Capability Assessment Management System (CAMS) Redesign

Donn MacMillan
Delivery Manager R&D Solutions
AMITA Corporation
1420 Blair Place, 5th Floor
Ottawa, Ontario
K1J 9L8

Arek Wojciechowski
AMITA Corporation

Defence Research and Development Canada – CSS

IMPORTANT INFORMATIVE STATEMENTS

AMITA.

Public Safety Canada OCIP/SARKMS/CSSKP Support

Capability Assessment Management System (CAMS) Redesign

Version 1.0

September 19, 2014

Document Control

Revision History Log

Date	Version	Description of Revision	Author(s)
September 19, 2014	1.0	Initial Release	Arek Wojciechowski

Retention Requirement: This document shall be read-only once approved except when authorized by Change Request. This document shall remain a permanent project artefact.

Distribution for Review and Approval:

Name	Location
Shaye Friesen	Risk Assessment & Capability Integration Section, Defence R&D Canada Centre for Security Science (CSS)
Donn MacMillan	AMITA Corporation Delivery Manager R&D Solutions

Acknowledgements:

(Note: Persons listed below, have provided comments on content and/or contributed material that is included in this document)

Name	Project Role
Shaye Friesen	Risk Assessment & Capability Integration Section, Defence R&D Canada Centre for Security Science (CSS), Original CAMS documentation
Pat Labelle	Enterprise Architect, AMITA Corporation

AMITA.

Table of Contents

AMITA.

Table of Figures

AMITA.

1   Document Purpose and Use

1.1   Background

The Capability Assessment Management System (CAMS) application was developed to support the AHRA and systematize capability assessment. The current CAMS web-based application includes the ability to:

- characterize scenarios
- maintain an inventory of "master events" and scenarios
- catalogue tasks and maintain a historical record of assessments
- capture subject matter expert (SME) judgment
- facilitate comparisons
- analyse capability gaps and requirements across the emergency management spectrum.

The current application is developed in Microsoft's ASP.NET and AMITA Corporation (AMITA) has been contracted to provide a basic architecture and design for converting the CAMS ASP.NET into a SharePoint Application, to be hosted within the CSS-TOP SharePoint Site Collection located in the High Resiliency Environment (HRE).

1.2   Purpose

In order to off load current support and maintenance costs, Defence R&D CSS has set forth the primary task of porting the CAMS application to the HRE SharePoint Farm.

This document presents the analysis of the current CAM's implementation and structure, a high level view of the proposed design, and budgetary details of the costs to implement the redesign in SharePoint technology in order for the CAMS' Technical Authority to formulate an informed decision on the way forward.

1.3   Audience

The target audience for this document is

- CAMS' Technical Authority: Use this Document to understand the scope of the re-design/conversion effort.

- AMITA's technical team: Use architecture and approach in the event that AMITA is contracted to implement the design and deploy the solution.

1.4   Scope

The scope of the proposed solution is defined as follows:

a. Port existing end user functionality to a SharePoint environment
b. Provide CAMS Users access to CAMS data and reports via the SharePoint User interface hosted in the High Resilience Environment
c. Provide CAMS Administrators access to manage Look up table data via the SharePoint User interface hosted in the High Resilience Environment
d. Provide CAMS Administrators ability to manage User access permissions via the SharePoint via the SharePoint User interface hosted in the High Resilience Environment

It must be noted that the scope as defined here presents the minimum steps and actions to port the current system to the new technical environment. This minimum is not equivalent to redesigning the prototype application for specific deployment to the target production environment.

1.5   Scope Exclusions

Existing Developer Role functionality, even if currently available to end users/administrators will not be available via the SharePoint user interface.

Specifically excluded functionality consists of:

a. Ability to modify SQL Server table structures
b. Ability to modify SQL Server Views
c. Any other functionality requiring direct access to the SQL database or the HRE Server file system.

The following use case diagram illustrates the current CAMS actors and their available functionality.

