Sustainable Built Environment National Research Centre

Collaborative Object Libraries

Project 3.1

Final Report

Version 1.0

8 November 2012

Table of Contents

1 Executive Summary

Close collaboration between the project partners, especially QUT, Smart Services CRC and Queensland Government, has produced a software platform that is a viable candidate for deployment as a National Object Library for the Australian design and construction sector. The software has been developed based on model-driven engineering (MDE) principles, with agile project management, in order to gain the greatest advantage of the experience of the partners in BIM as applied over the lifecycle of buildings, IFC and other BIM standards, Web services, repository technology, and model transformations.

The resulting Object Library (see Figure 1) stores CAD-tool-independent representations of BIM Objects, uploaded in a variety of fit-for-purpose file formats. Its user interface can be accessed via any modern Web browser, or started from within CAD tools, and offers designers powerful keyword- and property-based searches of manufactured objects. Selected objects are then transformed, before or during download, into CAD-specific formats and can be inserted directly into a building design. Universal indentifiers allow for object properties to be retained when exported from CAD to CAE, Cost Estimation or other tools, even when CAD tools do not support property customisation.

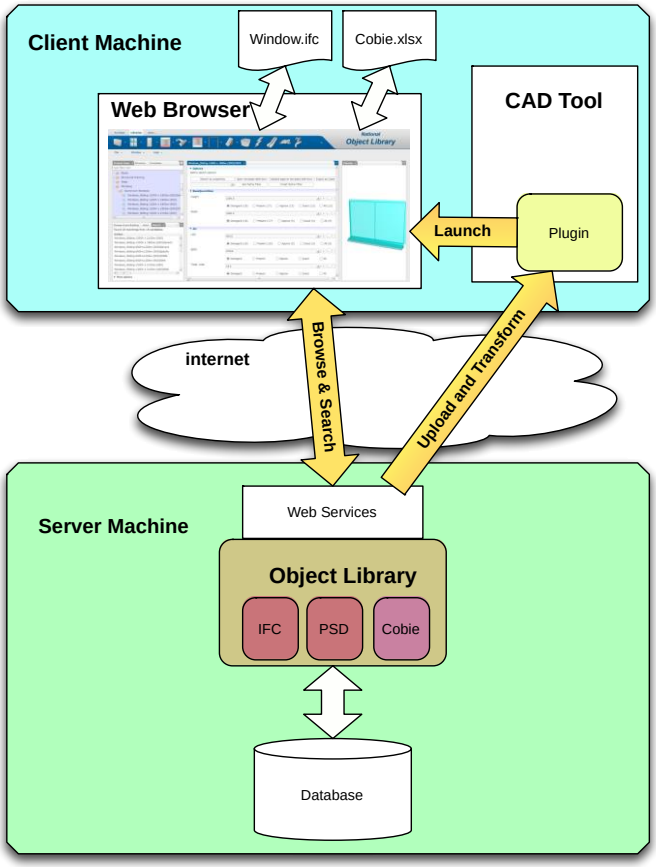

Figure 1: Overview of Object Library Implementation

Download of objects is currently supported for ArchiCAD and Revit, with support for Bentley Architecture in progress. We are have also experimented with the use of Object Library data in

engineering simulation and cost estimation software, as well as in parametric design tools. Instances of the Object Library server are running at Amazon Web Services and in Project Services in Queensland government, and they contain significant example templates and objects which demonstrate the functioning of the Library. Online help menus in the library refer users to web-based documentation on the use of its search, navigation and download functions. The project has resulted in three peer-reviewed international publications, with more in the pipeline.

1.1 Relationships to other documents

The following documents are available separately from SBE nrc.

The "National Object Library - Concept of Operations v1.0" document provides a guide to how the Object Library is intended to be used in a role of an Australian national construction product catalogue which contains BIM Objects in the form of IFC Geometry and Properties, which can be transformed into "objects" for use natively within various CAD software.

The "Instructions for Building and Installing the RaaS Object Library v1.0" document explains how the software source code is checked out from the various source code repositories which comprise the implementation, and how it is compiled, deployed and initialised to become a running online Object Library server.

The "Software Architecture the RaaS Object Library v1.0" document provides details of the components that make up the Object Library, and all the associated tools (such as the CAD plugins), open-source software framworks, models, generated code, and handwritten code, that provide the functionality available in the first full release. A summary of this document is provided in Section 4.

