



MBSE Usability Framework Structure and Context

MBSE Usability Group Online Meeting
December 7, 2010



Outline

- Recent History
- Context
 - A Usage Event
 - The Spaces of Usability
 - Framework: Minimalist View
 - Levels of Usage
 - Services Provided by Levels
 - Framework: An Expansive View
- Measures of Usage
- Preparing for International Workshop
 - Objectives for Workshop
 - Preparation



Recent History

- Lyells' Starting Point
- Cole's Extension



Lyells' Example:

Usability By Diagram Type, Concepts, Tasks,

	bdd	ibd	par	sd	act	pkg	Defn vs Usage	Model Mgt
Ease of Learning								
Effeciency of Use					why are operations in bdd not available as activities			
Memorability			Ref mtl must be close					
Error Frequency Severity	Lower level bdd's get reflected in upper level bdd							
Satisfaction								



Cole's Example:

Usability Matrix

	SysML			Integration Tools		
	BDD	IBD	Par	DB	Transformation	Spread - sheet
Establish Resources						
Ease of Learning	Come packaged with notions of inheritance and abstraction that may require education		Depends on learning working style – can be both cumbersome (lots of blocks) or elucidating (which parameters interact)			
Efficiency of Use Routine	Levels of allocation easy to show in single diagram for generic components; capture values for standard practice on margins, etc.	Model the analysis environment to see where updates to values will come from	Can show “tree math” for different levels of hierarchy; many SE’s can be frustrated with “I just want a simple sum!”			Quick to build; usability fades as length of time using same spreadsheet grows
Efficiency of Use Non-Routine		Easy to show how components interact – some elements (structural loading, mechanical power transfer) not naturally represented as “flows”	Specify sensitivities of one subsystem to another’s violation of allocation constraints; clear but not as simple as equation routines; hard to follow detailed calculations			Simple to develop, but often gets opaque once analysis is finished
Error Frequency					Low once debugged	High; hard to track pedigree



Cole's Example (cont. 1):

Usability Matrix

	SysML			Integration Tools		
	Activity	IBD	Par	DB	Transformation	Spread - sheet
Track Resources						
Ease of Learning	Easily understood by Sys Engineers; may take a little extra time to master		Depends on learning working style – can be both cumbersome (lots of blocks) or elucidating (which parameters interact)		Requires a different mindset for use; is in common with XSL	
Efficiency of Use Routine	Model development environment to see where information will be developed and with which tools	Model the analysis environment to see where updates to values will come from	Can show “tree math” for different levels of hierarchy; works better in abstract than concrete; many SE's can be frustrated with “I just want a simple sum!”	Concept is familiar; specifics of data entry / harvesting may be difficult	How does one provide a “tagged” handle of a property over to a disciplinarian for connection to his or her development effort?	Difficult to update, very manual process
Efficiency of Use Non-Routine			Previously captured sensitivity equations may be useful but hard to find / recall			
Error Frequency					Low once debugged	High; hard to track pedigree



Cole's Example (cont. 2):

