Framework for Component-Based Structural Engineering Software

Naveed Anwar, Worsak Kanok-Nukulchai

CR-AU-0248
Introduction
Structural Design Process

Principal

Design Management
Client Dealing

Cooperation

Engineer/Designer

Conception
Costing

Analysis
Detailing

Design

Design Management

Design Process
Conventional Application Architecture

- Pre-Processor
- Raw Input
- Processor
- Raw Output
- Post-Processor

External Interfaces
Visualizer
Database
Development of Component Based Framework
Component-Based Software Development

- Logical extension of object-oriented concepts and object technologies
- Components are developed independent of each other and assembled together to create new applications using as predefined framework
- Encourages and supports reuse of functionality at higher abstraction and packaging
- Helps collaborative and incremental and faster development
- Being widely used in several areas, particularly in business applications
No significant reference could be found in literature on use of Component Based Software Development in structural engineering.

Some references found in construction management applications, specially by Pena-Mora.

However, use of object oriented approach is becoming popular in structural engineering software, especially at research level and can serve as the basis for CBSD.
The information processing is taken as the basis for developing the application architecture and framework. This approach helps to integrate the application within as well as with information outside the domain. Each major information processing unit is perceived as an application package. The packages and components also retain the physical as well as computational divisions between design process.
Global Information Space

Information for Domain 1

Application 1/A

Application 1/B

Information for Domain 2

Application 2/A

Application 2/B

Inter Disciplinary Application
Information Flow in Integrated Structural Design

- Site Information
- Structural Design Process
- Ancillary Services
- Project Approval and Control Information
- Construction Information
- Project Management Information
- Architectural Planning Information
Structural Design Information Space

- Conception
- Modeling
- Analysis
- Design
- Detailing
- Drafting

The diagram illustrates the flow of information and processes in structural design, highlighting the interconnections between conception, modeling, analysis, design, detailing, and drafting.
• Level-1: The application made up of one or more packages. Each package handles one major task or step in the overall information processing

• Level-2: Packages made up of one or more integrated components. Each component provides specific services or handles dedicated tasks

• Level-3: The component made up of one or more objects. Each object provides specific services or handles specific information processing tasks
Basic Concept of Proposed Framework

- Application
  - Handles *overall* design process

- Packages
  - Handles *one step* of the design process

- Components
  - Handles *one aspect of* of the design process steps

- Objects
  - Handles *one portion of* of one aspect of design step

- Data/Information
  - Stores *information of* a portion of design aspect

- Services/Processes
  - Provides *functionality* for a portion of design aspect
The Software Package

Application Package

Integration Code

Component 1

Component Information

Objects

Data

Services

Component n

Component Information

Objects

Data

Services

Package Information
The Software Component
A Loosely Integrated Framework

Package Based Framework for Integrated Structural Design

1. Conceptual and Preliminary Design
2. Modeling and Analysis
3. Design and Detailing
4. Detailing and Drafting
5. Costing, BOQ, BOM, etc
6. Documentation and Reporting

Project Design Information
### Component Based Framework for Integrated Structural Design

<table>
<thead>
<tr>
<th>Package</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conceptual and Preliminary Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Modeling and Analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design and Detailing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detailing and Drafting</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Costing, BOQ, BOM, etc</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Documentation and Reporting</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Packages with their specific components using other composite and atomic components

#### Composite components using atomic components

<table>
<thead>
<tr>
<th>Component</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graphic Display</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual Reality</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design Information</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reporting</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graphs and Charts</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X-Section Analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Composite components using atomic components

<table>
<thead>
<tr>
<th>Component</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Math and Matrix</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Geometry</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Basic Meshing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Basic Graphics</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>String Processing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>General XML Parser</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Composite components using atomic components

Information Base
The Pattern for Packages

- Internalizer
- Externalizer
- User Interface
- Processor
The Conceptual and Preliminary Design Package

Diagram:

- **Internalizer**
- **System Selector**
- **Preliminary Designer**
- **Structure Model Builder**
- **Externalizer**

**User Interface**
The Modeling and Analysis Package

- Internalizer
- Pre-Processor
- Solver
- Post-Processor
- Externalizer

User Interface
Detailing and Drafting Package

User Interface

- Internalizer
- Member Detailers
- Beam
- Slab
- Externalizer
- Detailed Views
- Drawings
- CAD Output
The Cost Estimation and Documentation Package

User Interface

Costing and Documentation

Externalizer

Bill of Quantity (BOQ)  Bill of Material (BOM)  Costing  Documentation System

Internalizer

Pre-processor

Detailing Informal

Informal Preliminary Design

Pre-processor
Using Legacy Code in Components

Encapsulating an exiting Application

Encapsulating exiting code
No significant reference could be found in literature on the use of XML to represent the structural design or structural model information.

Some work is being done in construction management applications.

Several other disciplines such as Electrical Engineering, Chemical Engineering already have their XML information standards.

The best known effort in developing information schema is the CIS/2 from CIMsteel integration standards, but does not use XML.
XML to Represent Design Information

- Represent the structural design process information in XML
  - Develop information description and structure
  - Define Document Type Definition (DTD) and or Schema to use as basis for XML documents
  - Write actual information in XML documents using the DTD/Schema
  - Write parsers to read and interpret XML document

- The proposed The Structural Model Description Language (SMDL)

- This is similar in concept to CIS/2 from CIMsteel Integration Standards, USA, for steel fabrication, manufacturing and detailing
The Pattern for Packages

User Interface

Processor

Internalizer

XML to Internal

Externalizer

Internal to XML
Applications and Implementations
Architecture for Pre and Post Processing Component

The Graphic Pre and Post Processing Component

- Interactive 2D Graphics Editing Component
- General 3D Graphics Display Component
- Geometric Model Transformation Component
- Parametric Structure Generator Component

Geometric Model Information in GMF

XML to GMF to XML Converter

XML Information Bus
Usage Model

The Proposed Generalized Pre and Post Processing Component System

View Model Geometry Graphically
Transfer Geometry to External Formats

Pre-meshed structural geometry
Section Properties
Action-Deformation Curves

View Model Results Graphically
Compute and Check Capacity
Transfer Results to External Formats
Generate and View Stress Plots

Pre-processor
Post-processor

Raw Model Data
FEM Solver
Raw Model Results
Conclusions
Conclusions

- A component based framework has been developed with a view to standardize and streamline the development of structural design applications.
- The structural design information has been defined in several contexts and models, specially topological model, geometric model and analysis model.
- Schemas have been developed to describe various structural design information models in XML. Sample XML files have been written to actual information.
- Sample component has been developed to determine the response of composite cross-sections.
Conclusions

- The framework and the concepts presented in this research have been used in actual software development in the form of a “Structural Object Modeler” component that is being used by professional software.
- Based on the present research, additional work is in progress for development of comprehensive component design including class hierarchy and object model to handle the entire structural modeling, analysis and design application.