FUTURE DIRECTIONS FOR IFC-BASED INTEROPERABILITY

SUBMITTED: December 2002
REVISED: April 2003
PUBLISHED: July 2003 at http://www.itcon.org/2003/17
EDITOR: Väino Tarandi

Thomas Froese, Associate Professor
Dept. of Civil Engineering, University of British Columbia
email: tfroese@civil.ubc.ca

SUMMARY: *Interoperability is one of the major themes of research and development in information technology for the architecture, engineering, construction, and facilities management industries. A model-based approach to interoperability requires information structures that are standardized throughout the industry. The Industry Foundation Classes have been developed to provide this data exchange standard. The technology for exchanging information using Industry Foundation Classes has now been established, but many areas require additional development before comprehensive interoperability solutions are reached. These areas include: extending the scope to include a broader range of project information, for more types of projects, and more types of information; developing the exchange mechanisms layer below the data standards and the formalized transactions layer above; developing the range of software applications that implement model-based interoperability; and re-examining project management practices based on new integration technologies.*

KEYWORDS: *IFC, Interoperability, Future directions.*

1. INTRODUCTION

Architecture, engineering, construction, and facilities management (AEC/FM) are information intensive industries, and are increasingly dependant upon effective information technologies (IT). Various computer tools are used to support almost all AEC/FM design and management tasks, and the information entered into all of these tools describes the same physical project. However, this information is passed from one tool the next by producing paper-based or electronic documents which can only be interpreted by people, who must re-enter relevant information into the next computer tool. This manual data re-interpretation and entry is a non-value adding activity, can often introduce errors into the project, and inhibits the use of better computational tools. To address this problem of information communication and exchange, the topic of interoperability has been taken up as one of the primary areas for research and development in IT for AEC/FM. Interoperability—the ability for information to flow from one computer application to the next throughout the lifecycle of a project—relies on the development and use of common information structures throughout the AEC/FM industry.

1.1 The Industry Foundation Classes

The need for standard data exchange languages has been widely recognized throughout the AEC/FM IT community and a large-scale international effort has taken up this challenge. The *International Alliance for Interoperability (IAI)* (International Alliance, 2002; BLIS, 2002) is a global coalition of industry practitioners, software vendors, and researchers (over 600 companies around the world) working to support interoperability throughout the AEC/FM community by developing the *Industry Foundation Class (IFC)* standard. The IFCs are a high-level, object-oriented data model for the AEC/FM industry. The IFCs model all types of AEC/FM project information such as parts of a building, the geometry and material properties of building products, project costs, schedules, and organizations, etc. The information from almost any type of computer application that works with structured data about AEC building projects can be mapped into IFC data files. In this way, IFC data files provide a neutral file format that enable AEC/FM computer applications to efficiently share and exchange project information.

The IFCs, initiated in 1994, have now undergone four major releases, and commercial software tools for the AEC industry (such as Autodesk's Architectural Desktop, Graphisoft’s Archicad, Nemetschek’s Allplan, Microsoft's Visio, and Timberline Precision Estimator) are beginning to implement IFC file exchange capabilities.

1.2 Status of the IFCs

From the point of view of the basic technical ability to exchange AEC/FM information, it can be said that the IFCs have now been established as a viable interoperability technology. Significant portions of the IFCs are now mature, stable standards and numerous prototype and early commercial systems have demonstrated their extensive information exchange capabilities. From other points of view, however, the IFCs are still in a very early stage of development. Only recently have IFC-compatible software applications started to become commercially available, and, as yet, the IFCs have seen almost no use actual use in industry.

The current status and capabilities of the IFCs have been widely discussed is many forums, and will not be addressed further in this paper. Rather, the focus here is on the areas in which the technology itself requires further development to provide a comprehensive interoperability solution.

2. NEXT STEPS IN MODEL-BASED EXCHANGE

Although the basic product modeling capabilities of the IFCs have reached a level of maturity and stability that is sufficient to provide basic interoperability in a number of core project areas, this capability is far from a complete solution to the interoperability challenge. This paper outlines several areas in which the basic IFC approach to interoperability requires further development and extension. These extension areas are drawn from a synthesis of several IFC-related research and development projects ongoing at the University of British Columbia.

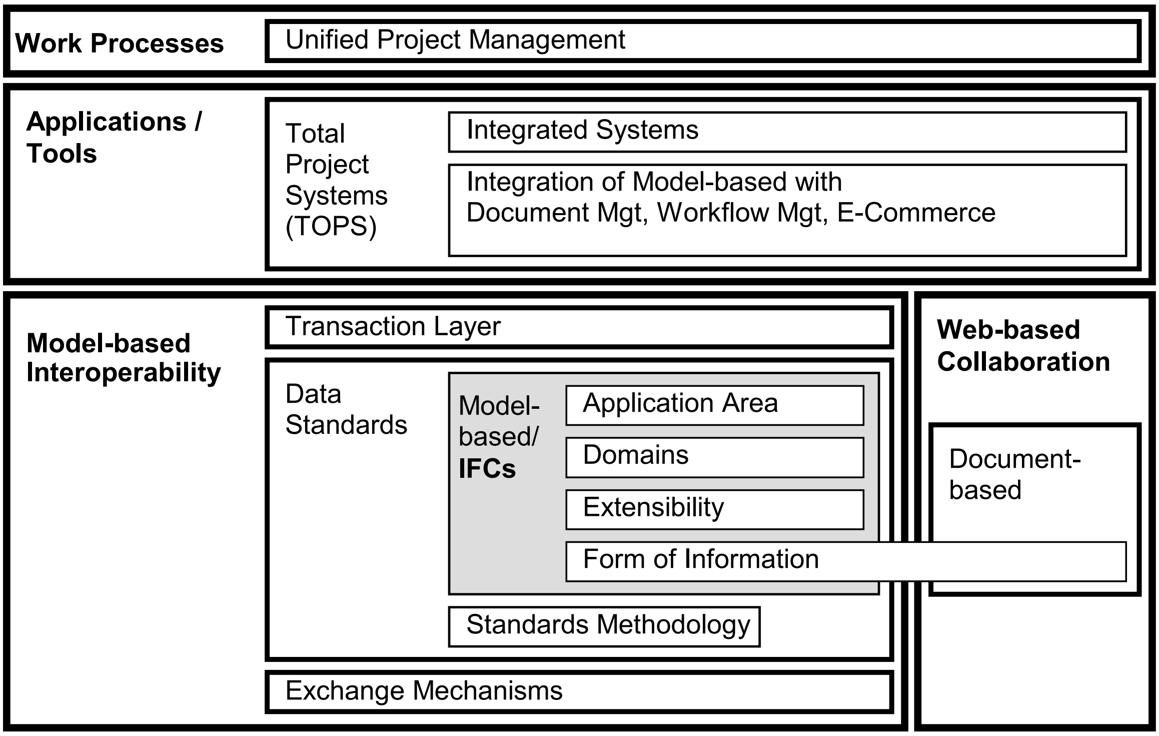


FIG. 1: The topic of IFCs within an overall IT framework

Fig. 1 illustrates the relationships between a basic IFC data exchange capability and the many areas of further extension discussed in this paper. The topic of “Model-based/IFCs” shown in a box in the middle of the figure represents the basic model-based data standards provided by the IFCs. Several boxes are shown within the IFCs box, representing areas in which the scope of the basic IFCs should be extended. Other boxes are shown encompassing the IFCs or adjacent to them, illustrating broader areas that must also be developer further in order to provide a complete solution to interoperability. Each of these boxes will be addressed in the following section of this paper. Fig. 1 is intended to illustrate our specific areas of research and development interest, yet it does provide some more generic structure to interoperability technologies.