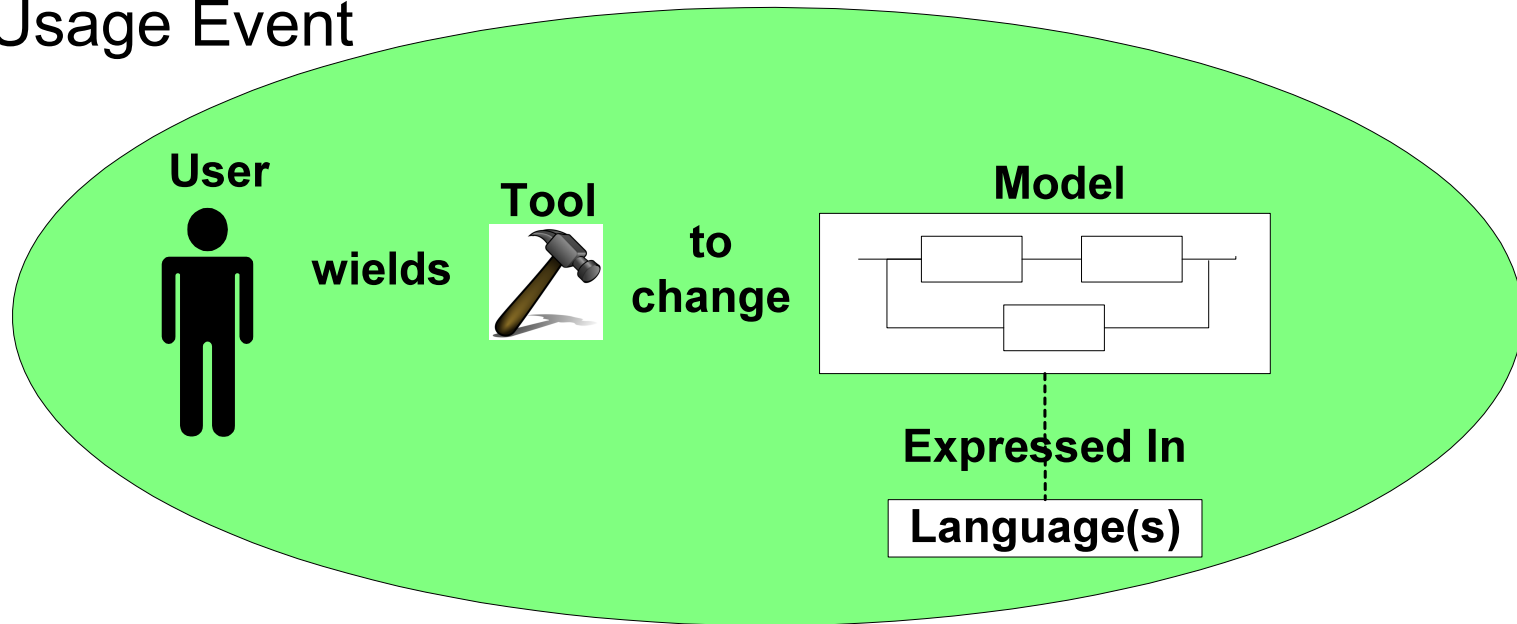
Usability Matrix

	SysML			Integration Tools		
	BDD	IBD	Par	DB	Transformation	Spread - sheet
Barter Resources						
Ease of Learning			Depends on learning working style – can be both cumbersome (lots of blocks) or elucidating (which parameters interact)			
Efficiency of Use Routine		Show the paths for redirected surplus resources from one subsystem to others; examine current usage	See parts of system dynamical models laid out, see where to tune parameters; hard to see the integrated system of equations at once across many blocks and values		Match resource use predicted by model into the same terms as the operating reports; generate reports and queries on current use	Nearly impossible unless the spreadsheet is in current use to understand how the system is modelled
Efficiency of Use Non-Routine		Identify damaged / reconfigured components; hard to mark up IBD				
Error Frequency					Low once debugged	High; hard to track pedigree

