The Tao of SysML
SysML的道

Dr. Bruce “Zen Master” Douglass, Ph.D.

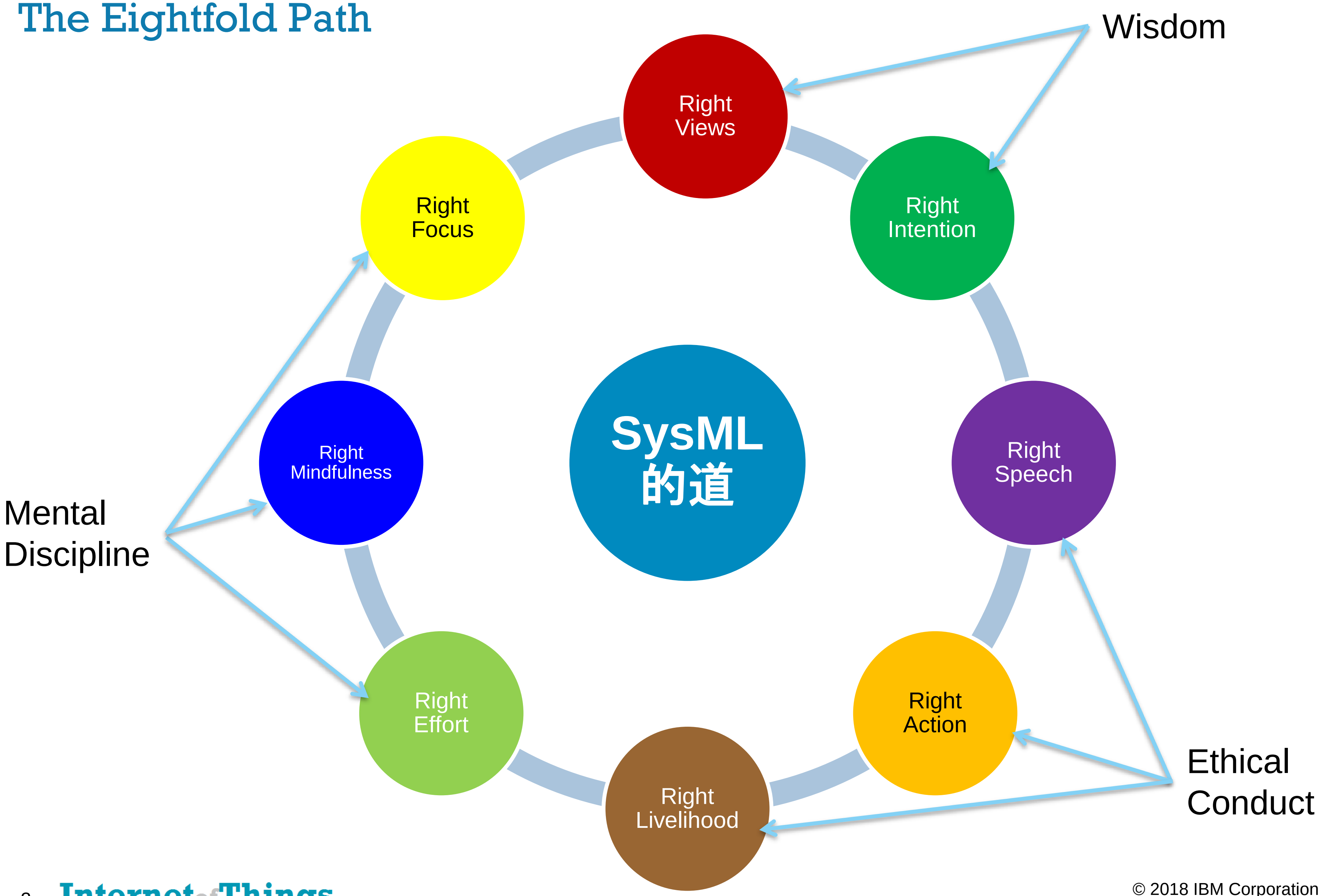
Chief Evangelist

IBM IoT

