System Analysis
Requirements for SysML 2

SysML2WG Meeting, OMG Technical Meeting
Cambridge MA, Sep 24, 2015

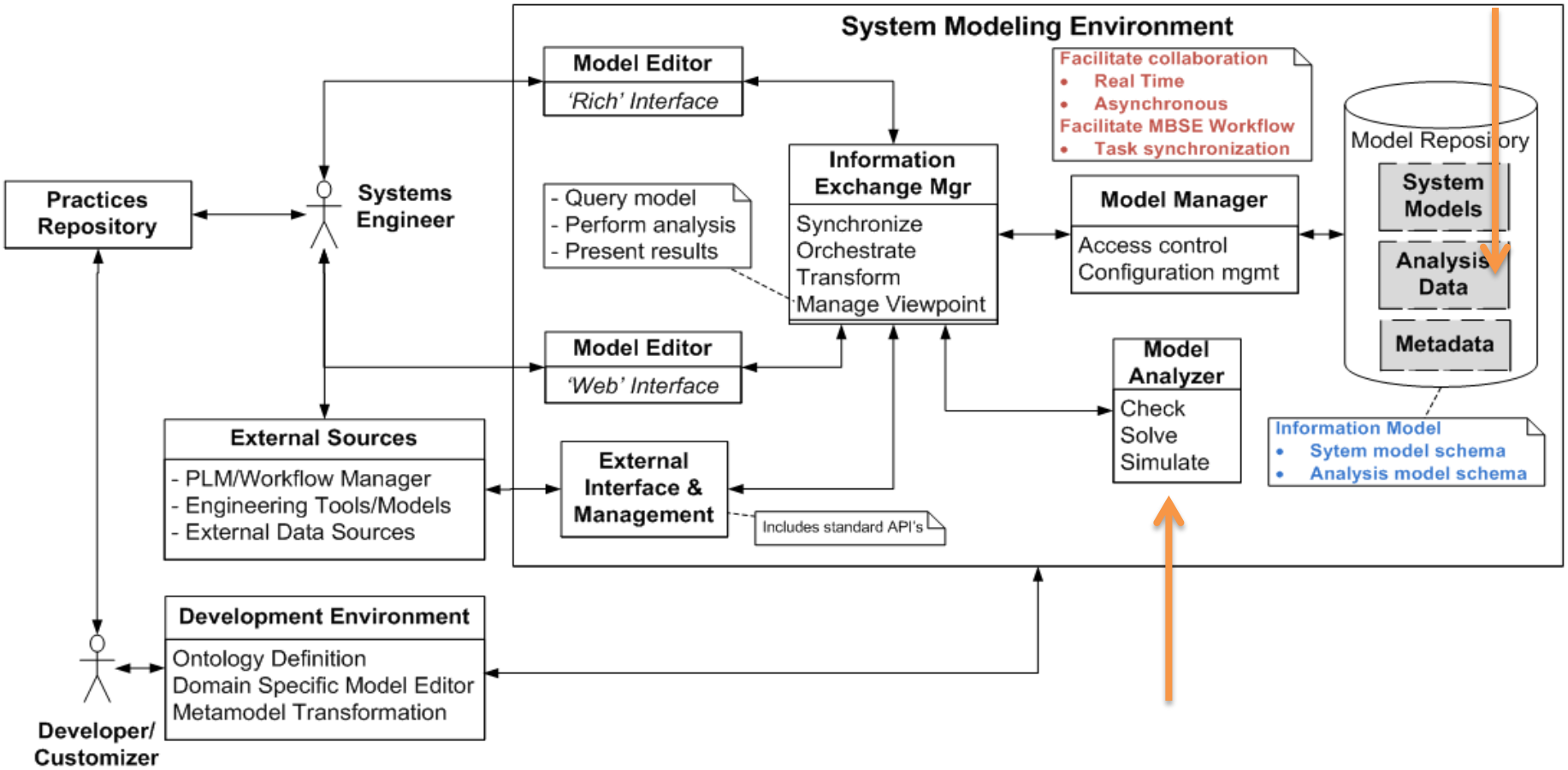
Manas Bajaj, PhD
Co-Founder & Chief Systems Officer
manas@intercax.com
www.intercax.com

Starting Point

Evolving SysML and the System Modeling Environment to Support MBSE
Draft (February 1, 2015)
S. Friedenthal/R. Burkhart

