Proceedings, 5th AOSFST, Newcastle, Australia, 2001
Editors: M.A. Delichatsios, B.Z. Dlugogorski and E.M. Kennedy

THE DEVELOPMENT OF A WEB-BASED DATABASE OF RATE OF HEAT RELEASE MEASUREMENTS USING A MARK-UP LANGUAGE

M. J. Spearpoint

Department of Civil Engineering, University of Canterbury

NEW ZEALAND

ABSTRACT

The application of most computer-based fire models is dependent on the user supplying the rate of heat release data that describes the design fire for a chosen scenario. Having access to a database of rate of heat release measurements assists the user in making their selection. The ability to use that data in a wide variety of general and specialised computer tools enables effective use of the data. This paper examines the current developments in database implementation using mark-up languages and describes how they can be applied to engineered fire protection design. In particular, the increased use of XML technology as a method of information storage, retrieval and manipulation in several engineering related fields are discussed since these developments are very likely to have an impact on fire engineering.

An XML-based schema for the implementation of a database of single and multiple item rates of heat release measurements is presented and an online database has been created using the schema. The database and its underlying schema can be viewed in a web browser. The database can also be queried via a client program using broad search criteria. The relevant matches can be viewed graphically and a selected dataset can be extracted in a format suitable for further processing. A number of transformations have been developed that allow a selected dataset to be converted from XML to an alternate format suitable for commonly used fire models or more general computer software tools.

Although the database currently operates as a stand-alone entity, the work is also aimed towards integration with the developments taking place in interoperability between a wide range of engineering-related software tools. This paper shows where this web-based database fits in with these developments.

INTRODUCTION

Data selection

When using most of the currently available computer-based fire models, it is the effect of the rate of heat release that has the greatest impact on the calculation of the conditions within a compartment[1]. Therefore, the selection of appropriate data is of primary importance.

In many fire-modelling situations the scenario that is being examined does not include a burning item (or items) that exactly matches those that have been tested and measured in the laboratory. Therefore, when searching for an appropriate heat release curve for use in a fire model, the user may not have a specific item or test in mind but may be looking for an item that best matches their particular scenario. For example, in the study by this author[2] the scenario to be modelled included a loveseat but it did not exactly match any of those loveseats that have been tested and reported in the literature. In order to model the scenario, a nearest match item was identified and the heat release rate for that item was used as input to the fire models.

Data exchange

The exchange of data between various electronic tools can be a problem common to any area of modern life where computers are used to store and manipulate data. Recently, and in particular with the explosion in web-based information exchange, this area of data exchange has been of particular importance. Without a standard format for the content and transfer of data between software tools, conversion processes are necessary. Each conversion process may ‘devalue’ the data as the content of the data has to match the lowest common format. Furthermore, ambiguities may occur in the data which cannot be resolved during the conversion The use of software tools across a whole range of engineering disciplines means interoperability between these tools is becoming a critical issue. The ability to efficiently exchange data increases productivity and reduces errors.

Rate of heat release catalogues

Although there is a range of rate of heat release data available in the published literature, it can be time consuming to find the data for a desired item and it may require the user to digitise the heat release curve before it can be used in a model. The concept of having a database of test data and also software to convert that data into different formats are not new ideas. Several attempts have been made at achieving both. Examples of databases include those found accompanying fire models as part of the package[3,4], the work of Särdqvist[5], the Fire Data Management System[6] (FDMS) and the online database initiated at NIST[7]. An example of conversion is the DCS program developed at SINTEF[8]. Previous unpublished work by this author[9] of a database of heat release curves and a conversion program also fit into these examples. It is not necessarily the intention of the work described in this paper to replace this earlier work, but instead to examine how the latest developments in information technology can be applied to the storage and exchange of such material.

Requirements

A number of requirements for a database of rate of heat release measurements can be identified:

- The use of a standardised data description technology for the database structure.
- Make the database widely available and easily accessible.
- Allow broad search criteria to be applied to the database.
- The ability to view, extract and process selected data using a variety of general and specialised software tools.
- Allow the scope of the database to be extended without making earlier software redundant.
- The ability to integrate the database with other tools across a range of disciplines.

