



Hierarchical Model Composition

Introduction and Controversy

Lucian Smith

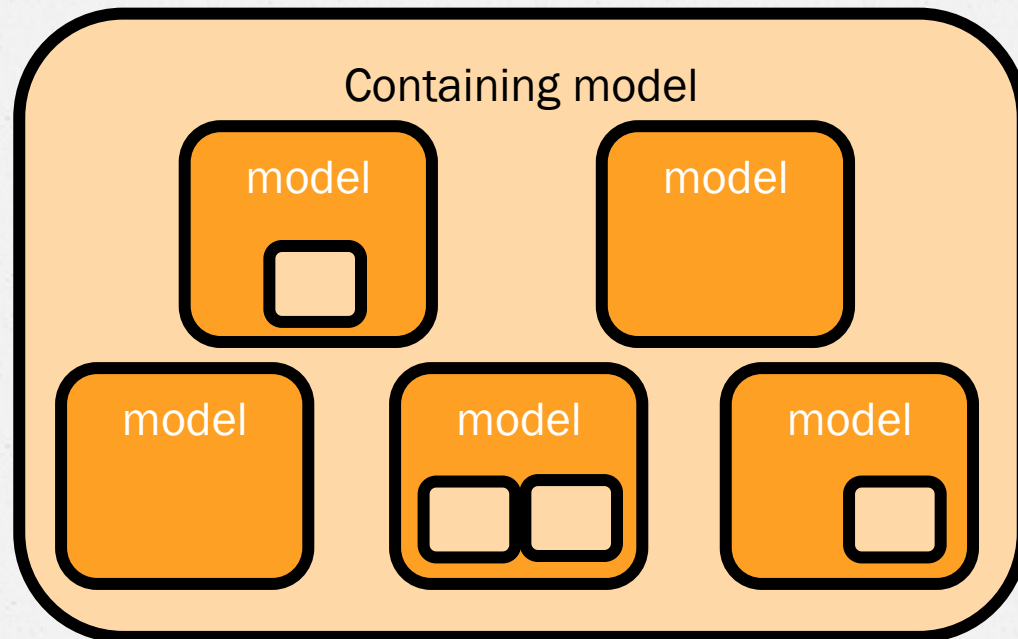
HARMONY 2011

What is model composition?