The **next-generation modeling language** must include **precise semantics** that avoid ambiguity and enable a concise representation of the concepts. SysML currently leverages the UML metamodel for much of its semantic foundations. The language must be based on a well-specified logical formalism that can leverage the model for a broad range of analysis and model checking. This includes the **ability to validate that the model is correct and consistent, and the ability to answer questions such as the impact of a requirement or design change, or the assessment of how a failure could propagate through a system**. The language and tools must also **integrate with a diverse range of equation solvers and execution environments. The language must also be able to represent, relate, and present quantitative data sets that can vary with time and space, and have probability distributions**.

System Modeling Environment



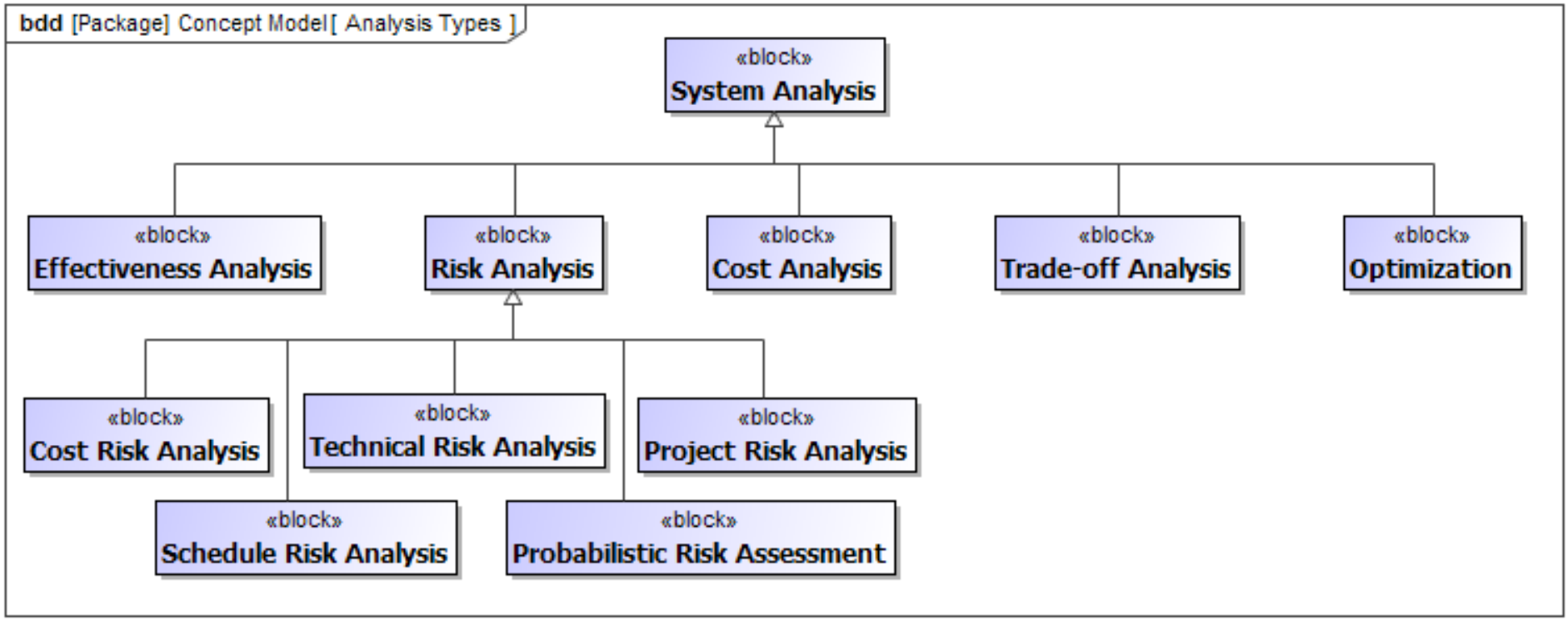
Goals

- Analysis must be seamless integrated with system development and operation
- Support various types of analyses and execution tools
- Manage analysis models and relate results to decisions
- Improved user interaction to define/generate, execute, archive analysis models (analysis lifecycle)

What is Analysis?

