

# Green Building XML: Open XML Schema for Sustainable Design

Prepared for NIST Project by  
Weichao Wang  
UNC Charlotte

# Learning Objectives

- Learn the basics of XML
- Learn the unique features of gbXML and how they help the BIM system
- Through examples of gbXML, learn its capabilities and usage in BIM
- Get hands-on experiences in gbXML interactive visualization tools



# Outline

- Green Building XML History
- Basics of XML
- Basics of gbXML
- gbXML examples for BIM
- Hands-on exercises



# Green Building XML History

- Data exchange between 3D-CAD and Green analysis applications
- The xml file should be viewer and editing tool independent
- Development started Dec. 1999
- Published June 2000
- Funding – development and promotion
  - California Energy Commission
  - Green Building Studio, Inc.
  - Pacific Gas & Electric



# Ownership and Support

- Non-proprietary open data standard
- Autodesk acquired Green Building Studio in 2008
  - gbXML was part of the acquisition but is not “owned” by ADSK
  - A new advisory board was formed in Q2 of 2009 to promote to the industry as a whole
- Maintained by an industry consortium
  - 200+ members
  - 20+ applications
  - 11 advisory board members
  - Industrial partners try to add their preferred standards and models into the documents
  - The GitHub page is at: [Green Building XML \(gbXML\) · GitHub](#)



# XML: Definition and Example

- XML is a text-based markup language that is fast becoming a standard of data interchange
  - An open standard from W3C
  - A direct descendant from SGML

## Example: Product Inventory Data

```
<Product>  
  <Name>Refrigerator</Name>  
  <Model Number>R3456d2h</Model Number>  
  <Manufacturer>General Electric</Manufacturer>  
  <Price>1290.00</Price>  
  <Quantity>1200</Quantity>  
</Product>
```

See the example on the left, each field has a unique name and associated value in it. It becomes easy for an automatic analyzer to extract and analyze information.



# XML: Data Interchange

- XML's key role is data interchange
- Two business partners want to exchange customer data
  - Agree on a set of tags
  - Exchange data without having to change internal databases
- Other business partners can participate by using the same tag-set
  - New tags can be added to extend the functionality

Key to successful data interchange is building consensus and standardizing of tag sets



# XML: Structure

- Prolog
  - Instructs the parser as to what it is parsing
  - Contains processing instructions for processor
- Body
  - Tags - Entities
  - Attributes - Properties of Entities
  - Comments - Statements for clarification in the document



# XML: Structure

## Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?> ← Prolog
```

```
<contact>
```

```
  <name>
```

```
    <first name>Sanjay</first name>
```

```
    <last name>Goel</last name>
```

```
  </name>
```

```
  <address>
```

```
    <street>56 Della Street</street>
```

```
    <city>Phoenix</city>
```

```
    <state>AZ</state>
```

```
    <zip>15784</zip>
```

```
  </address>
```

```
</contact>
```

← Body



# XML: Syntax: Elements & Attributes

- Uses less-than and greater-than characters (<...>) as delimiters
- Every opening tag must have an accompanying closing tag
  - <First Name>Sanjay</First Name>
  - Empty tags do not require an accompanying closing tag.
  - Empty tags have a forward slash before the greater-than sign e.g. <Name/>
- Tags can have attributes which must be enclosed in double quotes
  - <name first="Sanjay" last="Goel">
- Elements should be properly nested
  - The nesting can not be interleaved
  - Each document must have one single root element
- Elements and attribute names are case sensitive



# Tree Structure: Elements

- XML documents have a tree structure containing multiple levels of nested tags.
  - Root element is a single XML element which encloses all of the remaining XML elements and data in the document
  - All other elements are children of the root element



# Tree Structure: Elements

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
```

```
<contact>
```

```
<name>
```

```
<first name>Sanjay</first name>
```

```
<last name>Goel</last name>
```

```
</name>
```

```
<address>
```

```
<street>56 Della Street</street>
```

```
<city>Phoenix</city>
```

```
<state>AZ</state>
```

```
<zip>15784</zip>
```

```
</address>
```

```
</contact>
```

← Root Element

← Child Elements



# Attributes: Definition and Example

- Attributes are properties associated with an element
- Each attribute is a name value pair
  - No element may contain two attributes with same name
  - Name and value are strings

## Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<contact>
  <name first="Sanjay" last="Goel"></name> ← Attributes
  <address>
    <street>56 Della Street</street> ← Nested Elements
    <city>Phoenix</city>
    <state>AZ</state>
    <zip>15784</zip>
  </address>
</contact>
```



# Elements vs. Attributes: Comparison

- Data should be stored in Elements
- Information about data (meta-data) should be stored in attributes
  - When in doubt use elements
- Rules of thumb
  - Elements should have information which some one may want to read.
  - Attributes are appropriate for information about document that has nothing to do with content of document
    - e.g. URLs, units, references, ids belong to attributes
  - What is your meta-data may be some one else's data



# Student Exercise: XML file

- Please see the following XML file segment.
  - What does the XML segment try to describe?
  - Identify at least two errors in the XML segment

```
<?xml version="1.0" encoding="utf-16"?>
<movies>
  <movie id="56225">
    <title>Love Story</title>
    <title></title>
    <year>1980</year>
    <_director name='Coppola'></_director>
    <comment text="Five start" text="Average"/>
    <xml>Introduce XML content</xml>
    <newcomment text="An <important> text">Oscar</newcomment>
    <comment lang=de>&copy; 1980 Warner Bros.</comment>
    <!-- Famous movie of the --80s -->
  </Movie>