The success of previous efforts to create catalogues of rate of heat release measurements and tools to manipulate them were limited by the technology available at the time. The web provides a convenient resource for storing and distributing a database of rate of heat release measurements and the development of associated mark-up languages offer potential ways in which the above requirements can be met.

EXTENSIBLE MARK-UP LANGUAGE

Specification

Extensible Mark-up Language (XML)[10] is a meta mark-up language that provides a format for describing structured data. This facilitates more precise declarations of content and more meaningful search results. XML is defined by the World Wide Web Consortium (W3C) and is independent of software tools or vendors. XML is similar in many ways to HTML and HTML can be viewed as a special subset of XML. However, unlike HTML, XML separates the data from its presentation and processing which allows the integration of data from diverse sources.

There are a number of advantages in using XML as the basis for a database:

- The extensibility of the XML language means that new fields can be easily added. If a client application does not recognise a field, it is simply ignored. An XML document can therefore contain much more information than a particular client application requires but this excess information does not render the client inoperative.
- The XML document can be easily and efficiently searched via a pattern query using a number of techniques and programming languages. Libraries and parsers for creating and manipulating XML are already available.
- The document structure is in a human readable form allowing it to be edited in a text editor. However, more sophisticated editing tools specifically designed for XML are also available.

Document structure

An XML document consists of a collection of nodes. Nodes can be one of several types including elements or attributes. Unless an element is empty, it consists of a pair of tags plus its content. XML allows the definition of an unlimited set of tags and XML provides a data standard that can encode the content, semantics and schema of a dataset. Elements can have any number of sub-elements nested to any depth and any amount of text can be included. Additional information can be added to an element by using attributes. Attributes have a name and a value. Values are optional and an attribute can exist as a name only.

Schema

The schema describes the structure of the XML document in terms of the relationship between elements and attributes and the types of data that can be stored by them. A specific XML document can be automatically checked against a schema for conformity. An XML document can be described as 'well-formed' and also 'valid'. A well-formed document is one that conforms to the physical structure of XML but it has not been validated against a schema. A valid XML document is a well-formed document that has been verified against a specific schema. A schema can be automatically inferred from the structure of an XML document. If the document is subsequently modified, the schema can be updated appropriately. This allows a more flexible approach to the development and enhancement of a database without losing the ability to perform conformity checks.

XML Schema is an agreed "schema of schemas" published by W3C[15]. It is an XML document that defines the content and semantics of particular vocabularies and the structure of XML documents that use those vocabularies.

Transformations

One of the primary aims of XML is to enable the efficient sharing of data and documents by agreeing on a single schema for a particular field of interest. However, the likelihood of a single vocabulary fitting the needs of all interested organisations is very small. In order to promote interoperability between distinct XML vocabularies, transformations are required. Extensible Style-sheet Language Transformations (XSLT) describe these transformations and can be used to convert an XML document into another XML vocabulary or any other text-based document such as HTML, comma-separated values (CSV) etc.

Figure 1 shows the relationship between an XML document, its schema (where one exists) and how an XSLT can be used to create a new document. Because of the way in which XML separates data from its processing and presentation, each component can exist remotely as separate entities. This ability to separate the components is one reason why XML technology is particularly suited to web-based data exchange.

XML use in engineering

The technical merits of XML discussed above have been recognised by several other engineering fields and developments in these areas will most likely have an impact on fire engineering. Most importantly is the work being carried out by the International Alliance for Interoperability (IAI)[11]. This international group of engineering professionals, software developers and researchers is developing a family of schemas (referred to as the Industry Foundation Classes or IFCs) that permit an object-oriented description of many aspects of buildings and related services. Although their work does not use XML directly, they recognise its importance and implement an XML version of their schemas[12]. Other areas in which XML schemas are being developed include the representation of information in the Architecture, Engineering and Construction (AEC) industry[13] and a schema that facilitates the exchange of data created during the land planning, civil engineering and land survey process[14]. These examples illustrate the diverse uses of XML technology in engineering and were an important consideration in the selection of XML for the development of the rate of heat release database described in this paper.

