intercax

# System Analysis

## Requirements for SysML 2.0

SysML2WG Meeting, OMG Technical Meeting
Reston, VA, Mar 15, 2016

**Manas Bajaj, PhD**

Co-Founder & Chief Systems Officer

**manas@intercax.com**

**www.intercax.com**

# Team

- Manas Bajaj, Intercax [lead for SysML 2 WG]
- Bjorn Cole, NASA JPL
- Ahsan Qamar, Ford
- George Walley, Ford

How do we work? = Google Drive space (hosted by Intercax) for all members to jot down their ideas and thoughts. Team meetings to refine and merge ideas into a coherent story.

If you would like to contribute, please email manas@intercax.com

# Contents

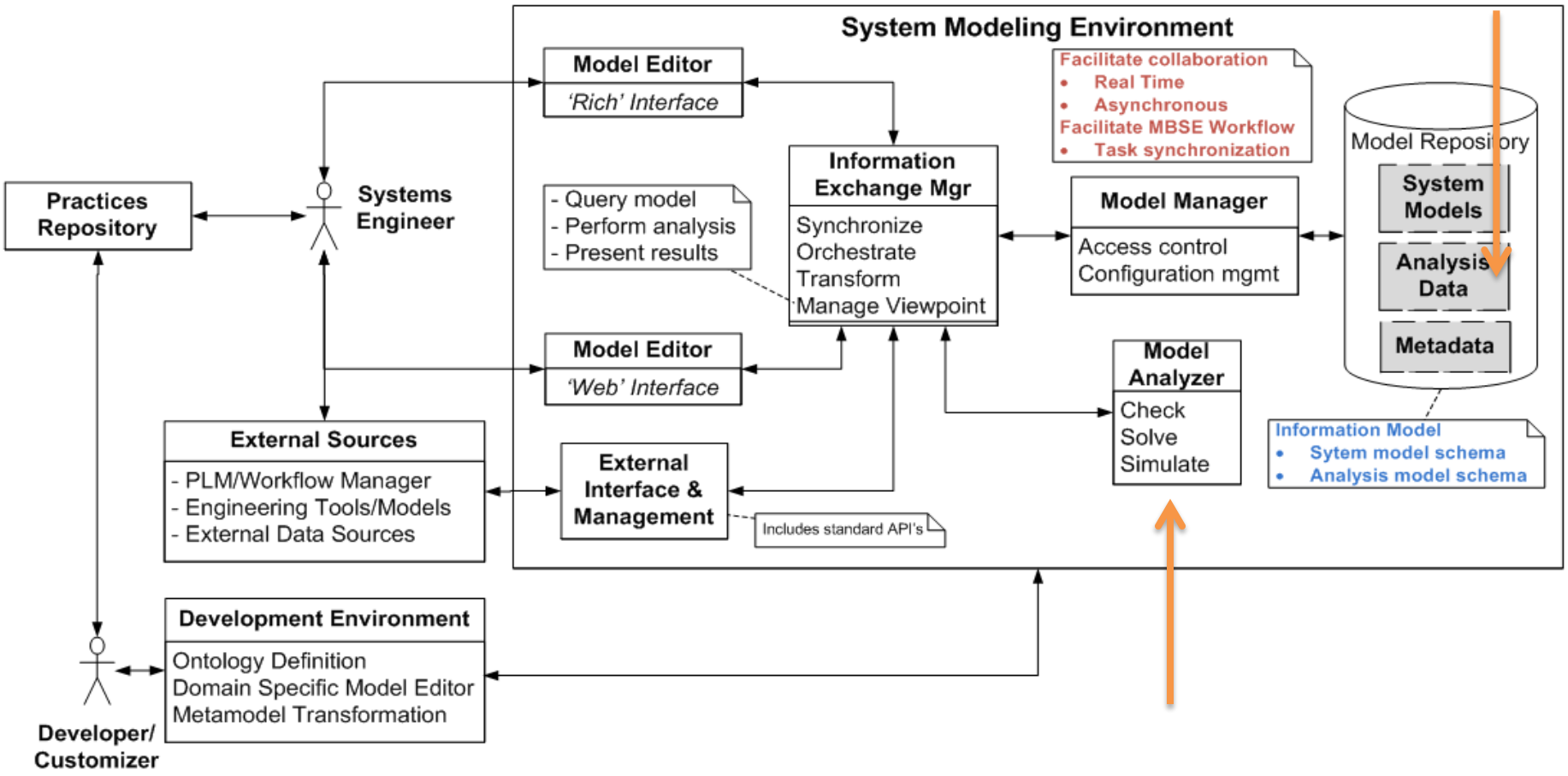
- Goal 
- Analysis
  - Concept
  - Issues in SysML 1.4
  - Effectiveness Measures
  - SME Services to Support Analysis
  - Hybrid SUV Change Scenario
- How can you get involved?

