management Body of Knowledge (PMBOK).

1.4   Assumptions, Dependencies, and Constraints

The Requirements Traceability Matrix and Gap Analysis deliverable is based on the following assumptions, dependencies, or constraints:

- The full set of requirements for the VoteCal System has been developed and was defined in Section VI (Project Management, Business, and Technical Requirements) of the VoteCal Request for Proposals (RFP). Therefore requirements development is not required.
- The SOS Project Team and the Catalyst Functional Team will conduct a Requirements Validation session to confirm the requirements defined in Section VI and to affirm Catalyst understanding of the requirements (see Section 3.1 of this document).
- The VoteCal System is a custom-developed solution for the California voter registration system and not a configuration of a packaged solution. Therefore, no gap analysis between the functions and features of a packaged solution and the VoteCal requirements is necessary.

1.5   Document Control

Requirements Management is a dynamic process that occurs throughout a project's life cycle. Accordingly, at a minimum, the requirements management process will be reviewed at the end of each project phase, and the Requirements Management Plan will be updated as required.

This document contains a revision history log. When changes occur, the version number will be incremented and the date, name of the person authoring the change, and a description of the change will be recorded in the revision history log of the document.

As with other work products of the VoteCal project, the approved Requirements Management Plan will be placed under configuration management in accordance with the Document Management Plan (a subset of the Project Management Plan). Also, in accordance with the Document Management Plan, the Requirements Management Plan will be stored on the SharePoint server and available to the SPS Project Team, the Independent Project Oversight Consultant (IPOC), Independent Verification and Validation (IV&V) vendor, and SOS senior management.

2. Roles and Responsibilities

The following roles and responsibilities have been identified for the requirements management process.

2.1 Catalyst Requirements Manager

The Catalyst Requirements Manager will have the responsibility to maintain requirements traceability throughout the VoteCal Project lifecycle. The Requirements Manager will have the responsibility for maintaining the integrity of the Requirements Traceability tool (RequisitePro), including entering the requirements, and updating the tool with traceability information, including design, development, and testing artifacts.

2.2 Catalyst Functional Lead

The Catalyst Functional Lead will have the responsibility to lead the Requirements Validation session to clarify the requirements with SOS and to prepare the Requirements Validation Document. Subsequently the Catalyst Functional Lead will have the responsibility to translate the validated requirements to VoteCal System functions and features.

2.3 SOS Project Team

The SOS Project Team will have the responsibility to participate in the Requirements Validation Session and to clarify the requirements as defined in Section VI of the VoteCal System RFP. The Project Team will have the responsibility to review and recommend approval of the Requirements Validation Document in accordance with the Deliverable Review and Approval Process agreed to by SOS and Catalyst (see the Project Management Plan and Schedule) for the details of that process.

2.4 SOS Project Director

The SOS Project Director will have the responsibility to approve the Requirements Validation Document in accordance with the Deliverable Review and Approval Process.

3. Approach

The requirements management function processes include the following five processes:

- Obtain an understanding of the VoteCal requirements
- Obtain a commitment to the requirements
- Maintain bidirectional traceability of the requirements
- Manage changes to the requirements

3.1 Obtain an Understanding of the Requirements

The first step in requirements management is to obtain an understanding of the requirements. For the VoteCal project this understanding will come in the form of a Requirements Validation session between Catalyst and the SOS Project Team. The purpose of the Requirements Validation sessions will be to clarify the requirements and affirm a common understanding of the requirements among all project participants.

In a project like VoteCal where the requirements are pre-defined, the overall process to clarify the requirements and affirm a common understanding of the requirements is called *requirements validation*.

3.1.1 Prepare for VoteCal Requirements Validation

The starting point for validating the VoteCal system requirements is the preparation for the requirements validation process. The process is initiated by configuring the RequisitePro tool to support the Requirements Management Process. This is followed by entering approximately 500 requirements, including the business and technical requirements, as described in the VoteCal RFP into the RequisitePro tool. The entry of requirements into the RequisitePro tool is then validated by the Catalyst team members to ensure that all the RFP requirements have been correctly entered. Other activities executed during this process include identifying all the participants who will be attending the requirements validation sessions and generating the requirement lists from the RequisitePro tool.