1.2 Structure of this document

The next section "Background" is a reiteration of the initial requirements for an Object Library, as specificied in the SBE Project Overview document. This is repeated here so that the reader can understand the context in which the research and development was conducted, and compare the stated goals with the actual outcomes.

Following this in Section 3 is a summary of the outcomes of the project, which are given in three categories: outcomes anticipated at the beginning of the work, additional outcomes, and features not pursued (of which there is only one).

After Section 4, which explains the main components of the delivered software, documentation and design artifacts delivered, Section 5 to 7 explain the details of deliverables in three categories summarised in Section 3.

Section 8 provides a brief conclusion.

2 Background

2.1 About the Collaborative Object Libraries Project

The project to develop a candidate National Object Library is called *Collaborative Object Libraries*. It is being conducted within the Sustainable Built Environment National Research Centre at Queensland University of Technology, with a number of other partners of the centre, notably Swinburne University of Technology, Queensland Government and the Smart Services CRC.

2.1.1 The aims of the project

The *Collaborative Object Libraries* project is one of the funded projects of SBEnrc, and its plan states the following aim, and frames the problems within the construction industry with respect to the lack of shared libraries of digital virtual building model objects:

> ***Aim:***
>
> *The aim of this project is to define and develop for the civil, commercial and residential construction industry a standard for a new system of creating, capturing and re-using knowledge.*
>
> *This will be achieved by extending the current paradigm of computer-aided design (CAD) libraries in new and powerful ways through collaborative building object libraries. These will support the design, construction, facilities management and demolition/re-use information across multiple disciplines.*
>
> ***Industry Need***
>
> *The fundamental problem faced by the industry is that the objects created and included in models are duplicated by each practice and mostly cannot be effectively shared between practices and projects. The objects are expensive to create and the current position compromises the interoperability and efficiency of the industry significantly.*
>
> *The problem affects all areas of civil and building construction. It is most acute for building projects, where design professionals rely on software libraries to maintain standardisation of object definitions and to increase productivity and improve quality throughout the development lifecycle thereby reducing costs and improving delivery times for projects.*
>
> *Every construction project uses "libraries" of products and processes. These capture information about the project that is used across multiple projects (industry wide) or within a single project (project specific). The current range of computer software used for design and analysis each address these libraries in individual ways, with no indication from the vendors of a "neutral" approach to libraries.*
>
> *This:*
>
> - *Prevents rationalisation and re-use within organisations, within projects (across organisations) and across the industry;*
> - *Creates inefficiencies as business are hindered in the transfer of data between systems – for example, between architectural and HVAC design systems, where the information required by each discipline is stored separately, with no consistency checking between;*
> - *Creates a barrier to SME adoption of this technology;*
> - *Results in a loss of productivity to designers and constructors;*
> - *Becomes costly to maintain object libraries in facility management systems using current industry practices and tools.*
>
> *The efficiency of digital modelling processes will improve enormously if it becomes possible:*
>
> - *To share object libraries across the different softwares thereby reducing effort required by individual organisations to exploit the capabilities of digital modelling*

- *For SMEs within this industry to adopt and benefit from the digital technology available.*
- *For object libraries to be adopted within the procurement supply chain process and in facility management systems.*

2.1.2 Scope

The aims of the project are very broad, and encompass cradle-to-cradle lifecycle of construction projects over a building's lifecycle. The ultimate vision of using digital building models across tools and stakeholders, throughout the life and changing ownership and occupancy of the buildings that are constructed is still some way from realisation. However, the development of an online, openly accessible, product library of manufactured products available within the Australian industry will go some way towards the sharing of digital models over time, and between industry sectors.

The scope of the National Object Library includes:

- A mechanism for defining templates for digital objects, including required properties, e.g. 'U' value, SHGC, etc., with ranges or choices of values, as well as placeholders for geometry for use within digital design tools.
- The classification of digital objects according to multiple classification schemes relevant within various sectors of the Australian and international building industries.
- The ability for manufacturers of building products to fill in these templates to represent their products and/or product lines or systems, and publish them to the Library. This includes the ability to upgrade and deprecate product descriptions, while retaining a record of any product actually deployed in a building for later use by the building owners and maintainers.
- The ability for designers and specifiers to browse or search the Library using various parameters to match products or product lines or systems that meet their criteria, and to download the objects which describe them in formats that are suitable for the design and analysis software tools that they use.