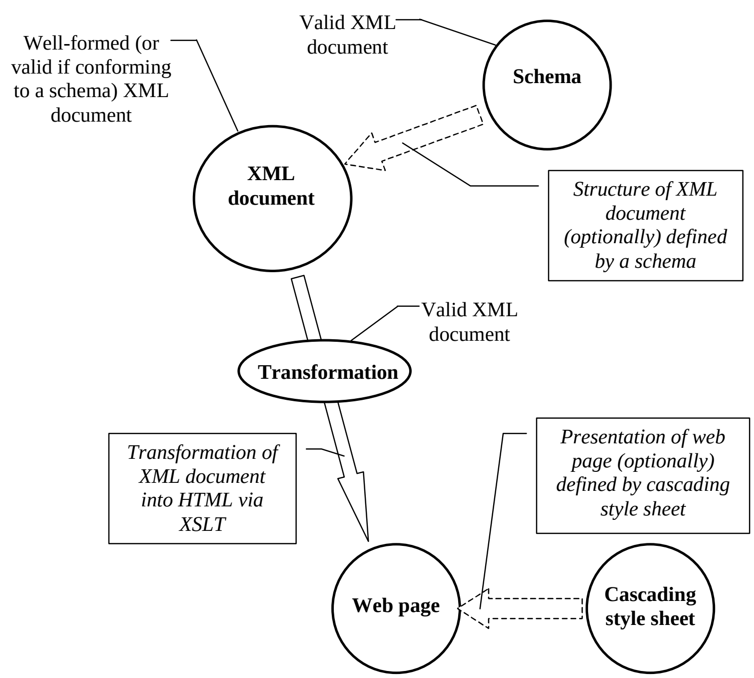

Figure 1: The relationship between an XML document, its schema and a transformation to another document (in this case a web-page).

RATE OF HEAT RELEASE DATABASE SCHEMA

Description

The development of schemas is one of the primary tasks required for interoperability. A schema for a database of rate of heat release measurements has been created and is referred to as the "FireBaseXML" schema. It includes fields that describe a particular database plus one or more records (Figure 2).

Each record in the FireBaseXML schema can be broken down into three general components:

- The test data. This includes the rate of heat release data plus optional data for the initial mass of the item and its average heat of combustion.

- The test description. A short (one line) description of the test arrangement and a more detailed multi-line description of the test are stored with each record. Each record also has an associated set of 'item attributes' that describe the entry in generic terms. By using these generic attributes associated with each record, the user can make context meaningful searches for a particular item. For example a search can be made for all types of single seat chair, or alternatively for only upholstered single seat chairs.

  The classification of groups and categories used in this database is similar to the BSAB format quoted by Särdqvist[5]. Items are classified into generic groups and then each group is sub-divided into a number of different categories. For example, groups include: *chairs, easy chairs, love seats, sofa*s and the chairs group can be categorised as *metal, plasti*c or *wood*. This system of classification and the use of XML mean that additional groups and categories can be easily added.

- The source publication. Each record in the database includes details of the source publication from where the data was taken from. Where an online version of the source document is available, a link to the electronic document can be specified.

The FireBaseXML schema can be easily extended to allow for data obtained from small-scale test methods such as the Cone Calorimeter or for the inclusion of other suitable information such as toxic gas yields by defining additional nodes in the FireBaseXML schema.

Schema design

The flexibility of XML means that there is not necessarily only one way to develop a document structure so as to achieve a particular aim. In developing the FireBaseXML schema, assessments were made as to the most appropriate structure.

Specific attention was paid to the method in which the test data should be stored. Several different options were considered in which the level of detail provided by elements was examined. A comma-separated format that provides sufficient descriptive detail but minimises the overall size of the document is used. The general form of the structure for data is shown in a representational form in Figure 3.