3.1.2 Conduct Requirements Validation Sessions

During this step, the Catalyst Project Team members will conduct a series of requirements validation session meetings with the SOS Project Team members, and other individuals deemed appropriate by the SOS. These requirement validation session meetings will give the Catalyst Project Team members an opportunity to further clarify the VoteCal RFP requirements. During these meetings, any ambiguities and/or redundancies in the RFP requirements will be discussed to be sure that the Catalyst team members have the same understanding of the requirements as the SOS Project Team members.

The intent of the requirements validation meeting is to clarify the requirements and confirm that the Project Team has a common understanding of them. The intent is not to eliminate existing requirements or to create new ones. In the event that SOS needs have changed and a requirement is considered no longer necessary, it will remain in the RequisitePro repository but be tagged with an attribute.

The requirements validation effort will result in a set of clarified requirements that will drive the total scope of the VoteCal System.

3.1.3 Define Hierarchical Relationships

During this process, the Catalyst PMO team will set up a hierarchical or "parent-child" structure for the confirmed RFP Requirements. A "child" requirement is a more specific expression of the parent requirement. Parent-child relationships are helpful since top-level or a "parent" view is used in developing business rules and scenarios in functional specifications, while the "child" view is used so that the teams understand the detail behind the requirement during the testing and implementation of the application. The naming convention used for parent-child relationships in RequisitePro is a common "root" number, which is the number of the parent, and a unique identifier for each child (e.g., a parent requirement would be numbered FU 300 and its two children would be numbered FU 300.1 and FU 300.2). This approach allows for the requirements traceability matrix to include all sub-requirements in a logical, organized manner.

3.2 Obtain Commitment to the Requirements

With a clear and common understanding of the requirements by both SOS and Catalyst, the project participants should make a commitment to those requirements. This commitment is made through the development of a *Requirements Validation Document* by Catalyst which is then reviewed and approved

through the deliverable review and approval process. Upon approval of the Requirements Validation Document, the Catalyst Requirements Manager will establish the baseline.

3.2.1 Develop the Requirements Validation Document

Catalyst will conduct one or more requirements validation sessions with the SOS Project Team to confirm a common understanding of the requirements. To avoid the need to go through each of the 300 or more requirements, Catalyst is conducting a self-review of the requirements prior to the requirements validation sessions so that the sessions can focus only on those requirements that Catalyst feels need clarification.

For each requirement addressed, the requirements validation session(s) will consist of a presentation of the requirement, a presentation of Catalyst's understanding of the requirement, and a discussion period.

Based on the discussions from the requirements validations sessions, Catalyst will prepare and submit a Deliverable Expectation Document to identify the content and acceptance criteria for the Requirements Validation Document. The Requirements Validation Document will be a re-iteration of the requirements from the VoteCal RFP with any clarifying language presented as discussed in the requirements validation sessions.

3.2.2 Approve the Requirements Validation Document

Catalyst will submit the Requirements Validation Document in draft form for review by the SOS Project Team. Based on comments from the Project Team, Catalyst will finalize the Requirements Validation Document and submit it for SOS approval.

3.2.3 Establish Baseline

Once the Requirements Validation Document has been approved as described above, Catalyst will document the approved requirements as the requirements baseline. Requirements with a baseline will be identified using an attribute in RequisitePro.

The requirements will be documented in RequisitePro prior to the requirements validation sessions. The description entered for each requirement will be the text from the RFP. On completion of the requirements validation sessions, any clarification of the requirement will be entered through the *Discussions* feature of RequisitePro. Any change to the actual wording of the requirement must be approved by the SOS Project Team through the agreed upon review and approval process.

3.3 Maintain Bidirectional Traceability of Requirements

As the work progresses, the VoteCal System will be designed and developed through a progressive elaboration process. The baseline requirements will be translated into VoteCal functions and features which will form the basis of the VoteCal system design, which will be used to develop the VoteCal system and form the basis for testing the VoteCal system. At each step, the Project Team must be able to determine the relationship between the requirement and implementation of that requirement into design, implementation or testing artifacts. Similarly, the Project Team must be able to trace the design, implementation or testing artifact back to the requirement or set of requirements that it fulfills.