# 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 *system* analyses and execution tools
- Manage analysis models and relate results to decisions
- Improved user interaction to define/generate, execute, archive analysis models (analysis lifecycle)

# Contents

- Goal
- Analysis
  - Concept 
  - Issues in SysML 1.4
  - Effectiveness Measures
  - SME Services to Support Analysis
  - Hybrid SUV Change Scenario
- How can you get involved?

# 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?

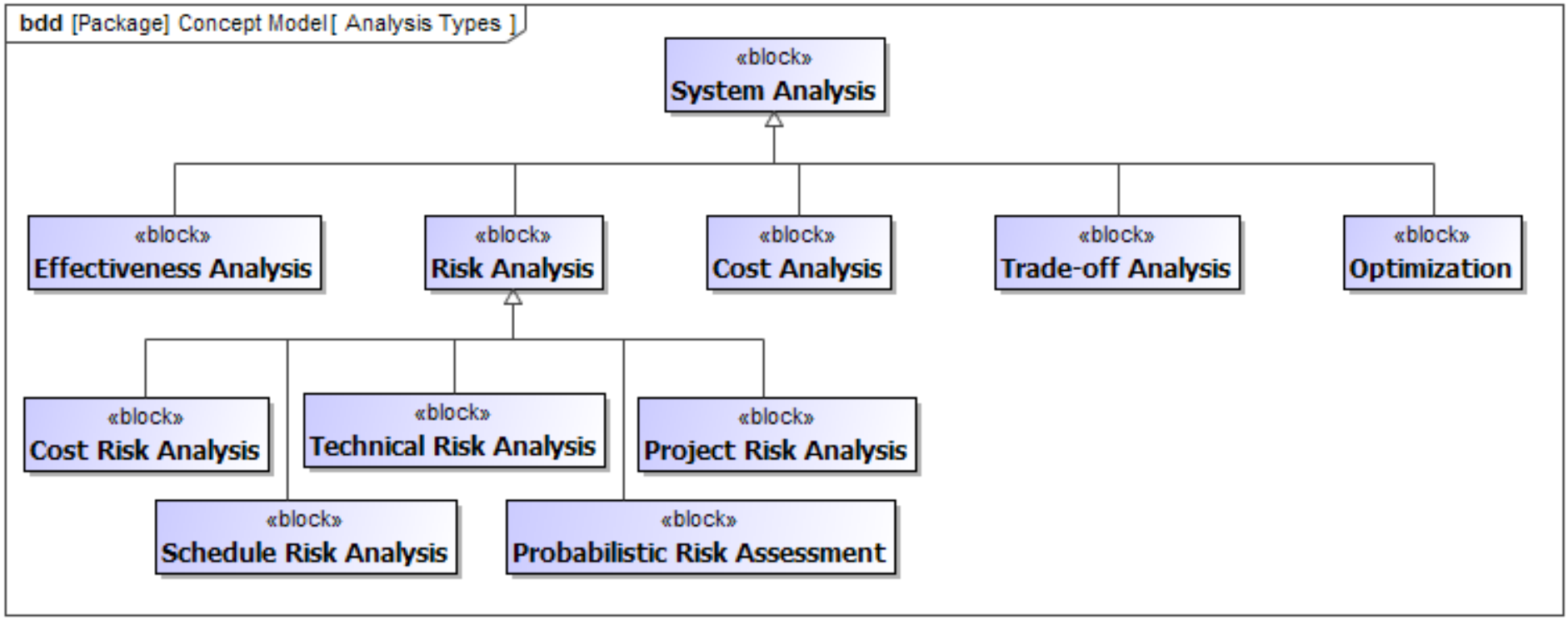
# Scope = System-Level Analyses

- Analysis that has wide-ranging or cross-cutting effect in a complicated system - greater than 100's of parts or involving more than one academic discipline of engineering
  - Simulation of science return over time from an interplanetary probe
  - Evaluation of fuel economy over a course
  - Impact assessment of change in a high-level requirement
- Analysis in each domain (mechanical, electrical, software) considered because of their system-level impact but not the primary subject of our investigation

# System Analysis Roles

- **Customer** = Engineer / manager who requires technical input before a decision is made.
  - **Focus** = Use analysis to rapidly explore the problem space, visualize results, and develop key insights
- **Performer** = Engineer who formulates the approach, selects or builds appropriate models of physics or behavior, and applies skills to providing technical insight
  - **Focus** = Study the domain, and develop analysis models of varying-fidelities, deep knowledge of domain and tools
- Typically
  - Customer and Performer may be different groups of individuals in large organizations, OR
  - Same person in a small organization

# Types of Analyses




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

# Types of Analyses
## Requirements Analysis [new]

- *Analyzing functional scenarios and requirements captured in the model*
- Requirements Quality Analysis
  - Are there situation/scenarios the system will encounter that are not covered by the formulated requirements [Coverage]?
  - Are there conflicting, or overlapping requirements between different enterprise systems [Logical Consistency]?
  - Are there requirements/constraint that have no mapping to a physical realization [Design Completeness]?
  - Is there any design space left? [Physical Consistency]
  - Did we blow out the cost or corrupt the business case? [Viability]
  - Did we make statements that have a precise, logical condition that can be tested? [Verifiability]
  - Can we simulate a feasible concept or two for the stakeholders for their acceptance [Validation]?