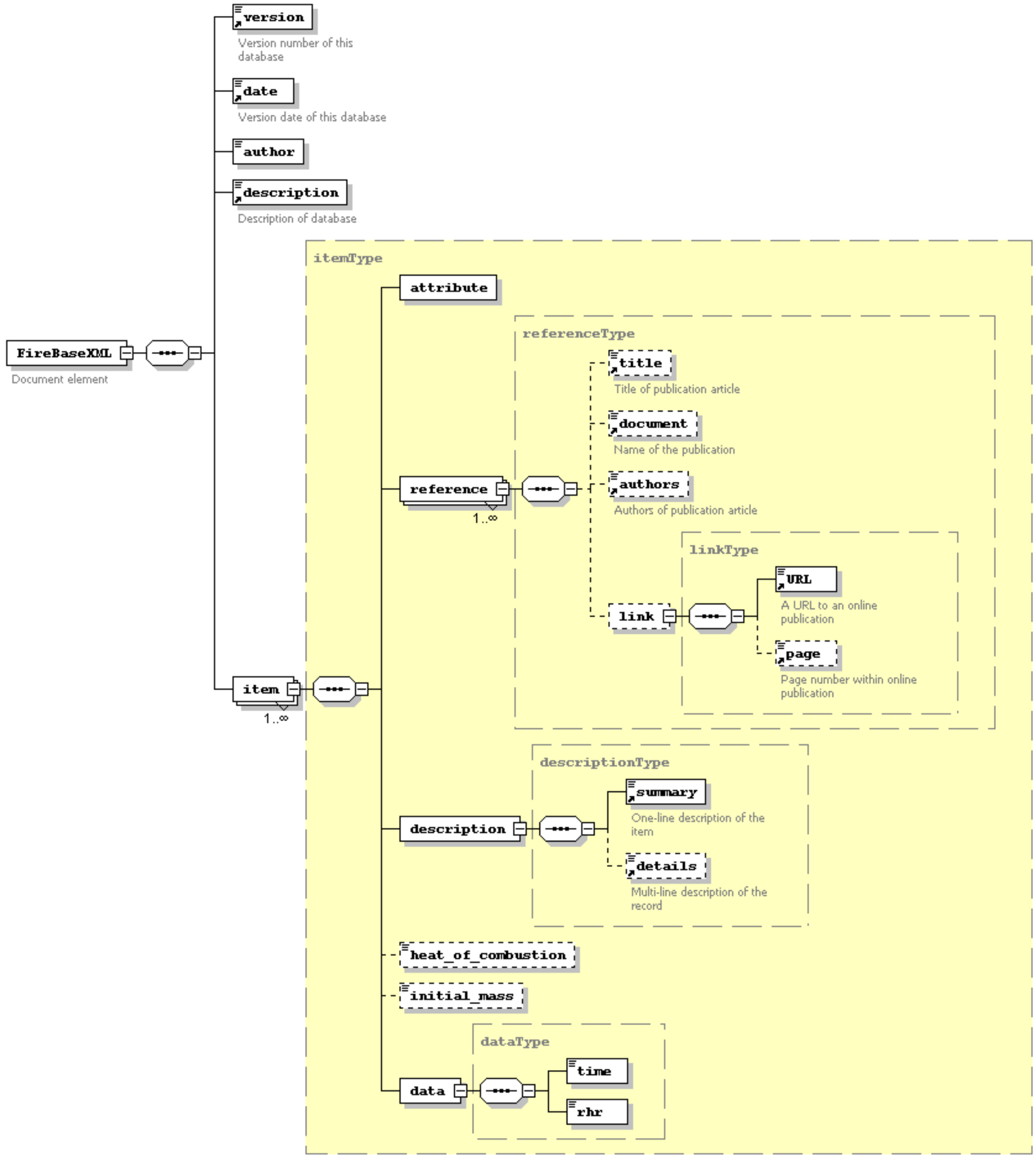

Figure 2: An overview of the FireBaseXML schema. Optional elements are shown by boxes drawn with dotted lines.

```
<dataset>
  <entity name='T1'>
    <data type='def-type' units='def-unit'> 0, 1, 2, 3 </data>
    <data type='def-type' units='def-unit'> 20, 22, 28, 30 </data>
  </entity>
  <entity name='T2'>
    <data type='def-type' units=' def-unit'> 0, 1, 2, 3 </data>
    <data type='def-type' units='def-unit'> 293, 298, 299, 305 </data>
  </entity>
</dataset>
```

Figure 3: Representational structure for data elements.