3.4 Manage Changes to Requirements

As needs change and as work proceeds, additional requirements may be identified and changes to existing requirements may be identified. Any changes to requirements in the baseline must be approved through the change control process (see the VoteCal System Change Control Plan for details on the change control process).

4. Maintain Consistency with the Requirements

The Catalyst Project Team will use the RequisitePro tool to manage the consistency of the design, development, and testing artifacts with the VoteCal requirements throughout the VoteCal Project lifecycle. The tool serves as a single location to capture significant detail related to the requirements and design, and the tool promotes communication and collaboration among team members and reduces project risk.

The robust architecture of the RequisitePro tool will allow the maintenance of the requirements and design and test artifacts that are dynamically linked to a repository database within the tool for powerful sort and query capabilities. This will allow the Catalyst Project Team to easily trace relationships between the VoteCal Project requirements and design and test artifacts and track changes that affect them. The tool's robust traceability features visually indicate how changes affect the project, thereby giving the Catalyst Project Team the ability to perform real-time impact analysis and allowing them to make informed decisions for scope management. These traceability features, for example, allow the Catalyst Requirements Manager to identify requirements that have not been traced to design or testing artifacts.

As a result, the Catalyst Project Team members will be better able to maintain consistency between the requirements and the design. The RequisitePro tool will also capture the change history for each design detail, thereby providing an audit of the VoteCal system evolution.

The Catalyst Quality Manager and the Catalyst Requirements Manager share joint responsibility to identify inconsistencies between the requirements and the design artifacts and to escalate the issue through creation of an action item that must be acted upon by the Catalyst Project Team.

The SOS Project Team and the IV&V team will be provided read only access to the RequisitePro tool to verify that the system and software requirements are correct, complete, traceable and testable. The tool will allow assessing the traceability of the system requirements according to design, code, test, and training documentation.

4.1 Organization of the VoteCal Project Requirements, Design, and Other Detail

Using RequisitePro, VoteCal System requirements, design specification, and other detail will be organized in "packages" that contain related information. Packages allow for logical groupings and classification of important data related to the system. Each package contains artifacts that will be organized in the following order: documents (alphabetically by name), view (by type and alphabetically within the type), and requirements (by type and then by tag). Potential packages or groupings may include the following:

- VoteCal General Requirements
- VoteCal Functional Requirements

- VoteCal Technical Requirements

- External Interface Requirements

- Functional Requirement Specifications

- Test Case Scenarios

- Obsolete Requirements

- Views

- General Documentation

- Supplementary Requirements

- Business Rules

- Design Documents

- User Documentation

The SOS Project Team and the Catalyst Project Team will work together to create the final set of groupings.

Requirements, their attributes, and their relationships with other requirements will be displayed and managed in views. Through these attributes, the VoteCal Project Team can trace the allocation of requirements to VoteCal system components through the design documents through to system documentation.

The RequisitePro tool includes query functions for filtering and sorting the requirements and their attributes in views. All view commands are located in the menu bar. Toolbar buttons are also available for quick access to commands. A view will present information about requirements in a table (matrix) or in an outline tree. The Catalyst Project Team members will create views to display requirement attributes, such as status and priority, or to show the relationships between requirements. RequisitePro will allow creation of the following two types of views:

- Attribute Matrix
- Traceability Matrix

4.1.1   Attribute Matrix

The Attribute Matrix view displayed in Figure 4-1 will allow the display of all requirements and design specifications of a specified type. The requirements/specifications will be listed in rows, and their attributes will appear in columns. This view will assist the SOS and Catalyst Project Team members to organize and prioritize requirements/specifications; for example, the tool will allow the Project Team members to sort based on priority or risk.