# Types of Analyses
## Requirements Analysis [new] – cont.

- Timing-based Requirements
  - Can we represent requirement as a plot of desired system response versus time?
  - Can we designate performance regions in this plot and identify system states?

# Types of Analyses
## Programmatic Analysis [new]

- Programmatic analysis supports project management by providing metrics on the rate of progress, the estimated amount of work to go, and possible divergences between planned and expected technical outcomes. Examples of this ask the following questions:
  - How many requirements are tested or not satisfied?
  - Are there concerns that are not addressed yet?
  - How to define a measure on progress based on the information available at the system-level?
  - How much work has been done so far?
  - How much work is left to go?
  - How much rework will a given change generate [impact assessment]?
  - Is the work planned well-allocated or is it overemphasizing one area at the cost of others?

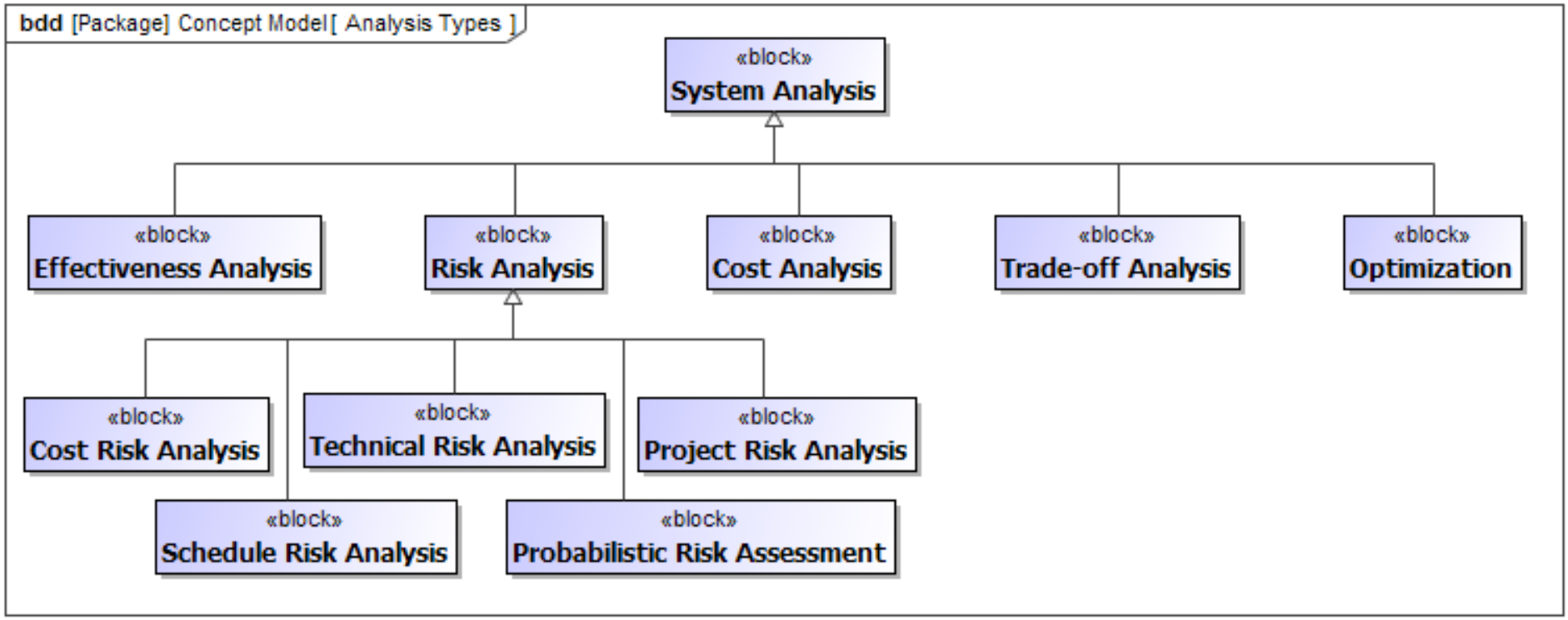
# Types of Analyses
## Resource Analysis [new]

- Determine if appropriate resources are being made available to achieve project goals.
  - Technical resources, e.g., mass, power, fuel.
  - External resource, e.g. cost or schedule (overlap with programmatic analysis).
  - In a more enterprise- or market-oriented project, the resources examined may include useful infrastructure such as a power grid capacity or fueling network.