</movies>
```



# gbXML Main Objective

- Sustainable building analyses
- Minimize human interpretation mistakes
- Reduce cost
- Improve efficiency
- Relationship to XML
  - XML based
  - 306 XML elements
  - 104 specific enumerations
  - Predefined property sets
  - No inherent extension mechanism



# Sustainable Building Analyses

- Whole building energy use & costs
- Water use & costs (indoor, outdoor, and catchment)
- Carbon emissions
- Heating and cooling load analysis
- LCA
- Renewable energy
- HVAC equipment sizing
- Lighting analysis
- Fire analysis
- Solar/shading analysis
- Energy code compliance
- Thermal comfort



# Data Capabilities

- 3D planar polygon geometry
- 2D rectangular polygon geometry
- Space boundaries
- Opaque constructions and materials
  - Thermal and emission properties, recycled content,
  - Costs including LCA (embodied, first, and future)
  - Fire properties
- Glazing, shades, and their operation
- Internal and external equipment
  - Energy, power, efficiencies, water use, physical characteristics
  - Costs including LCA (embodied, first, and future)
- Lighting and controls

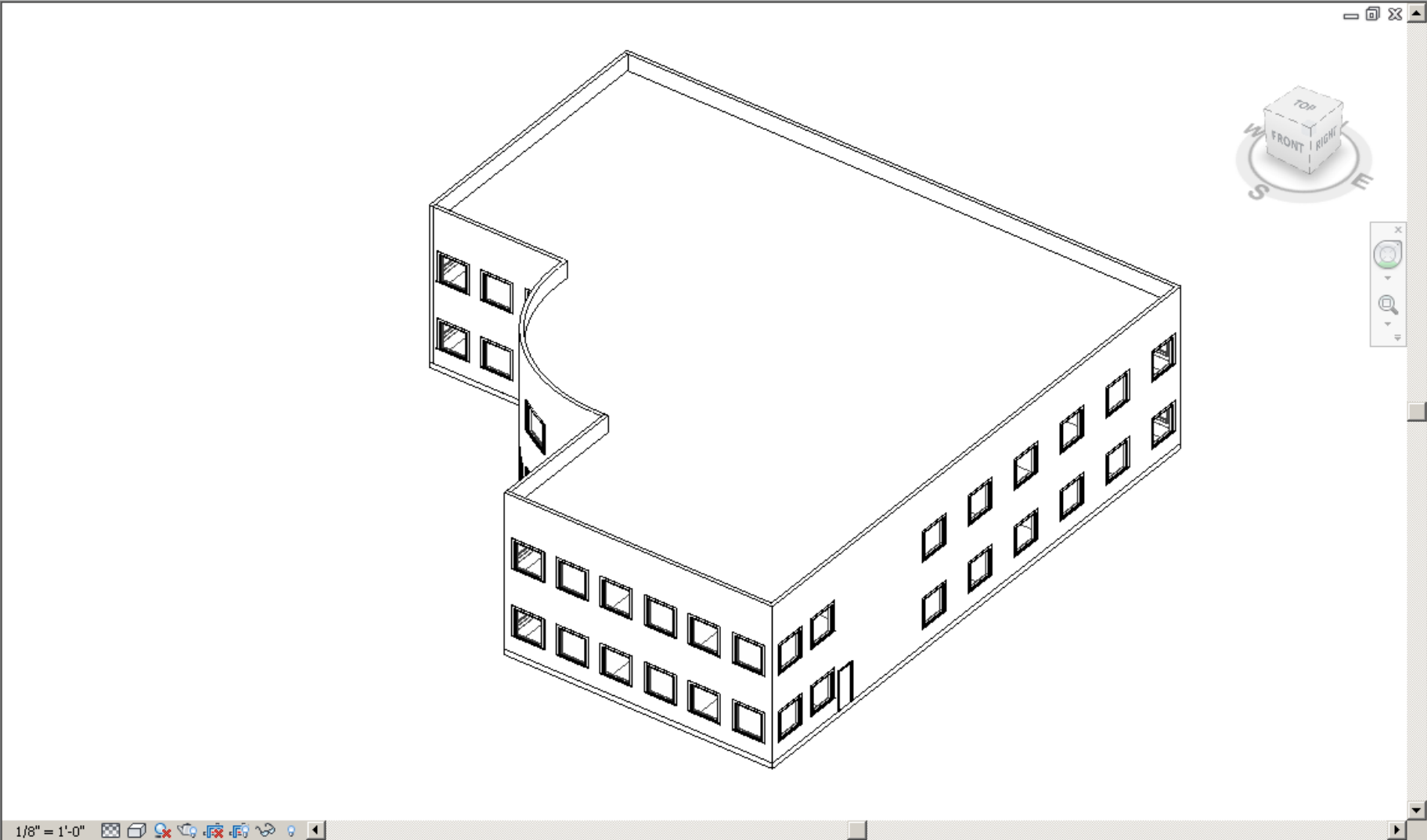


# Data Capabilities (continued)

- HVAC equipment
- Weather design data
- Ventilation requirements
- Operation schedules
  - Occupancy, lighting, equipment, HVAC, temperatures.
- Infiltration
- Transportation types, location, & schedule