- Systematic investigation of a real or planned system to (a) compare and select candidate system architectures, or (b) determine causes & resolutions of failures / exceptions
  - SEBoK - http://goo.gl/RCtAKt
  - NASA SE Handbook (2007) - http://goo.gl/iVBVES
- Examples (design, manufacturing, operation)
  - Compare the mileage/cost for a set of car designs
  - What is the design impact of a requirement change?
  - Optimize manufacturing process to max yield/cost
  - What parts may have caused a function to fail?

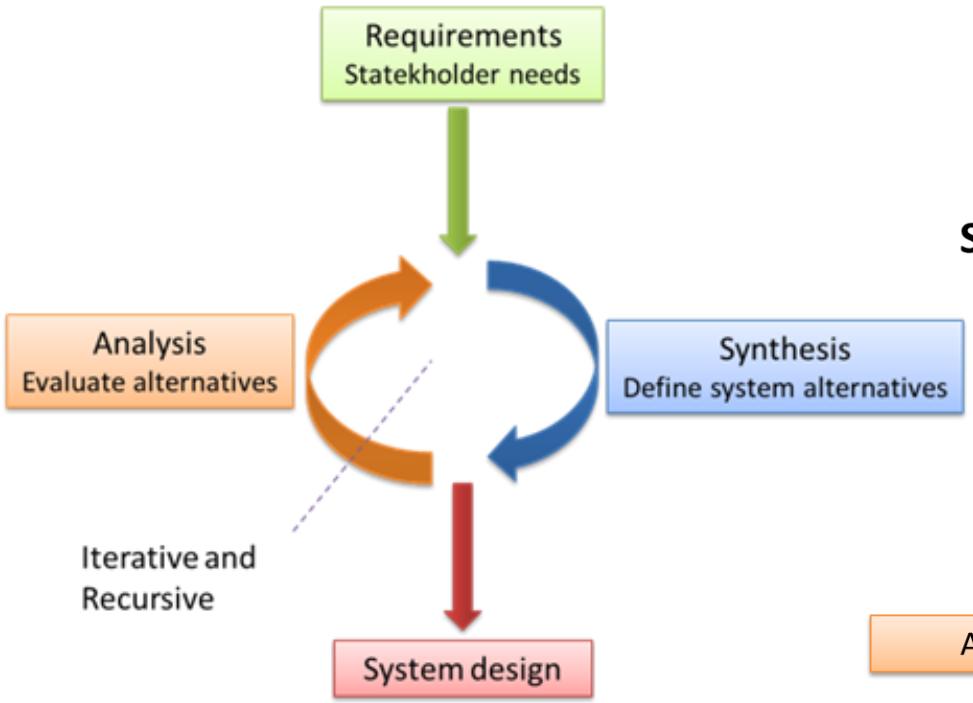
Types of Analyses



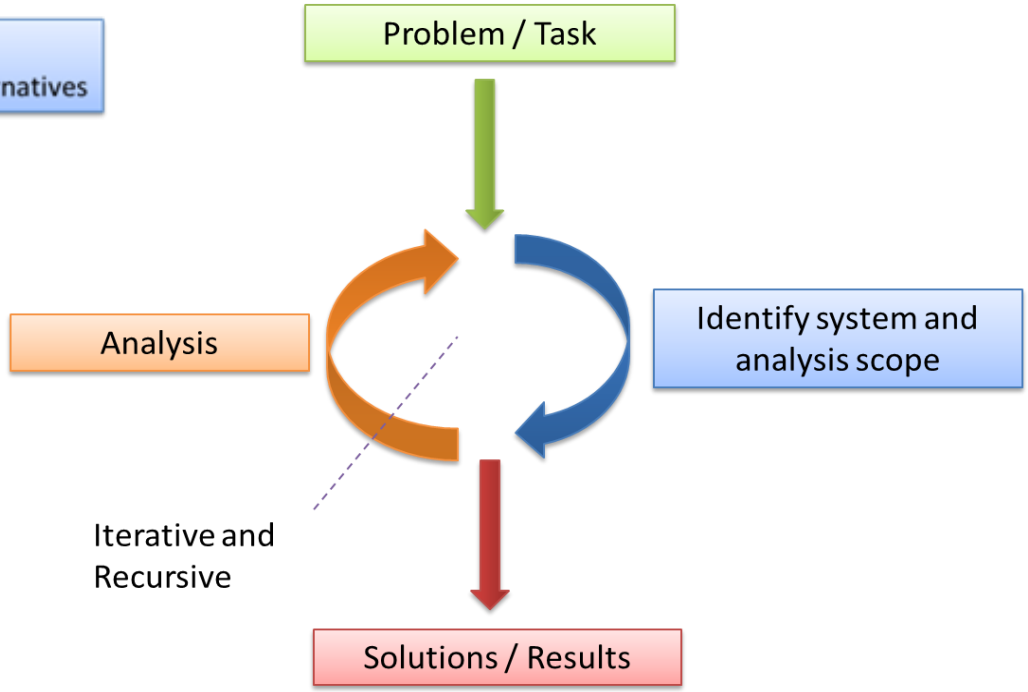
- **Quantitative Analyses**, e.g. computing MoEs
- **Qualitative Analyses**, e.g. If I change (increase) X, will Y change (increase/decrease)?