2.1 Application Areas

Systems that implement IFC-based data exchange should move beyond product data into project management data.

The scope of the IFCs includes product information: it models the physical parts that make up a building, including the semantic identification of all the building's systems and elements, their geometry, design properties, etc. Within the IFC's, the representation of certain building systems (e.g., basic architectural features such as walls, doors, floors, etc.) is fairly extensive while other building systems (e.g., electrical systems) have received very little development to date (IFC release 2X2, currently in development at the time of writing, will extend many of these areas).

The scope also includes non-product information, such as costs, schedules, people and organizations, resources, documents, etc. The largest effort to date in implementing IFCs has been in the area of product information, such as building geometry. Many of the IFC-compatible systems that have been developed to date do work with non-product information. However, in almost all cases, these systems use product information as an input to non-product applications. For example, the product model is used to input geometry into an energy simulation application, or to input a quantity takeoff into an estimating application. Very few systems have written non-product information back into IFC files and used these to exchange non-product data.

We are primarily interested in the use of IFCs to support project management-related tasks. We have worked within the IAI on the development of the project management components of the IFCs, and have developed prototypes systems for IFC-based exchange of project information relating to: costs (Froese et al., 1999), schedules (Froese and Yu, 1999), facilities management information (Hassanain, Froese and Vanier, 2001; Yu, Froese and Grobler, 2000), project specifications, materials management, and references to external documents.

The exchange of cost data provides an interesting illustration. The IFCs support the exchange of various types of costs associated with objects in the project model, and the assembly of these costs into costs schedules. As one of the few commercial IFC-compatible software products that supports the exchange of non-product information, Timberline Software's PECAD product (Timberline, 2002) produces cost estimate data from an IFC file and writes the costs back into the model. To date, however, we are not aware of any other applications that can read and use these costs. Indeed, it may be argued that costs are generally private information, and there is no great need for the exchange of cost information. We do not accept this argument. Costs information is often sensitive, but costs, prices and other types of financial information are central to much of the activity carried out on a project and many information transactions involve some type of cost information (both within an organization and among external participants). We have recently initiated a research project that assesses the range and demand for cost-related data exchange on AEC/FM projects.

Another example of interoperability of non-product information is provided by our recent work on facilities management. Fig. 2 provides a high-level view of the processes involved in maintenance management. We have developed a simple integrated maintenance management prototype (Hassanain 2002) that illustrates the opportunities for IFC-based interoperability within this domain. The first step involves identifying all of the elements within some inventory of buildings (e.g., the roofing sections, mechanical devices, flooring surfaces, etc.) that need to be treated as assets for active maintenance management. Here, IFC project data can be used to import product models of the buildings, and the elements of those buildings that require maintenance management are identified. The system will then treat these objects as assets, represented using the IfcAsset class. Maintenance management involves tracking the condition of these assets relative to prescribed performance requirements, so the second step involves describing these performance requirements. The system represents these requirements using the IFC property set mechanism. In the third step, the actual condition of the assets relative to the performance requirements is assessed and tracked within the system. Specialty condition assessment software is often used for this purpose. In the integrated prototype, IFC data exchange is used to load the inventory of assets into a condition assessment tool, and then to return the measured asset condition information back into the integrated system. Based on all of this information, the fourth step involves strategic planning to decide what maintenance activities to carry out, and when. This is a potentially vast task that was excluded from the scope of this prototype. In the final step, however, the planned maintenance activities are carried out and must be managed on a day-to-day basis. In this step, the integrated prototype used IFCs to export planned maintenance activities to construction scheduling software. A similar approach could be used to interact with costing and resource planning systems.

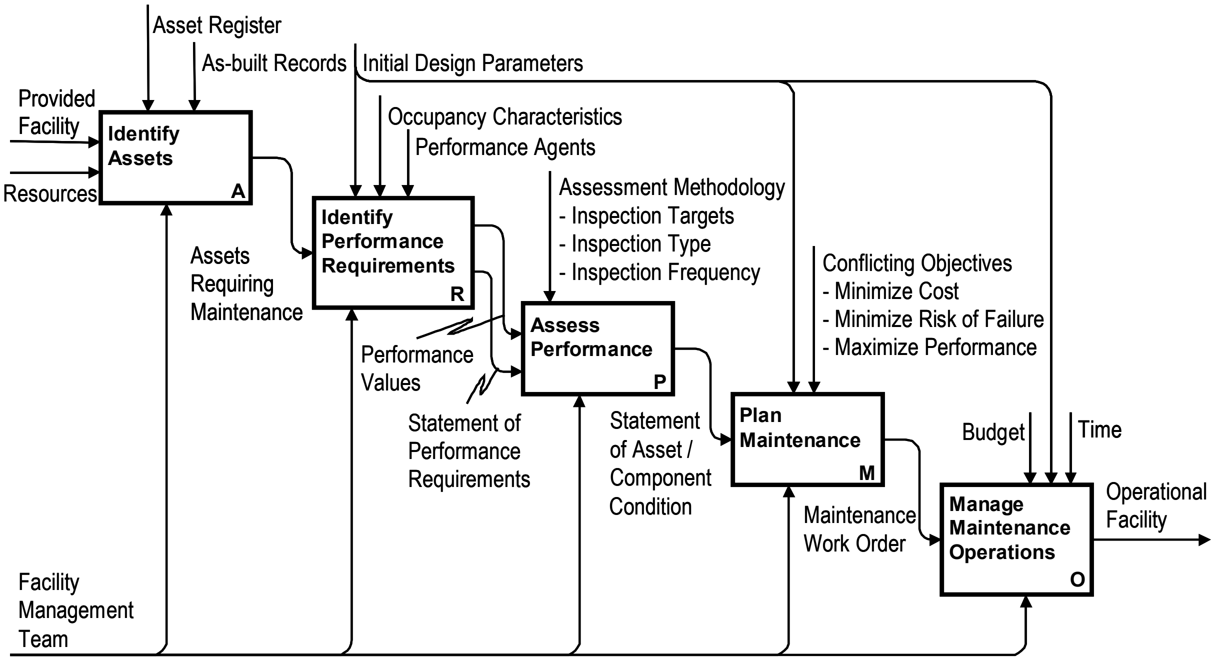


FIG. 2: Processes involved in maintenance management (from Hassanain 2002)

2.2 Domain

The scope of the IFCs should be extended beyond buildings to include a broad range of civil infrastructure.

