Usability Thoughts

R. D. Welling to the MBSE Usability Team 12 July 2010

Usability uses

- Usability for development control
- Usability for introduction, instruction in MBSE

Usability Questions

- What are we using?
- How are we using it?
- How easy is it to use, whatever it is?
- How do we measure usability?

MBSE tool uses

- Which is the primary goal of a modeling tool, model validity or usability?
- If the goal is validity, then does usability mean the ease of building valid models?
- Does usability mean something different for experts and novices? Yes!
- The proof of the above is that very few who understand and use SysML!

Usability for development control

- Mechanical and electrical drawings are models
- Classic engineering status/control mechanism: drawing completion
- Quantity: Number of "approved" drawings
- Quality: Approval asks, "Is this drawing correct?"
- System models also must answer correctness questions

Model navigability

- Finding one's way around an existing or in-process model
- Navigating to determine
  - Model validity: Is the model syntactically correct?
  - Model completeness: How much work do I have left?

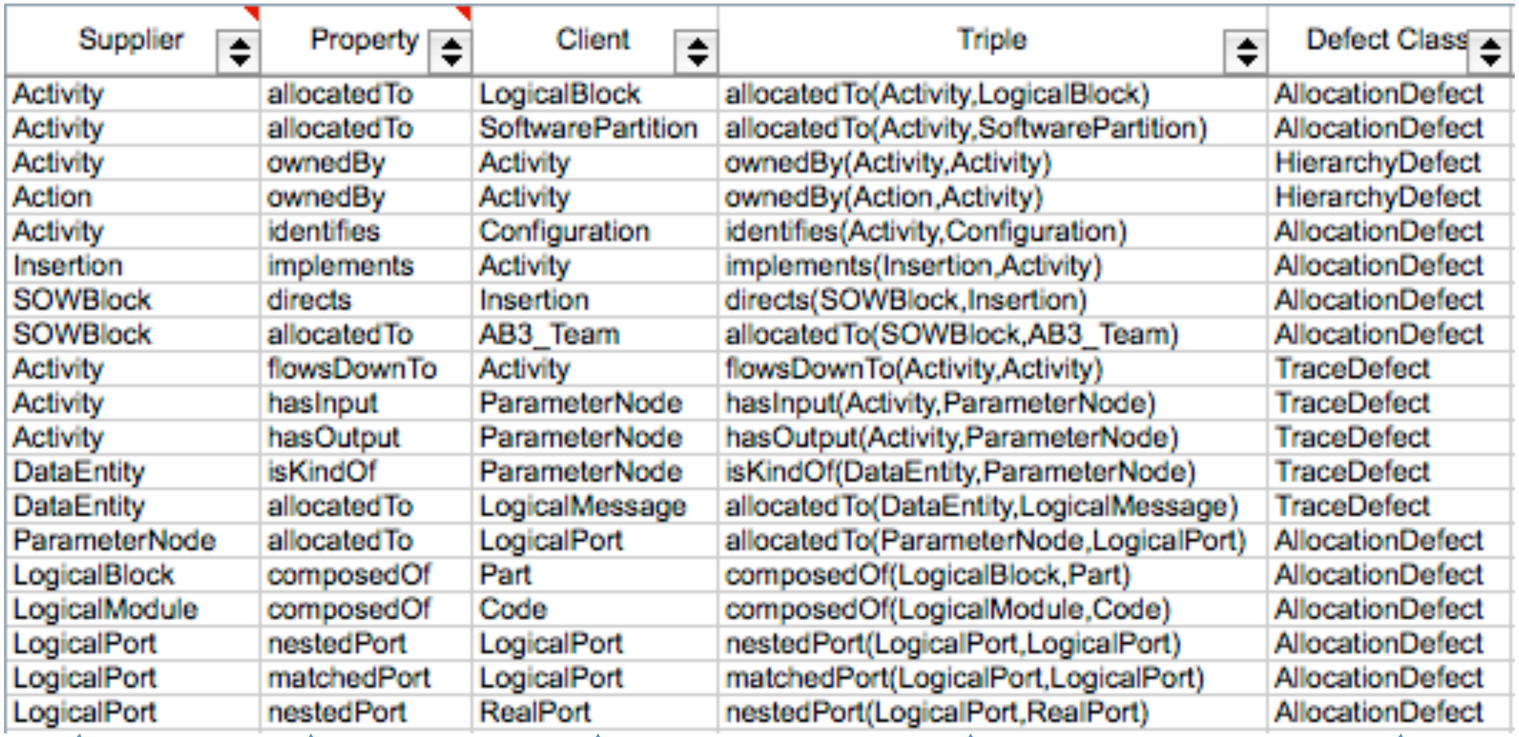
Example: Usability for control

- Sandy Friendenthal has talked about SysML-Lite as a possible path for improving usability
- “SysML-Lite” happened in 2006 mapping a SLATE tool schema to the SysML metamodel
- Diagrams used SLATE Visio API
- Briefings began with this question:
  “How do we gain and maintain control of the developing system?”

Usability for control

- SLATE/SysML-Lite used basic SysML constructs: Activities, “Logical” Blocks (almost everything in SLATE is a block)
- Quantity: Weekly parsing of the developing model
- Syntactical testing: Valid relations between object types