Bruce.Douglass@us.ibm.com

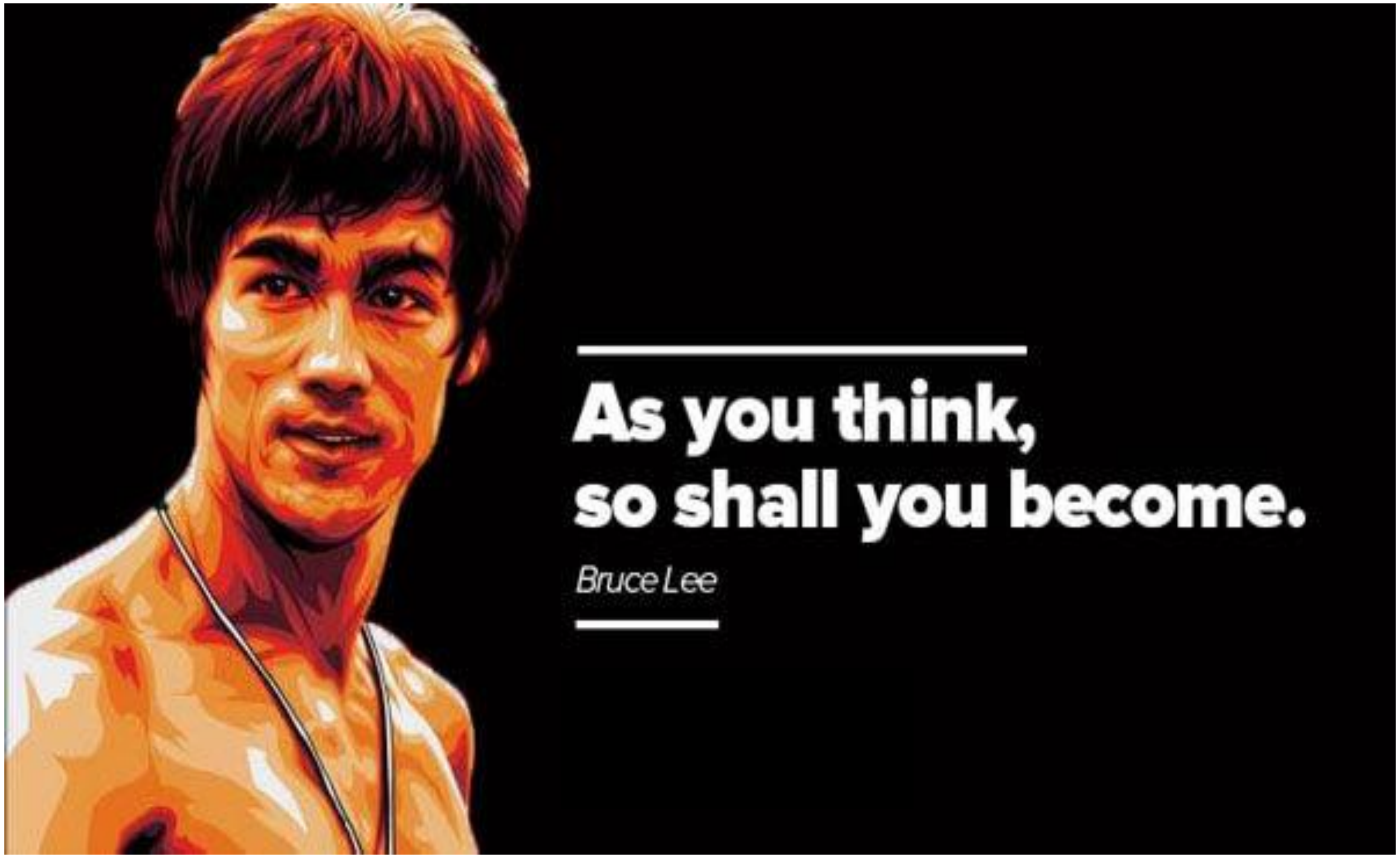
www.Bruce-Douglass.com

The Eightfold Path



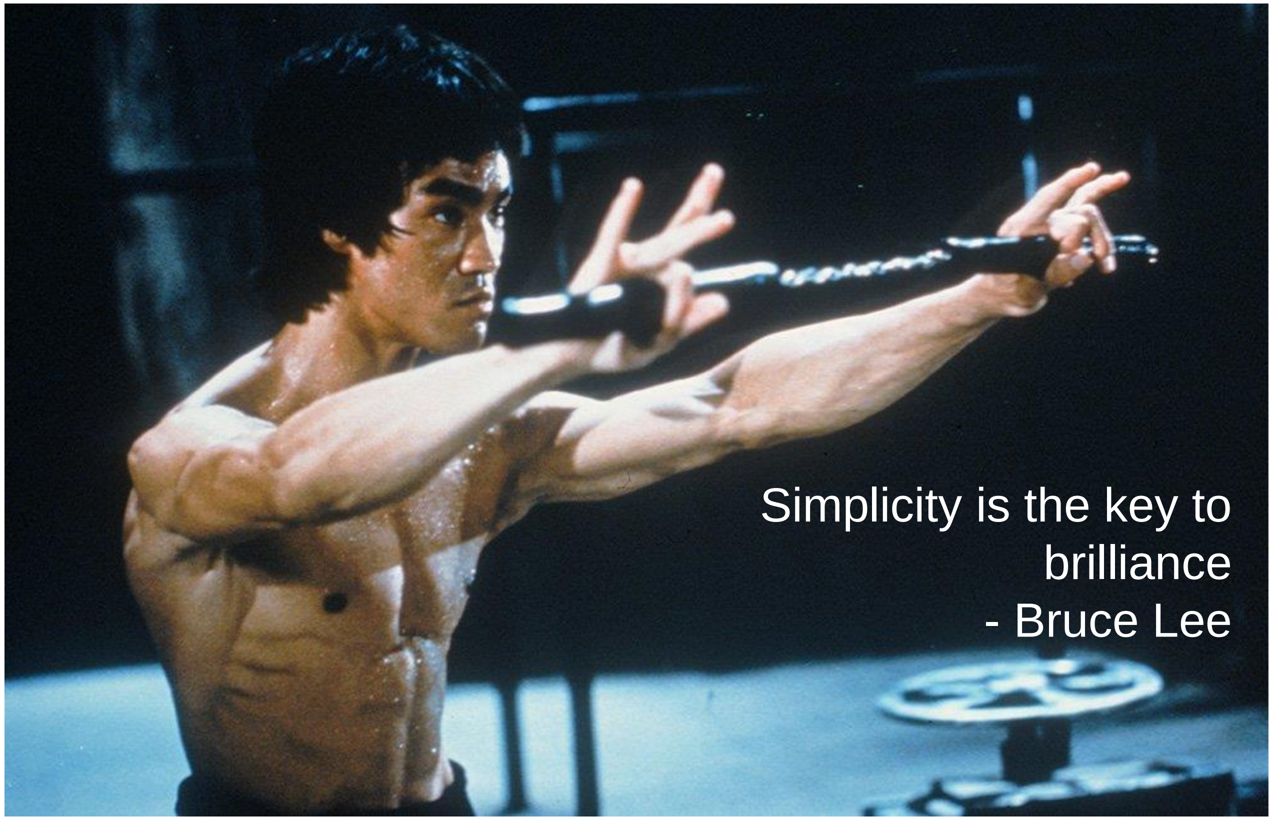
Ethical Conduct: Right Speech

- Precision Modeling (Drawing ≠ Modeling)
- Use Diagrams Correctly
- Subset SysML