The IFC model specifically addresses building construction. In many ways, this is an arbitrary boundary to the scope of the standard, and much value could be found in extending the scope to include the entire built environment. The contents of the IFC model can be divided into three broad areas: 1) a high-level core model that defines basic concepts, relationships, and modeling constructs, 2) a detailed product model for buildings, and 3) a generic model of non-product, project information. Much of section 2 is specific to buildings and would not be very relevant to other domains. However, most of sections 1 and 3 are just as relevant to any other type of civil infrastructure project as they are to buildings. Thus, to address interoperability for projects such as road building, underground utility maintenance, or bridge construction, it would be reasonable to extend the IFC model to include these types of projects within their scope.

Furthermore, even building projects involve the exchange of information from beyond the scope of the specific building itself, such as surrounding landscaping, connections to municipal services, planning issues that consider the building with respect to its surrounding neighborhood, etc. For example, some cities (such as Berlin) are working with 3D models of all buildings within the city core, and the IFCs could be extended to encompass all of this regional planning information.

Some of the challenges that will have to be addressed in order to extend the scope of the IFCs to civil infrastructure domains include developing the detailed product models for non-building projects; developing a basic representation scheme for projects that are essentially linear-based (e.g., roads) or area-based (e.g., earthworks) rather than the element-based approach underlying the IFC building product model; interoperability with GIS-based systems; and approaches to address the potential sprawl of a standard that is already vast in its scope (this topic is discussed later in this paper).

We are currently involved in projects that combine parts of the IFCs with data models being developed to support highways, earth works, and bridge structures (Halfawy et al. 2002; Froese and Halfawy, 2002), and several other groups around the world are also pursuing some of these extensions.

2.3 Extensibility

Systems that implement IFC data exchange should move beyond a fixed scope data structure and implement the capabilities that exist within the IFCs to work with dynamically extensible data, such as additional designer and supplier information.

The IFCs contain a detailed project model for representing construction projects. Never-the-less, this model still represents only a small portion of all of the important data that can exist for any project element. To accommodate information that is not explicitly modeled, the IFCs have a mechanism for representing additional object data through property sets. Property sets can be defined in advance, through an agreement between two software vendors or by representatives from specialty group within the industry, for example (and many systems that have implemented IFC data exchange use these pre-defined property sets). Given appropriate software, property sets could also be defined dynamically at any time by end users.

We are participating in a project by the IAI's North American Project Management Domain Committee entitled "PM-3, Material Selection, Specification, and Procurement", which is reviewing and enhancing the ability to extend the explicit IFC model. The focus of this project is on product specifications, which are generally analogous to "product data sheets" that describe the myriad of individual products used in building construction (a distinction is made between these product specifications and the more broadly scoped project specification documents used to fully specify the design of a building). As illustrated in Fig. 3, this project considers the process through which a designer might specify properties required for products (creating a "requirement" property set associated with an IFC object). Additionally, product manufactures and suppliers might use a similar method to describe the properties of their products (the "available" property set).

With this information, systems could be developed to help support the product selection process, searching available products for those that meet the design requirements. Once a specific type of product is selected, an instance of the product is procured and used on the project. This will lead to an "as build" property set associated with the IFC object. For the requirement and available property sets, properties might be expressed as a range of values, whereas the as-built properties would normally involve exact property values. The approach must support processes where there is only a partial match between required and available properties, where alternatives are proposed, etc. This work is also relates to efforts within the Omniclass Construction Classification System (OCCS) to develop classification systems for the types and definitions of properties that might be used in defining products (Omniclass Table 12) (OCCS Development Committee 2002).

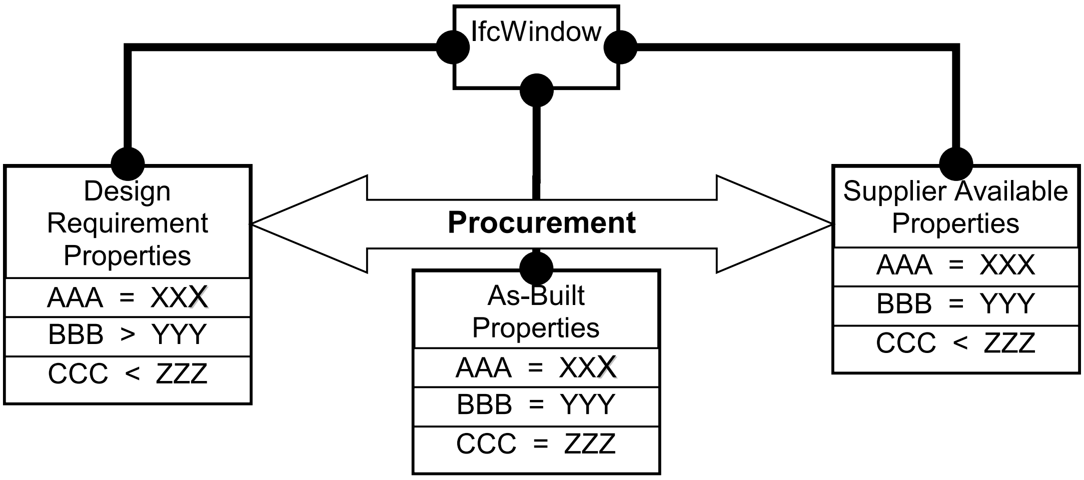


FIG. 3: Material selection and specification properties associated with an IFC object.

2.4 Form of Information and Document Management

Solutions for integration should address not only homogeneous, fully-structured, model-based information, but should extend to include a heterogeneous mix of many forms of information.

The IFCs deal with data that are fully structured according to a common standard. However, most information available on AEC/FM projects is unstructured or semi-structured documents (e.g., Word documents, spreadsheets, photographs, etc.). To fully address the IT interoperability needs of the AEC/FM industry, IFC-based approaches must find ways of integrating the structured model-based and the unstructured document-based worlds (Kosovac, Froese and Vanier, 2000).

There are several basic avenues for interfacing model-based and document-based information, corresponding to increasingly sophisticated approaches to document management.

- At the simplest level, document management need involve nothing more than a repository for placing and retrieving unstructured documents. Given only a basic ability to reference a document, the IFCs can associate these external document references with any object. This could be used, for example, to use a 3D view of a building as an index and graphical interface for accessing documents based on the parts of the building to which they relate.
- A more advanced approach to document management uses metadata to index and access documents. Metadata is structured data that describes a (typically unstructured) document. Generic standards exist for structuring document metadata (Dublin Core Metadata Initiative, 2002). On a construction project, however, there is good opportunity to define more AEC/FM-specific metadata (ISO, 2000). It should be possible to directly map any AEC/FM document metadata to IFC objects (Underwood and Watson, 2003). For example, if an unstructured text document is indexed according to project, phase of project, part of the building, subcontract, etc., all of this information forms metadata which can be described in terms of IFCs. This approach provides a direct interface between the world of model-based systems and the world of document management.
- An even more advanced approach to document management involves techniques such as text processing or picture recognition to extract partial structured and semantic information from unstructured documents. Such systems could again be tightly linked to IFC data by using the IFCs to structure the information extracted from the documents (Caldas and Soibelman, 2003).