# Broader Issues

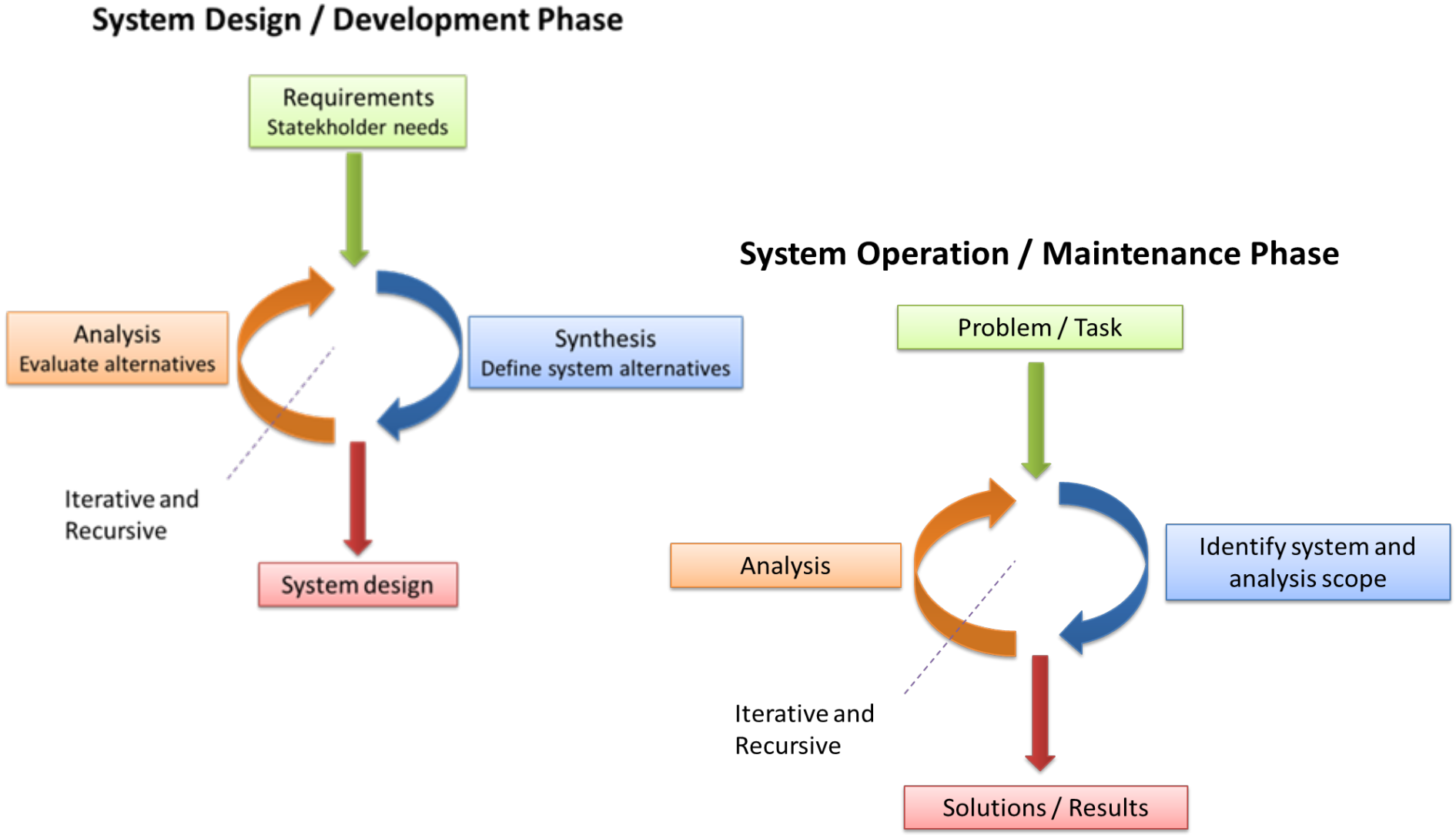
- How do we mathematically represent assumptions used in an analysis model?
- How do we mathematically represent the fidelity of analysis/analysis model, and use it to grade analysis models?
- How do we decide if we need a higher fidelity analysis for decision making, or is the current analysis “good enough”?