The `'def-type'` attribute could be 'time', 'temperature' etc. and the `'def-unit'` attributes associated with these types could be for example 's' for time and 'C' or 'K' or some other appropriately defined unit for temperature. The structure allows additional data types to be added to the schema by defining additional 'type' attributes. Units have to be explicitly defined so that ambiguity cannot occur.

The FireBaseXML schema was also designed to provide tight control over the specification of units to data but allowing for multiple forms of units if required. The units and measurement type are set as attributes on the main element. In this way the schema can be defined such that only valid measurement types and units can be associated to a particular variable. Figure 4 shows the schema fragment for the `<heat_of_combustion>` element.

Line 3 specifies the restriction for the content of the element as a number. Two attributes are associated with the element: `"units"` and `"type"` (lines 4 and 12). The units must be specified since their use is required and a list of restricted units are given in lines 7 and 8.

```
1: <xsd:complexType name="heat_of_combustionType">
2:     <xsd:simpleContent>
3:        <xsd:restriction base="xsd:number">
4:           <xsd:attribute name="units" use="required">
5:              <xsd:simpleType>
6:                 <xsd:restriction base="xsd:string">
7:                    <xsd:enumeration value="J/kg"/>
8:                    <xsd:enumeration value="kJ/kg"/>
9:                 </xsd:restriction>
10:             </xsd:simpleType>
11:          </xsd:attribute>
12:          <xsd:attribute type="xsd:string" name="type"
              use="fixed" value="available_energy"/>
13:       </xsd:restriction>
14:    </xsd:simpleContent>
15: </xsd:complexType>
```

Figure 4: The FireBaseXML <heat_of_combustion> schema fragment.

Additional unit enumerations could be appended to this list if necessary or the list could be reduced to a single enumeration so that only one form of unit is allowed. The type attribute (line 12) specifies the type of measurement and fixes a value to this type (in this case 'available energy'). Thus a valid <heat_of_combustion> that conforms to the schema would be.

```
<heat_of_combustion type="available_energy"
            units="J/kg">15000</heat_of_combustion>
```

This schema structure means that association of an incorrect unit is automatically discovered during the validation of an instance of the FireBaseXML schema. Interrogation or extraction of the data can include a check of the units that the data is stored in and appropriate action taken to convert those units to another type if necessary.

IMPLEMENTATION

A FireBaseXML database

The rate of heat release database developed in this study is a valid XML document that conforms to the FireBaseXML schema. The database consists of a collection of rate of heat release rate measurements taken from various sources in the literature. Currently this database resides on a server at the University of Canterbury. However, one of the advantages of this web-enabled technology is that several organisations may want to publish their own databases using the FireBaseXML schema as their basis. Software agents could search one or more of these databases concurrently and return a combined set of results with any replicate information removed.

Web integration

The close association of XML and the web means that a FireBaseXML database can be integrated with an XML-compliant web browser. XSLT can be used to manipulate the database to dynamically generate web pages that allow a number of different "views". For example, Figure 5 shows a web page that lists the items in a FireBaseXML database and an item extracted from the database is shown as a stand-alone XML document.