AMITA.
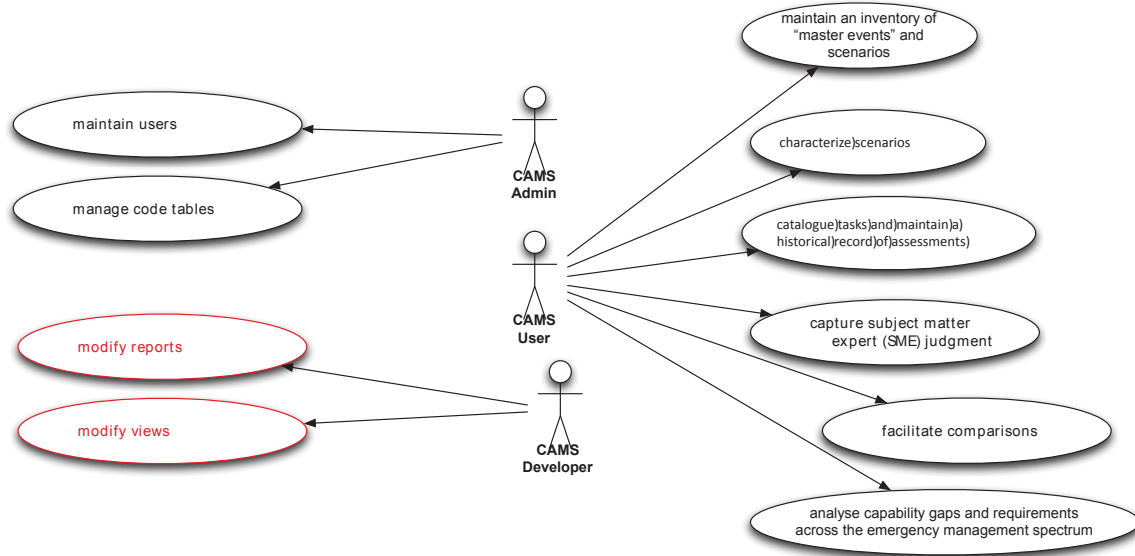
The functionality in red is outside the scope of the quote accompanying this report and will be addressed by the CSS-TOP delivering a release package to the HRE IT Support Contractor.

2   Constraints and Assumptions

2.1   Constraints

The following constraints were applied to the solution architecture when proposing a SharePoint conversion strategy for the CAMS application:

a. CAMS will become a site under the CSS-TOP Site Collection.
b. All existing CAMS Task and Event data must be retained and migrated into the upgraded solution.

2.2   Assumptions

The following assumptions were considered and applied to the solution architecture when proposing a SharePoint conversion strategy for the CAMS application:

a. The functionality of the current CAMS application is to be preserved, and no new functionality, beyond that required for SharePoint environment specific needs to be added.

AMITA.

b. The look and feel of the application will not be retained. Instead, a common SharePoint look and feel will be applied, and can further be modified by the application Administrator.

c. All currently assigned permissions will need to be manually migrated into the SharePoint permissions model post implementation (i.e., the CAMS application administrator will assign permissions post implementation).

d. The current User and Permissions management must be transformed to fit with SharePoint permissions model.

3   Review Findings

AMITA's review of the present CAMS application included a review of the source code, database schema, developer notes, and available documentation.

The current CAMS application is an ASP.net application, consisting of several web forms interacting with database tables stored in a SQL Server database, and several stand-alone (local) SQL Reporting Services Reports driven by complex views of the SQL tables. Reports are rendered by the ASP.net Report Viewer control.

3.1   Current Application Architecture

This sections provides a general overview of the existing CAMS applications.

3.1.1   References

The following CAMS reference materials were provided by DRDC-CSS:

1. All Hazards Risk Assessment Transition Project: Report on Capability Assessment Management System (CAMS) Automation C14-0324-1544 CAMS Contractor Report, DRDC-RDDC-2014-C35, April 2014

2. Overview of the All Hazards Risk Assessment (AHRA) Automated Application and Capability Assessment Management System (CAMS), Technical Note DRDC CSS TN 2013-031, November 2013