Some things out of scope for the initial deployment are:

- The deployment of similar company- or project-wide libraries which interoperate with the NOL and allow for the creation of composite objects made up of products from several manufacturers, or objects representing on-site construction. This is planned for phase two of the project.
- The support for arbitrary parametric designs.

3 Summary of Outcomes

The project has delivered the all of the outcomes anticipated at the beginning of the work, except one, which is listed below, and discussed in Section 7.

3.1 Anticipated Outcomes Delivered

- **A CAD-tool-independent object repository** for storing construction objects that manufacturers can upload to represent the geometry and properties of their products. This repository is based on the Industry Foundation Classes (IFC) and Property Set Definition (PSD) standards of buildingSmart International.
- **A Web-based application** which can run in any HTML5 Web browser and communicates to the repository. It provides facilities for:
    - navigation through the classes of objects available,
    - keyword- and property-based search functions to find objects
    - Interactive 3D previews of the geometry for one or more objects selected by search or navigation
    - the ability to upload objects via IFC Part21 files and COBie spreadsheets of property definitions, or zip files containing whole folders of objects
    - the ability to edit the properties of any object when acting in the "Librarian" role
- **A set of transformation tools** that permit objects stored as IFC to be exported using CAD-specific formats directly into CAD tools using their plugin APIs. This allows the Library application to be launched from within CAD tools, and for downloaded objects to be used immediately in open projects.
    - In the case of ArchiCAD, an IFC model can be transformed at the server into a GDL (Geometric Design Language) script, and this is uploaded via a plugin to the user's ArchiCAD at the client, and packaged into the native object format, GSM, in the CAD tool.
    - In the case of Revit, which does not publish its file formats, the tool plugin uses the Object Library's Web services interfaces to fetch the geometry and properties of objects, and associate them with the appropriate native object "Family" type, performing a geometry translation in the plugin at the client side.
    - The Bentley Architecture export facility is being investigated.
- **Design model artefacts** have been developed for testing, demonstrations, and proof-of-concept:
    - Object Templates for Windows, Beams, Doors, Floors, Roofs, Furniture and other object types
    - Object Definitions conforming to these Templates with geometry and properties: Windows, Beams, Doors, Floors, Roofs, Furniture and other object types at Levels Of Development (LOD) 200, 300, 400
- **Documentation** for the Object Library has been written using Web pages, which are accessible through the context-sensitive "Help" menus in the User Interface Application.

- **Multiple classification of objects** is facilitated by using the IFC entity type hierarchy as a primary classification, which can be compared with other classification schemes by navigating the ontology at the buildingSmart Data Dictionary (bSDD). The bSDD now contains concept classifications in various natural languages, IFC, COBie, MasterFormat, Omniclass, and an increasingly large range of other classification schemes. We are currently negotiating write-access to bSDD in order to upload Australian-specific property sets, which form the basis of object Templates, available directly through bSDD.
- **Certain "parametric" objects** in the library can be made adjusted according to design circumstances when exported to the CAD tools. These include beams and columns which in Revit "snap to" the nearest walls or ceilings, and in ArchiCAD allow to their dimensions to be edited using the X, Y and Z. We are monitoring the development of parametric capabilities for IFC within the buildingSmart standards, but these are currently experimental and provisional, and there is no draft standard that is being progressed by the organisation as yet. As IFC does not currently support parametrics, we change the fixed dimensions of some objects in the library to be variable and parametric within the transformations targeting each CAD platform.

3.2 Additional features not anticipated in the project plan

- **A scalable BIM repository for any IFC model**. The IFC-based repository can store BIM Models of arbitrary complexity and size from a single item of furniture to a whole office-block-sized design. Furthermore, the objects in a complex design can be accessed one at a time, or in groups via remote Web-service-based applications. This forms the basis for visualisation and manipulation on whole building designs for other projects, and in the Object Libraries context it is used as the basis of the geometry of a product description.
- **Superior implementation of IFC file (Part 21) import**. Our implementation of IFC file to model import is superior to almost all IFC-based readers in both commercial software, and open-source projects due to the innovative design of the storage to a relational database. We implement a new algorithm for creating linked IFC objects in Java-based models and leverage best-of-breed object/relational mapping technology from the Connected Data Objects (CDO) Eclipse open source project.
- **CAD tool plugins** which can launch the Object Library Web application, and allow a selected object in a CAD tool to be used as the basis for a search of library objects of the same type, and for the selected object in the CAD tool to be automatically updated to include some or all of the properties of a discovered library object.
- **Overcoming limitations in IFC export from CAD Tools**. The ability of IFC export functions in CAD tools currently cannot natively include custom properties, such as those attached to Object Library product descriptions, have been overcome by retaining the unique identifiers of library objects, and then re-attaching the associated properties to the IFC file after export from the CAD tool. This is done seamlessly using CAD plugins, and enables Object Library properties to be fully utilised down the tool chain by CAE and Cost Estimation software. Designs that are larger than can be stored in RAM can be similarly "patched" with correct properties if they are imported into the IFC repository.