The ability to link model-based and document-based information creates many opportunities for integration and useful applications, as discussed later in this paper. Furthermore, in a similar manner to the way that model-based information can be associated with document metadata, it can also be associated with on-line analytical processing (OLAP) approaches, providing a path to exploit data mining technologies within IFC model-based systems.

2.5 Web-Based Collaboration

Creating a link between model-based information and document-based information is perhaps equally important for a less-direct reason. We contend that there are currently two great trends in IT for AEC/FM: Model-based systems and web-based collaboration. Document management is one of several cornerstone technologies within web-based collaboration, along with other areas such as e-commerce. The link between model-based and document-based information is illustrative of the ways that these two important IT trends can be brought together.

2.6 Standards Methodology

The development methodology used for creating the IFCs should become more modular, allowing specific, narrowly scoped data exchange scenarios to be developed and deployed quickly as part of a loosely coupled overall IFC standard.

The IFCs have been described as adopting a *structuralist,* (or comprehensive) approach to data standards, meaning that they strive to develop a single, large data model to support most data exchange needs throughout the lifecycle of AEC/FM projects (Behrman, 2002). In some ways, the current IFC development methodology is modular, but only loosely so. It involves domain projects which draw from detailed industry data exchange scenarios, are developed by committees of domain experts, and lead to a specific set of data extension recommendations. However, these scenarios and domain projects are used to arrive at the extension recommendations only, they are not maintained as modules within the final IFCs. Furthermore, the final IFC classes are organized into numerous sub-models (schemas), but the underlying modeling methodology (adopted from the ISO 10303 standard STEP) combines all of these schema together into a single "long form" data model. For the most part, the resulting IFC model is tested by software implementers only after the standard has been released, and revisions are incorporated into future releases of the standards.

The large, comprehensive scope within the single IFC model provides the generality required to meet the IFC goals. Historically, however, few large-scale structuralist data standards have succeeded. Rather, successful data standards are typically smaller, more narrowly scoped, and more rapidly developed: a more ***minimalist*** approach. A minimalist approach to AEC/FM interoperability would abandon the IFCs and allow different groups to develop numerous, more-specific data exchange standards. We believe that this approach could not lead to the wide-spread interoperability envisioned for the IFCs. Yet the current IFC methodology suffers from many problems arising from its broad scope, large size, limited modularity, etc. It may be time to reengineer the IFC methodology to adopt some of the benefits of more minimalist approaches.

2.7 Exchange Mechanisms

IFC-based integration must move beyond file-based exchange to distributed system solutions.

Current implementations of IFC-based integration rely almost exclusively on the exchange of IFC files. This mode of transferring data is simple and effective. However, it is very limited in its ability to manage a large pool of shared project information that is accessed concurrently by many users, or to enable transactional forms of data exchange between project parties and applications. Rather, an approach that collects common data in a centralized database and offers a variety of data management services would be required for large scale, integrated, model-based systems. Various groups are currently in the process of developing IFC-based model server technologies for AEC/FM (Adachi, 2002). However, industry-wide integration cannot rely on a single product. Again, standards are needed at the layer of the exchange mechanism. These would take the form of some type of application programming interface for accessing model-based data and various data services. Various standards are available that could be suitable, such as the Standard Data Access Interface (SDAI), part of the ISO STEP technologies upon which the IFCs are based, or the CORBA common object request broker standard. Yet none of these standards has been developed and gained popularity within the AEC/FM industry. Recently emerging technologies in web services and XML may help lead the way to a more widely adopted interface standard.

We are currently carrying out research to further explore and develop the features required for system architectures for distributed integrated systems, and a modular approach to delivering these services based on standardized data access interfaces, which we are calling the Jigsaw system (Froese and Yu, 2000; Halfawy and Froese, 2002a). The basic architecture, illustrated in Fig. 4, is that data clients interact with data servers through a standard data access interface.

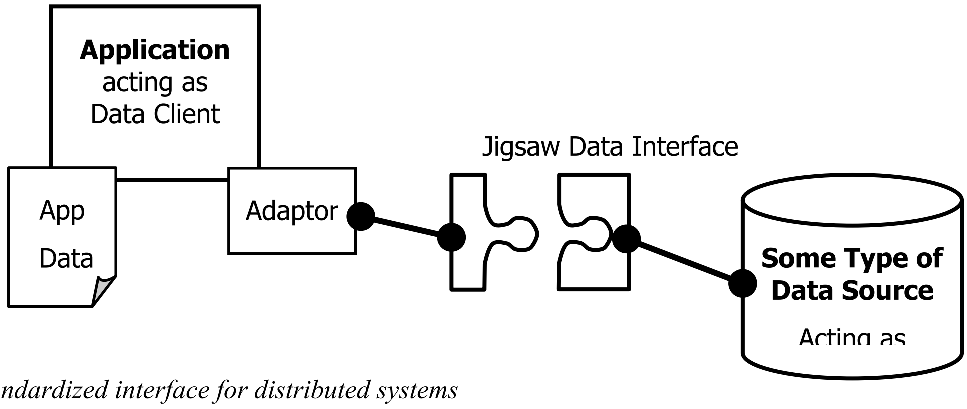


FIG. 4: Standardized interface for distributed systems

- Data Clients will normally be applications. These could be existing (legacy) applications, or they could be a new breed of application with new capabilities made possible only through access to the shared project model. They will typically have their own internal data model and work with their own data files. However, they will also be able to initiate data exchanges (read and/or write) with data sources through an adaptor to the Jigsaw data interface.
- Data servers are components that can respond to requests from data clients. There are many different types of components that could serve as data sources. Examples include simple components that store data as XML files, databases that maintain many project models within large relational or object-oriented databases, applications that are able to respond to data requests (thus creating a "peer-to-peer" approach for data exchange between applications), or components that access remote data sources across networks.
- The Jigsaw interface is intended to provide a flexible and stable standard data access interface. The interface implements several sets of data services, with information generally encoded as XML data. It is independent of any specific data content schema. There could be several different bindings of the Jigsaw interface to different implementation platforms (we are working with a Microsoft COM implementation and a Web Services implementation). The following are some of the data services that could be supported through the interface:
    - Handshaking: Once a data client has established a connection with a specific data server, they can carry out a handshaking procedure. Since the interface can support different data content schema, different data querying mechanisms, etc., the data client and data server can both query each other to ensure that they are compatible with each other for carrying out the data exchange.
    - Transactions: The interface supports the ability for the data client to pass both short and long transaction requests to the data server.
    - Data Sets: A data client must request the specific data set that it wishes to work with (e.g., the ID of a particular project). Data servers can also provide methods for managing data sets, such as creating new ones, copying data sets, etc.
    - Data Access: To read information from the server, a data client passes some type of data query to the server and receives a resulting data set. To write information to the server, the data client sends a data set to the server. The data set is encoded as XML, and the interface to the XML implements the XML DOM standard interface (W3C 2002). The Jigsaw interface does not assume any specific type of data query, it requires that data query information be passed on as an XML element. Data servers should generally implement at least a few basic query mechanisms, such as the ability to return a complete model, the ability to return a specific object instance only, and the ability to return all objects of a specified class. Some data servers may implement more advanced data queries like XSLT transformations filters.
    - In addition to these basic data services, the interface could also be extended to support services such as security and authorization, meta-data services, etc.
- Using the standard interface, several intermediate components could be "chained together" between the data client and the final data server. For example, Fig. 5 illustrates the following scenario:

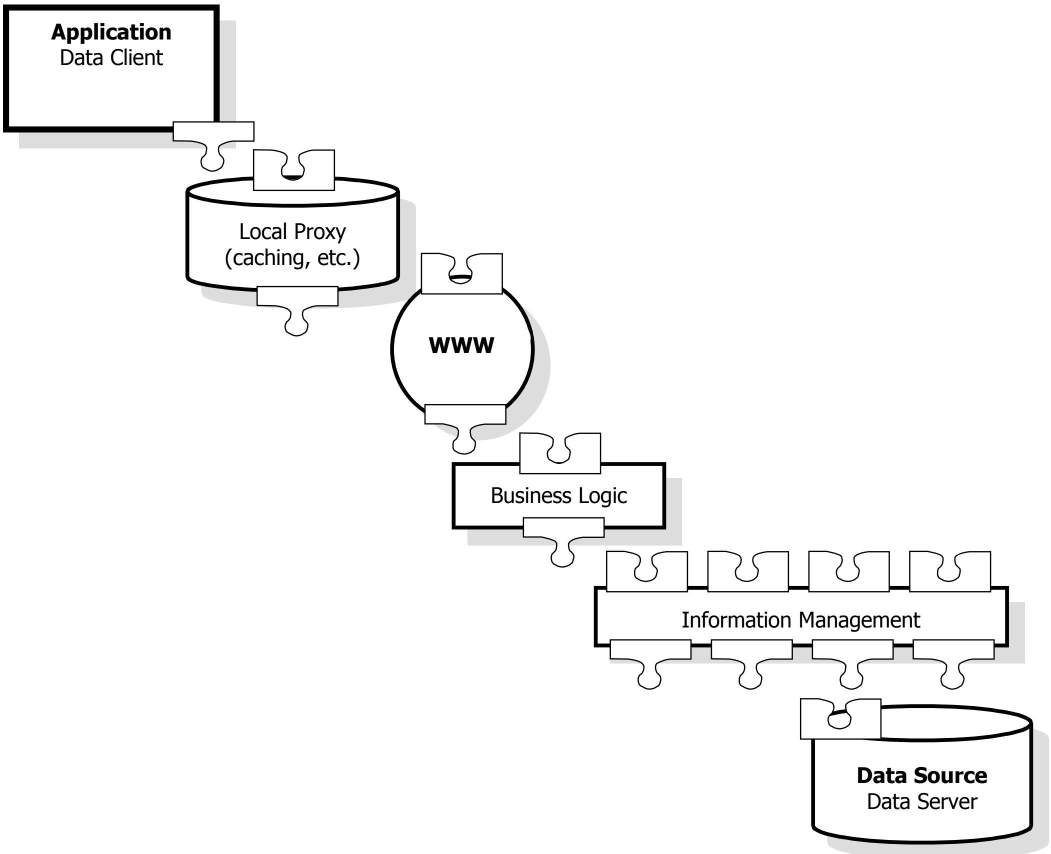


FIG. 5: A series of components linked together through standard interfaces

- An application initiates a data request to a local proxy component.
- The local proxy acts as a local copy of the shared central database. One of the important functions of the local proxy is to manage intelligent data caching. Complete project models are very large and access time over networks will always be an issue. However, most parts of a project model will not change very frequently, so there is an opportunity to cache objects on a local computer.
- Assuming that the local proxy does not have the requested information cached, it passes the data request on through a component that links to a centralized data source available over the Internet.
- The network server component receives the data request and passes it on to the server-based data source. However, it does not access the underlying database directly, but passes through middle layers of "business logic".
- A business logic component can implement object functionality that is better implemented on the server than on the local client. For example, this component may calculate derived property values from underlying stored property values of related objects, or it may perform validation of submitted data.
- The business logic component may access an information management component, which provides general, system-wide information management features. For example, this could provide object brokerage, so that clients can request specific objects without necessarily knowing where those objects are stored, and the object broker would connect to the appropriate data source.
- Finally, the data request reaches an underlying central database in which the object instances are stored.

- In order to provide the flexibility to support a variety of data models, the jigsaw interface is "late bound" and is independent of any specific schema. However, we have found that the data client is often easier to develop with a specific "early bound" object model interface. Therefore, we have introduced an "early-bound adaptor" component, which can access the jigsaw interface and expose the data to the client through an IFC2X-specific data model.

To develop and explore the Jigsaw approach, we have implemented several of the above components. We have built several new applications and adaptors to legacy applications, and several different types of data servers. This topic area is also the subject of recently-initiated work called SABLE (Simple Access to Building Lifecycle Exchange) by the BLIS project (BLIS, 2003).

2.8 Communication Layer

The family of standards related to the IFCs should extend beyond the "data dictionary" standards to include transaction standards.

Interoperability can be viewed as a series of data exchanges or "transactions" between computer applications or other software components. While the IFC model standardizes the content of an information exchange transaction, it offers no guidance to the context of these transactions; the IFCs standardize the vocabulary of a business conversation, but not the dialogue. It is still left up to the two parties exchanging information to come up with ad-hoc agreements about what data are being exchanged, for what business purpose, with what constraints and obligations on each participant, etc. Like the information content, these context agreements can be left ad hoc when the transaction is being carried out between two people. Automated transactions between computer systems, however, require that this information be formalized, and in the fragmented AEC/FM industry, these formalisms must be standardized throughout the industry to ensure consistency between all participants on all projects.