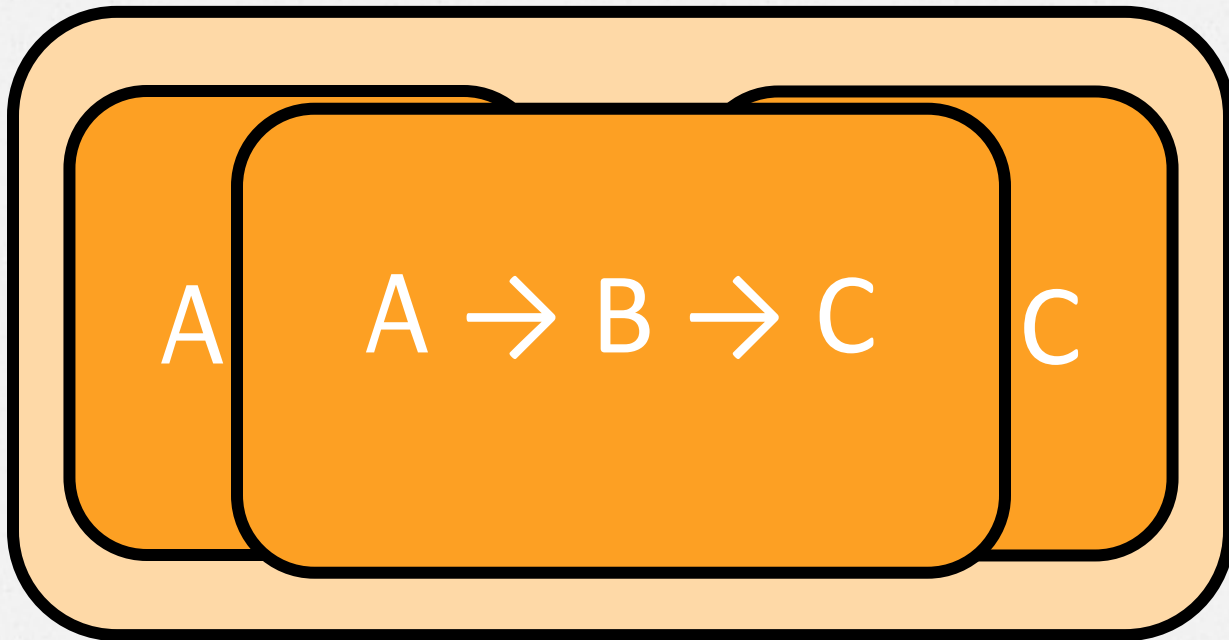
Simplest: Aggregation

- o Everything in the same bag,
- o Doing its own thing

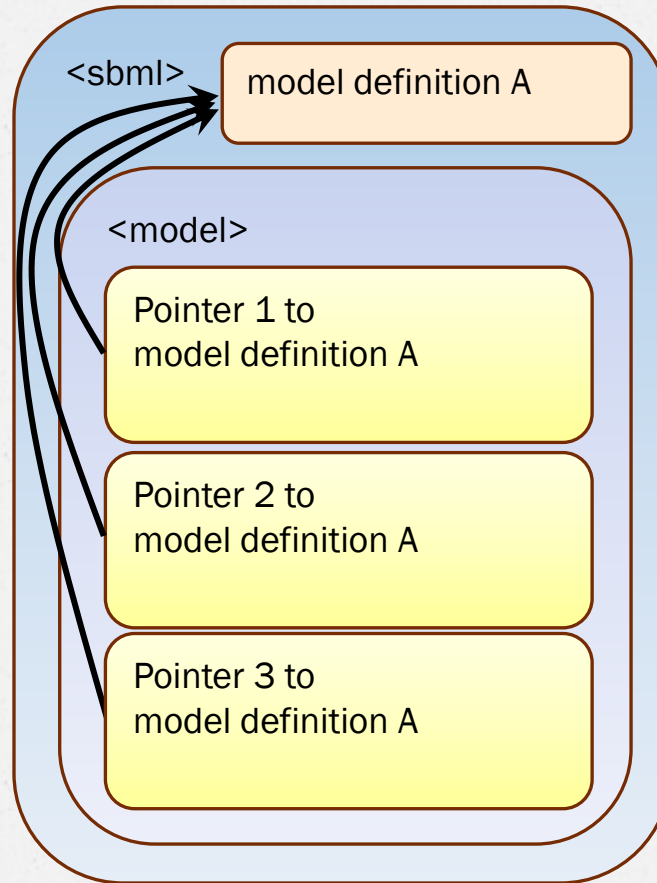


More helpful: connected

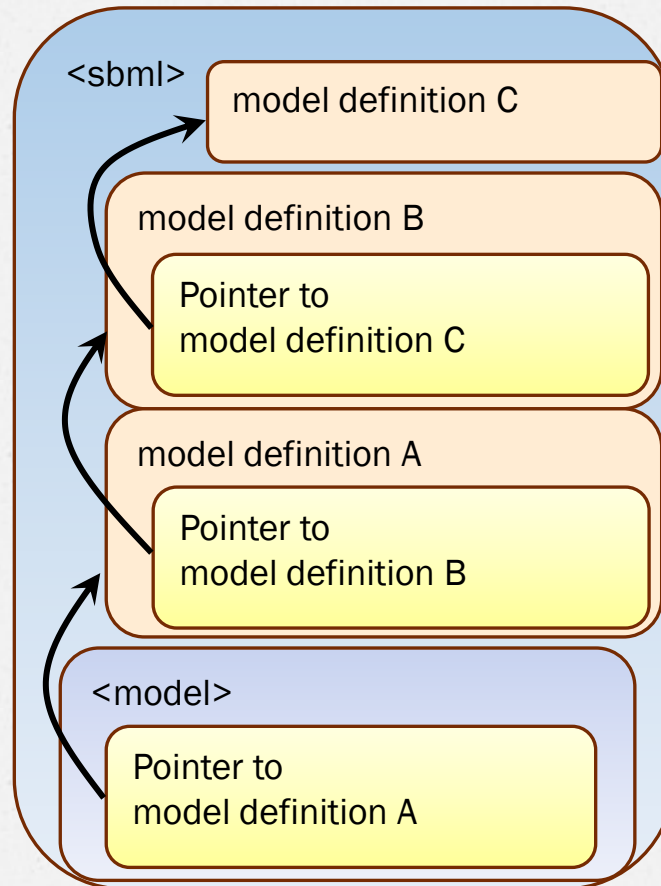
- o Rule: “This bit of model 1 is the same as this other bit of model 2.”