- Quality: Defect types to direct development

“SysML Lite” c. 2006 in SLATE



Supplier	Property	Client	Triple	Defect Class
Activity	allocatedTo	LogicalBlock	allocatedTo(Activity,LogicalBlock)	AllocationDefect
Activity	allocatedTo	SoftwarePartition	allocatedTo(Activity,SoftwarePartition)	AllocationDefect
Activity	ownedBy	Activity	ownedBy(Activity,Activity)	HierarchyDefect
Action	ownedBy	Activity	ownedBy(Action,Activity)	HierarchyDefect
Activity	identifies	Configuration	identifies(Activity,Configuration)	AllocationDefect
Insertion	implements	Activity	implements(Insertion,Activity)	AllocationDefect
SOWBlock	directs	Insertion	directs(SOWBlock,Insertion)	AllocationDefect
SOWBlock	allocatedTo	AB3_Team	allocatedTo(SOWBlock,AB3_Team)	AllocationDefect
Activity	flowsDownTo	Activity	flowsDownTo(Activity,Activity)	TraceDefect
Activity	hasInput	ParameterNode	hasInput(Activity,ParameterNode)	TraceDefect
Activity	hasOutput	ParameterNode	hasOutput(Activity,ParameterNode)	TraceDefect
DataEntity	isKindOf	ParameterNode	isKindOf(DataEntity,ParameterNode)	TraceDefect
DataEntity	allocatedTo	LogicalMessage	allocatedTo(DataEntity,LogicalMessage)	TraceDefect
ParameterNode	allocatedTo	LogicalPort	allocatedTo(ParameterNode,LogicalPort)	AllocationDefect
LogicalBlock	composedOf	Part	composedOf(LogicalBlock,Part)	AllocationDefect
LogicalModule	composedOf	Code	composedOf(LogicalModule,Code)	AllocationDefect
LogicalPort	nestedPort	LogicalPort	nestedPort(LogicalPort,LogicalPort)	AllocationDefect
LogicalPort	matchedPort	LogicalPort	matchedPort(LogicalPort,LogicalPort)	AllocationDefect
LogicalPort	nestedPort	RealPort	nestedPort(LogicalPort,RealPort)	AllocationDefect

SysML Object types

Associations (connectors?)

What object types get to be clients?

