Next Steps for R&D in IFC-based Integration

Integration of Project and Facilities Management Information in Distributed Environments

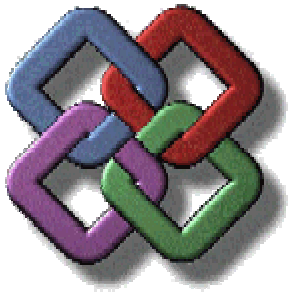
Dr. Thomas Froese
Department of Civil Engineering
University of British Columbia

IAI Seminar
VTT, Finland
April 24, 2002

Introduction

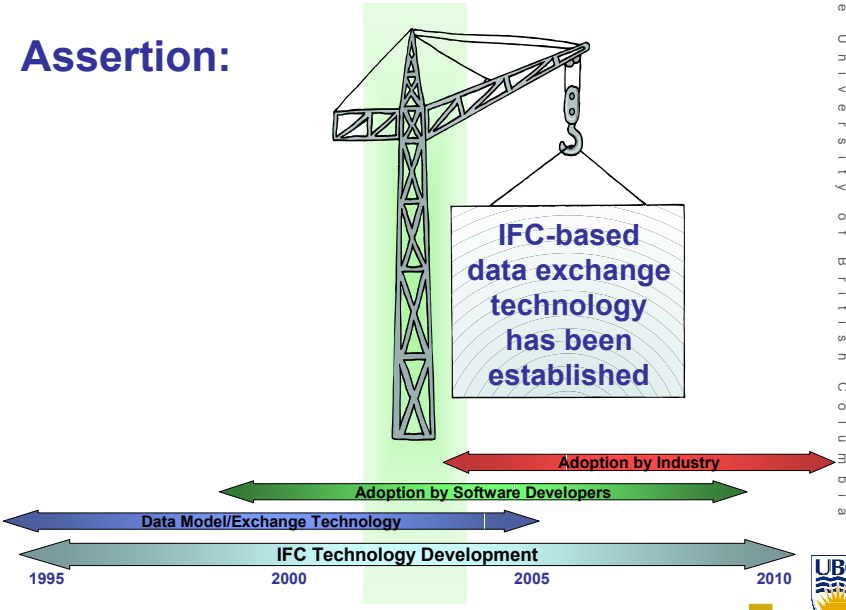
Industry Foundation Classes

- The IFCs provide an international standard for exchanging digital information about architecture, engineering, construction, and facilities management (AEC/FM) projects.



Introduction

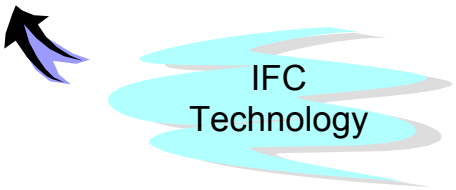
- **Assertion:**



Next Steps

- **Moving Beyond Basic IFC Data Exchange**

Application Area:
Beyond Product Data



Application Area: *Beyond the Product Model*

- The scope of the IFC's includes *product information*
- And also non-product information, such as:
  - Costs, Schedule, Organizations, Resources, Documents, etc.
- Systems have been developed that work with non-product information.
  - But these mostly use *product information* as an input to non-product applications.
  - Few systems have used the IFC's to exchange non-product data.

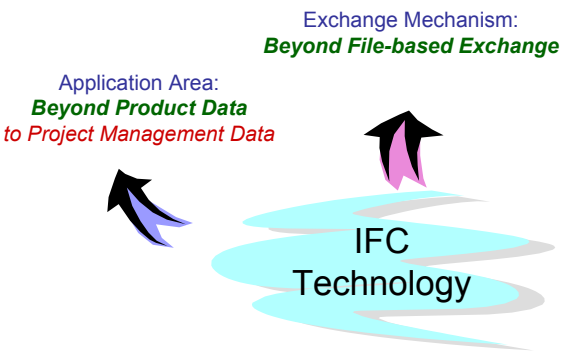
Next Steps: Application Area

Application Area: *Beyond the Product Model*

- Our primary interest is in the use of IFC's to support project management information exchange.
- We are experimenting with IFC-based exchange of:
  - Cost Data
  - Schedule Data
  - Facilities Management Data
  - Project Specifications
  - Materials Management Data
  - Document Reference Data

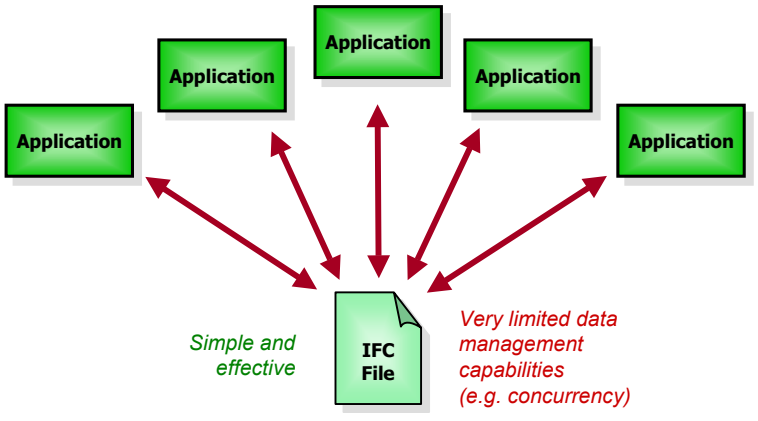
The University of British Columbia

Moving Beyond Basic IFC Data Exchange



Exchange Mechanism: Beyond File-based Exchange

- File-based data exchange



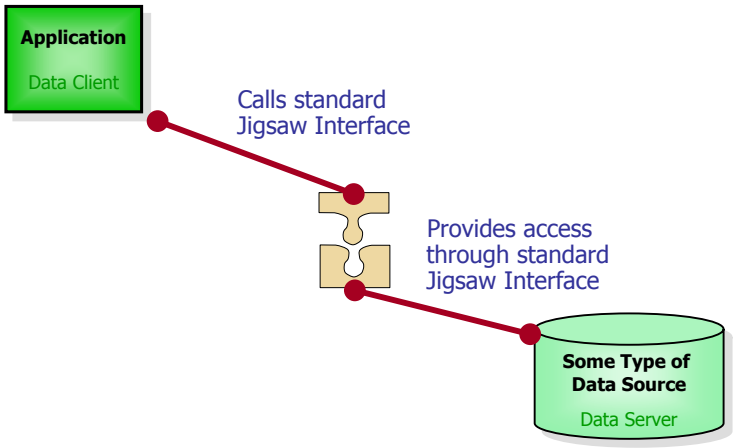
Next Steps: Exchange Mechanism

Jigsaw Distributed System Data Exchange

- Various data exchange mechanisms are possible, but none can be imposed as a requirement throughout the industry.
- The *UBC Jigsaw distributed system* provides a generic interface that can support a wide range of data exchange
  - From simple files to sophisticated on-line data access.

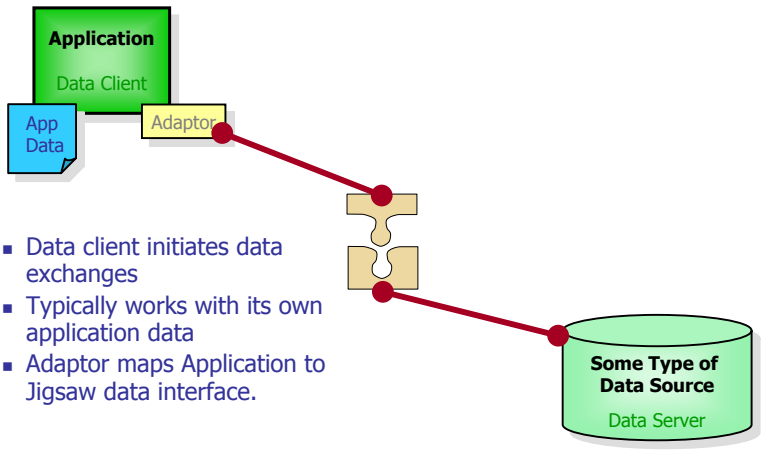
Next Steps: Exchange Mechanism

Basic Data Exchange Mechanism



Next Steps: Exchange Mechanism

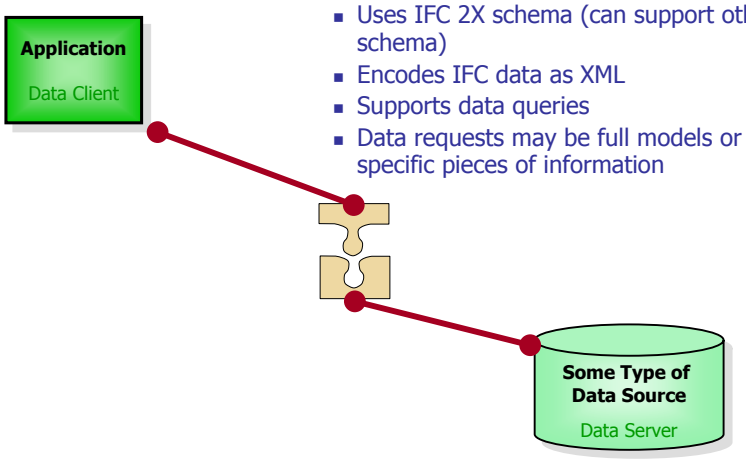
Data Clients



- Data client initiates data exchanges
- Typically works with its own application data
- Adaptor maps Application to Jigsaw data interface.