Analysis during system development and operations

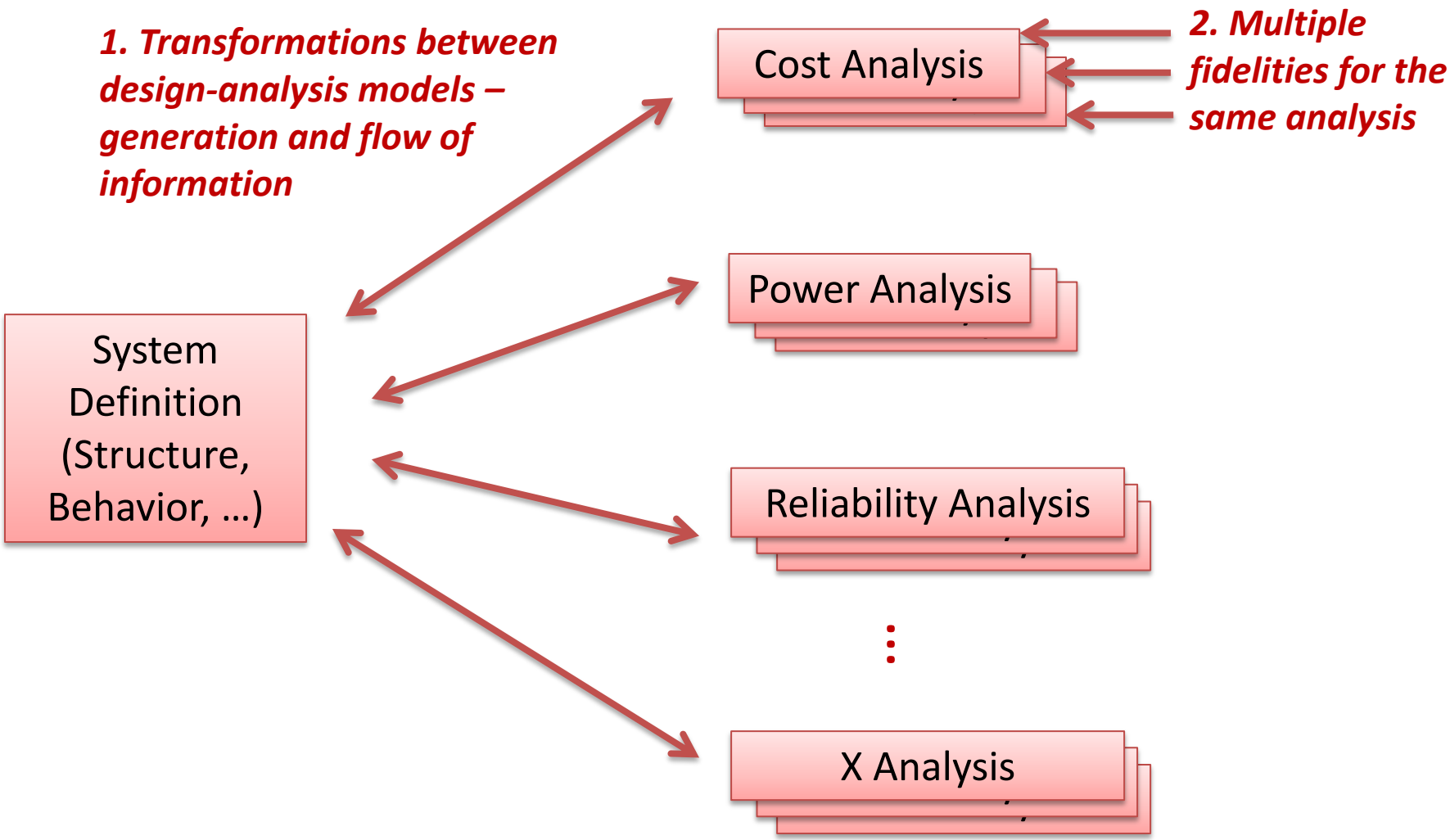
System Design / Development Phase



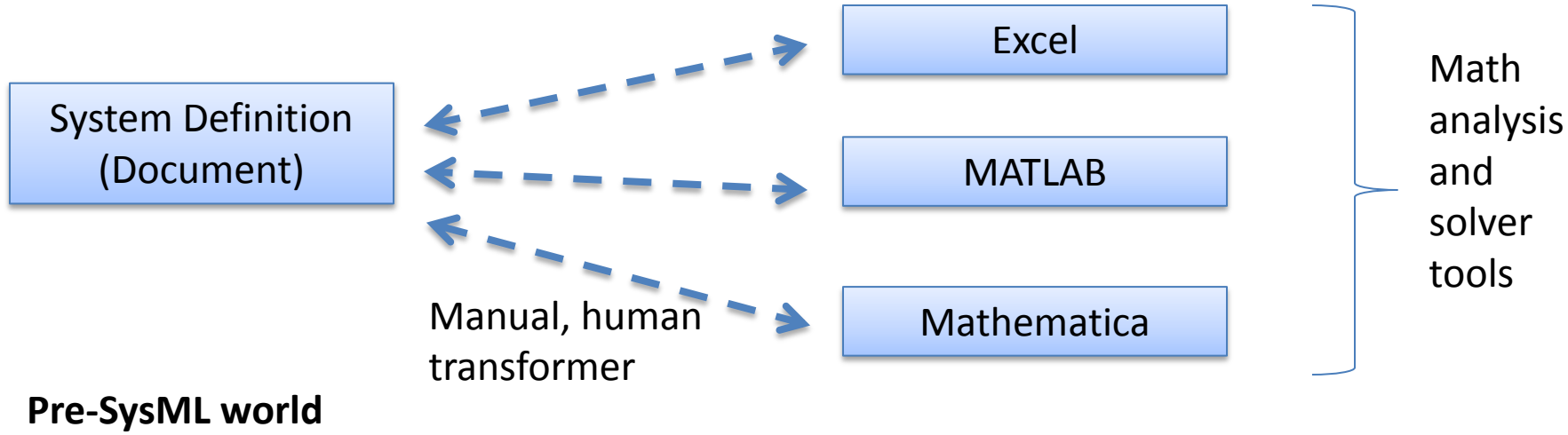
System Operation / Maintenance Phase

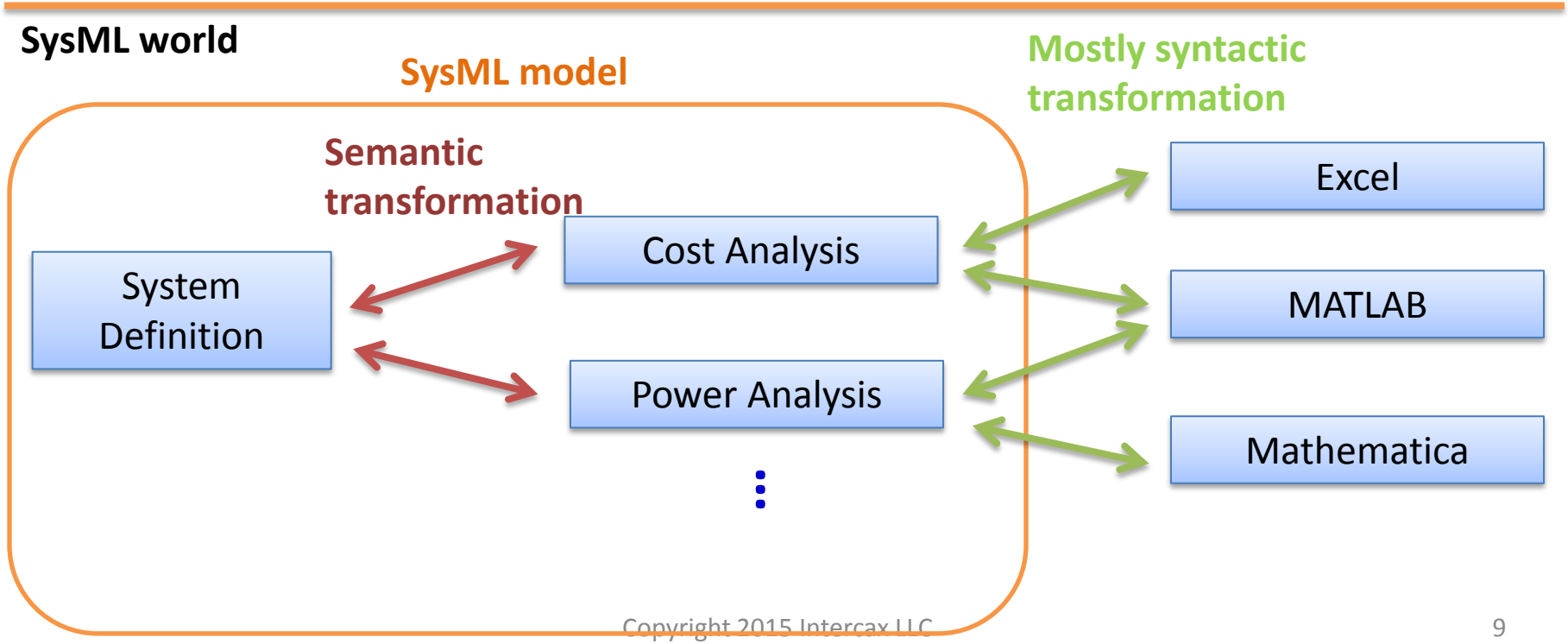


Design and Analysis are separate models



Where does the system analysis model live?





SYSTEM ANALYSIS-RELATED NEEDS FOR SYSML 2.0

Need #1 – Universal Unique ID (UUID)

- Universal Unique ID (UUID) to identify all system elements. We must first identify the system / view being analyzed, analysis model, results, and decisions related to that
- UUID concept needs to be a part of the spec and implemented by each SysML tool. Currently each tool defines its own ID system and some have multiple
- Necessary for SysML elements to interoperate and traceable with elements from non-SysML modeling tools and repositories (e.g. PLM, ALM, Databases, Resource on the web)