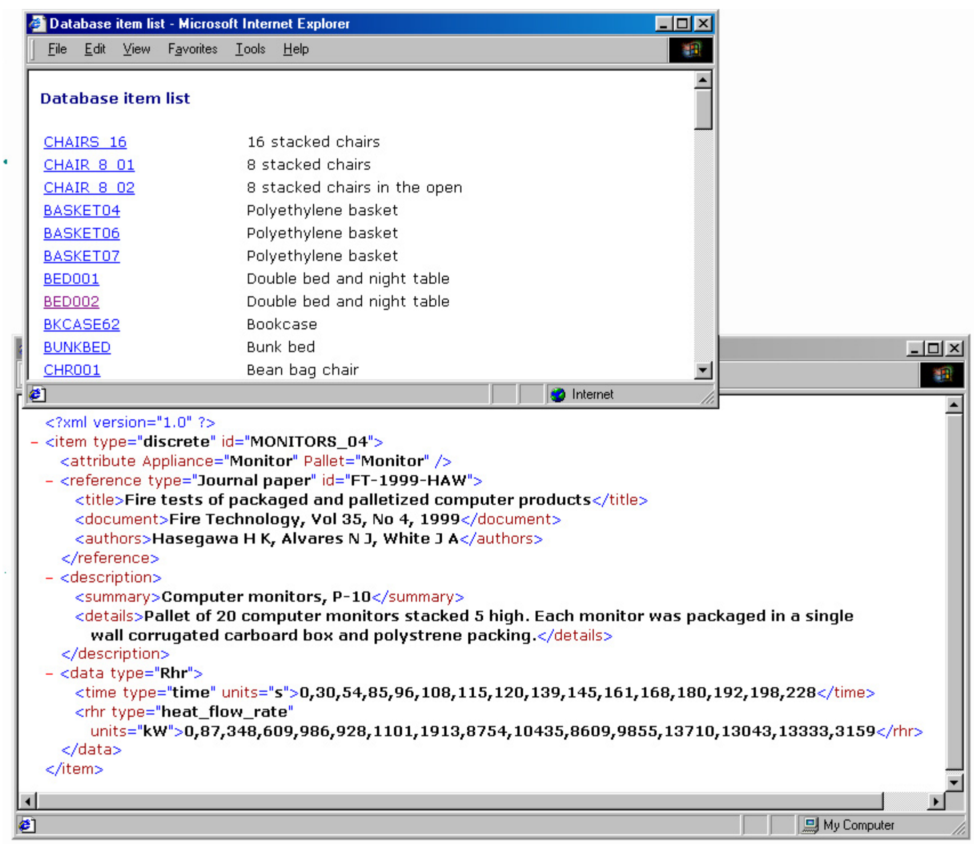

Figure 5: Web browser views of the records in a FireBaseXML database (top) and an extracted record (bottom) as an XML document.

Specific transformations

A number of specific XSLT documents have been developed so that records selected from a FireBaseXML database can be exchanged with general and specialised computer tools. These transformations are published on a web-server separate from any FireBaseXML database and remote from the eventual end-user. This has the advantage that a transformation can be delivered online at the time it is required. This means that the end-user does not have to ensure that their transformations are up-to-date. The transformation author need only upload a revised version of the transformation onto the appropriate server and this is delivered to the end-user when a transformation is requested. Currently transformations to process FireBaseXML records into files that can be used by a text editor, spreadsheet, FPETool[4] and a web browser have been completed. Additional transformations can be made available as the need arises.

The SelectFire client

The SelectFire program is a client application that allows a user to select a FireBaseXML database (either online or stored on a local disk) and then perform operations on that database. As a new database is accessed, the SelectFire program automatically scans it for the generic item attribute definitions allowing the program to dynamically construct the list of searchable item attributes. The user can therefore search for a particular generic group of items, view the rate of heat release curves of each match, consult any documentation related to a particular match if an online resource exits and make an appropriate selection (Figure 6). A selected match can be extracted from the database using n specified transformation so that it can be used in some further processing program whether that be a fire model or a more general package such as a spreadsheet.