- **WebDAV integration** allows for navigation of the contents of the Object Library, and insertion of contents into the Library, using the File Explorer interface in any version of the Windows operating system. The implementation allows access to "files" in the explorer to be translated into Web-service calls that access the repository, storing and retrieving the geometry or properties of Objects from the underlying database as if they were normal text or spread sheet files.

3.3 Aspects of the project plan that were not delivered

- **Rich parametrics**. There are two factors that made this approach currently too difficult: 1. There is no accepted parametric IFC standards at the moment, and 2. The majority of existing manufacturer product sheets do not express the variable parts of products as parameters, but rather show multiple specifications of the properties, dimensions and drawings of the available range of products. Therefore, due to resource constraints, and the likelihood of needing to re-implement this approach once parametric IFC is standardised, and tool vendors begin using it, we decided to pursue only a small set of demonstration objects with parametric qualities.

4 The Main Components of the Object Library Software

This section descibes the major pieces of software that were written and generated to implement the Object Library and its user interface, and the frameworks that support its operation. Figure 2 shows a server machine in green in which the main library is operated. The client machine in shown in blue, and is connected to the server using HTTP over TCP/IP (Web protocols running over Internet). All white arrows show communication using Internet protocols.

The fundamental development enviroment for most of the software is the Eclipse Java Integrated Development Environment (IDE). Eclipse is an open-source tool established by IBM, which is accompanied by a large ecosystem of "plugins" and other tools, mostly also available under the same open-source licence as the IDE. We use using the Eclipse Modeling Framework as the basis for models of building objects, their properties, and the directory structure of the Library. This modelling environment permits us to use various tools to partially or fully automate the construction of some of the software components of the Library, using an approach known as Model-Driven Engineering (MDE). The Object Library is written mostly in Java, however much of the Java code is automatically generated from the models and other artifacts like rule-based model transformations and generated serialisers and deserialisers that turn in-memory model graphs into files and vice versa.

In Figure 2 all white boxes show software that is not part of the Object Library. The main "out-of-the-box" components of interest, from top to bottom in the Figure are:

- A **Web Browser**. This can be any modern HTML5 browser.
- A **CAD Tool**. At the moment ArchiCAD and Revit are supported.
- **Tomcat** Application Server. This is one of the most widely deployed Web Application Servers. They host files and Web service-based applications that communicate using SOAP and REST protocols. Tomcat directs incoming requests to the correct part of

the Object Library implementation.

Running within Tomcat is a set of framework libraries from the Eclipse open source community:

- **RAP** is a user-interface technology that shows various panes in the user interface within the Web browser, and manages communications back to the Object Library based on user interactions with the Web browser.
- **JAX-B** is the Java framework that manages mappings from Java Objects to XML messages for Web services.
- **EMF** is the Eclipse Modeling Framework. It is the means by which we represent IFC and Property Set Definitions, as well as the Library's folder structure, as inter-linked Java objects whose classes and implementations are generated from the ".ecore" files shown as input to the RaaS.
- **CDO** is Connected Data Objects - an intelligent storage layer that manages the saving and loading of EMF objects into a database.

- **RaaS** is the Repository as a Service tool from Smart Services CRC, which generates code to coordinate the use of EMF, JAX-B and CDO and enables the Web Browser client and the CAD Tool Plugin to access the Objects in the Object Library remotely, as well as their storage in the database.

- **Database**. Any CDO-supported database can be used in this role.

The coloured boxes in Figure 2 are the implementation written and generated by the Collaborative Object Libraries project.

- **RAP Thin Client** is a generated user interface that is loaded into a web browser, and shows the user a customised version of the Eclipse IDE as a remote client. Its purpose is to display the contents of the Library, and allow for searching and display of the details of Objects in the Library.