Next Steps: Exchange Mechanism

Jigsaw Interface

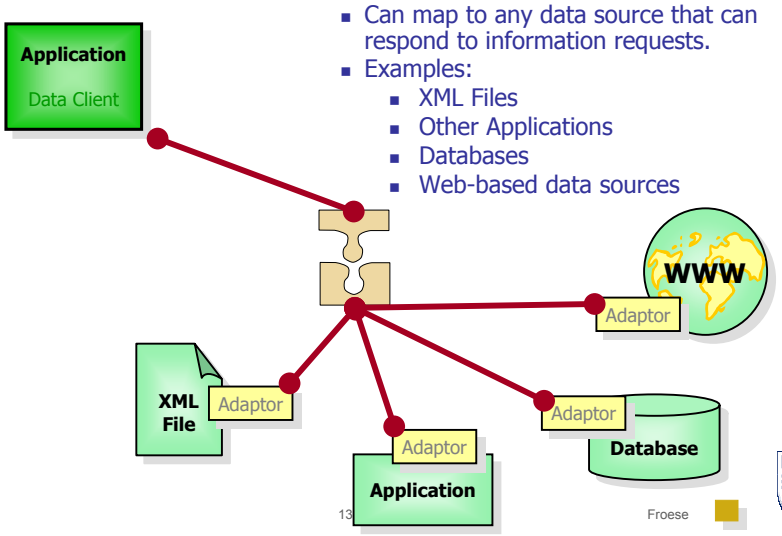


- Uses IFC 2X schema (can support other schema)
- Encodes IFC data as XML
- Supports data queries
- Data requests may be full models or specific pieces of information

Next Steps: Exchange Mechanism

Data Servers

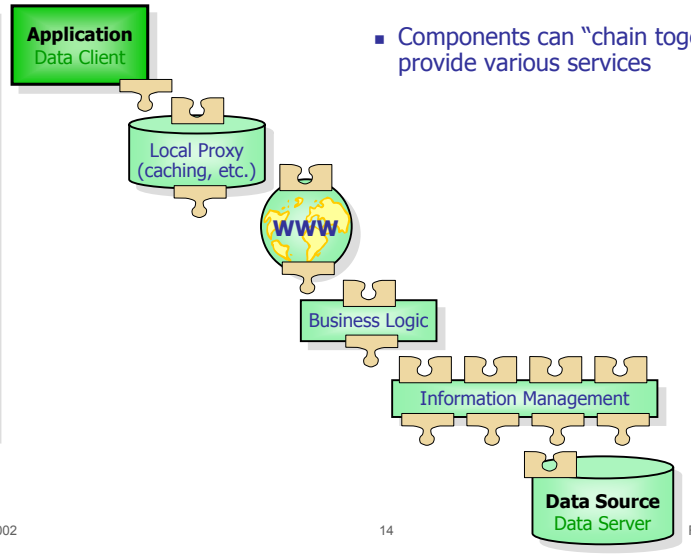
- Can map to any data source that can respond to information requests.
- Examples:
  - XML Files
  - Other Applications
  - Databases
  - Web-based data sources



Next Steps: Exchange Mechanism

Interconnecting Components

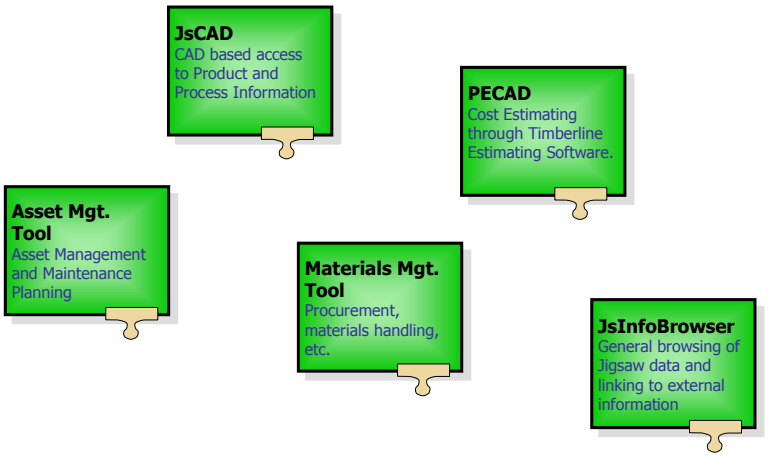
- Components can "chain together" to provide various services