# Types of Analyses

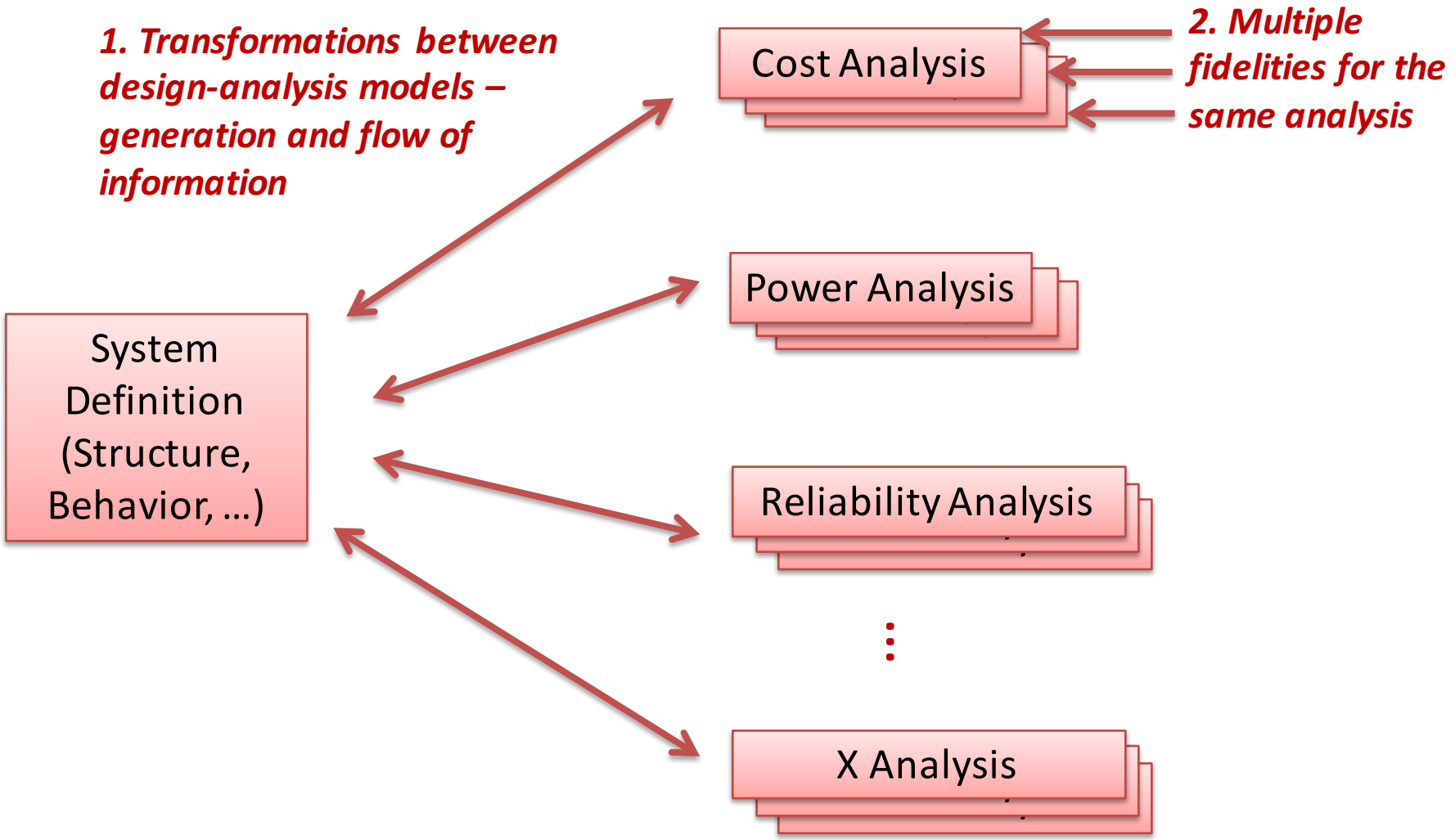



- **Quantitative Analyses**, e.g. computing MoEs
- **Qualitative Analyses**, e.g. If I change (increase) X, will Y change (increase/decrease)?

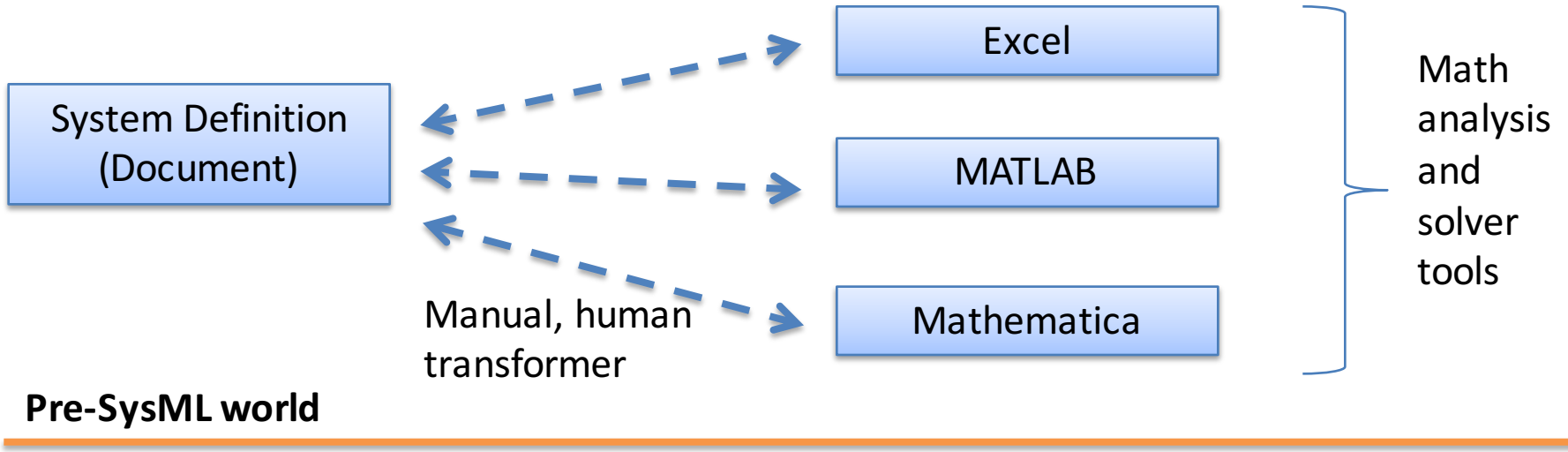
# Analysis during system development and operations



# Design and Analysis are separate models



# Where does the system analysis model live?



# Contents

- Goal
- Analysis
  - Concept
  - Issues in SysML 1.4 
  - Effectiveness Measures
  - SME Services to Support Analysis
  - Hybrid SUV Change Scenario
- How can you get involved?