- Vegetation types, location, & water use.
- Versioning and change history



# 3D BIM Model



# Analytical Model Derived from gbXML

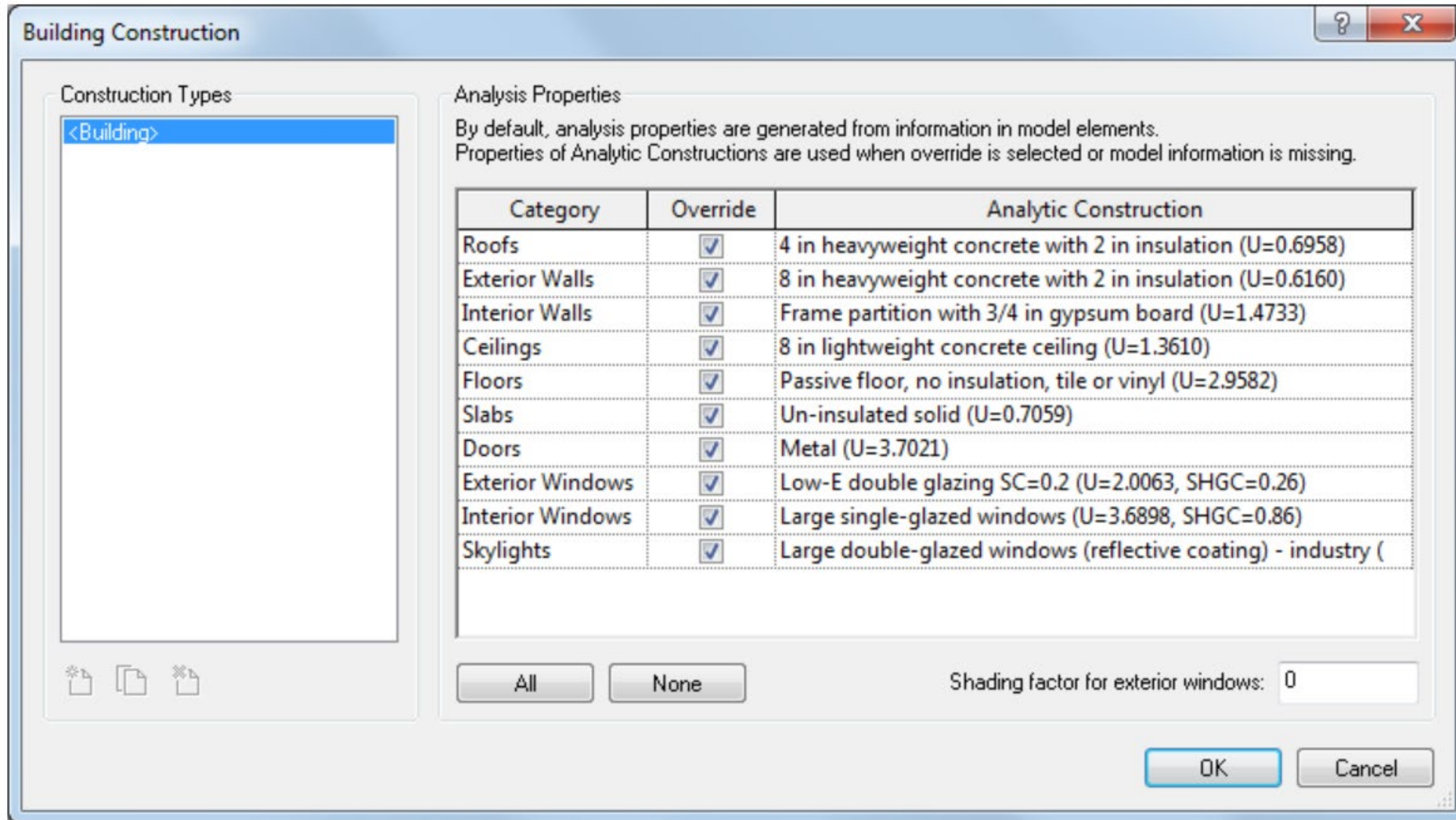
Heating and Cooling Loads

General Details

Parameter	Value
Building Type	Office
Location	Boston, MA
Building Service	VAV - Single Duct
Building Construction	<Building>
Building Infiltration Class	None
Report Type	Standard
Ground Plane	
Project Phase	Phase 1
Sliver Space Tolerance	1' 0"

Calculate OK Cancel

# AutoDesk software gbXML “Construction” dialog



# gbXML Support Software

- CAD/BIM
  - Autodesk
    - AutoCAD Architecture & MEP
    - Green Building Studio (GBS)
    - Ecotect
  - Bentley
    - Architecture
    - Building Mechanical Systems
    - Speedikon Architectural
  - Google
    - SketchUp
  - Graphisoft
    - ArchiCAD
    - Mac and Windows
- HVAC/Energy
  - CADLine
    - Cymap
  - Carrier
    - Hourly Analysis Program (HAP)
  - DOE-2.2 & eQuest (via GBS)
  - Elite Software
  - EnergyPlus (via GBS)
  - Energy Soft\*
  - Environmental Design Solutions Ltd.
    - Tas
  - IES, Ltd.
    - IES <Virtual Environment>
  - Trane
    - TRACE 700



# gbXML in Practice

- Most widely used format for sustainable analyses in the world
- Fastest method for exchanging sustainable analysis data



# Performance Data

- Glazing example
- Actual manufacturer data
- Analysis ready data