  In our user interface configuration shown in Figure 3 these include (left to right): listings of the folder hierarchy of Objects over a search results listing, detailed properties of a selected Object, and an interactive 3D preview of the Object's geometry. In addition, there are drop-down menus for managing files and window configurations, and a "ribbon" of popular kinds of objects for easy searching by category.

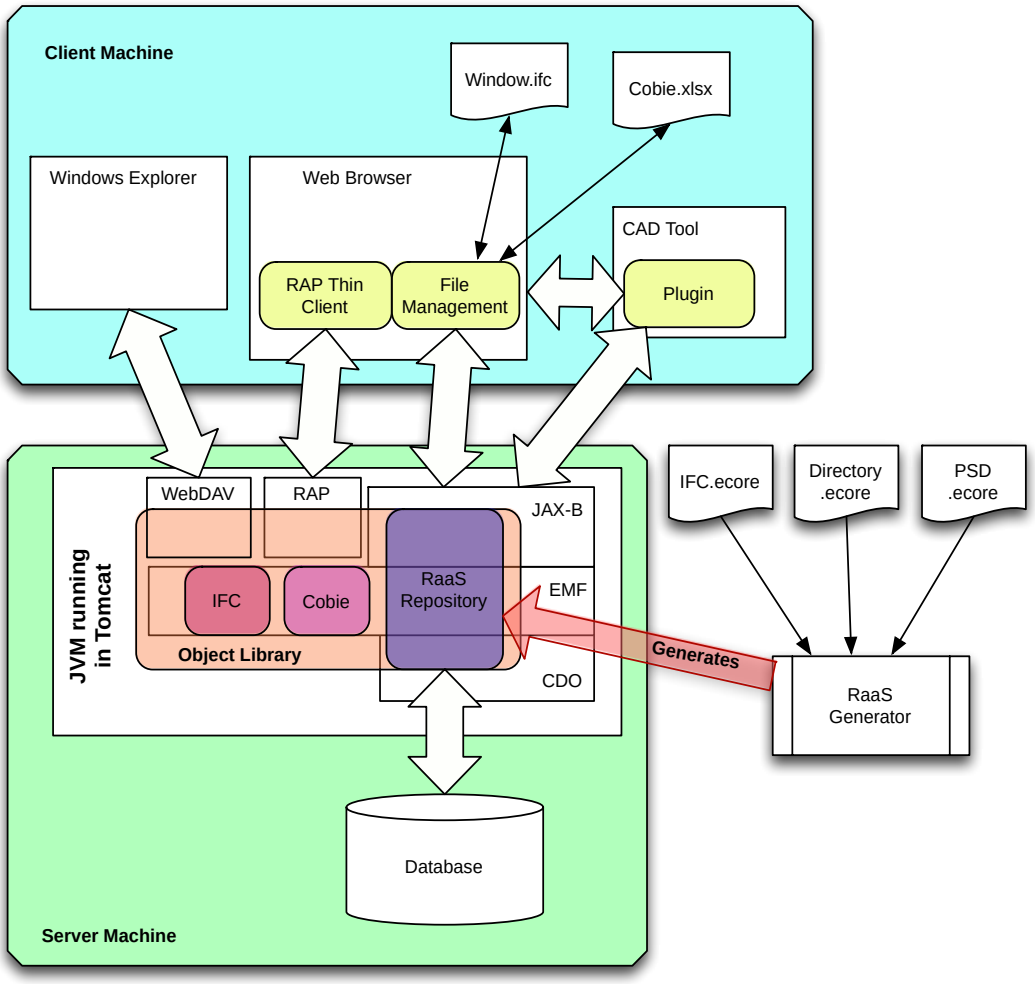

Figure 2: Object Library Software Architecture

- **File Management** is in reality just a set of links for downloads and a set of dialogs for uploads which directly invoke the RaaS-generated Web service interfaces, and allow content to be moved in and out of the Library. They are embedded inside the RAP client's display.

- **WebDAV** is a framework that exposes remote services to Windows users as a folder/file structure. We implement a WebDAV interface as part of the Object Library that allows clients of the Library to use Windows Explorer to see the product definitions in their folder structure, and to copy the components of the product descriptions to other place on their file system as IFC, COBie or other file types.

- The **Plugin** shown in the CAD Tool represents a per-CAD-Tool piece of code that can be loaded into the tool to allow searches of the Object Library to be triggered by from the CAD Tool's menus - which results in the opening of a Web Browser with the OL Interface loaded. In addition, the plugins may also provide contextual search parameters, depending on what design element is currently selected in the CAD tool.