Need #2 – Data Structures and Types

- Data structures, such as below, and operations (see Need #3) should be an integral part of the language
  - Arrays, Lists (ordered/unordered), Sets, …
  - Matrices – NxN
  - Map (key-value pairs)
  - Vector
  - Tensor
  - Mutable and Immutable lists, sets, etc.
- Types
  - Date
  - Time
  - Geographic map

Need #3 – Operators and Functions

- Operators, such as these, should be a core integral part of the language
  - Differentials $\partial/\partial t$ (of space, time, and other variables)
  - Integrals $\int$ (over space, time, and other variables)
  - Time (temporal properties)
  - Probabilities (property distributions)
  - Math functions
    - Trignometric, Logarithmic, Hyperbolic, …

Need #4 – Geometry

- Basic geometric concepts necessary to develop (1) system specifications, (2) spatial requirements, and (3) communicate with CAD systems should be an integral part of SysML
  - Co-ordinate systems (Rectangular, Polar, etc.)
  - Primitive 2D shapes (point, line, triangle, rectangle/square, pentagon,…,circle)
  - Primitive 3D shapes (3D point, plane, cuboid/cube, sphere, cylinder, cone,…)
  - Mass properties (mass, volume, density, bounding boxes, moments-of-inertia, …)

Need #5 – Transformations between SysML and non-SysML models

- Express mathematical transformations between SysML model and non-SysML models, especially for solvers, e.g.
  - equation-based models,
  - state-based models,
  - flow-based models,
  - graph-based models
- Currently this is no way to express these transformations except for some “tagging” using custom stereotypes
- Potentially extend the viewpoint concept with math formalisms to generate non-SysML models (views beyond static documents)