Based on this documentation, and a review of the CAMS source code and database schema, a high-level architecture of the existing CAMS application can be presented as follows:
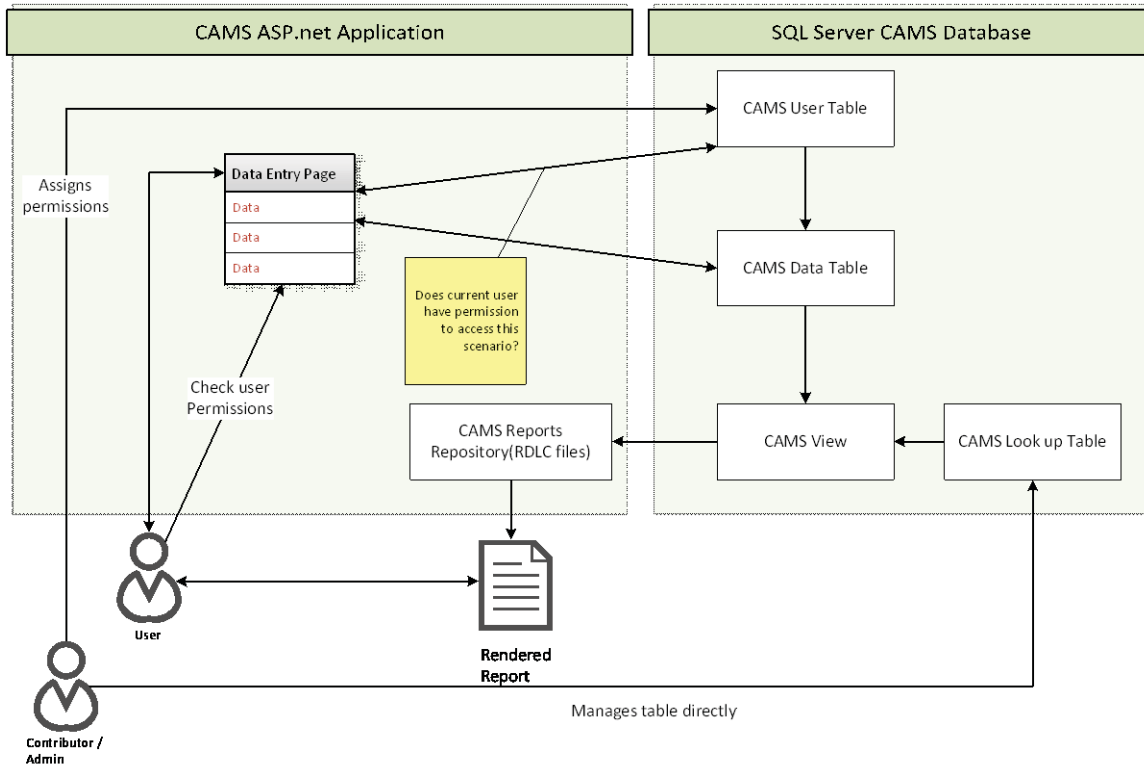
AMITA.

Figure 1 - Existing CAMS high level architecture

3.2   Existing Database

The interaction with the back-end SQL database is made possible by utilizing ASP.net Data Source objects and SQL statements. ASP.net web forms and controls are bound to the Data Source objects and allow users to add/edit/delete data in the tables.

3.3   Existing Look-up Tables

The CAMS application contains several look-up tables driving various pick-lists and drop-down lists on other forms. Some tables in the existing application have no data entry forms. The administrator must configure the look up values directly in the SQL database.

3.4   Existing Permissions Model

User access to the data is controlled by custom built logic. The current logged in Active Directory user is mapped to a user record in the Users SQL table with string comparison when the user access one of the ASP.net pages in the CAMS application. Data records are joined with the user table, and filtered for matching User records. Permissions are managed via an ASP.net data entry form connected to the User and User Permission tables. New users are automatically created on first access to the CAMS web application.

3.5 Existing Reports

The existing CAMS reports are rendered by the Report Viewer ASP.net control. This control allows for a stand-alone report (i.e., not requiring SQL Server Reporting Services) to access dynamic data via a Data Source. Existing reports query SQL Views in the CAMS database, which in turn use SQL tables. The reports are fixed in design and layout, with only the data being dynamic, and a Developer Role Actor is required to modify the report definition.

4 Proposed Solution

SharePoint Lists are not well suited to presenting data in a relational way for the purpose of complex reporting. As a result of this fact, and due to the complexity of the reports and views, the proposed solution will retain the SQL Database backend, as well as the stand-alone reports as they exist today, solely for the purpose of Reporting on the user entered data.

The redesign focuses on the data entry aspects of the application, namely the look up tables, Scenarios, Tasks, and Events. These entities lend themselves well to SharePoint list design. As a result they will be represented as SharePoint Lists in the CAMS SharePoint site, and will be able to make use of all available out of the box SharePoint features, such as item-level permissions, alerts, and workflows.

Data between the SharePoint Lists and the external SQL Database will be automatically synced by the system, facilitated by using workflows and/or event receivers, along with Business Connectivity Services (BCS) and External Content Types (ECTs).

4.1 Database

The structure of the original SQL tables will be retained in order to preserve existing Views, which drive the existing (and retained) reports in the CAMS system. Structural changes will be made to the database in so far as to facilitate BCS functionality.