We are pursuing the formalization and possible standardization of data exchange protocols to support IFC-based transactions in distributed and heterogeneous environments (Halfawy, Pouria and Froese, 2002; Pouria, Halfawy and Froese, 2002). This would be a separate standard that could be used in conjunction with the IFCs, or with other data standards. The following are some examples of AEC/FM transactions:

- On-line purchasing of materials
- Notifications of design modifications.
- Requests for information on a job site
- Reporting inspection results
- Submittals
- Quantity takeoff/estimating
- The content of the transaction specifications define the workflow models of various AEC/FM project processes, and the purpose and scope of each transaction. Specifically, it will describe the following:
- Introduction and scope: the definition and purpose of the business transaction. The scope of the transaction expressed as Unified Modeling Language (UML) use cases or similar.
- Process model/data flows: A flow diagram that uses UML "swim lane" activity diagrams or similar to demonstrate the sequences of document flows between participants, conditional logic, start/end/failure states, etc.
- Data content of the transaction, presented as a data model that will likely be drawn from existing data models (such as the IFCs), but is expressly presented in language appropriate to the data exchange domain. This data content model can then be mapped to specific data model standards such as the IFCs in the "Bindings" sections below.
- Transaction controls and characteristics: the maximum time allowed to acknowledge receipt, the maximum time allowed acknowledging acceptance, maximum time allowed to perform a task, authorization requirements, etc.

- Bindings to standard data models: mapping of the data model to one or more standard data models such as the IFCs, aecXML models, etc.
- Bindings to implementation protocols: mapping of the business transaction to specific messaging implementation mechanism, such as web services, BizTalk messaging, etc.

2.9 Applications and Tools – Total Project Systems

All of the above topics relate to a layer of "technological infrastructure" that sits behind user's individual applications. However, the applications themselves are influenced by the availability of shared project models. The range of issues relating to applications is very broad, and we will address only a few of them here.

We describe our vision for integrated systems as *Total Project Systems (TOPS),* which are defined by three characteristics:

- Comprehensiveness: the suite of applications that make up Total Project Systems span a high percentage of all of the tasks that must be completed during the course of the project.
- Integration: All applications have the ability to contribute to and draw from the shared project model.
- Flexibility: Total project systems cannot be a single integrated system from a single vendor, but must allow different users to use different applications in different ways, while still achieving the required integration.

2.10 Level of Integration

IFC-based approaches should encompass both interoperability and integrated systems.

Interoperability can be defined as the ability for tools to exchange data, and it offers one solution to the problem of integration. The IFCs adopt a standard data model approach to providing interoperability. However, a different solution to providing integration is that of integrated systems, which can be defined as multipurpose tools that combine many different views (both data and functionality). Both of these approaches lead to problems. With interoperability, it is difficult to manage the collective body of information since each tool works with only a limited view. Furthermore, the "business logic" associated with each view resides in different tools and can't interact. For example, data exchange through interoperability alone does not support the ability for one application to give immediate cost implication feedback to a user exploring design changes in another application. This would be possible within integrated systems, yet no one integrated system can support all of the views and users required for an AEC/FM project. It would be difficult to impose one specific system on all projects users, or to enter all the necessary information within one tool.

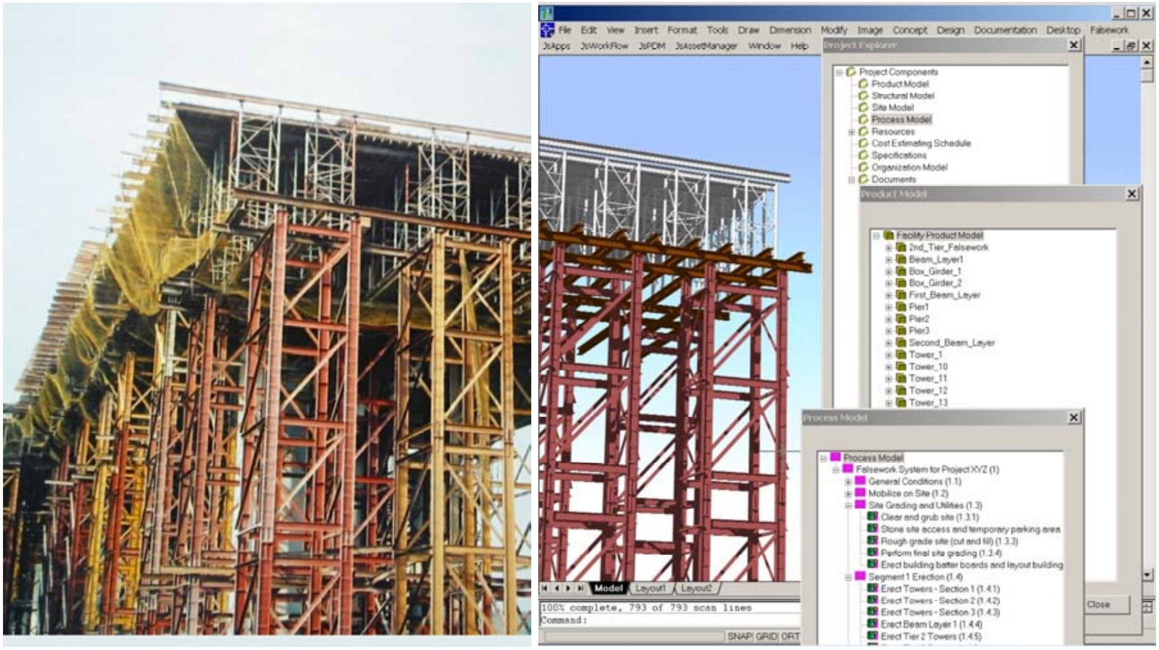
A general solution, then, is to combine both data interoperability and integrated systems: existing applications are able to exchange information through the data exchange standards, while new integrated tools are used for overall information management and functional integration.

Some of the general characteristics of integrated systems are that they cluster many functional views around an overall task of building, maintaining, and interacting with an integrated project database. They adopt a model-based approach: all information is structured around an object-oriented data model of the project. Generally, the product model (the physical components of the built facility) plays a central role, but this is interlinked with many other types of project information. Finally, most of the information contained within the system can be exchanged with other systems (i.e. the integrated system supports data interoperability).

At the core of these model-based integrated systems are the parametric AEC/FM objects (or "smart objects") (Halfawy and Froese, 2002b) that model project information throughout the project lifecycle. Like data exchange models, these objects support the representation of project information. Unlike data exchange models, however, they also implement the objects' behavioral characteristics or business logic. For example, an object's methods may allow it to automatically perform a quantity takeoff from its geometric properties.

We are developing prototype systems that use IFC-based interoperability to interact with a wide variety of project information, and then combine this information within an integrated system based on smart objects. We are currently focusing on two interrelated prototype systems, illustrated in Fig. 6 (Halfawy and Froese, 2002c):

- A falsework design system, which helps in the layout, design, and planning of heavy falsework systems for elevated highways. This system is based on a CAD platform and implements “smart falsework objects”. While the objects that represent the falsework sections themselves are not within the scope of the IFCs, many of the other related project information is based on IFC classes.
- A general-purpose project browser, which can import a variety of project information—product information, costs, schedules, documents, resource information, etc.—and interlinks them all within a CAD-based explorer environment. Because this system can link to any type of product, it can be used in conjunction with the falsework design system.



```
100% complete, 793 of 793 scan lines
Command:
```

FIG. 6: An elevated highway falsework system (left), is modeled in an integrated system (right) that models both the physical project and several non-physical aspects such as construction schedules, cost estimates, etc. The integrated system supports interoperability: it can exchange project information with other applications.