```
- <WindowType id="NPwindowType-210" DOELibIdRef="100001">
  <Name>PPG Double Pane SB70XL</Name>
  <Description>PPG Double Pane SB70XL</Description>
  <U-value unit="WPerSquareMeterK">1.625</U-value>
  <SolarHeatGainCoeff unit="Fraction">0.274</SolarHeatGainCoeff>
  <Transmittance type="Visible">0.643</Transmittance>
- <Glaze id="Glaze-L1-NPwindowType-210">
  <Name>PPG SB70XL Starphire 6</Name>
  <Description>PPG SB70XL Starphire 6</Description>
  <Thickness unit="Feet">0.01857</Thickness>
  <Conductivity unit="WPerMeterK">1</Conductivity>
  <Transmittance type="Solar" unit="Fraction">0.28</Transmittance>
  <Transmittance type="Visible" unit="Fraction">0.722</Transmittance>
  <Transmittance type="IR" unit="Fraction">0</Transmittance>
  <Reflectance type="ExtSolar" unit="Fraction">0.499</Reflectance>
  <Reflectance type="IntSolar" unit="Fraction">0.559</Reflectance>
  <Reflectance type="ExtVisible" unit="Fraction">0.068</Reflectance>
  <Reflectance type="IntVisible" unit="Fraction">0.051</Reflectance>
  <Emittance type="ExtIR" unit="Fraction">0.841</Emittance>
  <Emittance type="IntIR" unit="Fraction">0.029</Emittance>
</Glaze>