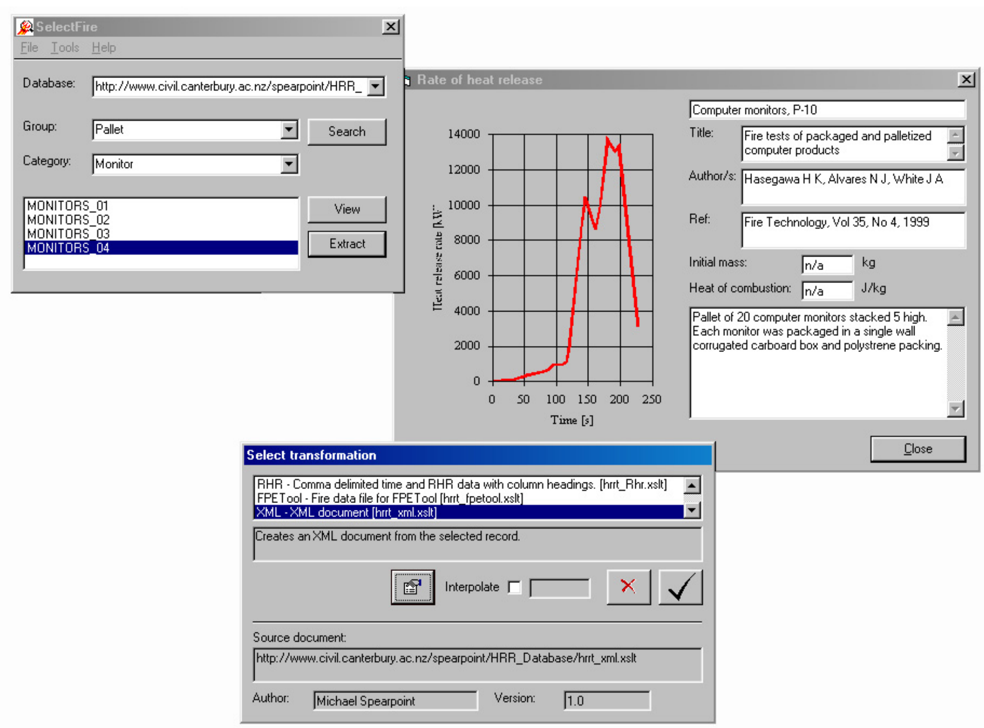

Figure 6: The SelectFire client with a selected record being viewed prior to its transformation.

Integration framework

Mowrer[16] proposed the integration of Computer Aided Design (CAD) information with fire specific properties of a building and its contents. The key features identified by Mowrer required by CAD systems to permit integration included object-orientation, the association of attributes with objects and the ability to extract attributes from a CAD-developed drawing database.

The work described in this paper and that being carried out by the IAI mean that the opportunity to implement the proposals made by Mowrer can be realised. A building and its contents can be described using the IFC schema and delivered as an XML document. The contents of the building can then be linked to an XML database of fire specific information. This can then all be delivered to a fire modelling application ready for any additional user input prior to computation. Although the development and implementation of the FireBaseXML schema is currently a stand-alone entity, it is intended that it will be a source of fire specific data that can be integrated with the IAI schema developments. Figure 7 shows how fire engineering-related aspects can be integrated within the IAI architecture.

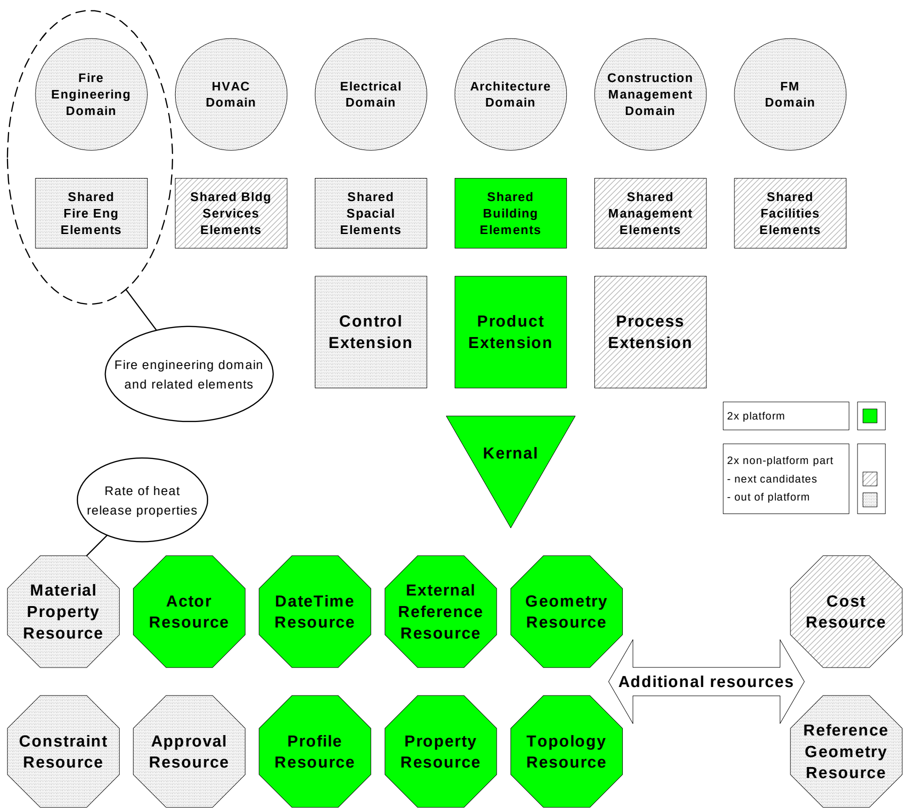

Figure 7: Integration of fire engineering with the IAI IFC release 2x architecture, adapted from Liebich & Wix[11].