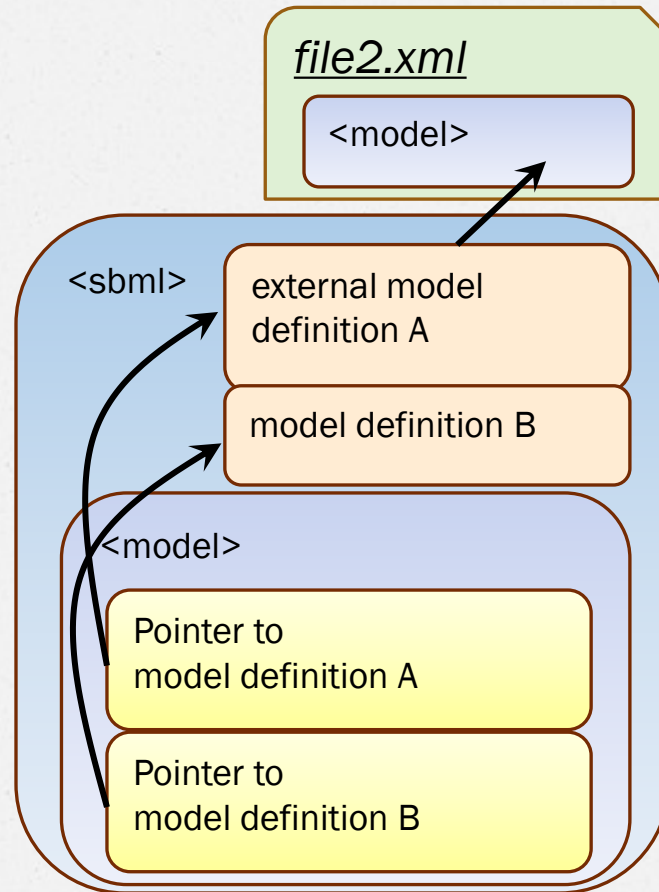
Examples of hierarchy



Examples of hierarchy



Examples of hierarchy



And here's where we sweep a lot of details under the rug...

- o You can link any type of thing as long as it makes syntactic sense in the end.
- o You can delete things from submodels you don't want in the containing model
- o Links actually need something in the containing model to link to:
 - o Rule1: B replaces mod1.B ← Effectively links mod1.B to mod2.B
 - o Rule2: B replaces mod2.B

Progress

- o Finished:
 - o Implemented reading, writing, getting, and setting of most elements in a libsbml package
 - o Light testing
- o To-do
 - o Resolve final issues with spec itself.
 - o Implement more semantic API
 - o Implement more error checking
 - o Implement conversion to/from Antimony

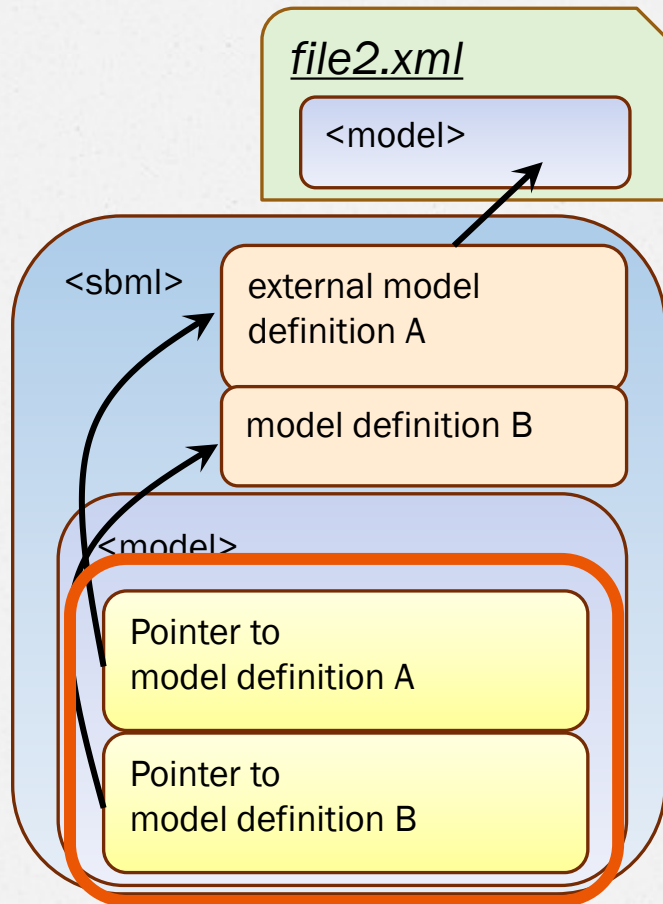
Progress

- o Finished:
 - o Implemented reading, writing, getting, and setting of most elements in a libsbml package
 - o Light testing
- o To-do
 - o **Resolve final issues with spec itself.**
 - o Implement more semantic API
 - o Implement more error checking
 - o Implement conversion to/from Antimony