- <Gap id="Gas-L2-NPwindowType-210">
  <Name>KRYPTON 3MM</Name>
  <Description>Krypton</Description>
  <Thickness unit="Feet">0.0105</Thickness>
  <Conductivity unit="WPerMeterK">0.1</Conductivity>
</Gap>
- <Glaze id="Glaze-L3-NPwindowType-210">
  <Name>PPG CLEAR 6</Name>
  <Description>PPG CLEAR 6</Description>
  <Thickness unit="Feet">0.01857</Thickness>
  <Conductivity unit="WPerMeterK">1</Conductivity>
  <Transmittance type="Solar" unit="Fraction">0.771</Transmittance>
  <Transmittance type="Visible" unit="Fraction">0.886</Transmittance>
  <Transmittance type="IR" unit="Fraction">0</Transmittance>
  <Reflectance type="ExtSolar" unit="Fraction">0.072</Reflectance>
  <Reflectance type="IntSolar" unit="Fraction">0.073</Reflectance>
  <Reflectance type="ExtVisible" unit="Fraction">0.085</Reflectance>
  <Reflectance type="IntVisible" unit="Fraction">0.085</Reflectance>
  <Emittance type="ExtIR" unit="Fraction">0.84</Emittance>
  <Emittance type="IntIR" unit="Fraction">0.84</Emittance>
</Glaze>
</WindowType>
```



# Student Exercise: gbXML reading and exercises

- Please go to this link to see the elements supported by gbXML
  - [Help | gbXML Element | Autodesk](#)