The University of British Columbia

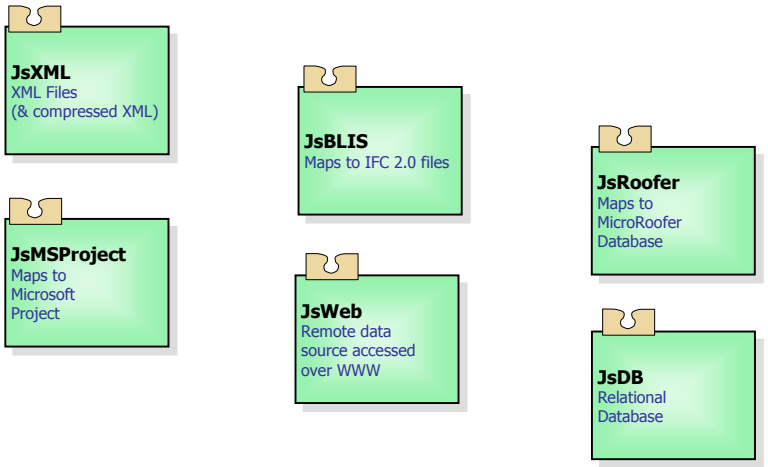
Next Steps: Exchange Mechanism

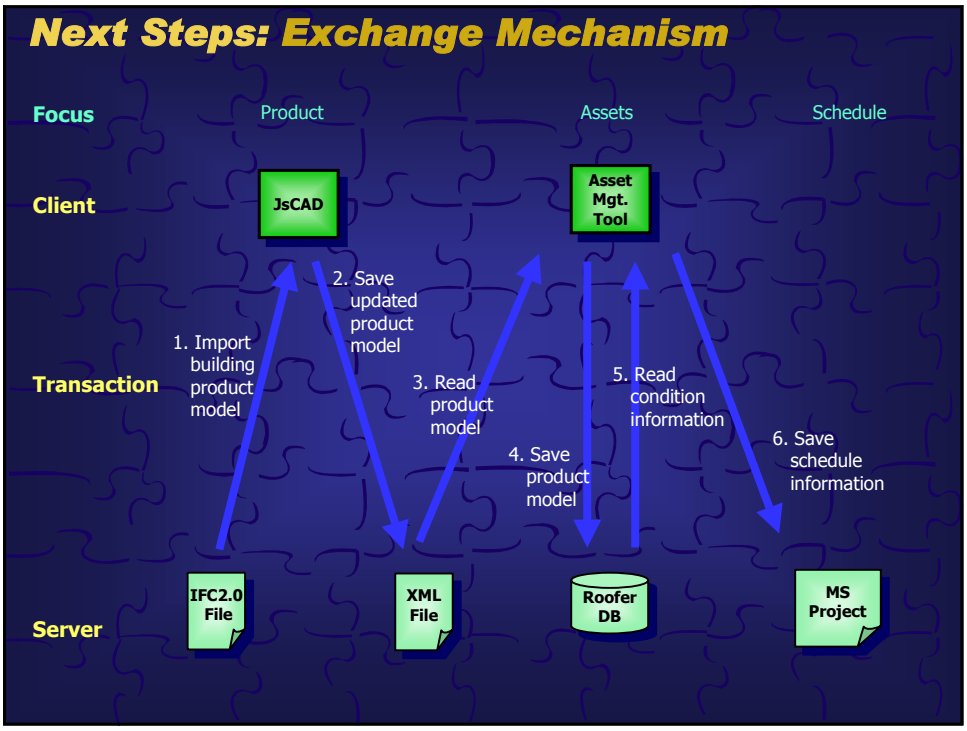
Jigsaw Version 0.6 – Data Clients

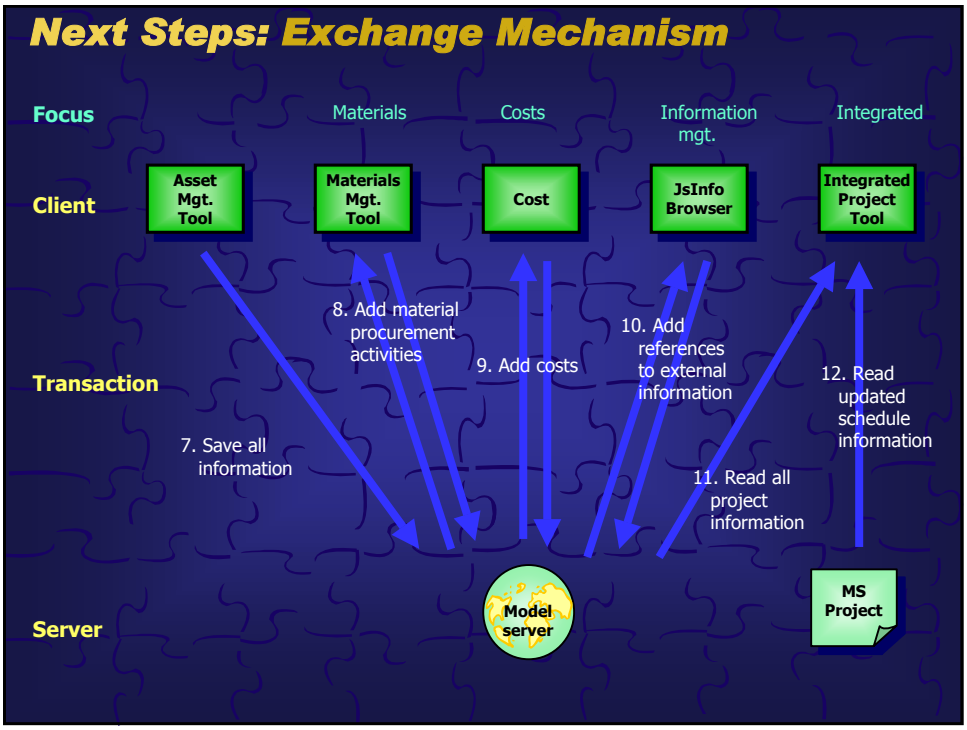


Next Steps: Exchange Mechanism

Jigsaw Version 0.6 – Data Servers



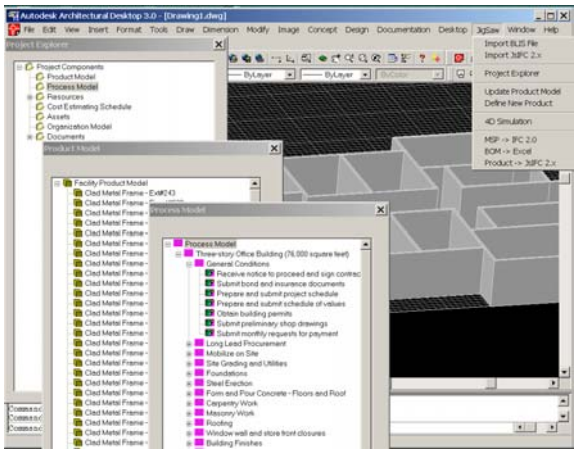

Next Steps: Exchange Mechanism
Focus
Product
Assets
Schedule
Client
JsCAD
Asset Mgt. Tool
Transaction
1. Import building product model
2. Save updated product model
3. Read product model
4. Save product model
5. Read condition information
6. Save schedule information
Server
IFC2.0 File
XML File
Roofer DB
MS Project


Next Steps: Exchange Mechanism
Focus
Materials
Costs
Information mgt.
Integrated
Client
Asset Mgt. Tool
Materials Mgt. Tool
Cost
JsInfo Browser
Integrated Project Tool
Transaction
7. Save all information
8. Add material procurement activities
9. Add costs
10. Add references to external information
11. Read all project information
12. Read updated schedule information
Server
Model server
MS Project

Next Steps: Exchange Mechanism

JsCAD

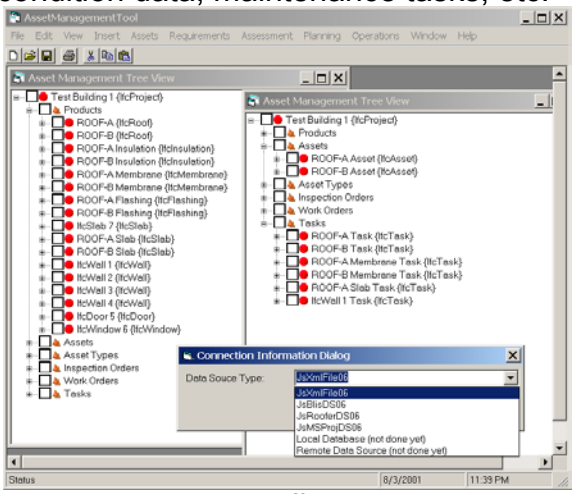
- Import and/or Develop Product Model
  - Can read/write IFC 2.0 files and other Jigsaw Data Sources
- Works with process information in a CAD-based platform
  - Schedule, resources, costs, documents, etc.



Next Steps: Exchange Mechanism

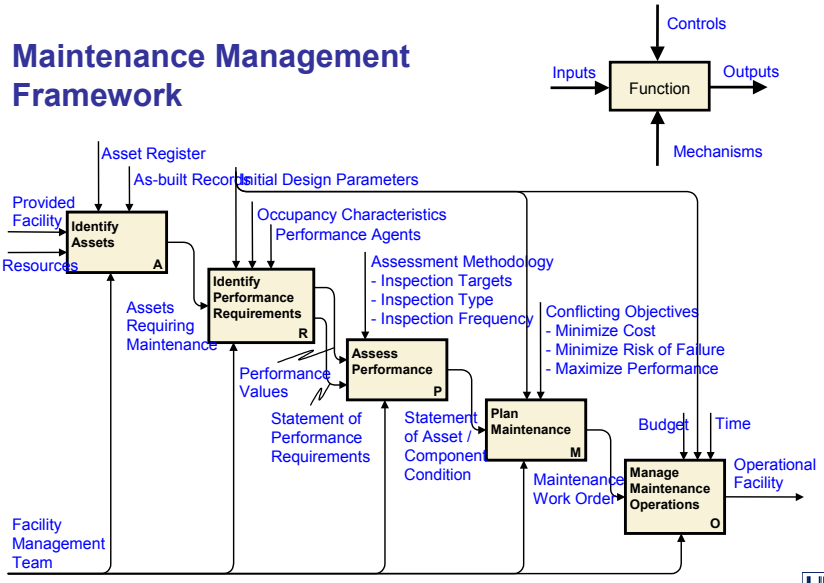
Asset Management Tool

- Associates product model with assets, asset condition data, maintenance tasks, etc.



Next Steps: Exchange Mechanism

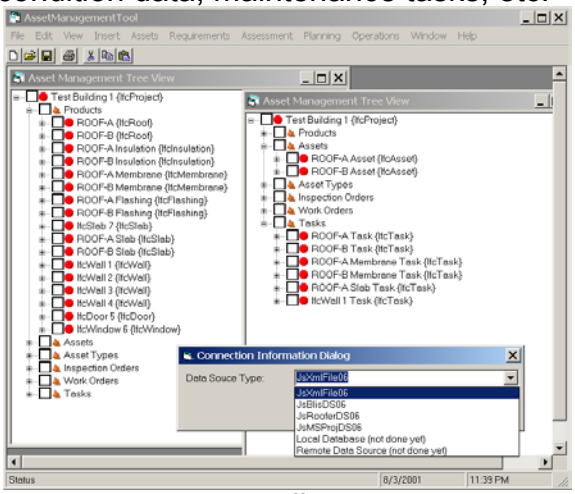
Maintenance Management Framework



Next Steps: Exchange Mechanism

Asset Management Tool

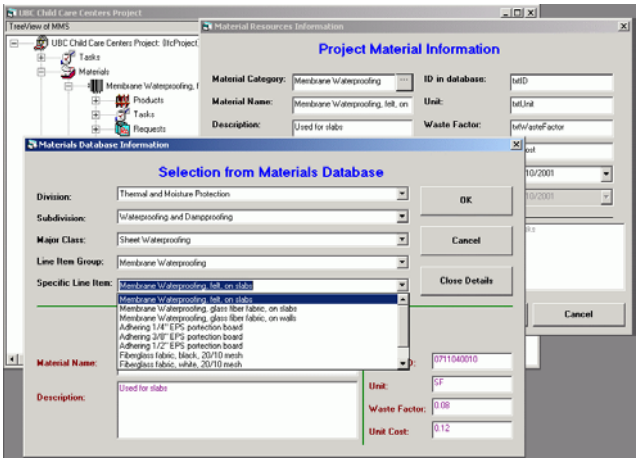
- Associates product model with assets, asset condition data, maintenance tasks, etc.



Next Steps: Exchange Mechanism

Materials Management Tool

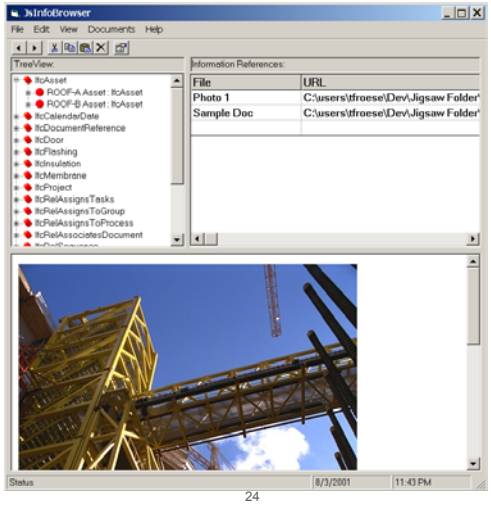
- Maintains database of materials
- Links products, materials and material properties, material resources, tasks, costs, etc.