RDF-style triples implemented in Tcl

Control and feedback

Issues for original SysML-Lite

- Weekly parsing for status and direction (control) was almost agile
- The system model as a "principal artifact" was and is a foreign concept
- How do we get engineers to communicate thoughts in an abstract language outside their domain?
- Means of improving interpretation: trees, graphs, temperature charts

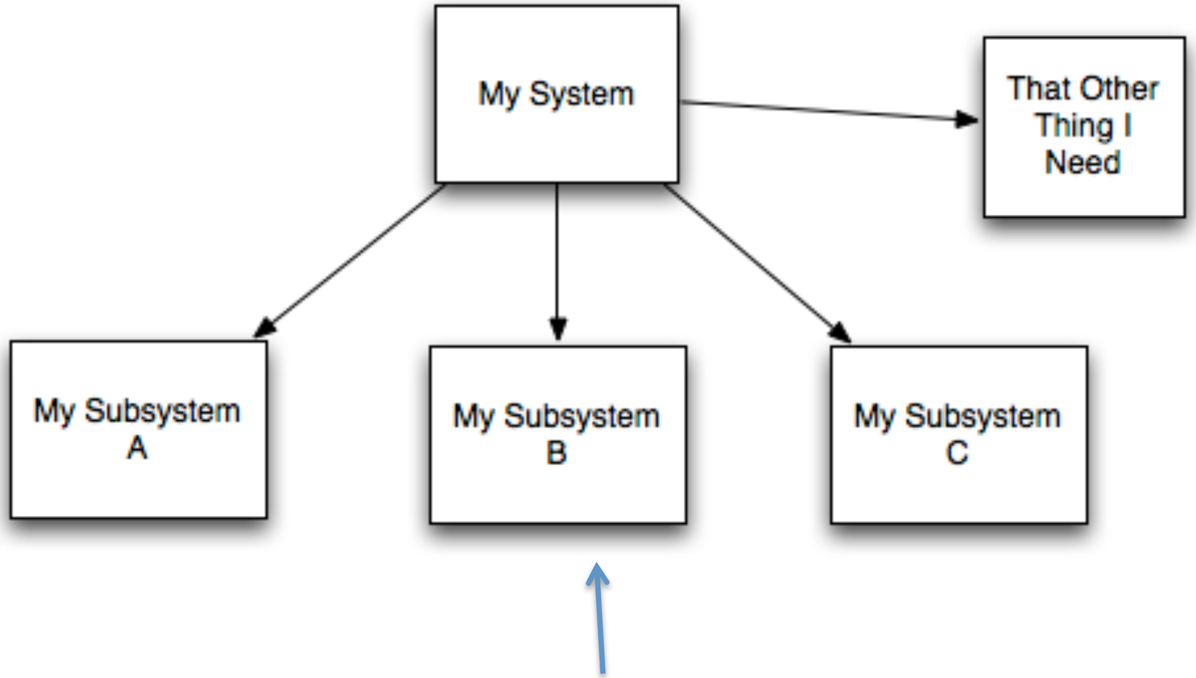
Sandy’s note (extract)

- Are there ways we can significantly reduce the time to develop a basic level of competency [for the new user who is just learning SysML]
- We have talked about defining a SysML Lite… the new SysML certification includes a basic feature set that could be a starter.
- …we are thinking of including only a subset of the diagrams in SysML Lite such as bdd, ibd, activity, requirements, and use case.