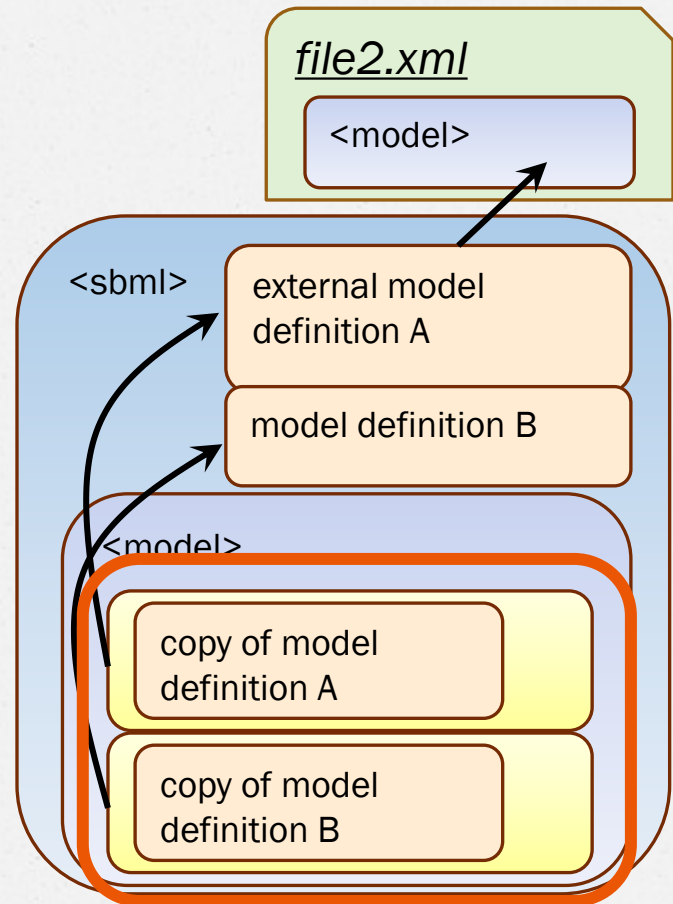
Remaining Issues

- o Is a 'comp' model a recipe or a cake?
- o Should models, and not documents, define their submodels?
- o How necessary is xpath to create references?
- o What more do we need to define 'port contracts'?

Is a 'comp' model a recipe or a cake?