Next Steps: Exchange Mechanism

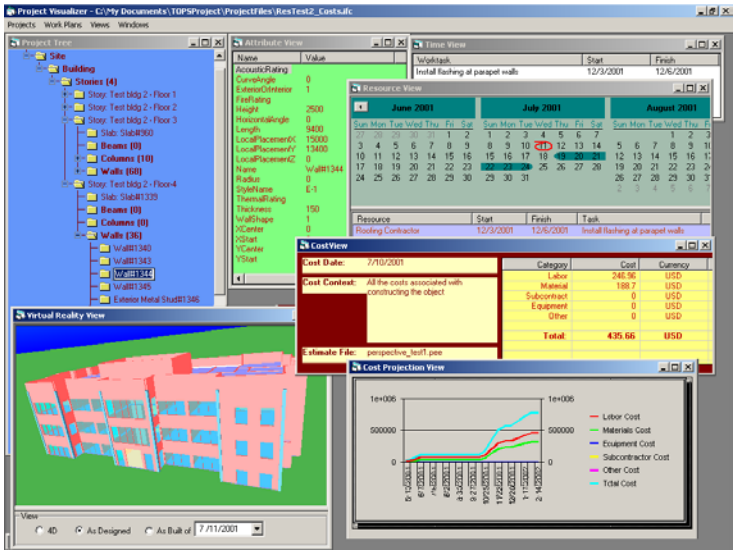
Information Browser

- Associates unstructured documents with any structured objects.



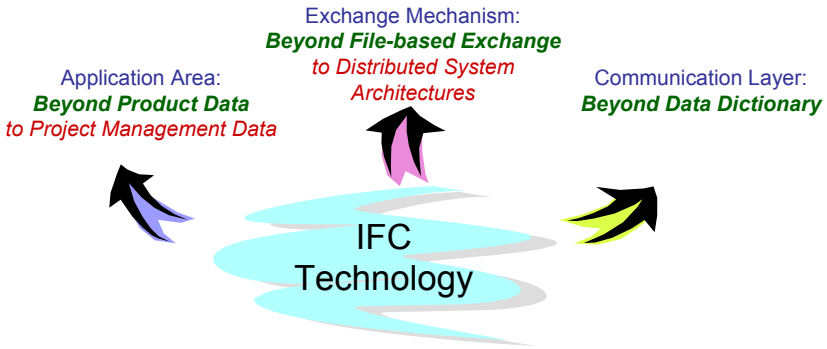
Next Steps: Exchange Mechanism

Integrated Project Browser Tool



Next Steps

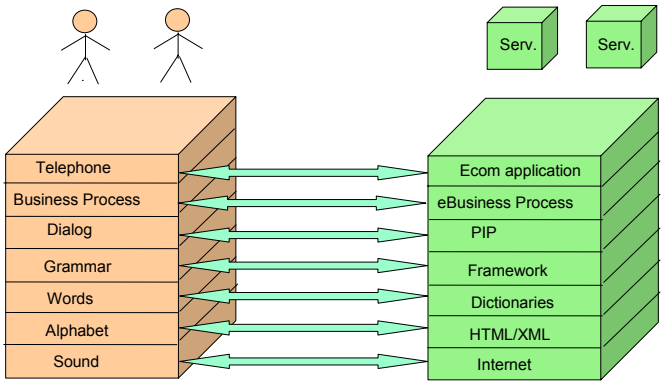
Moving Beyond Basic IFC Data Exchange



Next Steps: Communication Layer

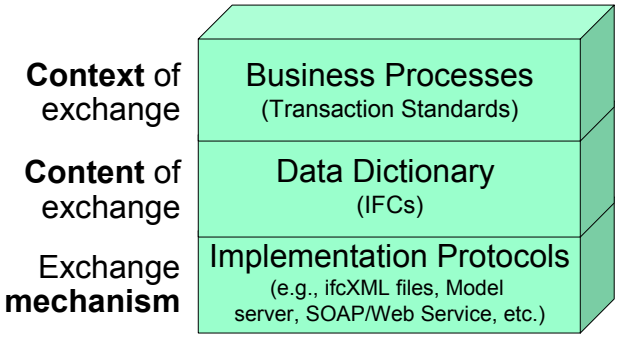
Analogy *from RosettaNet*

- RosettaNet is an initiative for electronic industry which aims at defining electronic business interfaces. The RosettaNet scope is shown below.



Next Steps: Communication Layer

Main Communication Layers of Interest



Next Steps: Communication Layer

Limitations of "Data Dictionary" Layer

- With IFCs, the exchange *content* is structured and standardized, but the exchange *context* is still ad-hoc.
  - Content needs to be structured and formalized because we have moved from human-interpreted communication to computer-interpreted communication.
  - Similarly, as we move towards communications between computer applications, the context of an information exchange, or transaction, needs to become more formalized and standardized.
- Most existing IFC-based data exchange scenarios are based on a general approach of:
  - "Here is the complete product model"
  - In reality, data exchanges typically involve very specific pieces of information (partial model exchange).

Next Steps: Communication Layer

Transactions, Examples

- Interoperability
  - Can be viewed as a series of data exchanges between computer applications or other software components.
- → "Transactions"
  - Examples might include:
    - On-line purchasing of materials
    - Notifications of design modifications.
    - Requests for information on a job site
    - Reporting inspection results
    - Submittals
    - Quantity takeoff/estimating

Next Steps: Communication Layer

Content of Transactions

- To exchange information electronically (conduct a transaction), users must first agree upon many different aspects of the communication:
  1. *Business Processes*: the purpose and scope of the transaction, the roles and responsibilities of the transaction participants, etc.
  2. *Data Content*: the data content to be exchanged and the schema for structuring the data.
  3. *Implementation Procedures and Protocols*: the protocols for encoding the data, transporting between systems, initiating and responding to data exchanges sessions, system security issues, etc..

Next Steps: Communication Layer

Characteristics of Transaction Standards

- Some of their characteristics would be as follows:
  - Would formalize the context and requirements for transactions.
  - The primary audience would be solution providers, but the process of formalization would have implications throughout the project.
  - Mainly human-readable, include computer-readable elements.
  - They would apply to data standards such as IFC's, aecXML,etc.
    - E.g., they may specify the data set exchange requirements in terms of IFC objects.
  - Would provide libraries of transactions standards.
    - Standards available for use as appropriate, they are NOT intended to constrain a company's business processes.
    - The transactions standards in the library may overlap, may be replaced as a data exchange technologies and business practices involved, etc.

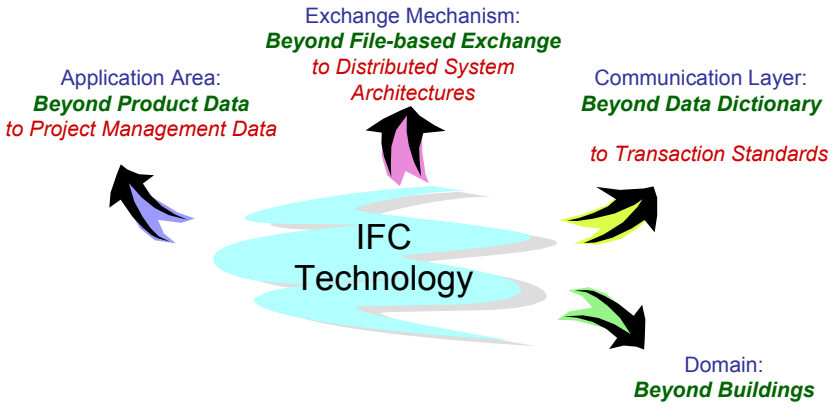
Next Steps: Communication Layer

Makeup of a Transaction Standard

- 1. Business Process
  - Introduction: definition, purpose
  - Scope of the transaction expressed as UML Use Cases
  - Process Model / Data Flows:
  - Data Content / Data Dictionary
  - Transaction Controls and Characteristics
  - Examples
- 2. Bindings to standard data models:
  - Maps the data model to one or more standard data models such as the IFC's, aecXML models, etc.
- 3. Bindings to implementation protocols:
  - Maps the business transaction to specific messaging implementation mechanism, such as Biztalk messaging.