Engineers and models

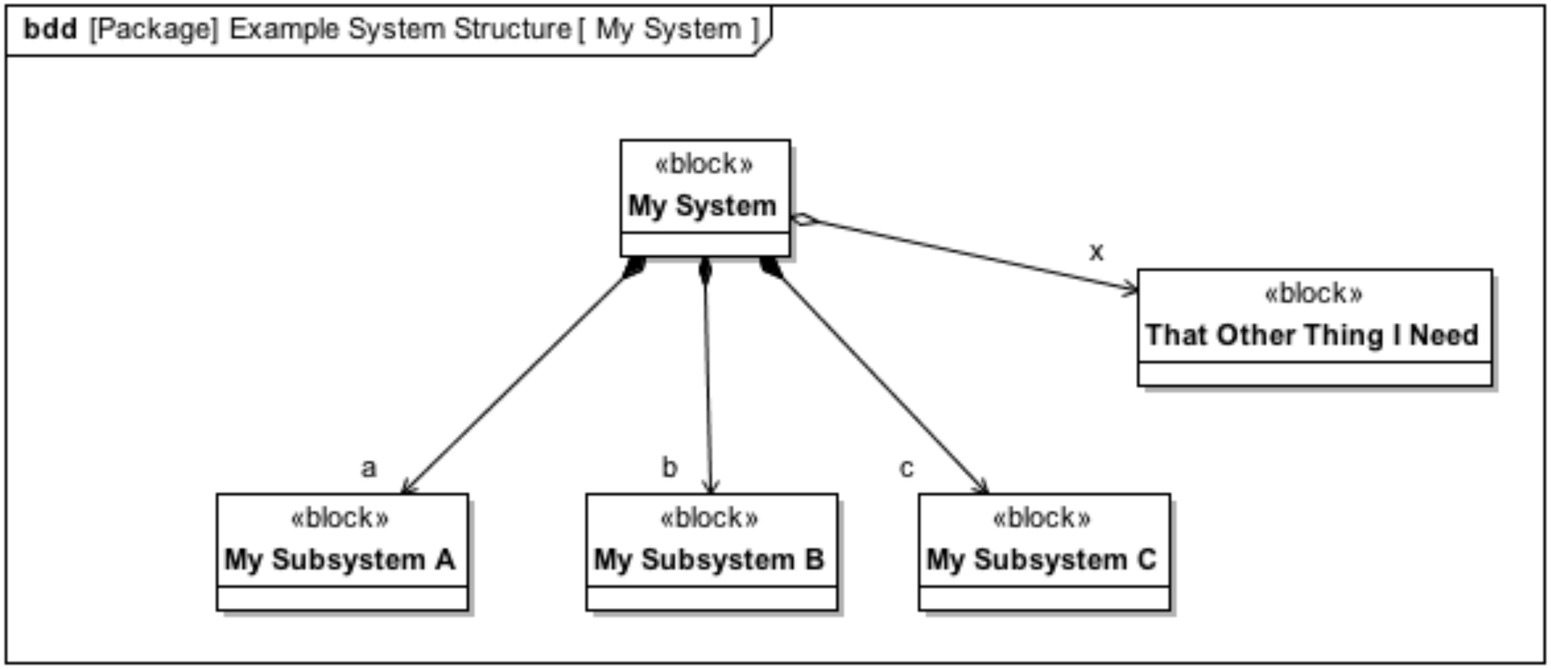
- Engineers are not familiar with the formalism of generalized systems
- Modelica, e.g., uses objects exhibiting specific behaviors (pumps, relays, etc.)
- VLSI tools, e.g., result in mechanical layouts
- Generalizing the many domain-specific concepts sets a gap in understanding
- Is MBSE usability concerned with closing the gap between specific and abstract knowledge?

White board model

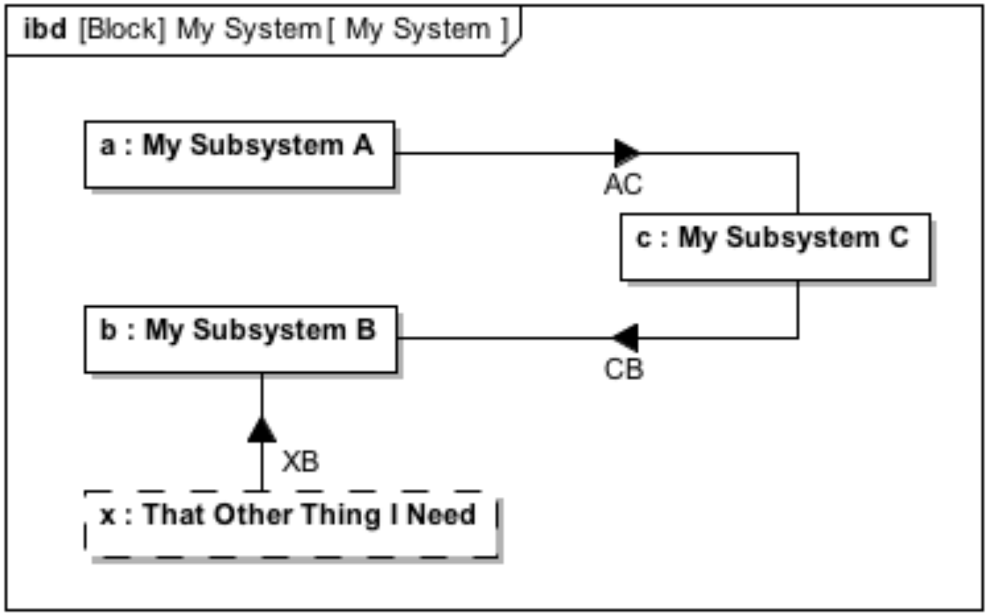


Naïve Modeling Domain = White Board Engineering (WBE)