- Please refer to the “windowtype” element in the previous slide. Search on the internet to figure out the usage of “Glaze” and “gap” elements
- Refer to the “LightingSystem” element example at the following link, write the “LightingSystem” gbXML segment for the current classroom.
  - [Help | LightingSystem Element | Autodesk](#)



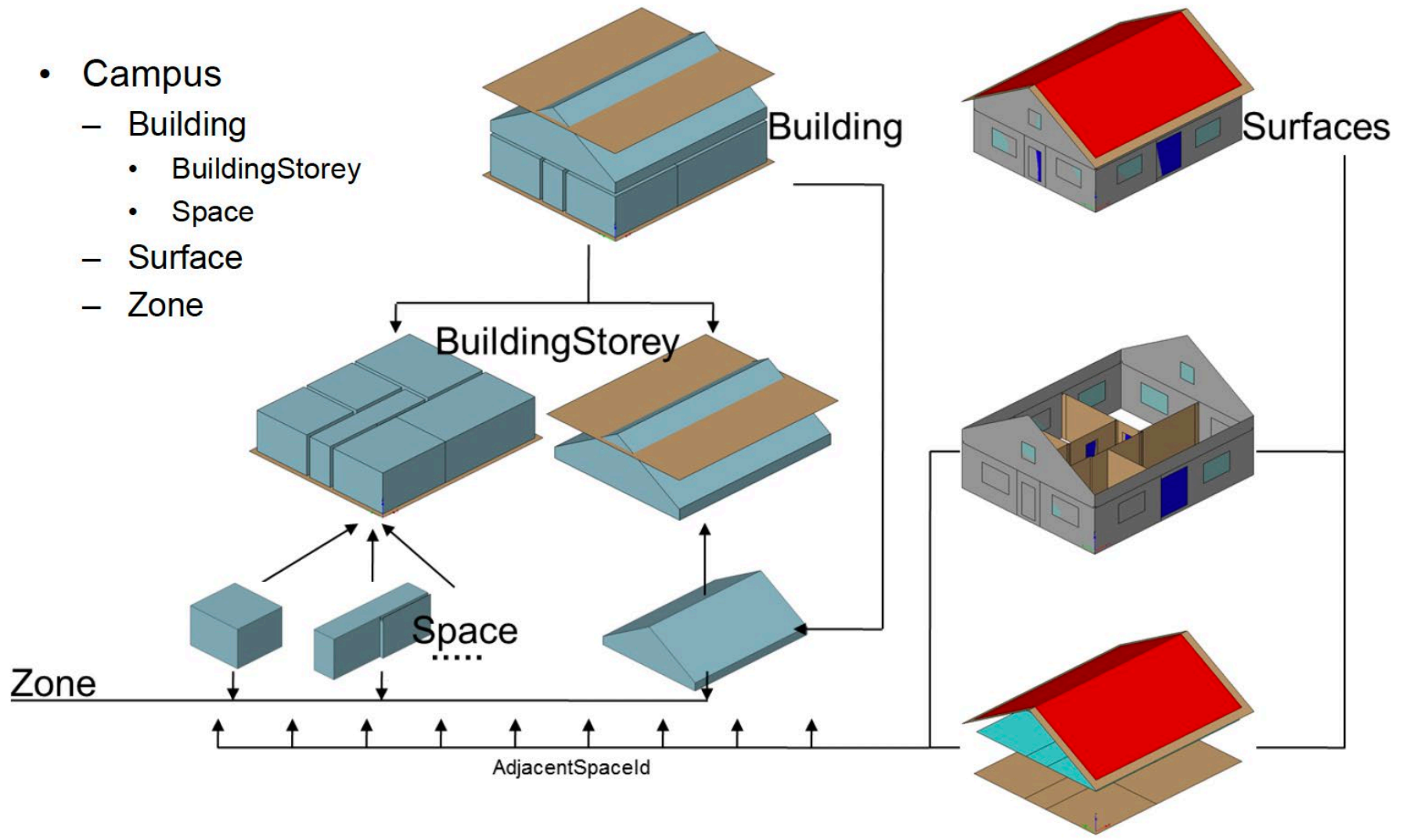
# gbXML Adoption Examples

- UC Berkeley – Bancroft Library \$38 mil. renovation
- Days saved with using AutoCAD-gbXML-TRACE 700
- Firm's practice changed to fully utilize gbXML
- Nearly every project has BIM model built, gbXML file exported and imported into Trace 700
- Average of two days saved on every project

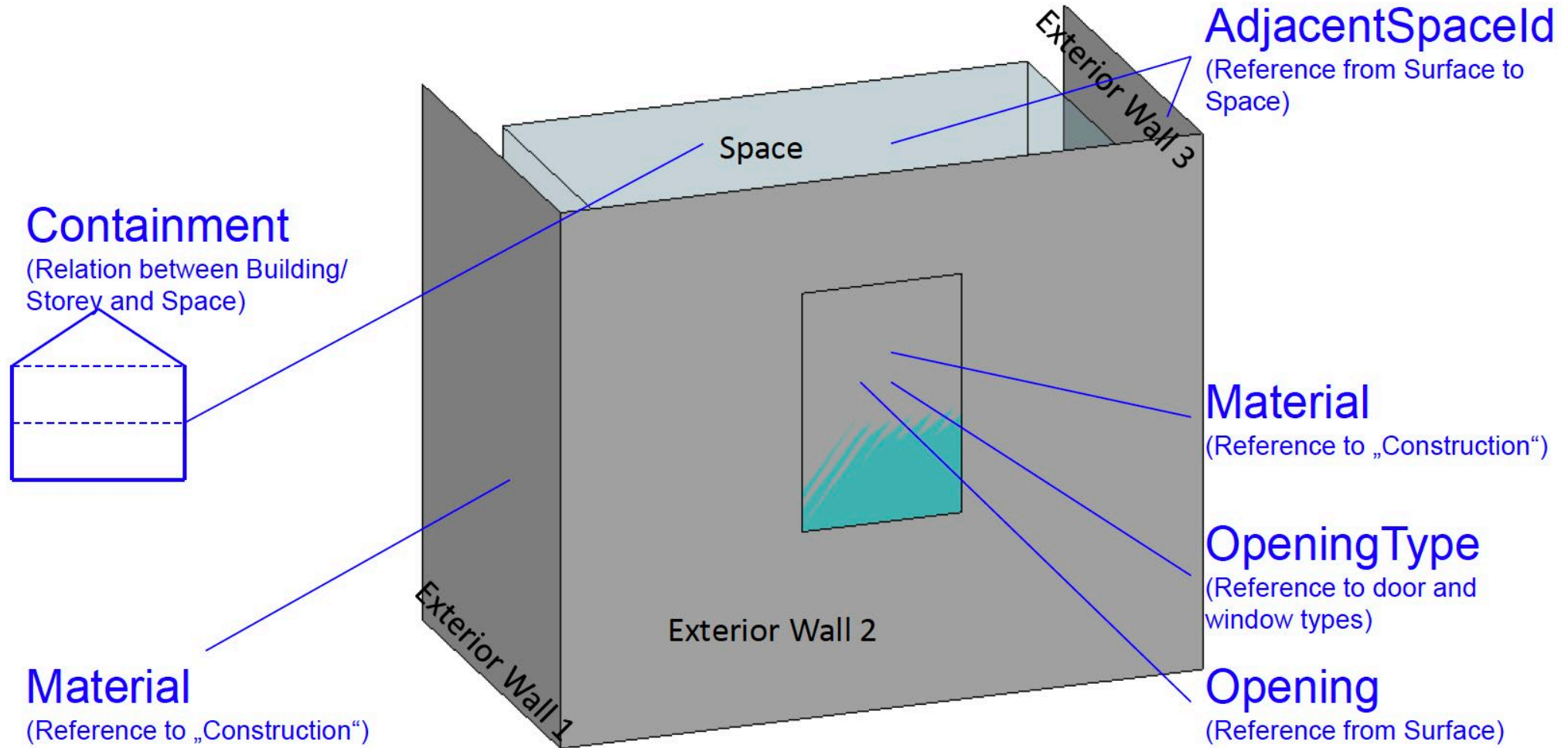


# gbXML example: Campus structure

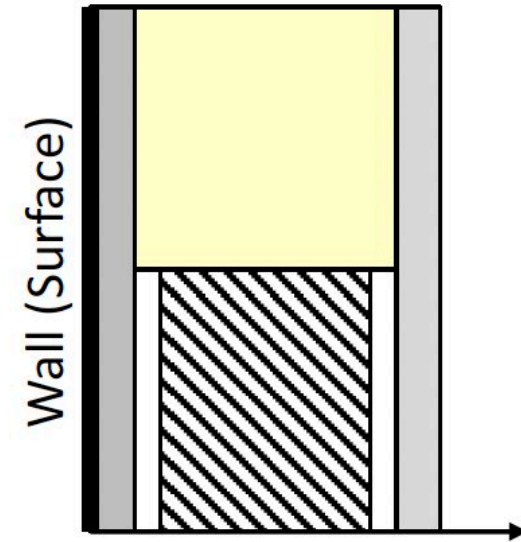
- Campus
  - Building
    - BuildingStorey
    - Space
  - Surface
  - Zone



# gbXML example: surfaces



# gbXML example: construction



Layer 1

Layer 2

Material 1 50% thickness 30 cm

Material 2 50% thickness 24 cm

Layer 3

Layered material structure with different material thickness

# gbXML example: nested elements of a “Ceiling”