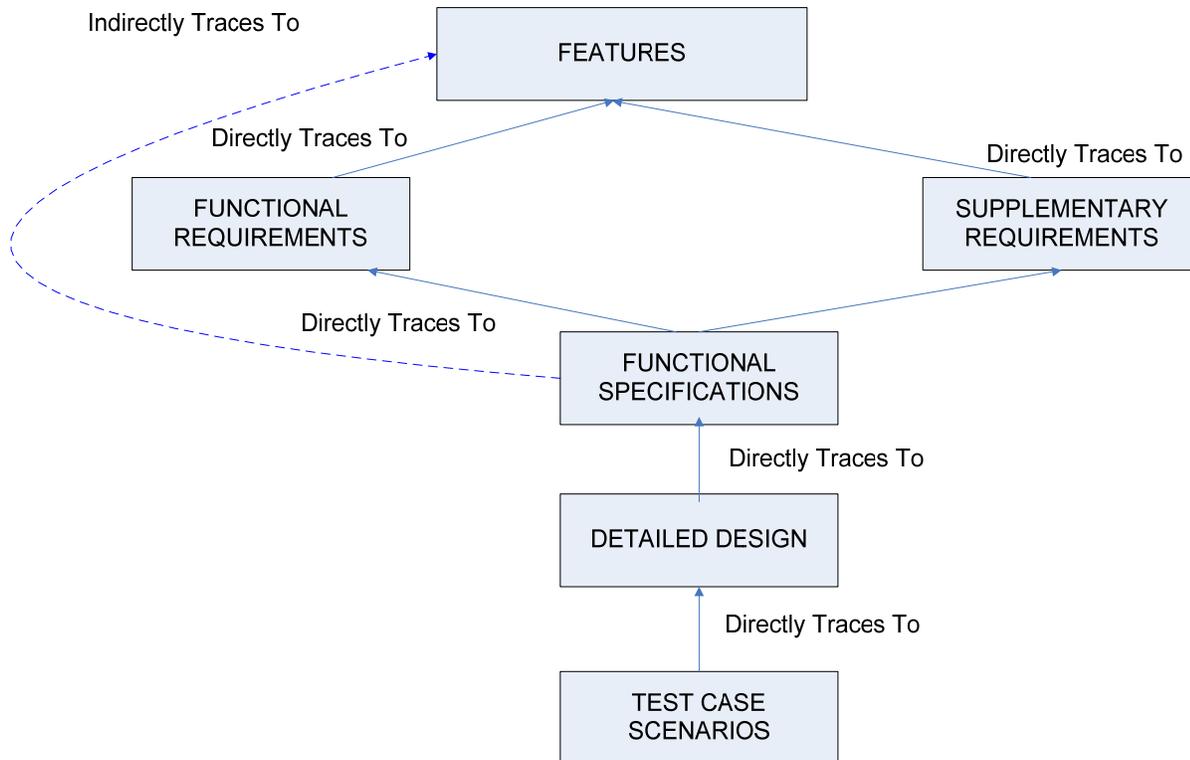**Figure 4-1 – Displaying Requirements of All Types**
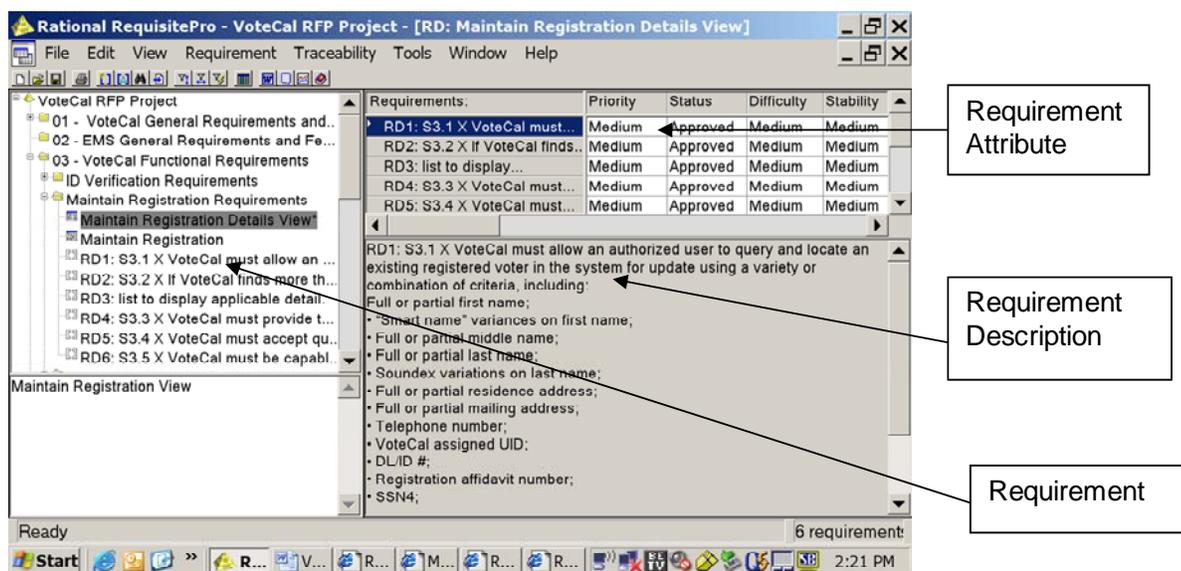
4.1.2 Traceability Matrix