The Revit Plugin also performs the conversion from IFC geometry to native Revit geometry, and assists an IFC export from the tool in re-attaching the property sets from the Library to the objects downloaded from the Library (as Revit does not allow IFC export of custom "parameters", and does not support the grouping of "parameters" into sets.)

The ArchiCAD plugin performs similar geometry conversions at the server side, expressing these in the GDL language, and then downloads the GDL geometry and object's properties from the Library, and packages them as an ArchiCAD GSM object.

- The **Object Library** implementation is shown as a set of nested coloured boxes representing software components overlaid on top of the frameworks that they use.

  The indigo part is the RaaS-generated code, which coordinates the storage and retrieval of EMF objects to the database using CDO, and exposes Web services interfaces to the clients to access these objects (more correctly graphs of linked objects).

  The pink **IFC** and **COBie** components allow for the reading and writing of ".ifc" (Part 21) files, and .xlsx (COBie spread sheet) files into and out of EMF object graphs. These components are used indirectly by the File Management part of the user interface to allow users to get access to Library Object information in standard file formats, and to upload new Object definitions into the Library.

  The orange box that surrounds these other components is all the other code that glues the Library together as a coherent whole.

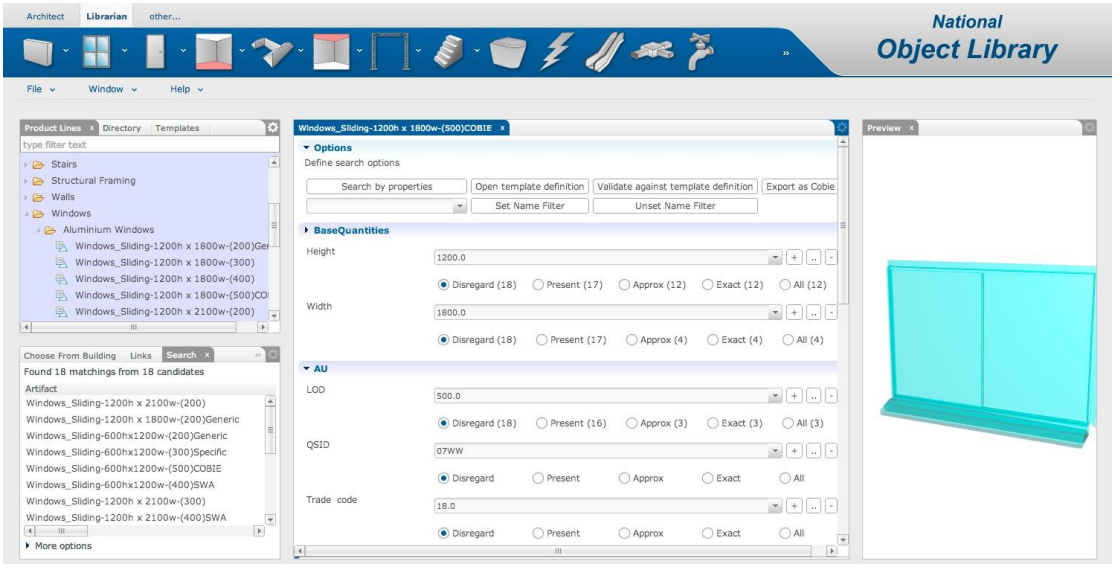

Figure 3: Object Library User Interface

5 Detailed Description of Deliverables in Project Plan

5.1 A CAD-tool-independent Repository

The Object Library is implemented primarily using the Repository as a Service (RaaS) technology from Smart Services CRC, which is a partner in the project. The standard interchange model for CAD and CAE tools, Industry Foundation Classes (IFC) from buildingSmart International, has been used as the primary language for representing the geometry of objects in the Library. This langauge is written as an EXPRESS Shema, and the QUT BIM Tools implementation of the Eclipse Modelling Framework (EMF) representation for IFC has been used as the foundation for generating a repository which supports (a) file uploads in the standard IFC plain text format (b) the generation of a smart database layer to store objects read in from IFC files, and (c) a Web services layer to retrieve graphs of connected objects from the Library to some remote machine connected via the internet.

In addition, a model of buildingSmart Property Set Definitions (PSD) has been used to generate an additional part of the repository, also using the RaaS technology, so that in cases where the geometry for objects is not needed, the properties of the obejcts can be transmitted independently from IFC geometry.