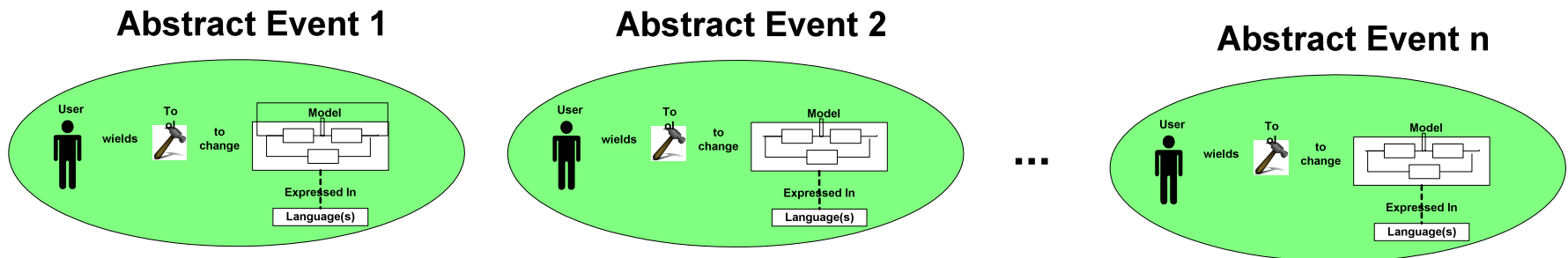


Usage

□ Usage Event

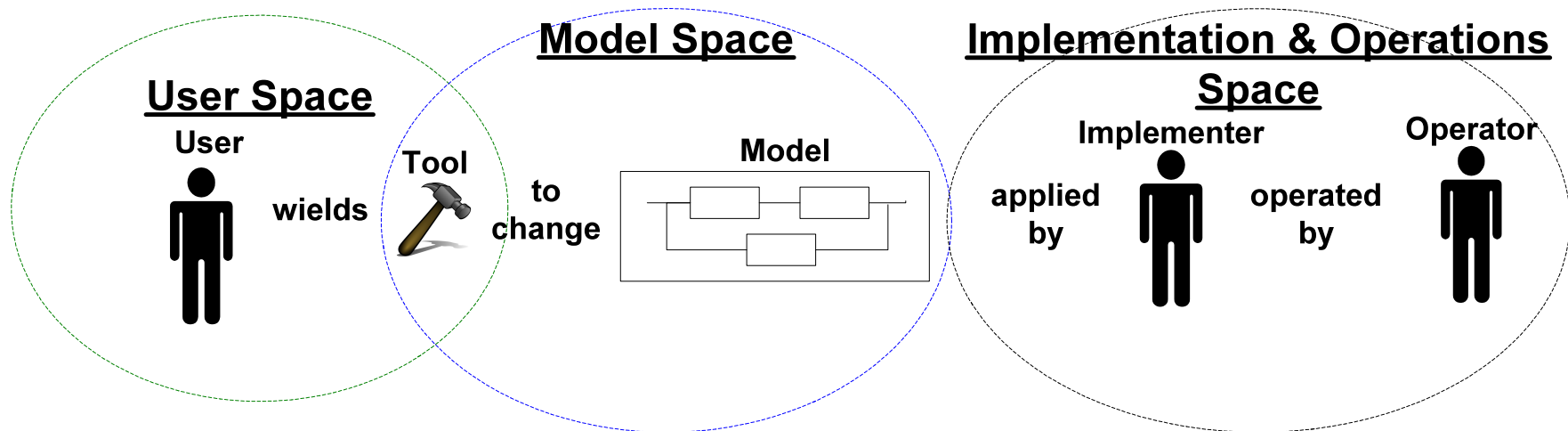


□ Usage Process



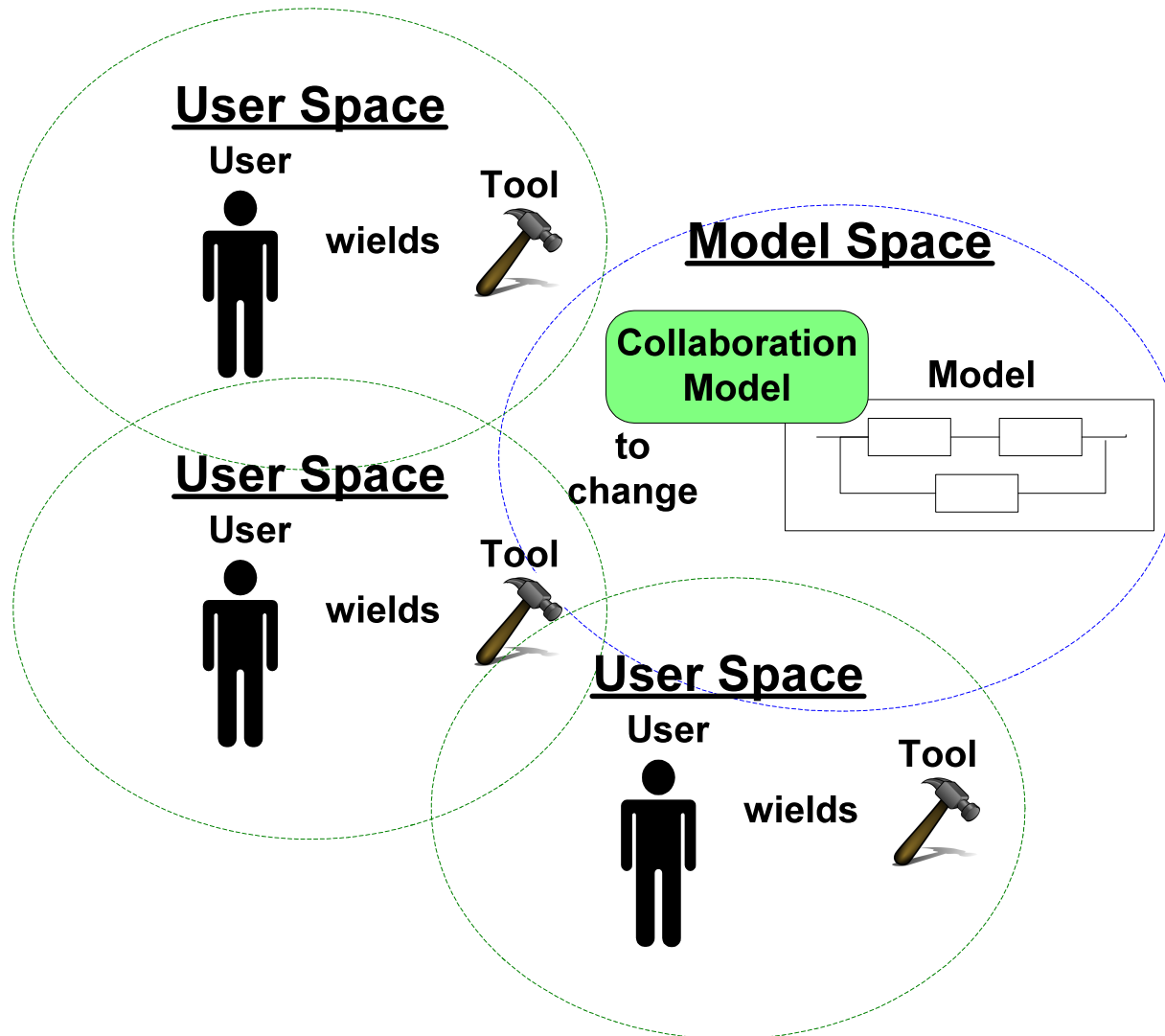


The Spaces of Usability





Collaboration





A Minimalist View

Usability Categories	Tool	Model	Process
Ease of Learning			
Efficiency of Use	MagicDraw	Process Flow	P&ID Design
Memorability			
Error Propensity			
Satisfaction			



Levels of Usage

- ❑ Lifecycle Processes
- ❑ Systems Engineering Processes
 - V Model
 - Hybrid Model:
 - Requirements Development
 - Architecture
 - Construction, Integration, Test
 - Bjorn's Example: Resource Management
 - Establish / Allocate Resources
 - Track Resources v. Allocations
 - Barter Resources
- ❑ Model Building Level
 - Definition & Connection
 - White Board Engineering
- ❑ Modeling Tasks (ex: Construct a block diagram)
- ❑ Elementary Actions (ex: Place a block in a diagram)



Services Provided by Levels

- Each Level provides services to the level above
- Bjorn's Example: Resource Management
 - Establish / Allocate Resources
 - Track Resources v. Allocations
 - Barter Resources

- These are services that are desired from the modeling level



Usability Framework:

A More Expansive View (Fragment)

	Collaboration Processes			Model Meta-Data			Collaboration Tools		
	Proj. Planng	Config. Control	Review & Comment	Model Files	Change Flags	Comments	JAZZ	Shared Calendar	Situation Awareness
	SE Process			Model Data in Views			SE Process Tools		
	Reqs. Devel.	Architectre	Resource Mgmt.	SysML BDD	SysML IBD	SysML Para	Workflow	CM (ex. DOORS)	