Ethical Conduct: Right Action

- Build Semantically Complete Models
- Manage Your Models



Semantically Complete For Purpose

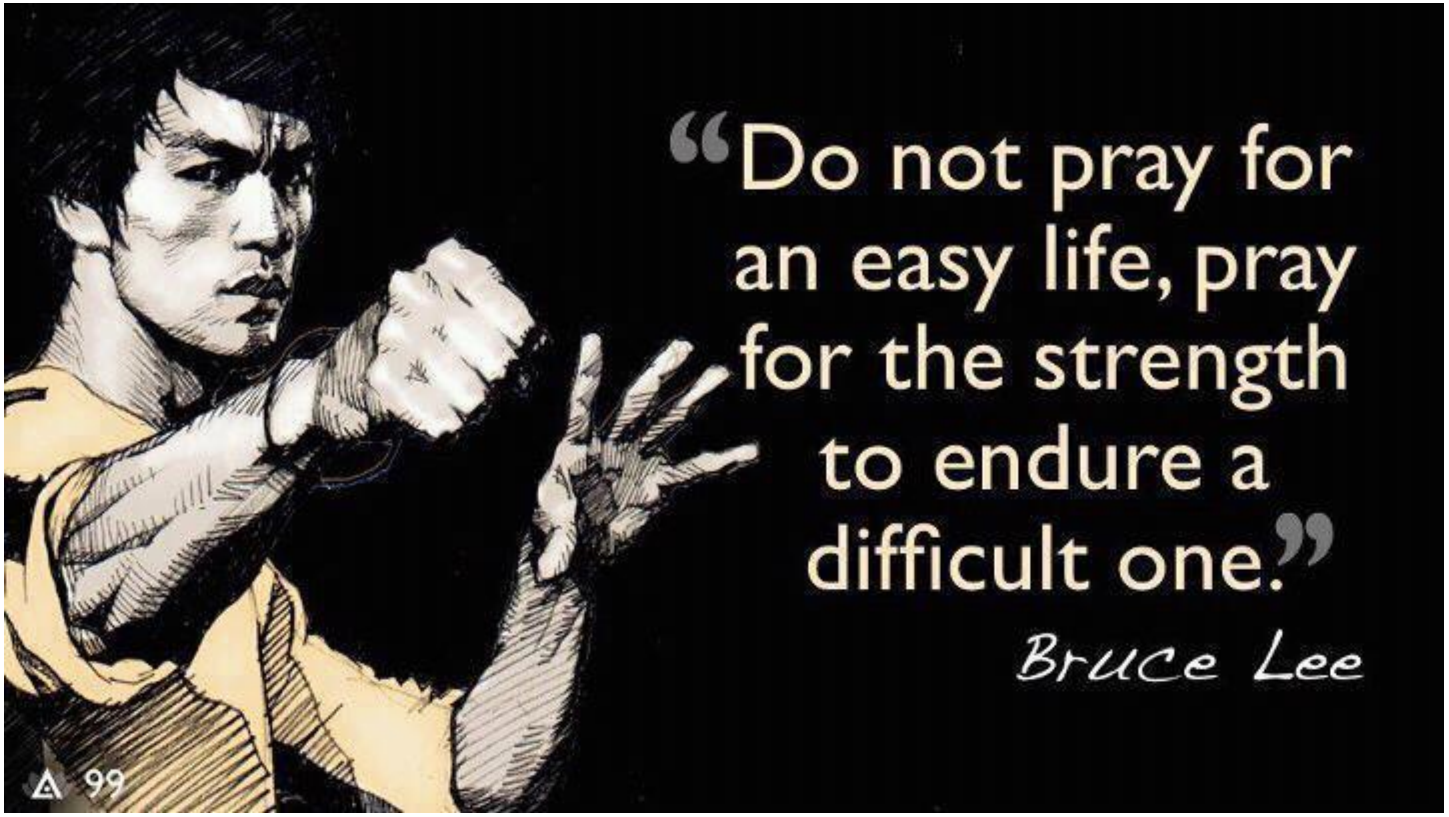
Ask – What information is necessary?
- Abstraction level
  - System scope?
  - Subsystem scope?
  - Design element scope?
- Functionality – input-output control/data transformation
- Structure
- Precision
  - Accuracy
  - Fidelity
- Behavior

Ask – Who needs this information?
- Stakeholders?
- Designers?
- Testers?
- Managers?