A Traceability Matrix view reflects the relationships between two different types of requirements. A traceability matrix view will be used to create, modify, and delete traceability relationships so that requirements can be traced throughout the development lifecycle (from the request for proposals through deployment). This will provide an efficient mechanism to check requirements coverage and identify gaps in requirements coverage earlier in the development life cycle where it is easier to make adjustments and corrections and perform an impact analysis.

The Traceability Matrix view will allow the display of both direct and indirect traceability relationships between two types of requirements or requirements of the same type as indicated in Figure 4-2. A traceability relationship is direct when it traces from one requirement to another. A traceability relationship is indirect when a requirement traces to an intermediate requirement, which in turn traces to another requirement. The Traceability Matrix view will be used to create, modify, and delete traceability relationships. The Traceability Matrix view will also be used to filter and sort the row requirements and column requirements

Figure 4-2 – Viewing Direct and Indirect Relationships

```
Indirectly Traces To
                                    ┌──────────────────┐
                                    │     FEATURES     │
                                    └──────────────────┘
          Directly Traces To                                 Directly Traces To

   ┌──────────────────┐                              ┌──────────────────────┐
   │   FUNCTIONAL      │                              │    SUPPLEMENTARY     │
   │  REQUIREMENTS     │                              │    REQUIREMENTS      │
   └──────────────────┘                              └──────────────────────┘

          Directly Traces To
                           ┌──────────────────────┐
                           │      FUNCTIONAL       │
                           │    SPECIFICATIONS     │
                           └──────────────────────┘

                              Directly Traces To

                           ┌──────────────────────┐
                           │    DETAILED DESIGN    │
                           └──────────────────────┘

                              Directly Traces To

                           ┌──────────────────────┐
                           │      TEST CASE        │
                           │      SCENARIOS        │
                           └──────────────────────┘
```

When the arrow ⬆ appears in a cell (the intersection between a column and a row), it means that a requirement listed in that column can be traced to the requirement listed in that row. The direction of the arrow indicates whether a requirement is traced to or traced from another requirement. Traceability Matrix views will be created in the RequisitePro tool to trace each baseline requirement through to the functional specifications, design documents, object components, and test scenarios. RequisitePro's traceability feature will help assure the quality and completeness of the VoteCal system. The RequisitePro tool will allow linking abstract requirements, such as product features, to requirements such as hardware and software functional specifications. The tool will allow tracing requirements from the functional specifications to the actual test scenarios. Once the test scenario is successfully executed, all the requirements linked to the test scenario will have deemed to have been successfully implemented in the VoteCal system for system acceptance purposes.

The Traceability Matrix view as illustrated below in Figure 4-3 displays requirements and design specifications mapped to test case scenarios. For example, in the illustration, requirements S3.1 and S3.3 are mapped to test case TC1.