The interaction with a back-end SQL database is made possible by utilizing SharePoint Business Connectivity Services (BCS) and External Content Types that map SQL Server tables to SharePoint recognized list sources. The external lists will be updated automatically when users update the SharePoint list counterpart. This database will be inaccessible to the CAMS users, essentially becoming a read-only reporting data source.

It is assumed that the currently existing views and reporting functionality is correct and functioning properly. The quote attached to this report does not include optimization or alterations to the existing reports. Any alterations will have to be performed by the CAMS vendor and supplied as a release package to the HRE IT Support team for a controlled deployment cycle.

AMITA.

4.2   Look-up Tables

Once converted, Look-up tables will be accessed by users (view, add, edit) via standard SharePoint lists and View and Edit Item forms. Permissions to modify these tables and lists as a whole can be assigned by the CAMS application administrator.

4.3   Item Level Permissions

SharePoint solutions are very well suited to permissions trimming and item level security. However, this functionality does not extend to External Content Types (ECTs). Out of the box, SharePoint only supports list level permissions on ECTs, allowing the administrator to set access and read/write permissions on the entire list only. This does not work for the CAMS application, as users must be able to be limited in which Scenarios they can view and edit. To overcome this limitation, a hybrid List/Table approach will be implemented.

The proposed approach preserves the current security model of the application, such that any currently accepted SOS, and PIA safeguards in the application are not altered. The application as proposed leverages infrastructure hardware, software, and networking that is currently in place and as a result we do not perceive a change in the security posture of the destination SharePoint farm.

4.3.1   Integration of Security with SharePoint Permissions

Due to the nature of how SharePoint applies permissions to External Content Types, out of the box integration with item-level permissions is not possible. The solution to bridge this functionality gap is to combine SQL Server database tables (External Content Type) with a SharePoint list counterpart. This approach will provide the administrators an integrated user management experience as well as built in SharePoint list permission trimming.

Administrators/Contributors can set item level permissions on the CAMS Scenario list (and other entities) directly in the SharePoint list. The associated SQL records in the CAMS Database are then filtered based on the permission trimmed SharePoint List items and by logged in User extracted from the SharePoint user context, and passed into the Report Definition as parameters.

4.4   Reports

All report logic will remain as is, but the RDLC local report definitions will be converted to RDL files and uploaded to the central HRE Report Server. Appropriate permissions will be applied, allowing CAMS application administrators to further manage, edit, and define additional reports, based on existing tables and views available in the CAMS SQL database. Access to the reports will be via the Report Viewer Web Part, and will be filtered on the current user context to show only the data that the current user is allowed to access.

AMITA.

4.5   High-Level Architecture

The following diagrams show the high-level architecture for the proposed solution:
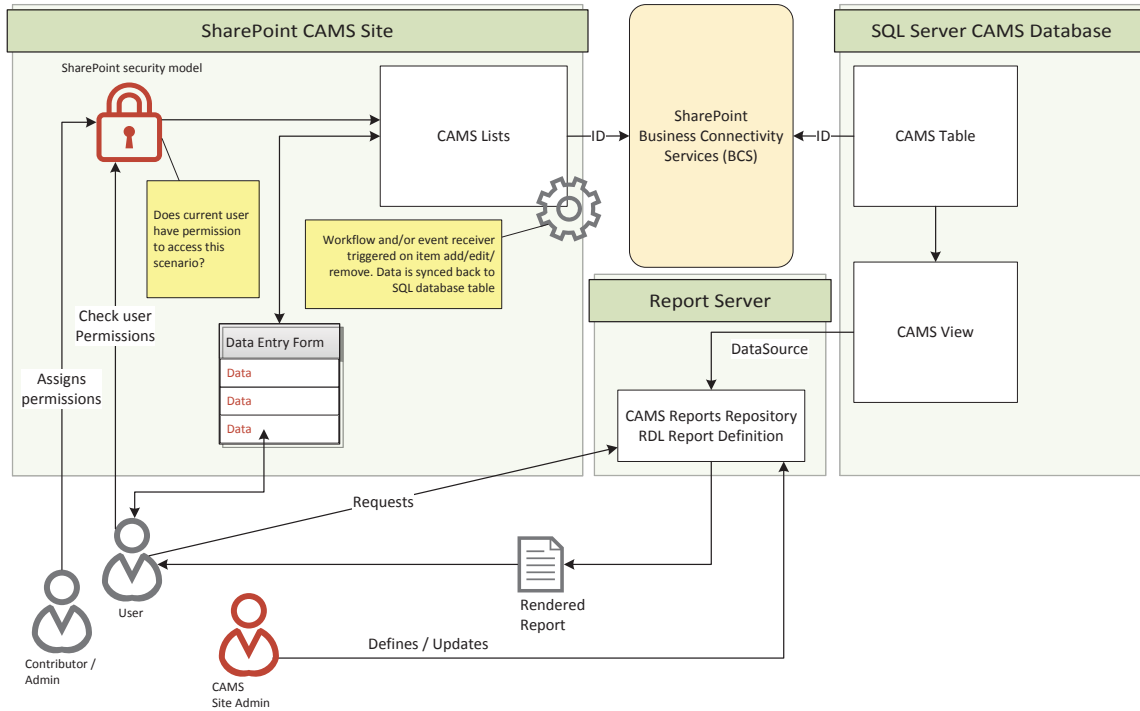