# Issue #1 – Analysis is a missing concept, Where do I start?

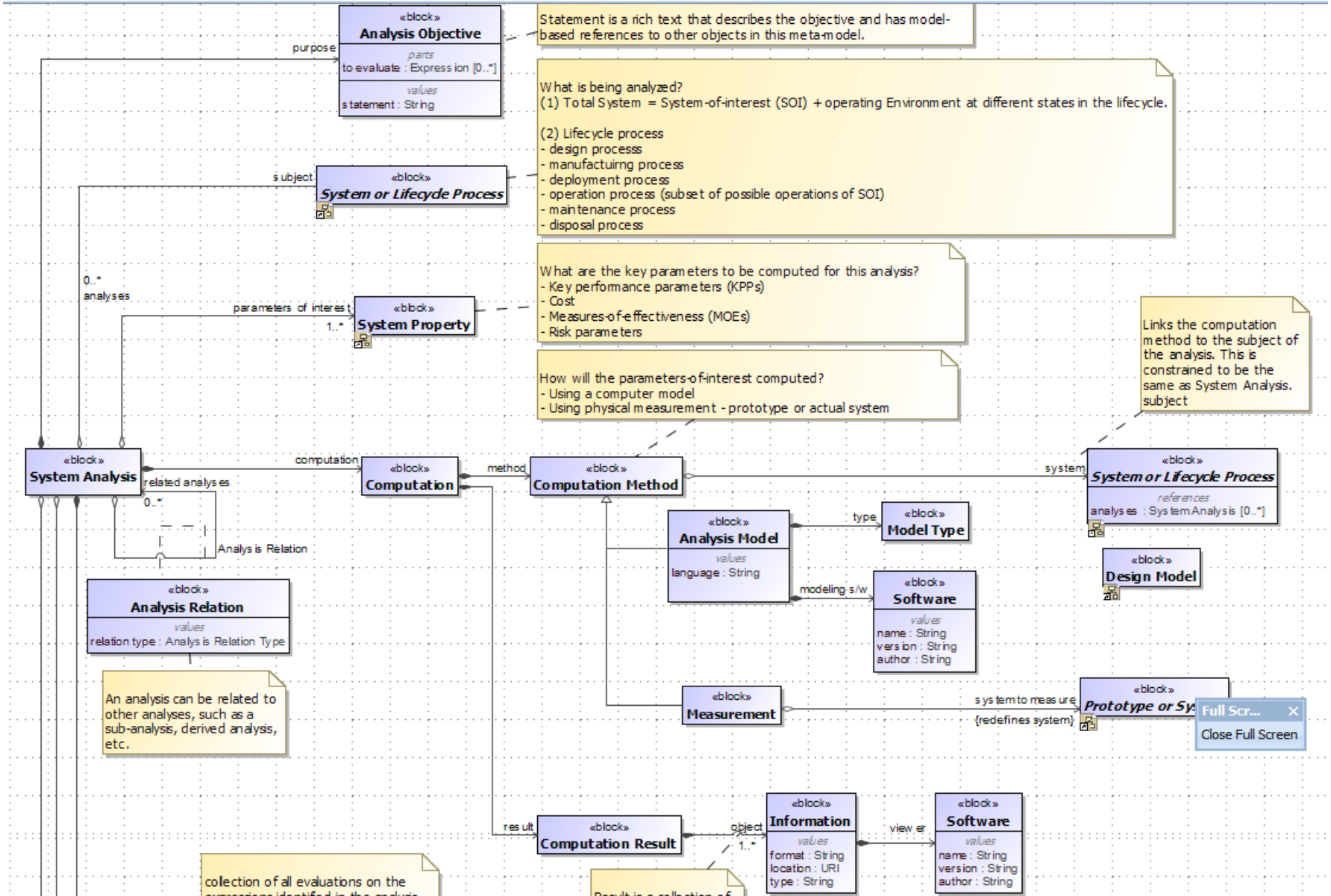
- Analysis is a missing concept in SysML
  - SysML provides low-level modeling constructs (e.g. blocks, constraint blocks, activities) that can be used for representing various SE artifacts. But,
  - Missing high-level SE concepts such as “System”, “Analysis”, “Decision”, “Trade Study”
- Given a system definition/design, can’t easily query
  - What *analyses* will be or have been performed on the system?
  - What are the *results* of the analyses?
  - What issues did the analysis reveal?
  - What architecture decisions were taken from the analyses?
  - Where is the revised version of the architecture?