Finally, as the Object Library is structured like a catalogue of products, in which each product is placed in a folder appropriate to its category, the catalogue structure is also created using RaaS, and used as the way of getting product descriptions in and out of the repository as files, which are then converted into Java Objects within the Eclipse Modelling Framework, for display, search and transformation into CAD-specific formats.

5.2 A Web-based application

The Rich Application Protol (RAP) tools which are released as open-source software with the Eclipse Java IDE allows for web-based applications to be created which can run in any HTML5 Web browser and communicate back to the repository.

The RAP conifguration that we have created for the web interface to the Objecy Library supports rich navigation through the product description objects available, through the catalogue-like structure of the Library. It also provides a number of keyword- and property-based search functions to find products that the user is interested in finding and comparing. Many useful user tips are provided in the interface, such as: suggestions for property values that will match against actual products; the relevance of any property in finding the matched set of product descriptions; and the ability to instantly update search matches when a user changes the value of a product property.

Interactive 3D previews of the geometry for one or more objects that the user selects is available in a preview pane.

The Librarian can also upload objects via IFC Part21 files and COBie spreadsheets of property definitions, or zip files containing whole folders of objects. The librarian may also edit the properties of any object.

5.3 A set of Transformation Tools

Our preferred approach to the transformation of design objects between IFC-based descriptions, and CAD-specific formats is the also the most resource efficient and fast from the user point of view. That is: we import an IFC file and associated property sets (in PSD or COBie or SPie format)

and then immediately perform the tranformation into any required CAD-specific formats, so that this work is done once, and done early, and makes the transformed object available without delay to designers wishing to download it. However, a prerequisite for this approach is the ability to have a known file format for the CAD-specific version from which we can abstract an object-oriented model, and then perform a rule-based transformation between the model representation of the IFC classes and the model representation of the CAD format (with file importers/exporters to facilitate exchange acrosss the internet). We use this approach for the creation of ArchiCAD GSM files, which contain snippets of Geometric Description Langauge (GDL) to describe the geometry, and some text and images to represent properties and object previews.

In the case of Revit, which does not publish its file formats, the tool plugin uses the Object Library's Web services interfaces to fetch the geometry and properties of objects, and associate them with the appropriate native object "Family" type, performing a geometry translation in the plugin at the client side each time an object is imported from the library.

The Bentley Architecture export facility is being investigated, along with other tools, including Rhino and DDS.

5.4 Design Model Artifacts

During the development and testing of the Object Library, we have created templates and objects of many different product categories for testing purposes. All of the SPie template files, containing property sets and example generic geometry have been imported into the library in order to show the diversity of object types that will eventually be represented by real manufacturer-provided product descriptions.

In addition, all of the objects used in the design of a residential apartment building have been separately exported from Revit as IFC files, and these have been imported into the library, so that example objects are available for almost all kinds building products used in residential construction.

Finally, windows have been a primary class of objects used for testing and demonstration, and a number of objects that relfect the designs of commercially available windows in Australia have been uploaded to the Library. Windows, doors, beams and other object types have been the basis for experiments based on including different Levels of Development (LOD) for objects required at different stages of a CAD design project. We have also experimented with composite objects ranging from roofs up to entire built-in kitchen units.

5.5 Online Documentation for the Object Library Interface

The "Help" menu in the Object Libarary web interface automatically directs the user to documentation Web pages, based on the context within which the user is currently using the Object Library. The documentation pages also contain small instructional video clips that can assist the user in better utilising the OL's Web interface and menus.

5.6 Multiple classification of objects

The buildingSmart Data Dictionary (bSDD) is an online ontology and data disctionary for terms used in the construction industry worldwide. Each term is defined within a context - for example COBie, or the UK Building Code, or Omniclass, and may include translations into many natural languages. In addition, terms can be grouped into property sets which describe, for example, the performance characteristics of a Window in the Australian construction industry.

IFC is used as the primary classification for objects within the Object Library. IFC's Classes describing building objects are also populated into the bSDD, and on that basis, the equivalence of any object's classification in the Object Library with respect to many other classification schemes can be investigated using the bSDD online interface. We are currently negotiating write-access to bSDD in order to upload Australian-specific property sets, which form the basis of object Templates, available directly through bSDD, and to allow users to search for product templates and categories from within the Object Library.