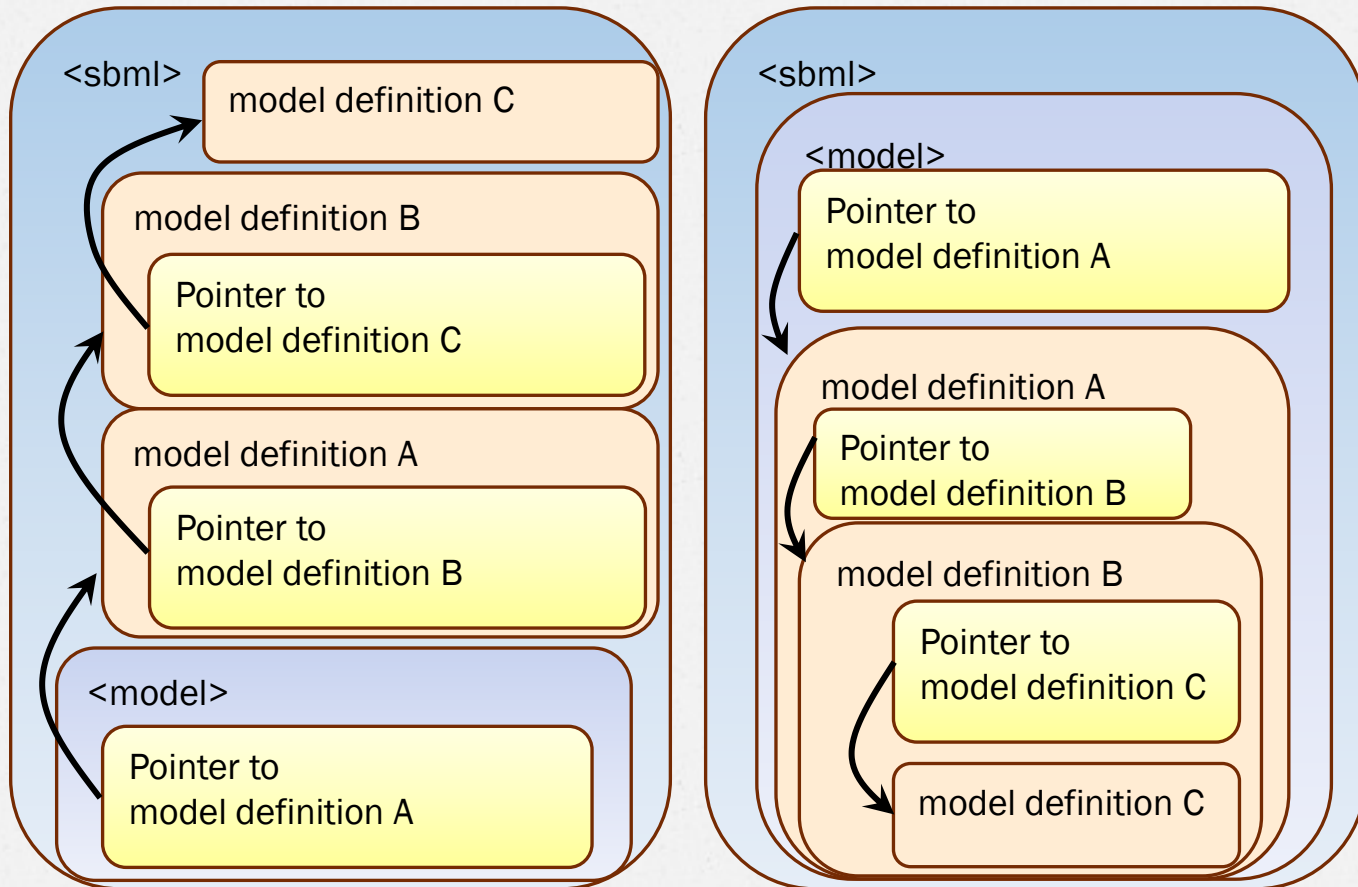


Rule



Full Model

Should models, and not documents, define their submodels?



Analogy:

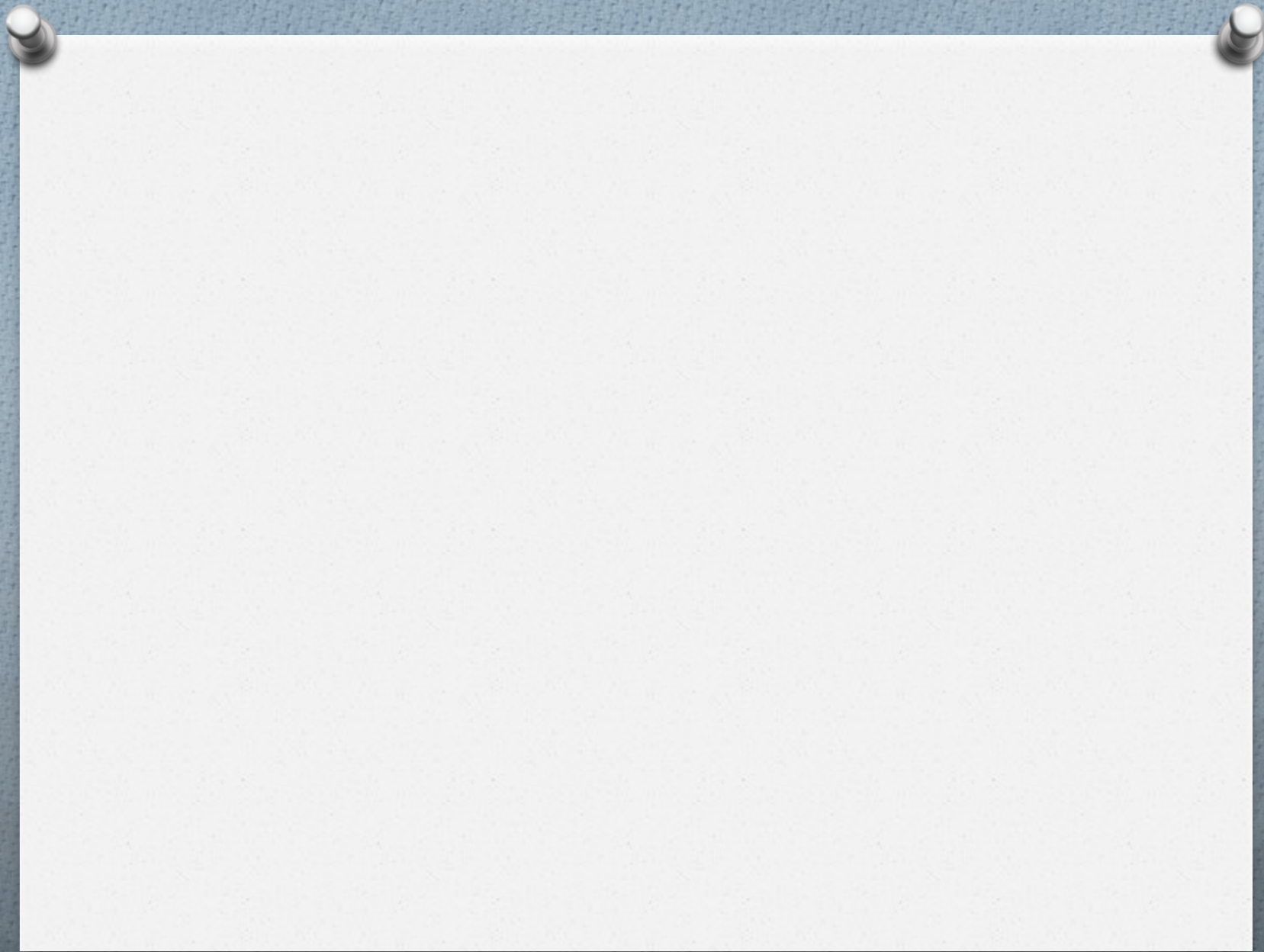
- o Kinetic laws can use locally-defined parameters and globally-defined parameters
- o Should models be able to use locally-defined submodels and globally-defined submodels? Just one or the other?
- o Complication: locally-defined submodels could potentially define their own locally-defined submodels. Should those have access to their parent's submodels?
- o Current proposal: globally-defined submodels only