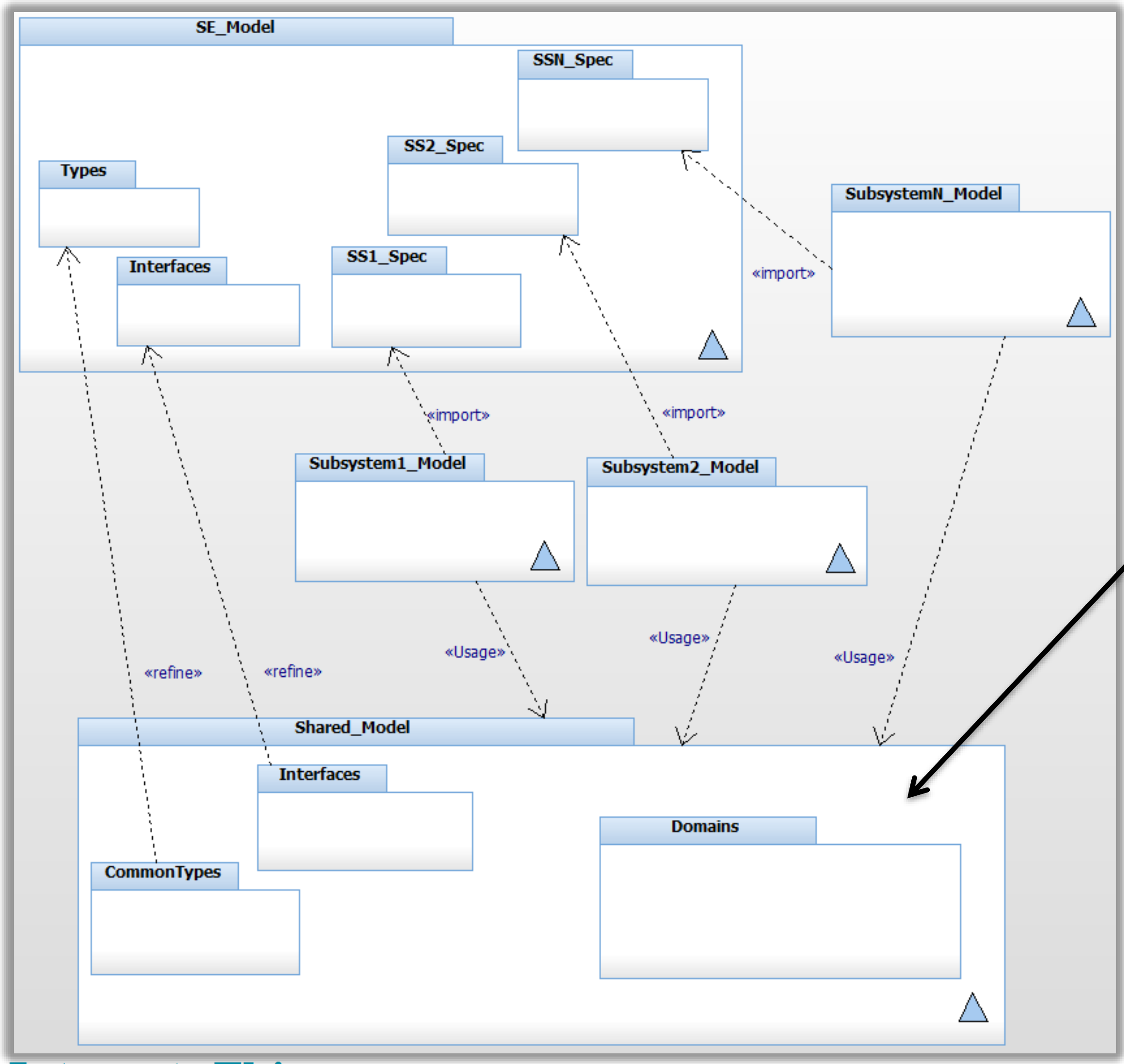
Ask – What outcomes does this information support?

Ethical Conduct: Right Livelihood

- Model Organization
- Verify Model Content


"Do not pray for an easy life, pray for the strength to endure a difficult one."
Bruce Lee
99

Canonical Model Organization



Subdivided into nested subject-oriented packages to store reusable software types and classes

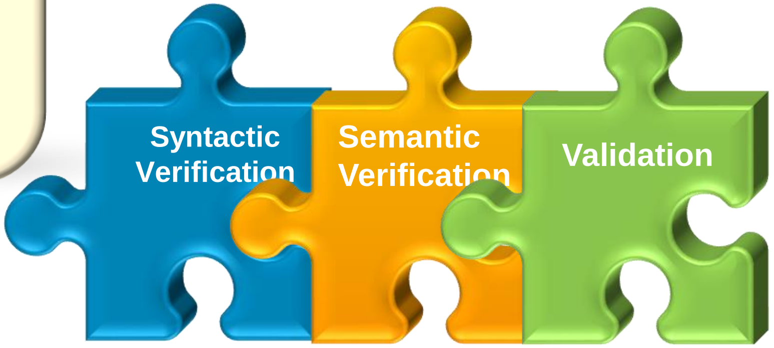
Right Livelihood: Verify Your Models

Semantic Verification

- "correct" (*compliance in meaning*)
  Performed by engineering personnel

Three basic techniques

- **Semantic review** (subject matter expert & peer) – most common, weakest means
- **Testing** – requires executability of work products, impossible to fully verify
- **Formal methods** – strongest but hard to do and subject to invariant violation