	Model Building Strategies			Model Fragments			Modeling Tools		
	White Board Eng.	Define & Connect	ReFactrng	Marker / Whitebrd	Partial Diagram	Class Tree	Ent. Arch.	MagicDraw	Excel
	Tasks & Task Primitives			Language Elements & Aggregates			Task Affordances & Primitives		
	Specialization	Generalization	Apply Arch. Pattern	Blocks & Connectors	Model Fragments	Patterns	Place an Element	Make a Connection	Define a Pattern
Usability Categories									
Ease of Learning									
Efficiency of Use									
Memorability									
Error Propensity									
Severity									
Frequencey									
Satisfaction									



Measures of Usage

- Categories
 - Ease of Learning
 - Efficiency of Use
 - Memorability
 - Error Propensity
 - Frequency
 - Severity
 - Satisfaction
- Discussion



Preparing for International Workshop

- Objectives
 - Identify key problems (Populate the matrix)
 - Establish Priorities
 - Engage Larger Community by:
 - Get people committed to solving particular problems
 - Cloud Sourcing?
- Preparation for IW
 - ½ Day for our meetings
 - Presentations
 - Tentative Key Problems & Discussion
 - Tentative Priorities & Discussion
 - Discuss Path Forward
 - Volunteers



Summary & Discussion

- ❑ Recent History
- ❑ Context
 - A Usage Event
 - The Spaces of Usability
 - Framework: Minimalist View
 - Levels of Usage
 - Services Provided by Levels
 - Framework: An Expansive View
- ❑ Measures of Usage
- ❑ Preparing for International Workshop
 - Objectives for Workshop
 - Preparation