# Issue #2 – No clear way to represent and query artifacts related to analyses

- No direct and clear way to represent:
  - Objective of the analysis
  - System MOEs being analyzed
  - Analysis models (multiple fidelities) for computing MOEs
  - Execution of analysis models (tools, versions, …)
  - Results of executing analysis models
  - Decisions taken from analysis models
  - Analysis relations
    - Decompose a complex analysis into sub-analyses
    - Upstream and downstream analyses

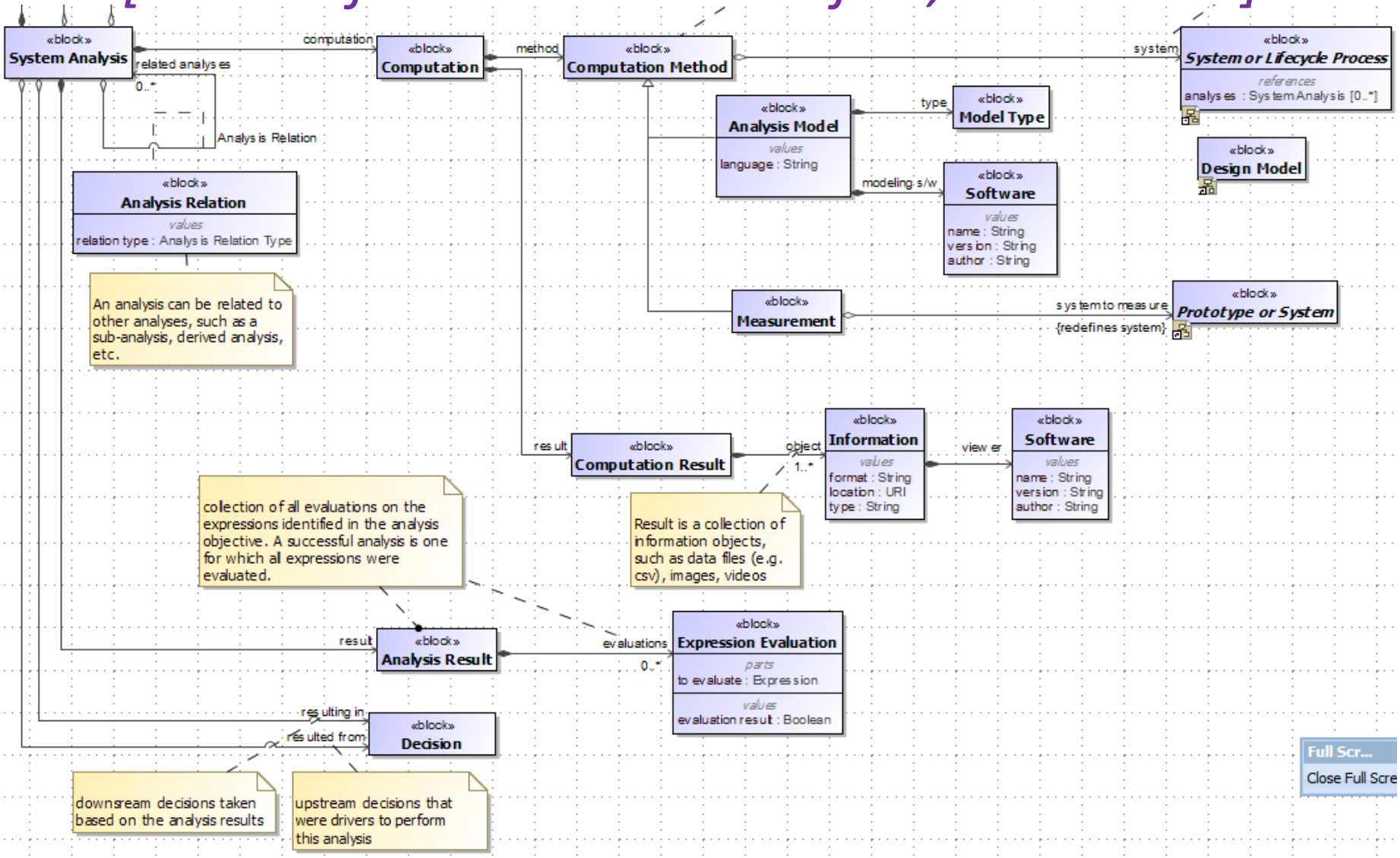
# System Analysis Meta-Model (1/2)

*[Result of NIST-Intercax Project, 2014-2015]*



# System Analysis Meta-Model (2/2)

*[Result of NIST-Intercax Project, 2014-2015]*



# Issue #3 – Design <-> Analysis Model Transformations for Simulation-Based Design

- Where do I express model transformations between design and analysis models?
- Design -> Analysis Model Transformations
  - Knowledge capture -- What assumptions does an analyst take?
  - Executable transforms -- Change the design model, update the analysis model automatically
- Analysis -> Design Model Transformations
  - Relate analysis results of the system design
  - Reconstruct design models from analysis models
- Reference
  - *Knowledge Composition Methodology for Efficient Analysis Problem Formulation in Simulation-based Design (Bajaj, PhD Dissertation, 2008, Georgia Tech)*
    - https://smartech.gatech.edu/handle/1853/26639