Next Steps

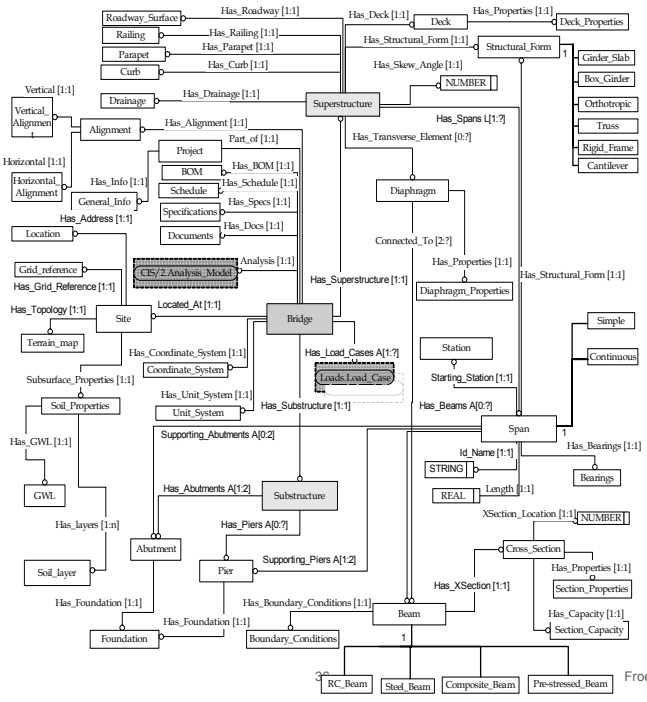
Moving Beyond Basic IFC Data Exchange



Next Steps: Domain

Data Exchange Models for Civil Works

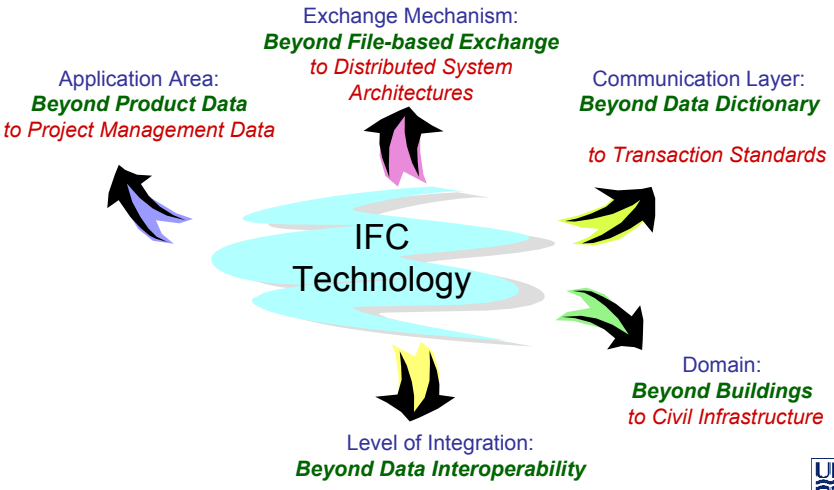
- Data Model for bridges
- Data Exchange model for Highway Design and Maintenance.



The University of British Columbia

Next Steps

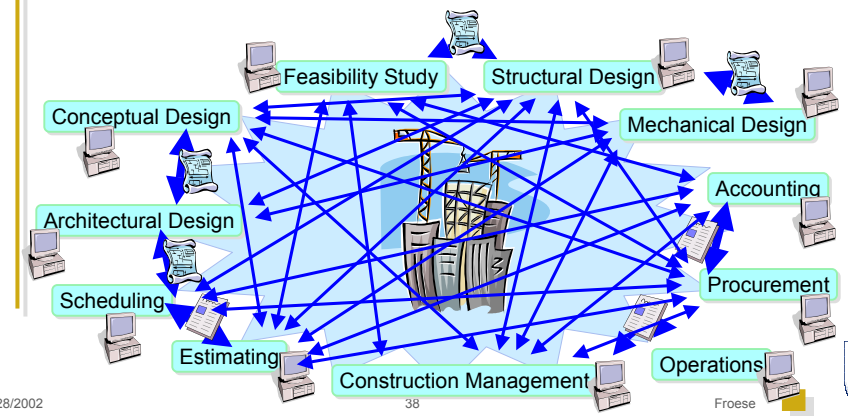
Moving Beyond Basic IFC Data Exchange



Next Steps: Level of Integration

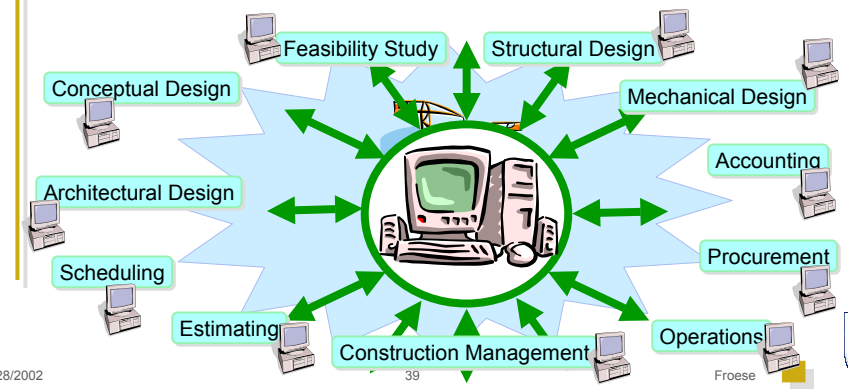
The Integration Problem

- We want to go from ad-hoc, many-to-many, human-reinterpreted information flows…



The Integration Problem

- We want to go from ad-hoc, many-to-many, human-reinterpreted information flows…
- …to shared, standardized data.



Solutions to Integration

Interoperability

- Definition: an ability for tools to exchange data
- Techniques: Data Exchange Standards – *Industry Foundation Classes*

- Definition: Tools that combine many different views (data & functionality)

Integrated Systems

Next Steps: Level of Integration

Solutions to Integration

Interoperability

- Problems:
- Difficult to manage collective body of information if each tool works with only a limited view.
- The "business logic" associated with each view resides in the different tools and can't interact.
  - *e.g., exploring the cost impacts of various design changes*

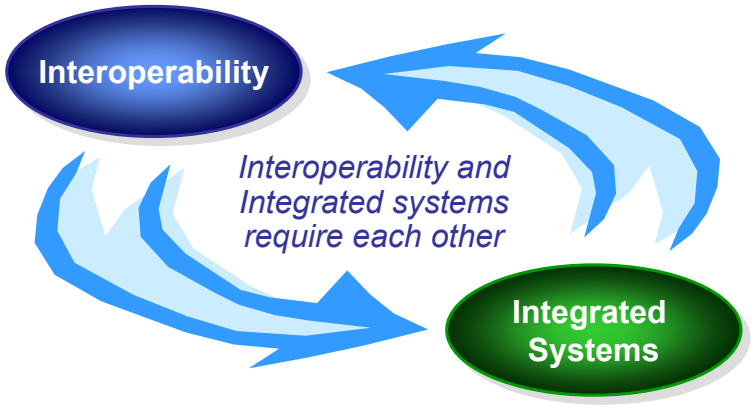
- Problems:
- One system can't support all views and users.
- Can't impose a specific system on all users on a projects.
- Can't enter or create all necessary information within one tool.

Integrated Systems

The University of British Columbia

Next Steps: Level of Integration

Solutions to Integration



The University of British Columbia

Next Steps: Level of Integration

Characteristics of Integrated Systems

- Cluster many functional views around an overall task of building, maintaining, and interacting with an integrated project database.
- Model-based approach:
  - All information structured around an object-oriented data model of the project information
- The product model (the physical components of the built facility) plays a central role.
  - The product model is interlinked with many other types or project information.
- Most information can be exchanged with other systems (interoperability).

Next Steps: Level of Integration

Smart Objects

- At the core of the model-based approach are parametric AEC objects that can model project information throughout the project lifecycle.
- Like data exchange models..
  - these objects support the ***representation*** of various aspects of project information.
- Unlike data exchange models…
  - they also implement the object's behavioral characteristics or ***business logic***.
    - E.g., objects' methods can automatically perform quantity takeoff from the geometric representations.

Next Steps: Level of Integration

Prototypes

- We are interested in integrating a range of project management tasks with the basic project design information, implemented in a CAD-based environment.
- Two, inter-related prototype systems
  - Falsework Design System
  - Integrated Construction Planning System

Next Steps: Level of Integration

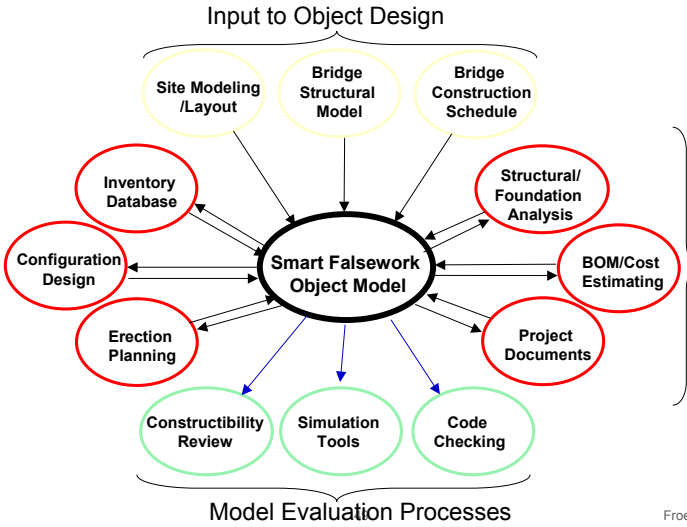
An Integrated Falsework Design And Management System

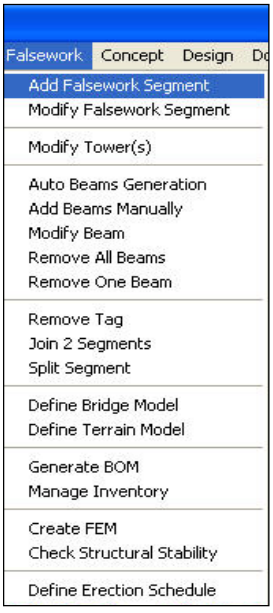
- Developed in collaboration with Hua Construction in Taiwan.
- Domain: Falsework systems for cast-in-place concrete box girder bridge structures.
- Layout, structural analysis, sequencing, cost, inventory, etc.