Integration with currently available fire engineering tools is already being undertaken. Appropriate fields in FDMS format data can be automatically mapped to the FireBaseXML schema by using a simple program. The ability to directly query a FireBaseXML database from the BRANZFIRE model[17] is being explored. Clearly the integration with a wider range of fire engineering tools may require access to program source codes.

CONCLUSIONS

The development of integrated computer tools means that there needs to be widespread agreement on the way in which data is exchanged. A set of requirements for a database of rate of heat release measurements have been identified and the use of XML as a means of storing and retrieving such data meets these requirements. Even if XML is not currently used by developers of fire models or those that supply data for these models, it is important for fire engineering to keep up-to-date with emerging technologies. As XML data exchange becomes more widespread then fire engineers will necessarily begin to interact with those in related disciplines that have embraced the technology.

Currently intermediate transformations are necessary to take a rate of heat release curve from a FireBaseXML database and use it in a fire model. By making the structure of the database open and by using a standard format it is possible that future fire models can interrogate such a database from within, without the need for the intermediate step.

REFERENCES

1. Babrauskas V., Peacock R.D., Heat release rate: The single most important variable in fire hazard. Fire Safety Journal. 1992 18 (3) 255-272.

2. Spearpoint, M.J., Mowrer F.W., McGrattan K., Simulation of a single compartment flashover fire using hand calculations, zone models and a field model. Third International Conference on Fire Research and Engineering. Chicago, October 1999.

3. Bukowski R.W., Peacock R. D., Jones W.W., Forney C.L., Software user's guide for the HAZARD I fire hazard assessment method. NIST Handbook 146, Volume I. Gaithersburg, National Institute of Standards and Technology, 1989.

4. Deal S., Technical reference guide for FPEtool version 3.2. NISTIR 5486-1. Gaithersburg, National Institute of Standards and Technology, April 1995.

5. Särdqvist S., Initial fires – RHR, smoke production and CO generation from single items and room fire tests. ISRN LUTVDG/TVBB-3070-SE, Lund University, Sweden, April 1993.

6. Portier R.W., Peacock R.D., Reneke P.A., Data structures for the Fire Data Management System, FDMS 2.0. NISTIR 6088, National Institute of Standards and Technology, Gaithersburg, 1997.

7. Fires on the web. http://fire.nist.gov/fire/fires/fires.html (accessed May 2001).

8. Lønvik L.E., Opstad K., DCS – Data converting system. SINTEF, Norway, 1992.

9. Spearpoint M.J., A collection of rate of heat release data suitable for computer fire models, Building Research Establishment, Garston, Private Communication, 1993.

10. Light R., Presenting XML. Sams.net, Indianapolis, IN, 1997.

11. Liebich T., Wix J. (ed.), IFC technical guide, Industry Foundation Classes – Release 2x, International Alliance for Interoperability, October 2000.

12. Liebich T., XML schema language binding of EXPRESS for ifcXML. MSG-01-001(Rev 4), International Alliance for Interoperability, 2001.

13. A framework for electronic communications for the AEC industries. aecXML Working Group. http://www.aecxml.org/docs/aecwhite.doc (accessed May 2001).

14. LandXML schema documentation (version 0.88). http://www.landxml.org/ (accessed May 2001).

15. XML Schema. World Wide Web Consortium, http://www.w3.org/XML/Schema (accessed May 2001).

16. Mowrer F.W., Williamson R B. Room fire modeling within a computer-aided design framework. International Association for Fire Safety Science. 2nd International Symposium. June 13-17, Tokyo, Japan, pp.453-462, 1988.

17. Wade C., BRANZFIRE: Engineering software for evaluating hazard of room lining materials. Proc. 8th International Interflam Conference, Volume 2. Interscience Communications Ltd., London, England, pp.1147-1152, 1999.