Syntactic Verification

– "well-formed" (*compliance in form*)
  Performed by quality assurance personnel

- **Audits** – work tasks are performed as per plan and guidelines
- **Syntactic review** – work products conform to standard for organization, structure and format

Validation

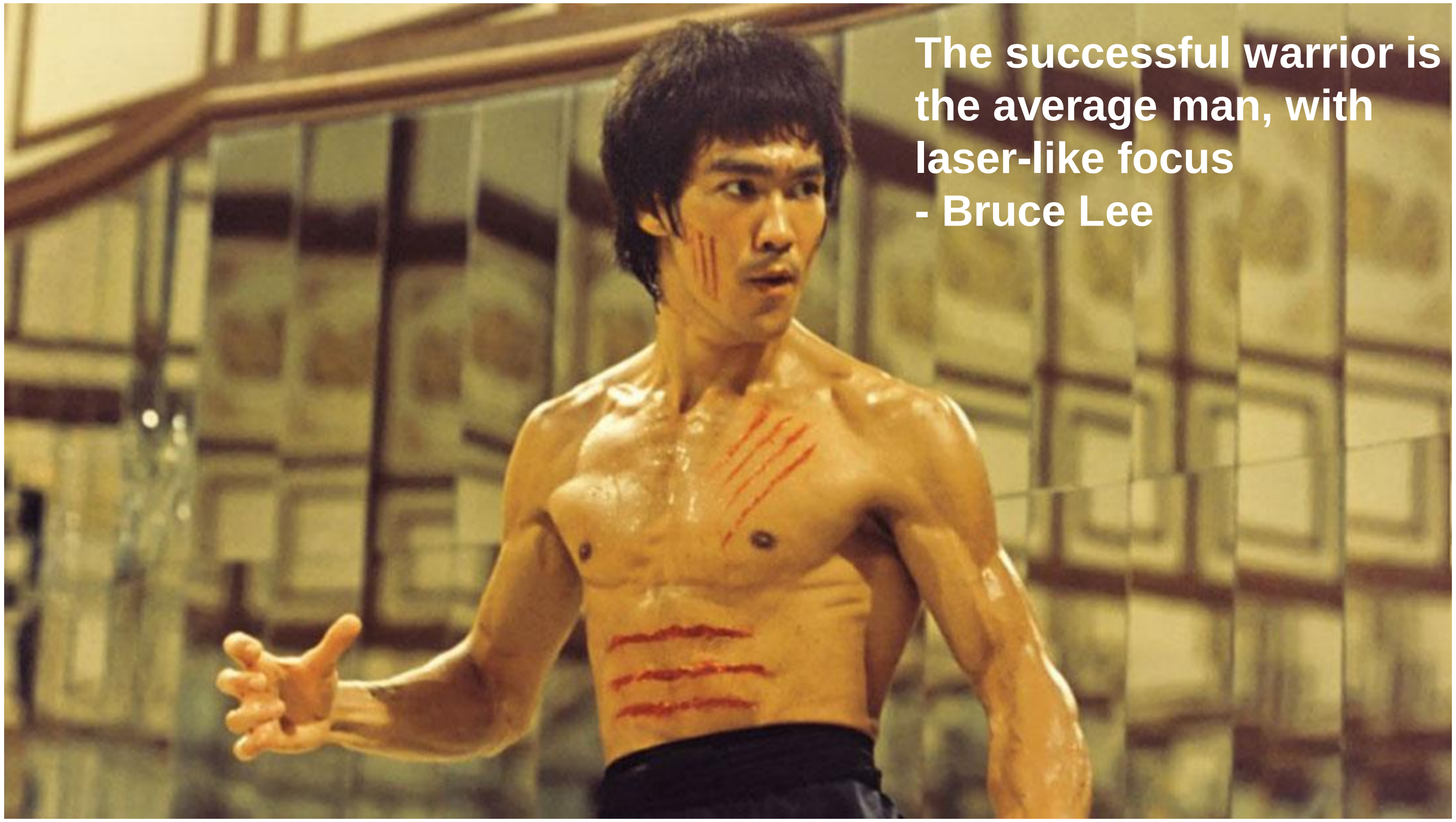
- "meets the stakeholder need"
  Performed by customer + engineering