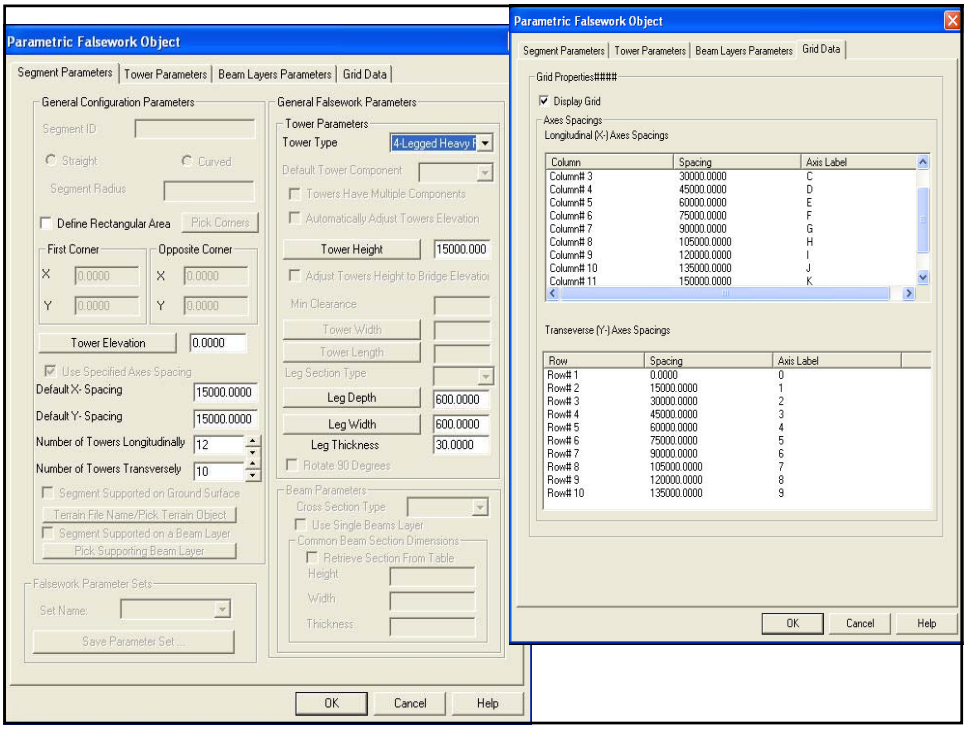
The University of British Columbia

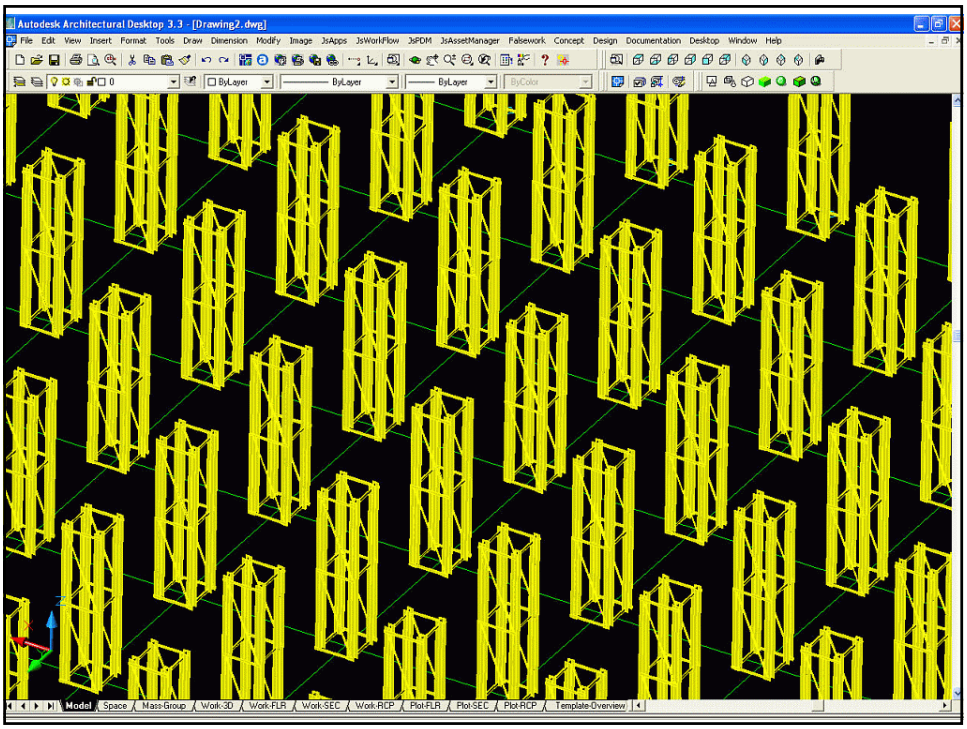
Next Steps: Level of Integration

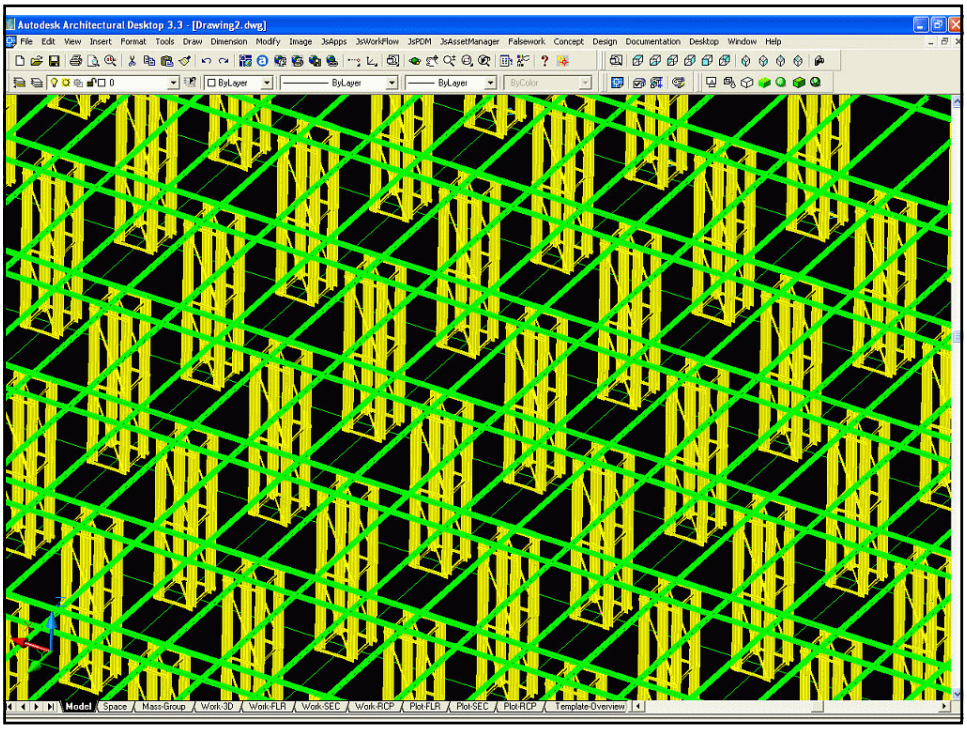
Product-Centric Approach to Integrate Falsework Project Processes