# Issue #3 (cont.)

- Express mathematical transformations between constructs in SysML, or SysML and non-SysML constructs, e.g.
  - SysML parametrics <-> equation-based models,
  - SysML state machines <-> state-based models,
  - SysML IBD/Activity <-> flow-based models,
  - SysML <-> 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)

# Issue #4 – Missing Types and Confusing Units

- System definition and analysis needs a rich representation of types
  - Arrays, Lists (ordered/unordered), Sets, …
  - Matrices (mxn)
  - Map (key-value pairs)
  - Tensors and Vectors
  - Mutable and Immutable objects (constants)
  - Date and Time
  - Geographic map
  - Probability Distributions

# Issue #4 – Missing Types and Confusing Units (cont.)

- Yes, we have a QUDV profile and a library of units but it hasn't been useful
  - Units and quantity definitions are fundamental, shouldn't be non-normative extensions. Not all SysML tool implement this.
  - Library of units only SI, missing FPS system
  - Confusing and difficult to create complex, derived units -- given value types *kg*, *m*, *s*, and *kg.m/s^2*, how do I create the value type *g.cm/s^2* leveraging the existing quantity kinds and dimensions?
- Verify if units are same and automated conversion

# Issue #5 – 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
    - Trigonometric, Logarithmic, Hyperbolic, …

# Issue #6 – Geometry

- Lack of geometric concepts needed for design, analysis, and requirements, e.g.:
  - *Distance between CG of comp A and comp B shall be no more than 4.5 cm*
  - *Shape of part A is a Sphere with radius 5 cm*
- Example concepts
  - 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, …)

# Issue #7 – Visualization of analysis results

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

# Issue #8 – 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)

# Issue #9 – Version and Configuration Management

- Version chain
  - What version of the system architecture was analyzed?
  - What version of the analysis model was used?
  - What version of the solver tool used for executing the analysis model?
- Configuration control
  - Who created the analysis and who can modify?
  - Who can run the analysis?
  - Who can view the results of the analysis?
  - Who can affect the system architecture based on the analysis?

# Contents

- Goal
- Analysis
  - Concept
  - Issues in SysML 1.4
  - Effectiveness Measures 
  - SME Services to Support Analysis
  - Hybrid SUV Change Scenario
- How can you get involved?

# Effectiveness Measures

- Can I unambiguously represent analysis and related artifacts for a system, as listed in Issue #2?
- Can I perform analysis-related queries on the system architecture model, as listed in Issue #1?
- Can I keep track of the analysis and related artifacts for future lookup and queries, as listed in Issues #1,2,9?
- Can I seamless generate analysis models in various tools/languages from the SysML architecture model, and can I update/sync the analysis model if the system architecture changes? (Issue #3)

# Contents

- Goal
- Analysis
  - Concept
  - Issues in SysML 1.4
  - Effectiveness Measures
  - SME Services to Support Analysis
    - *See latest version of the spreadsheet*
  - Hybrid SUV Change Scenario
- How can you get involved?

# Contents

- Goal
- Analysis
  - Concept
  - Issues in SysML 1.4
  - Effectiveness Measures
  - SME Services to Support Analysis
  - Hybrid SUV Change Scenario 
- How can you get involved?

# Scenario #1

## *Gov regulation to improve fuel efficiency*

- Analysis needs:
  - HSUV design: Size, Weight, Power,…
  - Environment: Road, Drag (Air), Traffic, Conditions (city/hwy)
  - Driver: Driving profile
- Analysis model
  - Acausal parametric model + trades and optimization
    - Given the design, environment, driver variables, compute fuel efficiency
    - Given the target fuel efficiency and avg values of environment and driver variables, compute max size/weight and min power
  - Fidelities
    - Simple analytical model (static equations)
    - Complex analytical model (dynamic, time-based integrals)
    - CFD model using 2D / 3D CAD of the HSUV
  - Analysis/Simulation Tool
    - Mathematica / MATLAB for equation-based analytical models
    - ABAQUS / ANSYS for CFD models

# Scenario #2

## *Vehicle fails to meet fuel efficiency requirement*

- Search model repository for fuel efficiency analysis models used, and result sets (HSUV design, environment, and drive profile used)
- Compare result set with the actual prototyped / built vehicle.
- Setup and perform new analyses
- Provide recommendations to the design team

# Scenario #3
## *Recall catalytic converter*

- TBD

# Contents

- Goal
- Analysis
  - Concept
  - Issues in SysML 1.4
  - Effectiveness Measures
  - SME Services to Support Analysis
  - Hybrid SUV Change Scenario

# Next Steps

- Finish gathering system analyses needs [wish-list] from team members
- Develop a meta-model in SysML to represent the findings, e.g. types of analyses
- Develop a white paper with detailed documentation on system analysis needs for SysML 2 and issues in SysML 1.x – beyond slideware