xpath: completeness vs. complicated implementation

- o Must use xpath if an element of a model:
 - o Has no id
 - o Has no metald
 - o Is in a model you can't copy
 - o Is in a model you can't convince the owners/curators to change
- o How often does that happen?
- o Proposal 1: delay implementation; see if problem arises.
- o Proposal 2: only use subset of full xpath.

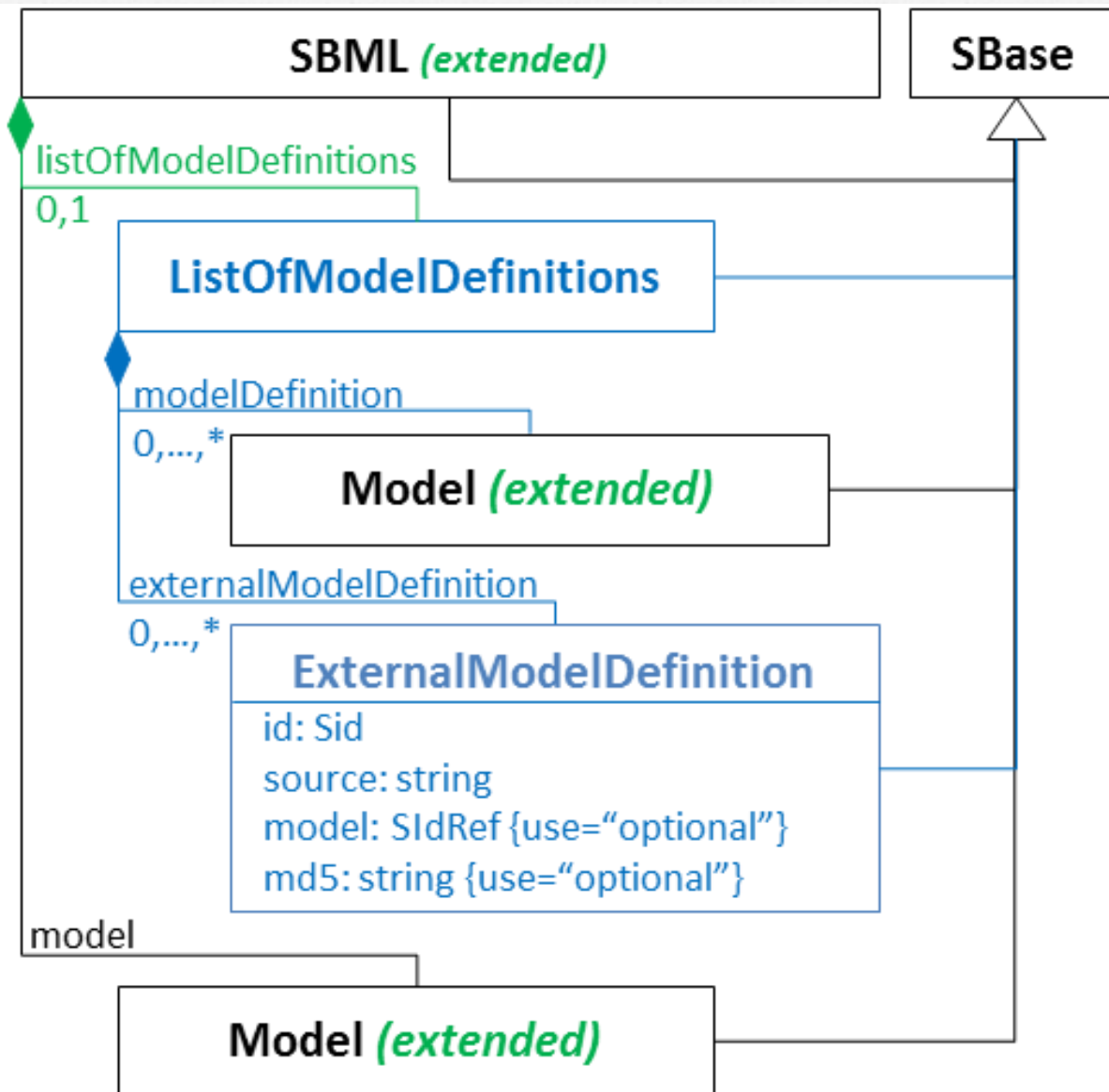
Black box modeling: port contracts

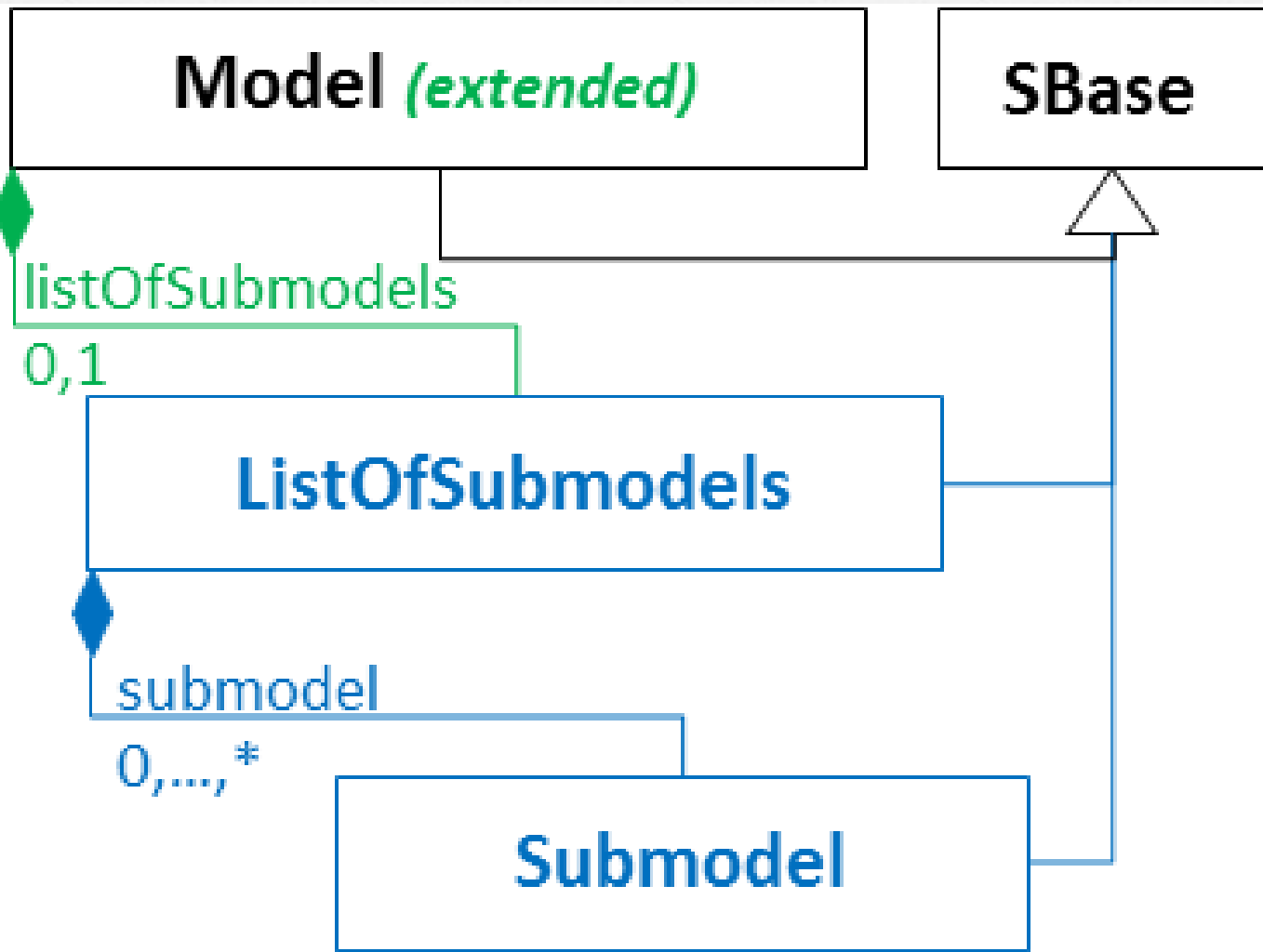
- o Currently, elements can be flagged as ports, but nothing more
- o What more is needed?