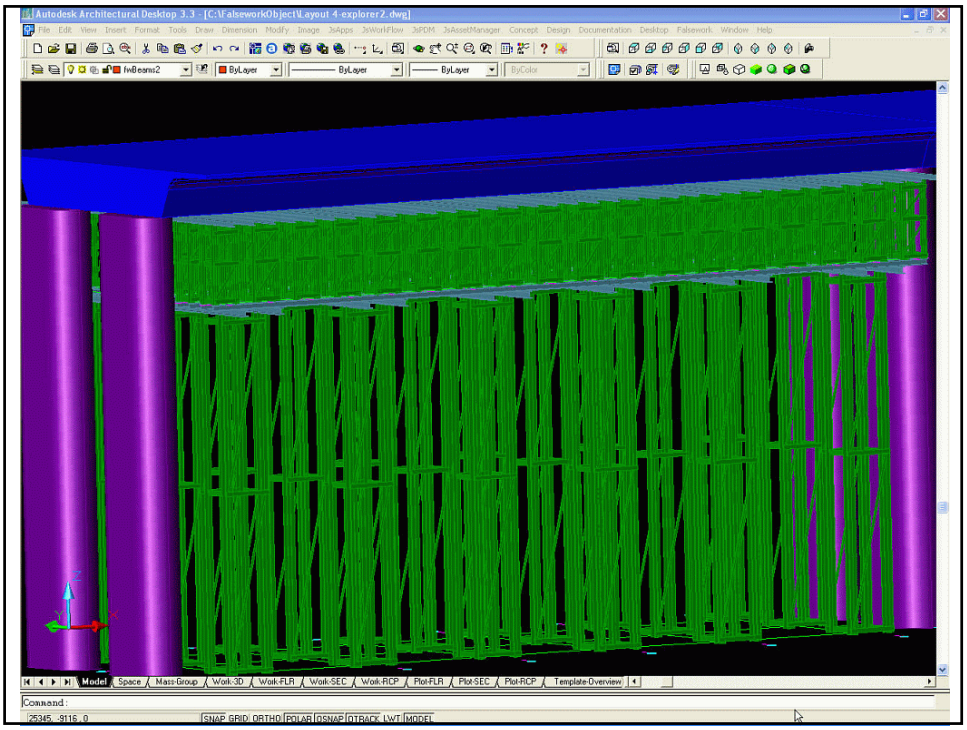


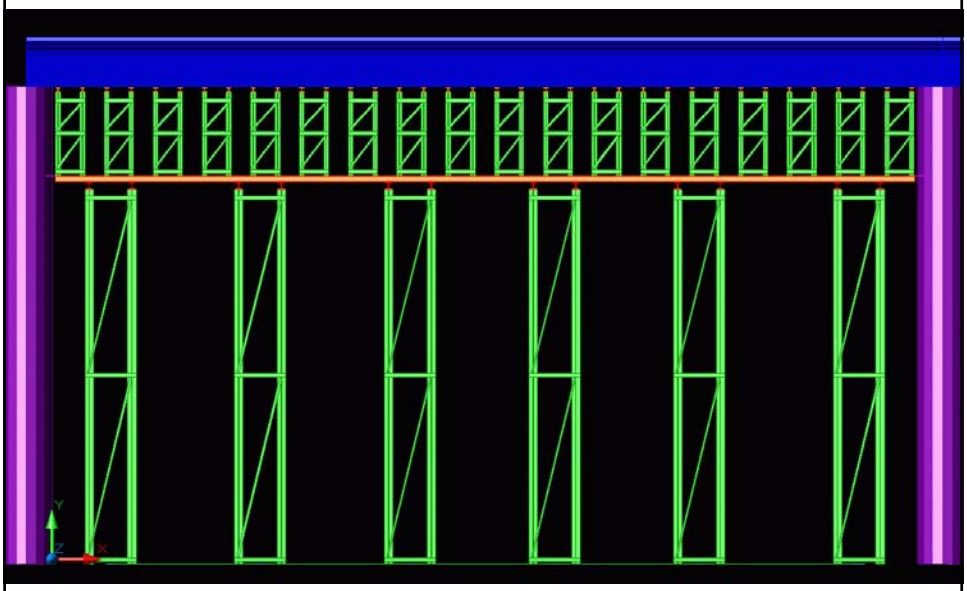


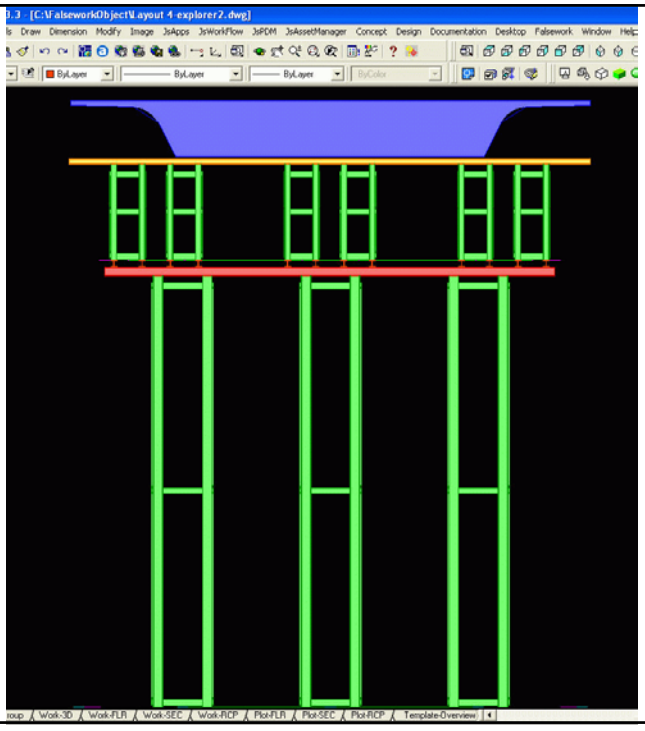

Parametric Falsework Object
Segment Parameters
Tower Parameters
Beam Layers Parameters
Grid Data
General Configuration Parameters
Segment ID
Straight
Curved
Segment Radius
Define Rectangular Area
Pick Corners
First Corner
Opposite Corner
Tower Elevation
0.0000
Use Specified Axes Spacing
Default X- Spacing
15000.0000
Default Y- Spacing
15000.0000
Number of Towers Longitudinally
12
Number of Towers Transversely
10
Segment Supported on Ground Surface
Terrain File Name/Pick Terrain Object
Segment Supported on a Beam Layer
Pick Supporting Beam Layer
Falsework Parameter Sets
Set Name:
Save Parameter Set...
General Falsework Parameters
Tower Parameters
Tower Type
4-Legged Heavy F
Default Tower Component
Towers Have Multiple Components
Automatically Adjust Towers Elevation
Tower Height
15000.000
Adjust Towers Height to Bridge Elevatio
Min Clearance
Tower Width
Tower Length
Leg Section Type
Leg Depth
600.0000
Leg Width
600.0000
Leg Thickness
30.0000
Rotate 90 Degrees
Beam Parameters
Cross Section Type
Use Single Beams Layer
Common Beam Section Dimensions
Retrieve Section From Table
Height
Width
Thickness
OK
Cancel
Help
Parametric Falsework Object
Segment Parameters
Tower Parameters
Beam Layers Parameters
Grid Data
Grid Properties#####
Display Grid
Axes Spacings
Longitudinal (X-) Axes Spacings
Column
Spacing
Axis Label
Column# 3 30000.0000 C
Column# 4 45000.0000 D
Column# 5 60000.0000 E
Column# 6 75000.0000 F
Column# 7 90000.0000 G
Column# 8 105000.0000 H
Column# 9 120000.0000 I
Column# 10 135000.0000 J
Column# 11 150000.0000 K
Transeverse (Y-) Axes Spacings
Row
Spacing
Axis Label
Row# 1 0.0000 0
Row# 2 15000.0000 1
Row# 3 30000.0000 2
Row# 4 45000.0000 3
Row# 5 60000.0000 4
Row# 6 75000.0000 5
Row# 7 90000.0000 6
Row# 8 105000.0000 7
Row# 9 120000.0000 8
Row# 10 135000.0000 9
OK
Cancel
Help


Autodesk Architectural Desktop 3.3 - [Drawing2.dwg]
File Edit View Insert Format Tools Draw Dimension Modify Image JsApps JsWorkFlow JsPDM JsAssetManager Falsework Concept Design Documentation Desktop Window Help
Model
Space
Mass-Group
Work-3D
Work-FLR
Work-SEC
Work-RCP
Plot-FLR
Plot-SEC
Plot-RCP
Template-Overview


Autodesk Architectural Desktop 3.3 - [Drawing2.dwg]
File Edit View Insert Format Tools Draw Dimension Modify Image JsApps JsWorkFlow JsPDM JsAssetManager Falsework Concept Design Documentation Desktop Window Help
Model Space Mass-Group Work-3D Work-FLR Work-SEC Work-RCP Plot-FLR Plot-SEC Plot-RCP Template-Overview


Autodesk Architectural Desktop 3.3 - [C:\Falsework0bject\Layout 4-explorer2.dwg]
Model Space Mass-Group Work-3D Work-FLR Work-SEC Work-RCP Plot-FLR Plot-SEC Plot-RCP Template-Overview
Command :
SNAP GRID ORTHO POLAR OSNAP OTRACK LWT MODEL