Same system in SysML

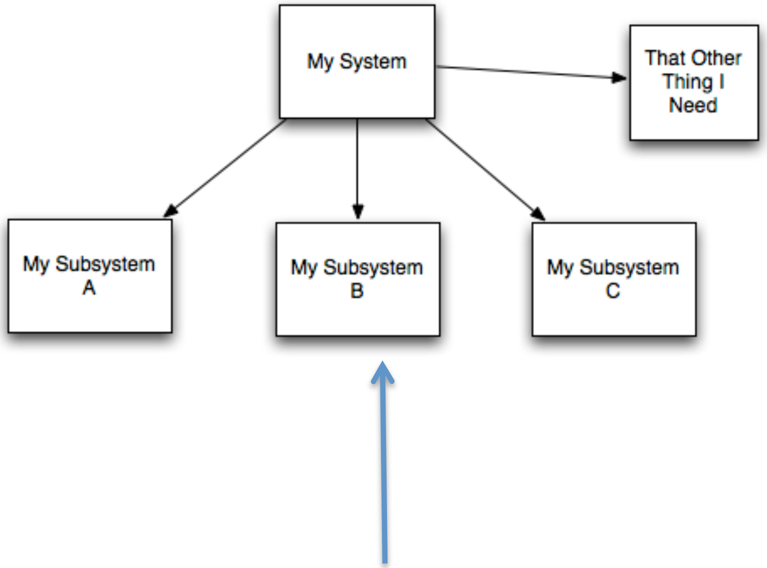


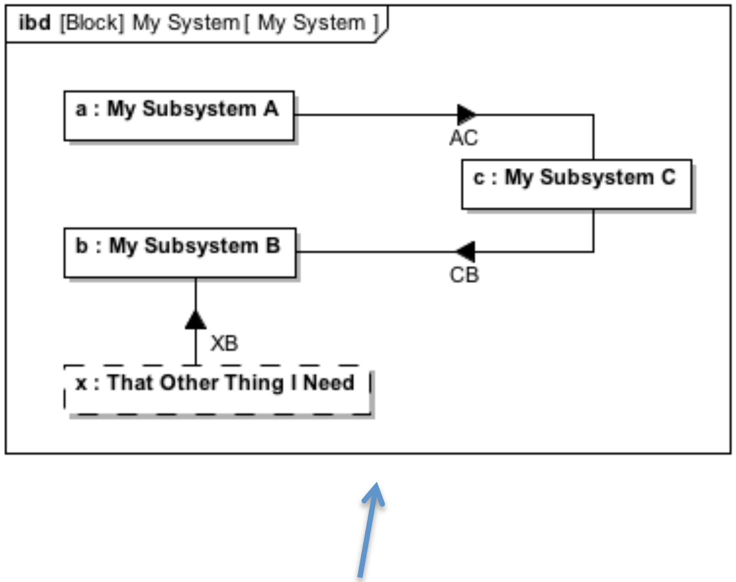
System has an internal arrangement


ibd [Block] My System [My System]
a : My Subsystem A
AC
c : My Subsystem C
b : My Subsystem B
CB
XB
x : That Other Thing I Need

Item flows? Reference properties? Yikes!

Levels of Abstraction




ibd [Block] My System [My System]
a : My Subsystem A
AC
c : My Subsystem C
b : My Subsystem B
CB
XB
x : That Other Thing I Need

- More intuitive
- Lower abstraction
- Limited information potential
- Domain specific (white board domain)

- Less intuitive
- Greater abstraction
- Rich information potential
- Domain independent (any system)

Transformations

- Making a tool usable amounts to changing the way that its authors are quite comfortable in using it.
- SysML is complex even for the experts.
- Non-expert users need only "SysML Lite"
- SysML Lite needs to conform to the SysML metamodel
- Enhancing usability is a matter data transformation: from the authors to the users.