Figure 4-3 Example of Test Case Scenarios Mapped to Requirements
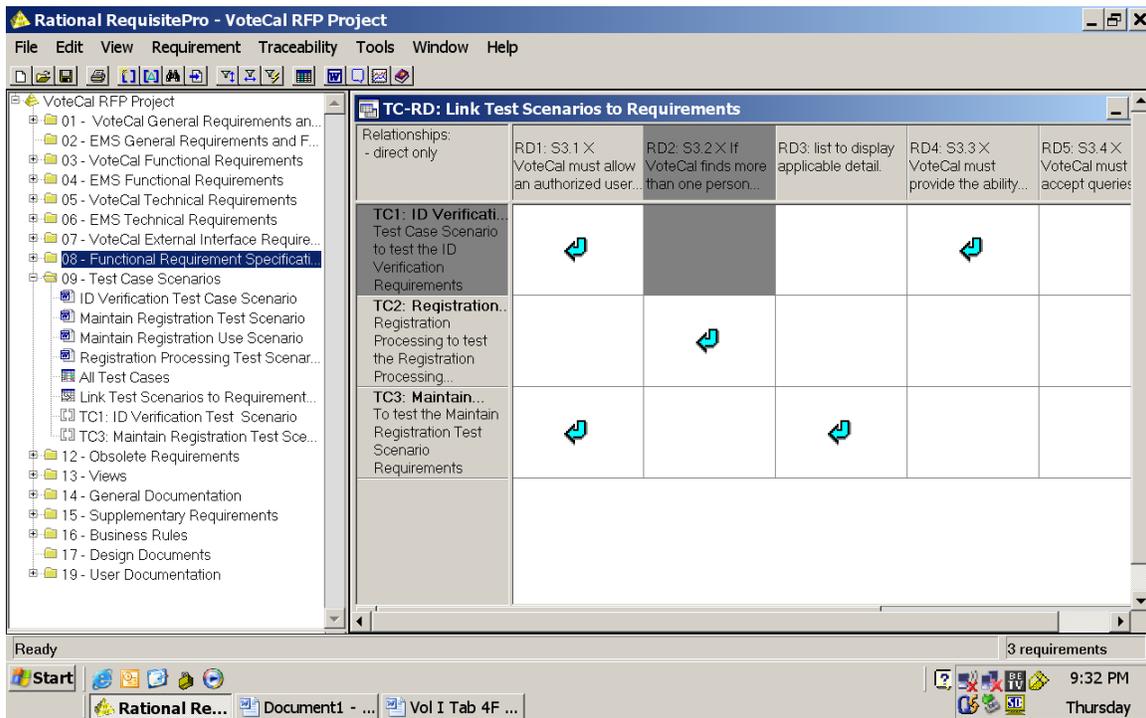


4.2 Data Elements and Attributes

As each requirement and design specification is documented, the Catalyst Project Team will capture a variety of information. Requirement and/or design specification data elements and attributes will be documented into the appropriate location within the RequisitePro tool. The attributes are completely configurable – attributes can be eliminated or added; and the values can be changed. The Catalyst Project Team will work with the SOS Project Team to identify the attributes to be used, the definition of the attribute, and the values associated with the attribute.

Some of the default attributes are illustrated in Figure 4-4 and described as follows:

- **Description** – Under the general tab, provides the RFP description of the requirement and provides additional definition about the requirement or specification derived from the requirements validation sessions.

- **Priority** - Identifies the importance of the requirement/specification in relation to others. Since all VoteCal requirements are of equal importance, this attribute would not likely be used.

- **Status.** Provides a means to categorize the current state of the requirement/specification, and to differentiate those that have been allocated to a design artifact, or those that have been implemented, versus those which are pending.

- **Difficulty.** Describes the level of effort related to meet the requirement/specification. Since this attribute is often associated with priority, it is another of the default attributes that would not be used.

- **Revision.** Serves as a reference point for a requirement/specification by identifying the history of the requirement/specification, and the number of times it has been modified or revised.