2.11 Integration of Model-Based Approaches with other Information Technologies

Model-based approaches should be integrated with other information technologies, particularly those relating to web-based collaboration, such as document management, workflow management, knowledge management, e-commerce, etc.

The practice of engineering is drastically different from what it was a generation ago. More than any other factor, this change is due to the development of computer-base tools for engineering analysis. However, the process of communicating the work between participants is not substantially different from traditional approaches. If the first generation of IT revolution was the use of computers to perform engineering analysis, then the second generation revolution will be the use of computers to provide a shared media for the work: the equivalent to “multi-dimensional paper” that allows participants to share a common, computer-interpretable view of the project.

Model-based integrated systems are central to the next generation of IT support for the AEC/FM industry. Yet they are not the only important IT trend. As mentioned previously, web-based collaboration is another major IT trend. It is important that these vital technologies not be developed in isolation of each other. Model-based systems and interoperability should be integrated with important web-based technologies such as document management, workflow management, knowledge management, and e-commerce applications. One approach to the integration of model-based systems and document management was presented earlier in this paper. We are embarking on several projects to investigate this and other forms of integration between model-based and web-based systems.

2.12 Work Processes – Unified Project Management

In addition to trying to fit interoperability solutions to current industry practice, new ways of organizing and managing projects should be developed to better leverage the IT and integration capabilities.

Although research and development work in the area of interoperability attempts to produce solutions that meet the needs of current industry practice, there are many ways in which the available technologies are not an ideal fit with practice. In part, this may be because of the relatively minor role that shared information resources and integrated processes play in current practice. In addition to trying to fit the IT solutions to current practice, it may be useful to also consider how current practice could be changed so that it will be able to take advantage of the new integrated solutions that are available. We are developing the concept of Unified Project Management (UPM) as a project organization and management technique that is better able to accommodate integrated processes and information technologies. Unified Project Management draws from several key reference areas:

- Trends in integration and model-based information technologies.
- Integration techniques in software engineering, in particular, the Unified Modeling Language (UML) (Object Management Group 2002), and UML-based software development methodologies (Unified Process, Kendall 2002)
- Concepts of value chains, lean construction (Lean Construction Institute 2002), etc.

In current project management practice, the overall design and construction project is broken into individual work packages that are assigned to different groups within different companies. The predominate focus for these groups is then on carrying out their work packages, with relatively little focus or structure for how the results of the work integrate with other work packages into the complete project.

In contrast, Unified Project Management introduces a framework that emphasizes the way that each work package fits within the overall project lifecycle and value chains, that makes explicit the relationship between work packages and the collective body of project information, and that emphasizes the cyclical, repetitive nature of work tasks rather than their "one-off" nature.

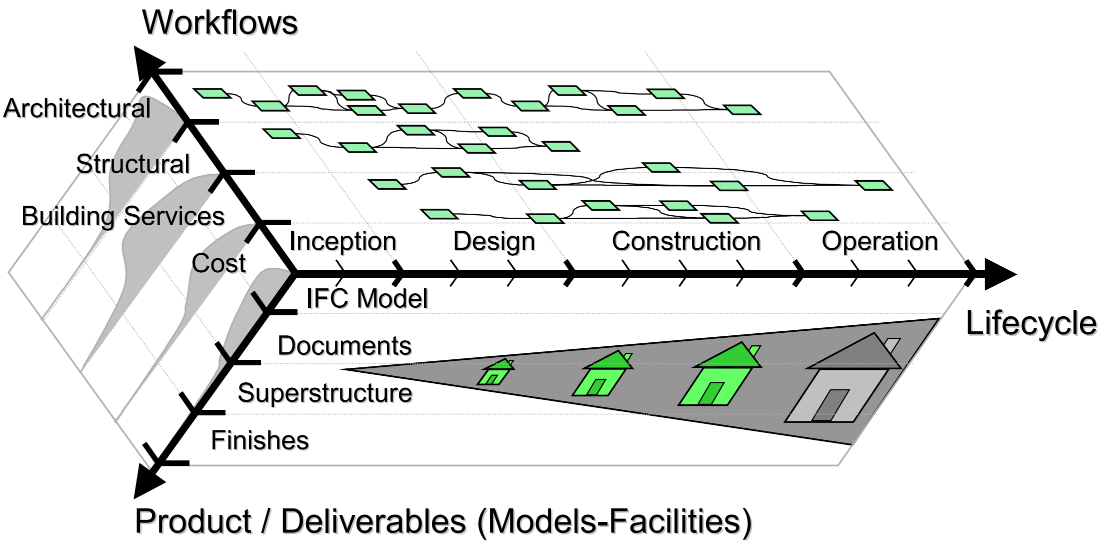


FIG. 7: Schematic of dimensions in Unified Project Management

The Unified Project Management approach defines three dimensions for organizing project work (illustrated in Fig. 7):

- The Project Lifecycle dimension organizes the project into well-defined project phases, which are further refined into iterations. Although projects are currently thought of as passing through typical phases, this is an informal notion that is not explicitly used in organizing the work.

- The Workflow dimension organizes the work into the various work disciplines required to complete the project. This is somewhat like the normal division of work into work packages, but rather than describing the tasks as discrete work packages, the work is organized as ongoing workflows, which can be further broken down into sequences or networks of sub tasks.
- The product/deliverable dimension organizes the results of work. These results are described as parts of the product being constructed: initially, the results are information about the physical facility, which make up the virtual product, while during the construction phase, the results are the construction of the physical components themselves. These deliverables, flowing from the virtual facility to the physical one, form a continuum that makes up the product of the project.

Given these three dimensions, the work can be further organized along the intersection of each pair of dimensions:

- Workflows along the project lifecycle: most workflows span several project lifecycle iterations/phases, but different amounts of work are required at different times. By considering the workflows vs. the project lifecycle, the amount of each workflow that should be carried out in each iteration can be planned.
- Product/Deliverables along the project lifecycle: similarly, different amounts of the various deliverables are developed during different project iterations/phases.
- Product/Deliverables vs. Workflows: The organization can also define which workflows should collaborate on each of the project deliverables.

With the Unified Project Management framework dimensions are planned explicitly and used as a primary organizational vehicle during the project, participants still carry out their individual work tasks, but their inter-dependencies with other tasks, other project phases, and a combined product model are much more evident and easier to attain.

Although the three dimensions seem appropriate for the overall organization of all project participants, the management specialists responsible specifically for the overall management of the project could consider additional simultaneous dimensions, such as costs, risks, resources, etc.

We are continuing to develop the concept of Unified Project Management, based on the underlying premise that new project organization and management frameworks may help work practices better fit the emerging technological tools.