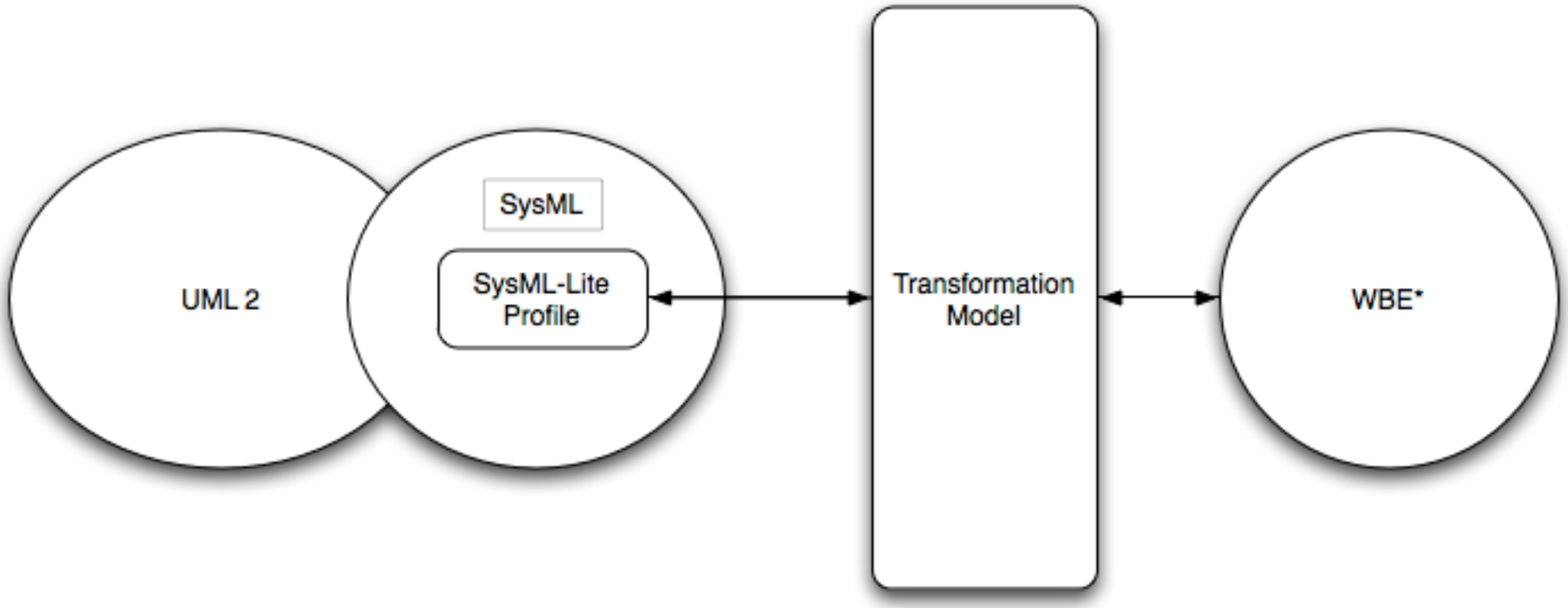
Usability for introduction, instruction in MBSE

- Method for new users to “wade” into MBSE
- “Naïve” or intuitive SysML (White Board Engineering)
- Automatic or aided (prompted) transformation to valid SysML

SysML-Lite as a DSL

- The UML2 extension mechanism allows for profiles of domain specific language (DSL)
- SysML is an extension of UML to avoid language proliferation
- A SysML-Lite profile would provide a controlled method for defining
  - Valid SysML-Lite
  - The transformation between SysML and SysML-Lite

SysML-Lite Profile



After J. Bankston, A. Shah, ISYE 8813 *Domain Specific Modeling of an ECAD System*, 2006

*White Board Engineering

RD Welling 7/5/10