6 Detailed Deliverables: Additional Features

6.1 IFC Repository for arbitrary building designs

The IFC-based repository can store BIM Models of arbitrary complexity and size from a single item of furniture to a whole office-block-sized design. The implementation of IFC file (Part 21) import is superior to almost all IFC-based readers in both commercial software, and open-source projects due to the innovative design of the storage to a relational database. We implement a new algorithm for creating linked IFC objects in Java-based models and leverage best-of-breed object/relational mapping technology from the Connected Data Objects (CDO) Eclipse open source project.

6.2 Launch of the Library from within CAD Tools

The plugins created for ArchiCAD and Revit allow the user of these CAD tools to select an additional menu item from within the tool to launch the web-borwser interface to the Object Library. In addition, if an object is currently selected by the designer within the CAD tool, that object's type and basic properties are used to perform an immediate search in the Library for the same kind of object, and shows the search results immediately upon launch. Then the designer can use any matching object description in the library to add and edit required properties of an object they are searching for, and by clicking the "Search by Properties" button, they can widen or narrow the matched list. Once they have located the required product description, the designer can then choose to replace the object selected in the CAD tool with the one they have located in the Library.

6.3 Rich Property Sets for IFC Exports

In Revit and ArchiCAD the set of attributes/properties of a design object is flat, and all properties are shown in a single list. This is a limitation that does not occur in the Object Library, where multiple named Property Sets can be associated with different kinds of products, so that they can be grouped for purposes of likeness, or identification of the source of the Set. For example, a Window may have certain properties that derive from the IFC specification (such as overall dimensions), as well as others that relate to performance characteristics, colour, materials, as well as manufacturer-specific data. We represent these in the flat property/attribute sets of the CAD tools by creating compound names, like "Performance.UValue", or "Finish.Colour" so that designers can see which set a given imported property/attribute comes from after import.

At the time when a designer wishes to export the BIM from the CAD tool as an IFC file for use with CAE or cost estimation software, the Plugins automatically re-create the structured property sets to be associated with the appropriate exported objects by matching up the GUIDs of the original Library objects. The library implementation also allows for IFC files that are exported without using this feature to be loaded into the Library, and have their objects' property sets re-created upon re-export from the Library.

6.4 WebDAV and ZIP integration

For users of Windows O/S the WebDAV framework allows external data sources to be "mounted" as additional drives, and the contents of these sources to be navigated as if they were folders and files. The Object Library supports WebDAV, and so the Library can be mounted as a drive for easy search/browse/navigation, as well as file import and export from the Library. Each product description in the Object Library can be viewed as a folder in which there exist at least two files: a COBie spreadsheet in .xlsx format, which contains the property sets of a product, and an IFC file which contains its geometry. These are automatically mapped by the WebDAV implementation to and from the native database formats used within the Object Library, and allow easy uploading of new product definitions via the file explorer in Windows, or extraction or editing of the contents of the Library as if they were files.

As a simple backup mechanism, an folder of product descriptions, or indeed the whole Library, may be exported as a Zipped file system, or a whole set of product decription files can uploaded into the library from a ZIP file.

7 Details of Undelivered Features

7.1 Rich parametrics

Although certain objects in the library, including beams and columns, can be made parametric when exported to the CAD tools, generally parametric objects are not supported in the first release of the Object Library. The supported parametric objects are supported in Revit by its "snap to" feature, whereby the nearest walls or ceilings are, and in ArchiCAD allow to their dimensions to be edited using the X, Y and Z properties.

We are monitoring the development of parametric capabilities for IFC within the builidngSmart standards, but these are currently experimental and provisional, and there is no draft standard that is being progressed by the organisation as yet. As IFC does not currently support parametrics, we change the fixed dimensions of some objects in the library to be variable and parametric within the trasnformations targeting each CAD platform.

8 Conclusion

This project has produced a polished and feature-rich Object Library web application, back-end repository and CAD tool plugins that form the viable basis for an Australian National Object Library. A suite of test Objects, documentation, demonstration and training materials has also been produced, along with a detailed proposal for Concept of Operations for such a National Library, which can be acted upon by an industry body to take the project from demonstrator to Library.

The project used Agile development methods, with very short development cycle, user-driven feature addition. The extensive use of standard models and model-driven development methodologies and tools has allowed a very small team (one software architect and one software engineer - working at 1.4 full time equivalent person hours - in collaboration with a small team of part-time involved designer BIM modellers, and some project management support from QUT) to create a robust server and application suite that surpassed the planned project objectives.