Figure 4-4 Example of Requirements Attributes
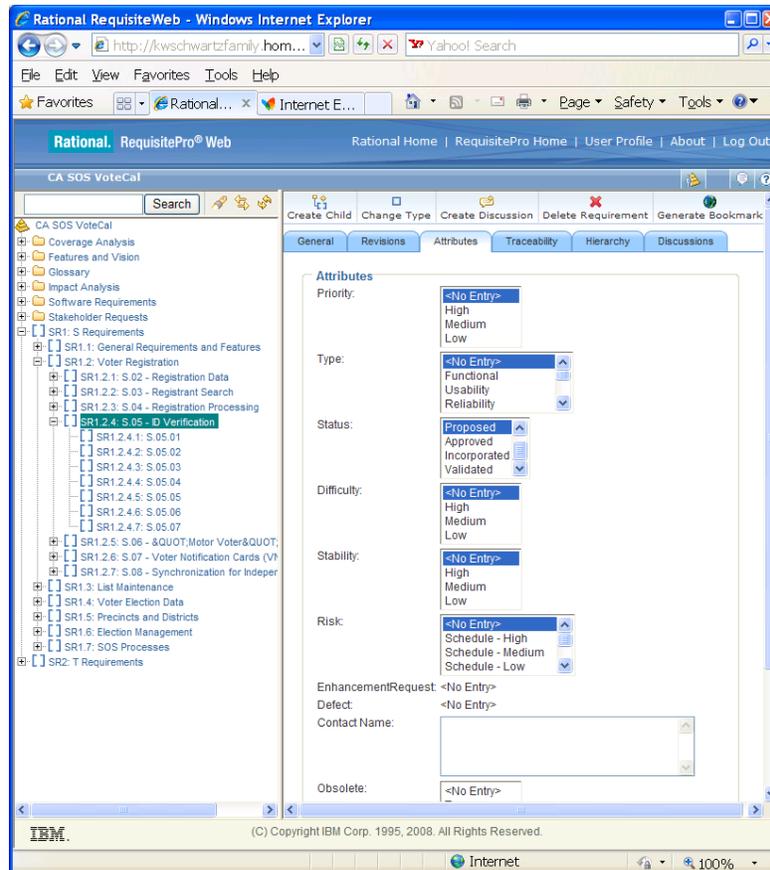
Again it is important to note that additional attributes and associated values can be defined by the SOS Project Team to assist in the traceability process.

4.3 Update Frequency

A key component of managing the requirements and design specifications includes regularly scheduled and ad hoc updates utilizing the requirements management tool (RequisitePro). As discussed in Section 3.0 Approach, Catalyst will populate the tool with requirements initially and then set a baseline after requirements validation sessions have been completed. Once requirements have a baseline, the team will continue developing detailed design specifications which support and further elaborate the

VoteCal System based on the requirements. Traceability from the requirements to these items is entered directly into RequisitePro.

The Requirements Management Plan will be reviewed and, if necessary, updated at the end of each phase. If changes are made to the requirements management and traceability processes, the Catalyst Project Team will also assess if corresponding changes are needed to the Change Control Plan and to the Configuration Management Plan.

The Catalyst Project Team will update design and test specifications in RequisitePro during the design, development, and test phases of the project.

The RequisitePro database will be backed up on a nightly basis, along with the databases for the other project management tools.

4.4   Accessibility of Requirements Traceability Matrix and Tool

The SOS Project Team will be provided read only access to the RequisitePro tool to verify that the system and software requirements are correct, complete, traceable and testable. The tool will allow assessing the traceability of the system requirements according to design, code, test, and training documentation.

The security for the tool is based on the permissions assigned by the tool administrator (the Catalyst Requirements Manager). For each individual (Catalyst and SOS) the Catalyst Requirements Manager will assign 1 to 4 permission: read, update, create, delete. All SOS and Catalyst Project Team members will have read access. Only the Catalyst Requirements Manager will have create and delete access. The Catalyst Functional and Test Teams will have the ability to upload documents for requirements traceability purposes.

If the SOS Project Team so desire, the Catalyst Project Team can export a copy of requirements in comma separate value (CSV) format, which can then be imported and utilized in other applications.