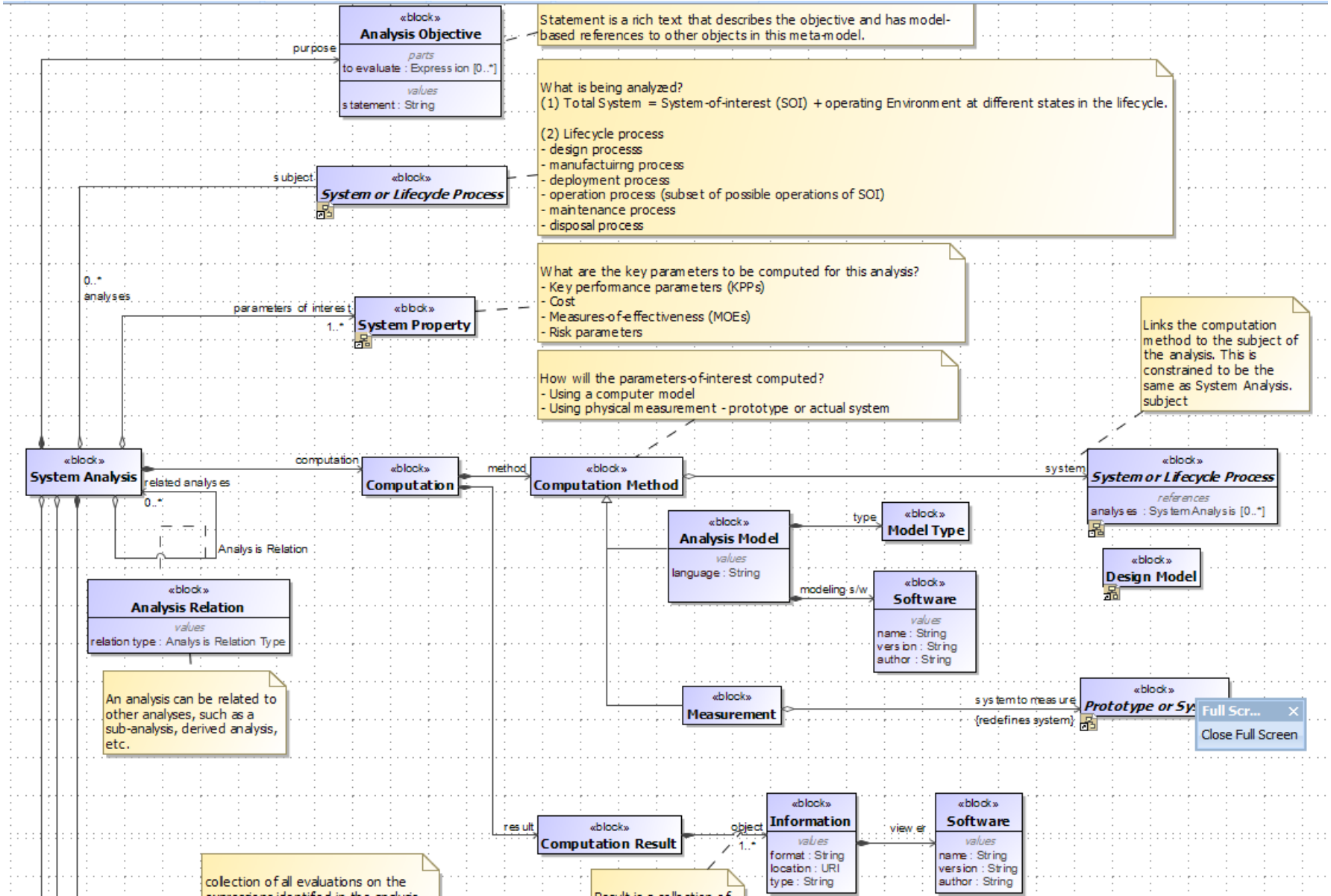
Need #6 – Visualization of analysis results

- Various forms of visualization of analysis results should be available
  - Tables
  - Plots (2D and 3D)
  - Custom charts (extensibility)