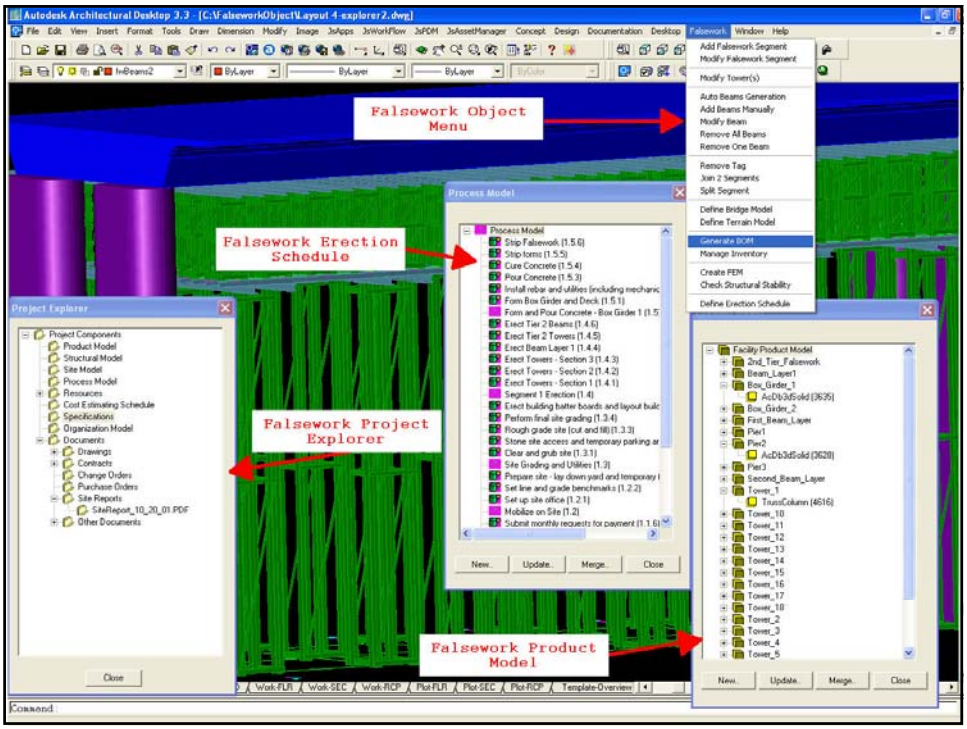


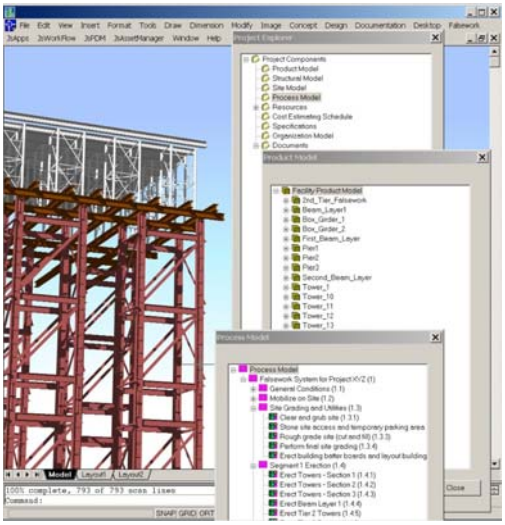

Draw Dimension Modify Image JsApps JsWorkFlow JsPDM JsAssetManager Concept Design Documentation Desktop Falsework Window
ByLayer
ByLayer
ByLayer
ByColor
Work-3D
Work-FLR
Work-SEC
Work-FICP
Plot-FLR
Plot-SEC
Plot-FICP
Template-Overview

Next Steps: Level of Integration

Integrated Construction Planning System

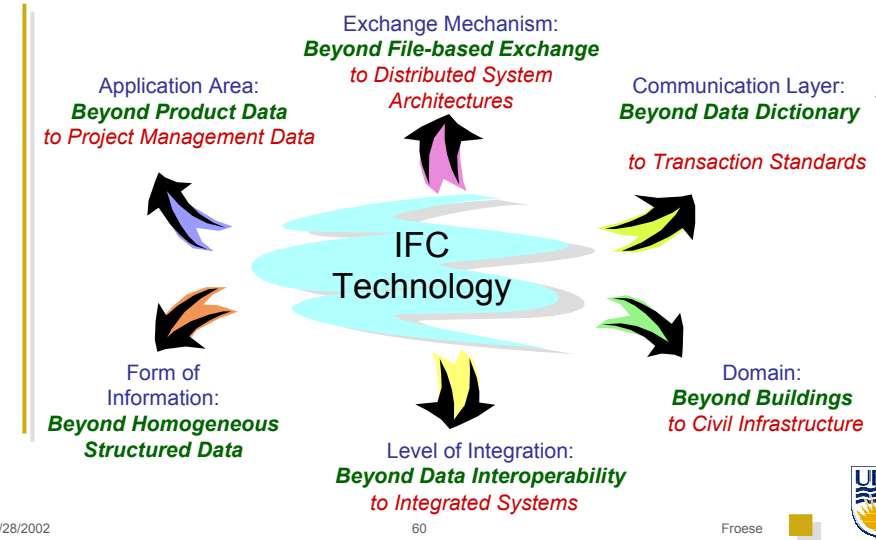
- Uses a generic “product” concept that can link into more specific product models for various types of facilities.
- Integrates schedule, cost, document, and other types of information with the product model for the purpose of construction planning.
- Used as a layer on top of the Falsework design system to provide much of the project-level integration.





Next Steps

Moving Beyond Basic IFC Data Exchange

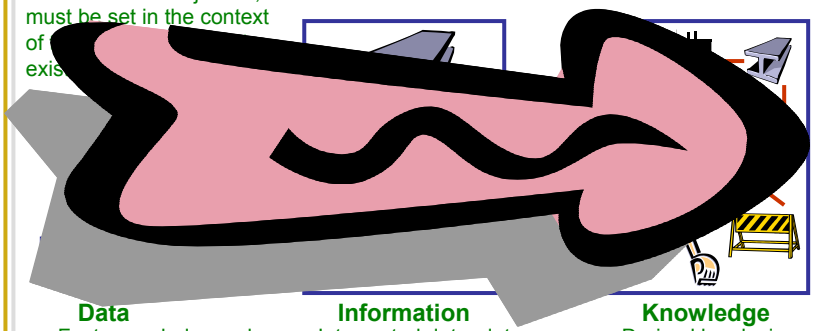


Forms of Information

- The IFCs deal with data that are fully structured according to a common standard.
- Most information available on AEC/FM projects is unstructured or semi-structured
  - e.g., Word documents, spreadsheets, photographs, etc.

Data, Information, and Knowledge

Subject
Information is subjective; it must be set in the context of
exis



Data
Facts; symbols used to represent something

Information
Interpreted data; data placed within a meaningful context.

Knowledge
Derived by placing information within existing knowledge

Knowledge Management

- ***Knowledge Management:*** turning large collections of "data" into useful "knowledge"
  - Easily find all data relevant to a given issue.
  - *(just one of many possible approaches).*
- Knowledge management systems must be able to access, search, filter, and present relevant information.
  - Makes extensive use of relationships.
- For unstructured data:
  - Uses structured meta-data (data *about* the information item)
  - Resolves partial structured data from the unstructured data (text processing techniques, drawing recognition, etc.)

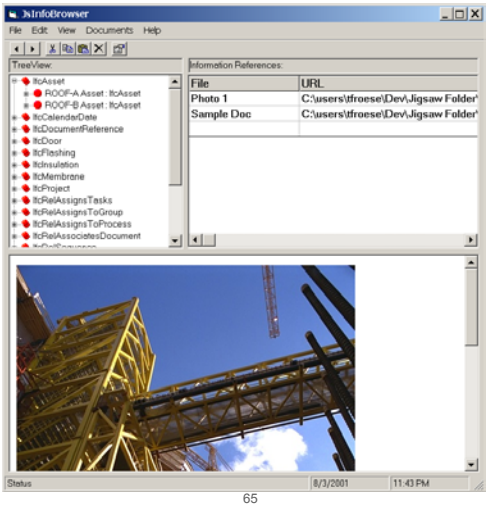
Role of IFCs for Heterogeneous Information and Knowledge Management

- Role of IFCs:
  - Structure Meta-data
    - E.g., index documents to products, tasks, organizations, costs, etc., which can all be IFC objects.
  - Link unstructured information into IFC models as external document references.
  - Structure partially structured data extracted from un-structured documents.

Next Steps: Forms of Information

Information Browser

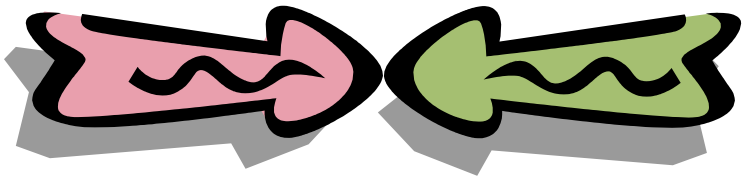
- Associates unstructured documents with any structured objects.



Next Steps: Forms of Information

Model / Document "Unification"

- This provides a path for the integration of IFC model-based technology with web-based project information systems.



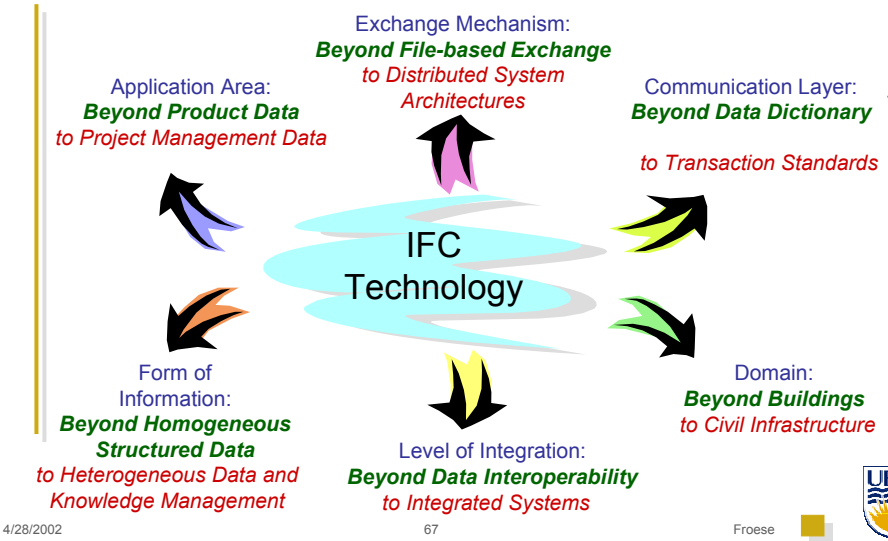
IFC Models

Web-based Project Information Systems

(DotCOMS)

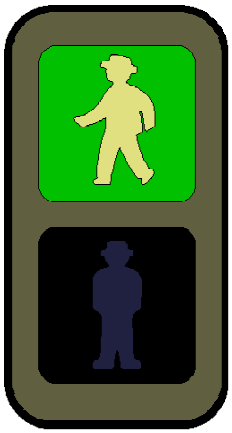
Next Steps

Moving Beyond Basic IFC Data Exchange



Conclusions

Conclusions



IFCs are ready for use

IFCs require significant additional development