3. CONCLUSIONS

AEC/FM projects have been described in terms of collections of information-intensive tasks connected through information transactions. IT influences the effectiveness with which these tasks and transactions are carried out. Two themes in emerging IT are internet-based collaboration and model-based approaches. While internet-based collaboration is ready for full-scale use,

Model-based systems and interoperability represent a major trend in information technologies for the AEC/FM industries. By providing a standard language for exchanging project data, the Industry Foundation Classes form an important cornerstone of these emerging technologies. Model-based approach are only beginning to reach practical viability, and the many directions in which further development is required have been outlined in this paper. They are ready, however, for AEC/FM companies to begin familiarizing themselves with these technologies and begin pilot projects. In this way, organizations will be well prepared to take up and exploit these emerging tools to improve the efficiency of their AEC/FM projects.

4. ACKNOWLEDGEMENTS

This paper presents a high-level summary of several different research projects, each of which involves several researchers—their significant contribution to this paper is gratefully acknowledged. Much of the work is supported by the Natural Sciences and Engineering Research Council of Canada, Collaborative Research Opportunities Program.

5. REFERENCES

Adachi, Y., 2002. “IFC Model Server Development Project”, web page at http://cic.vtt.fi/projects/ifcsvr/ [accessed Aug. 30, 2002].

Behrman, W., 2002. “Best Practices for the Development and Use of XML Data Interchange Standards,” Technical Report #131, Center for Integrated Facilities Engineering, Stanford University, USA, July, 2002

Caldas, C. and Soibelman, L., 2003. “Integration of Construction Documents in IFC Project Models” Winds of Change: Integration and Innovation in Construction, Proceedings of the Construction Research Congress, American Society of Civil Engineers, Honolulu, HI, March 2003.

BLIS, 2002. “BLIS home page”, web page at http://www.blis-project.org/ [accessed Aug. 30, 2002].

BLIS, 2003. “The Sable Project”, web page at: http://www.blis-project.org/~sable/ [accessed April 1, 2003].

Dublin Core Metadata Initiative, 2002. “Dublin Core Metadata Initiative (Home Page)” web page at http://dublincore.org/ [accessed December 1, 2002].

Froese, Fischer, Grobler, Ritzenthaler, Yu, Sutherland, Staub, Akinci, Akbas, Koo, Barron, and Kunz, 1999. "Industry Foundation Classes For Project Management—A Trial Implementation", Electronic Journal of Information Technology in Construction, Vol.4, 1999, pp.17-36. Available online at http://itcon.org/1999/2/

Froese, T. and Halfawy, M., 2002. “Information Integration Technologies for the Management of Bridge Projects,” Taiwan-Canada Workshop on Bridges, Taipei, Taiwan, April 8-9, 2002. pp. 169-176.

Froese, T., and Yu, K., 1999. “Industry Foundation Class Modelling For Estimating And Scheduling”, Durability Of Building Materials And Components 8. Vancouver, May 1999. Vol. 4, pp. 2825-2835.

Froese, T. and Yu, K., 2000, “Architecture Issues For Distributed AEC/FM Systems,” Computing in Civil & Building Engineering: Proc. of the 8th Int. Conf. (ICCCBE-VIII), ASCE, Stanford University, USA, August 14 – 17, 2000. Vol. 2, pp. 1285-1292.

Halfawy, M. and Froese, T., 2002a. “A Framework For AEC/FM Total Project Systems,” CE World, ASCE's First Virtual World Congress for Civil Engineering, available at http://www.ceworld.org/ [accessed Aug. 30, 2002]

Halfawy, M. and Froese, T., 2002b. “Modeling and Implementation of Smart AEC Objects: An IFC Perspective,” CIB W78 2002 Conference, Aarhus, Denmark, June 12-14, 2002

Halfawy, M. and Froese, T., 2002c. “A Model-Based Approach for Implementing Integrated Project Systems,” 9th International Conference on Computing in Civil and Building Engineering, Taipei, Taiwan, April 3-5, 2002. Vol. 2. pp. 1003-1008.

Halfawy, M., Hadipriono, F., Duane, J., Larew, R. and Froese, T., 2002. “Developing A Step-Based Core Information Model For Bridges” 6th International Conference on Short & Medium Span Bridges, Vancouver, BC, Jul. 31 - Aug 2, 2002.

Halfawy, M., Pouria, A. and Froese, T., 2002. “Developing Message-Based Interoperability Protocols for Distributed AEC/FM Systems,” CIB W78 2002 Conference, Aarhus, Denmark, June 12-14, 2002

Hassanain, M. 2002. Integrated Systems for Maintenance Management, Ph.D. Thesis, Dept. of Civil Eng., University of British Columbia, Feb. 2002.

Hassanain, M., Froese, T. and Vanier, D., 2001. “Development of a maintenance management model based on IAI standards,” Artificial Intelligence in Engineering. Special Issue on Conceptual Modeling in Design. Vol. 15, No. 3., pp. 177-194. 2001.

International Alliance for Interoperability, 2002. “IAI Web Site”, web page at http://www.iai-international.org [accessed Aug. 30, 2002].

ISO 2000. “ISO/TR 19033:2000, Technical product documentation -- Metadata for construction documentation” International Organization for Standardization, Geneva, Switzerland, 2000.

Kendall, S. 2002. The Unified Process Explained, Addison Wesley.

Kosovac, B., Froese, T. and Vanier, D., 2000. “Integrating Heterogeneous Data Representations In Model-Based AEC/FM Systems”, Construction Information Technology 2000, Proceedings of CIT 2000 – The CIB-W78, IABSE, EG-SEA-AI International Conference on Construction Information Technology, Reykjavik, Iceland, 28-30 June, 2000, Vol. 2, pp 556-567.

Lean Construction Institute 2002 “Lean Construction Institute (Home Page)” web page at: http://www.leanconstruction.org/ [accessed December 3, 2002].

Pouria, A., Halfawy, M. and Froese, T., 2002. “Developing AEC/FM Transaction Standards,” 3rd International Conference on Concurrent Engineering in Construction - CEC02, University of California at Berkeley, California, USA, July 1-3, 2002.

OCCS Development Committee, 2002. “OCCS Net, The Omniclass Construction Classification System”, web page at: http://www.occsnet.org/ [accessed December 1, 2002].

Object Management Group, 2002. “UMLResource Page”, web page at http://www.omg.org/uml/ [accessed December 3, 2002].

Timberline, 2002. “CAD Integrator”, web page at http://www.timberline.com/products/estimating/cad_integrator.htm [accessed Aug. 30, 2002].

Underwood, J., and Watson, A., 2003. “An XML Metadata Approach to Seamless Document Exchange,” on-line document available at: http://www.doclink.info/XML Metadata Document Exchange.pdf [accessed April 1, 2003].

W3C 2002. “Document Object Model (DOM)” web page at: http://www.w3.org/DOM/ [accessed December 3, 2002].

Yu, K., Froese, T. and Grobler, F., 2000. “A Development Framework for Data Models for Computer-Integrated Facilities Management” Automation in Construction, Vol 9, No. 2, 2000, pp.145-167.