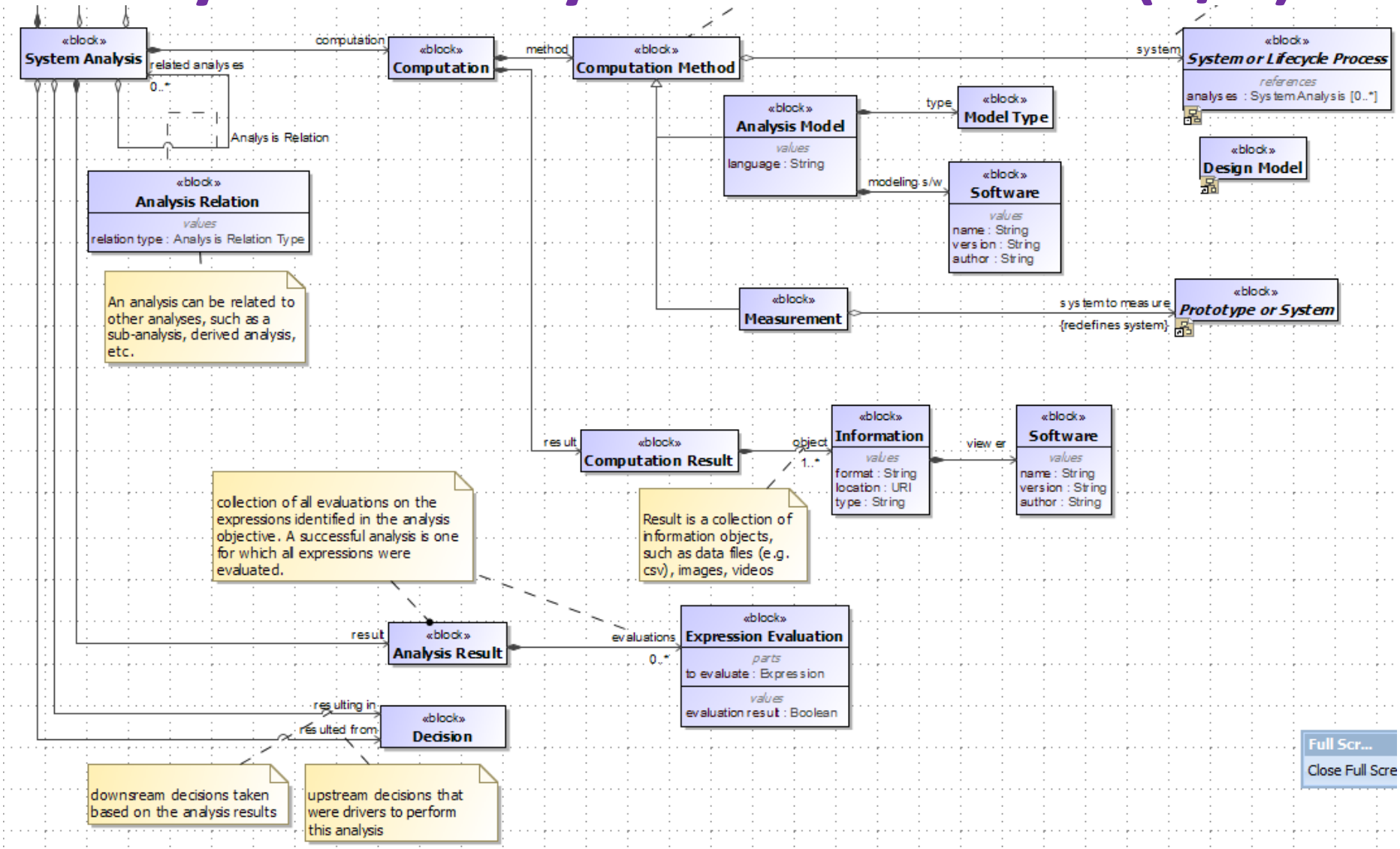
Need #7 – Analysis lifecycle

- "Analysis" as a separate concept in SysML 2 – represents the investigation that will be performed on a "System"
- Objective of the analysis
- System properties of interest
- Analysis models used
  - Multi-fidelity, multi-scale,
  - Formulation environment
  - Execution environment
- Relations to other analyses (upstream, downstream)
- Results of the analysis
- Decisions taken from the analysis (e.g. using OMG Decision Modeling Notation)

Elaborating Need #7
System Analysis Meta-Model (1/2)



Elaborating Need #7
System Analysis Meta-Model (2/2)



Next Steps

- Requesting feedback from
  - SysML 2 WG members
  - End user organizations
- Add to the needs based on the feedback
- Identify testbed end-user problems
- It would be great to:
  - Develop an early “profile” to represent analysis concepts
  - Use concepts to demo formulation and execution of analysis problems using COTS/open source math solvers