Figure 2 - CAMS SharePoint High-level Architecture

5   Key Points and Summary

The solution proposed in this document moves the existing CAMS ASP.net application into a SharePoint environment and retains all current functionality, while providing additional benefits that an integrated SharePoint environment has to offer. The key points of this solution are as follows:

a. The existing CAMS data is imported into SharePoint lists. All subsequent data maintenance is conducted via the SharePoint UI, with all inherent benefits, including Forms, List views, Workflows, Alerts, Item-level Permissions, etc.
b. The CAMS applications administrator can further enhance the application via available SharePoint tools and functionality, should sufficient permissions be established
c. Existing report definitions are preserved, and will continue to use the SQL Server database views. Report data can continue to be exported to MS Excel for further manipulation.
d. Additional and existing reports can be maintained by the CAMS application administrator, if sufficient permissions are assigned.
e. Data synchronization between the user maintained SharePoint lists and the report source SQL database is automatic and transparent.
f. Future growth potential and integration with other SharePoint tools part of the CSS-TOP site collection (for example, AHRA) becomes feasible and greatly simplified as all User data uses a common technology and platform.
g. Upgrade to SharePoint 2013 becomes well understood as all components are built using out of the box SharePoint functionality.

AMITA.

Annex A –User Permissions Management Alternative

Retain Current User Table

An alternative option was considered, but rejected for consideration in this solution. This option simply consisted of retaining the existing user and permissions tables in SQL Server, and allowing the Administrator to assign permissions to Scenarios as before.

This option does not provide any interface with SharePoint permissions, beyond passing the current login information to the system and filtering the data accordingly.

This option is not recommended for the production environment as it forces the CAMS users to maintain two separate User stores (SharePoint/AD users and CAMS Users), with no linking between these user entities, and does not fit into the CSS-TOP integrated common technologies platform.

Annex B – Data Storage and Management Alternative

Consideration was given to maintaining only the SQL Database as a data repository and facilitating access to the data from within SharePoint. However, the nature of such a solution, namely using External Content Types with SharePoint External Lists to maintain the data in the database was deemed too restrictive, particularly with respect to managing user permissions in a consistent SharePoint way, but also due to the numerous other limitations.

External Content Types and External Lists based on them, while extremely convenient for integrating existing Line of Business applications with SharePoint, are not well suited to a full data entry solution via the SharePoint interface. That is to say, in such a scenario, non-SharePoint data entry is retained in some capacity. The following is a summary of features not available to a SharePoint list based on an External Content Type:

- Datasheet view is not available
- Export to Excel option is not available
- Workflows cannot be configured for External Lists
- Cannot create Information Management Policies (bar codes, auditing, retention or labels)
- No version or version history
- Inline Editing is not available
- No Ratings widget
- Can't create Visio Diagrams
- Open with Access/Open with Project is not supported
- REST access via ListData.svc to External Lists is not available (All WCF services like RSS feeds are not available)
- No Look ups - No Content Types
- No Context Menu (ECB) Send To operations
- Events handlers are not supported
- LINQ to SharePoint spmetal.exe does not support external lists
- No Item Level Permissions
- No item or field level validation or Formula supported
- Document templates are not supported
- Alerts are not supported
- Attachments will not be available
- Additional data connections in InfoPath 2010 list forms not supported
- No Check In / Check Out options

AMITA.

- Drafts of items are not supported
- No write support for BLOB
- Metadata based navigation is not supported
- You cannot mix the SharePoint columns with External List columns since they are coming from external source

SharePoint 2013 addresses some of these limitations, but not to a satisfactory extent:

- Sorting & Filtering for External lists is improved
- Export to Excel option included
- Event Receiver on external lists is introduced
- Supports OData