```
<Construction id="ASHIF5" surfaceType="Ceiling">
  <Name>8 In. Light Weight Concrete Ceiling</Name>
  <Description>8 In. Light Weight Concrete Ceiling</Description>
  <LayerId layerIdRef="lay-ASHIF5" />
  <U-value unit="WPerSquareMeterK">1.3610</U-value>
</Construction>

<Layer id="lay-ASHIF5">
  <MaterialId materialIdRef="mat-AM13" />
</Layer>

<Material id="mat-AM13">
  <Name>8 in. lightweight concrete</Name>
  <Description>8 in. lightweight concrete</Description>
  <Thickness unit="Meters">0.2032</Thickness>
  <Conductivity unit="WPerMeterK">0.53</Conductivity>
  <Density unit="KgPerCubicM">1280</Density>
  <SpecificHeat unit="JPerKgK">840</SpecificHeat>
</Material>
```



# Geometry in gbXML

- Storeys optionally have an explicit geometrical representation
- The volume and/or the footprint area of a room may be represented
- Building elements like walls are modeled with surface geometry (planar polygons or parametric rectangles)
- The position of the surfaces depends on the building element's function (e.g. exterior or interior wall)





# The Future

- Continue to evolve with industry input
- Continue to standardize terminology
  - HVAC Components and Systems
- Interfaces with different automatic design tools and databases
- Support of future smart buildings



# Summary

- We introduced
  - Development history of Green Building XML
- Basics of XML
  - Grammar
  - Elements
  - Attributes
- Basics of gbXML
  - Elements and supported structures for BIM
- gbXML examples for BIM



# Assignment

- The assignment of this module consists of two parts
  - Generation of a gbXML file for a three-layer exterior wall (We will provide a starting gbXML file structure and corresponding information)
  - Hands-on exercise with gbXML Web browser



# Getting Started with Hands-on Experiences

Visit [www.gbxml.org](http://www.gbxml.org) for  
schema,  
more information, &  
Yahoo! gbXML group link.



# gbXML viewer through web browser

The screenshot displays the Aragog gbXML Viewer 12.39 web application. The browser address bar shows the URL: <https://www.ladybug.tools/spider/gbxml-viewer/r12/gv-app/gv-app.html>. The interface is divided into several sections:

- Left Menu:** Contains a header with a dropdown menu set to "Ladybug Tools home page", the application title "Aragog gbXML Viewer 12.39", a file upload section with a "Choose File" button and a file path input field, and a "Features" section with buttons for "Settings", "Reports", "Issues", "Numbers", and "reports".
- Top Navigation:** Includes buttons for "rotation", "surfaces", "edges", and "reset view".
- Right Menu:** Features a "toggle the visible items" section with buttons for "surface", "surfaces", "edges", and "all", and an "edit the surface" section with buttons for "delete surface", "modified by", and "save edits".
- Surface Details:** A dropdown menu for "surface" is open, showing a list of surface IDs (aim0126, aim0415, aim0438, aim0461, aim0484, aim0507, aim0530, aim0553). The selected surface (aim0126) has the following properties: id: aim0126, zoom: [input], type: SlabOnGrade, update: SlabOnGrade, cad object id: 4700, cad id: 4700, and area: 5893.8 ln 59.333 wd 99.333. An "update" button is located below the details.

The central 3D model shows a rectangular building with a red floor and brown walls. A blue line indicates the current view direction, and a green line points to a specific surface. The right menu also contains a note: "Model has only a single space, therefore there is no adjacent space data to be shown here." and a section for "For debugging surface appearance".

# gbXML Github page

The screenshot shows the Github page for Green Building XML (gbXML). The page header includes navigation links for Product, Solutions, Open Source, and Pricing, along with a search bar and Sign in/Sign up buttons. The repository profile for Green Building XML (gbXML) is displayed, featuring a profile picture, name, and a description: "Repositories for all things Green Building XML (gbXML) including schemas, sample gbXML files, validator source code, test cases, and more...". It also shows 6 followers, location (San Rafael, CA), website (http://www.gbxml.org), and email (info@carmelsoft.com). Navigation tabs include Overview, Repositories (22), Projects, Packages, and People (3).

**Pinned**

- TestCaseDevelopment\_GATech\_Phase1\_2018** (Public)  
gbXML test case development research performed by Georgia Tech - 2018/2019  
3 stars
- Sample\_gbXML\_Files** (Public)  
Includes a wide variety of sample gbXML files from different vendors and organizations.  
22 stars, 15 forks
- gbXML\_Schemas** (Public)  
Includes current and previous versions of the Green Building XML (gbXML) schema  
23 stars, 4 forks

**Repositories**

Find a repository... Type Language Sort

- buildingsync\_schema** (Public)  
BuildingSync® Schema  
Ruby 0 stars, 21 forks, 0 issues, 0 pull requests, Updated 3 weeks ago

**People**

Top languages

- C#
- JavaScript
- Ruby
- Python
- HTML

**Most used topics**

- gbxml
- aec



# Acknowledgement

- This work was supported by the Standards Coordination Office Curricula Development Cooperative Agreement Program of National Institute of Standards and Technology
- This set of slides are built upon the sources from:
  - Stephen Roth, PE, LEED AP, Principal of Carmel Software Corporation and gbXML Administrator
  - University of Albany course slides