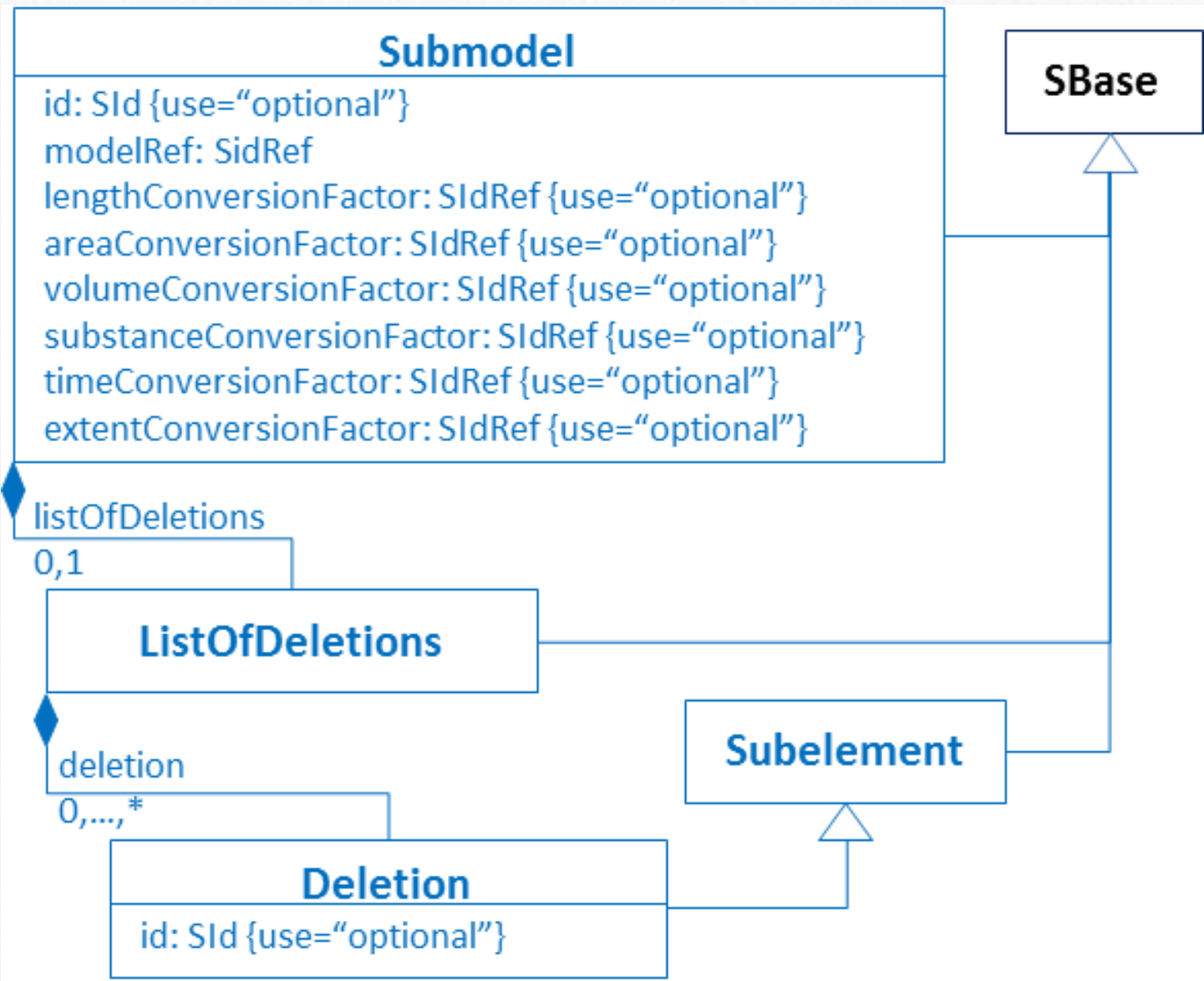


Remaining Issues: conclusion!

- o Is a 'comp' model a recipe or a cake?
 - o A recipe!
- o Should models, and not documents, define their submodels?
 - o No
- o How necessary is xpath to create references?
 - o Leave for comp I2 so we can get I1 implementations faster
- o What more do we need to define 'port contracts'?
 - o Either nothing at all, or <listofportcontracts> to act as 'playground' for future groups.







Subelement

submodel: SIdRef

port: PidRef {use="optional"}

idRef: SIdRef {use="optional"}

xpath:element: string {use="optional"}

deletion: SIdRef {use="optional"}

SBase

subelement

0,1

Subelement



SBase *(extended)*

portid: PId {use="optional"}

listOfReplacedElements

0,1

ListOfReplacedElements

replacedElement

0,...,*

ReplacedElement

identical: boolean {use="optional"}

conversionFactor: SIdRef {use="optional"}

Subelement