Some common techniques

- **Review** – (subject matter expert & customer) – most common, weakest
- **Simulation** – show simulated input → outputs
- **Sandbox** – exploratory usage in constrained environment
- **Flight test** – demonstration of system capabilities
- **Deployment –** early usage of system of partial capability

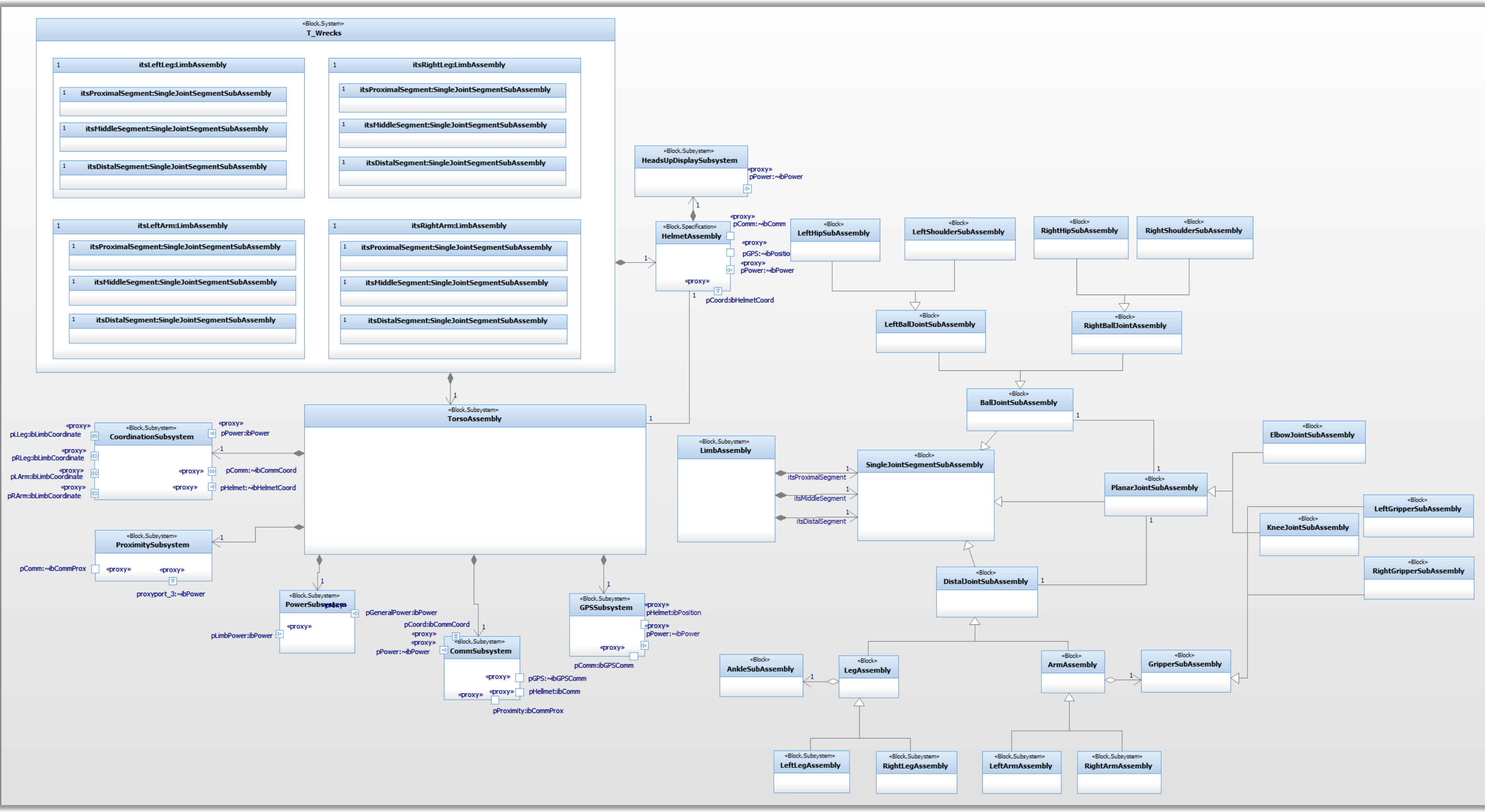
Mental Discipline: Right Focus

- Define Model Purpose and Scope
- Define Model Precision
- Define Abstraction Levels


The successful warrior is the average man, with laser-like focus
- Bruce Lee

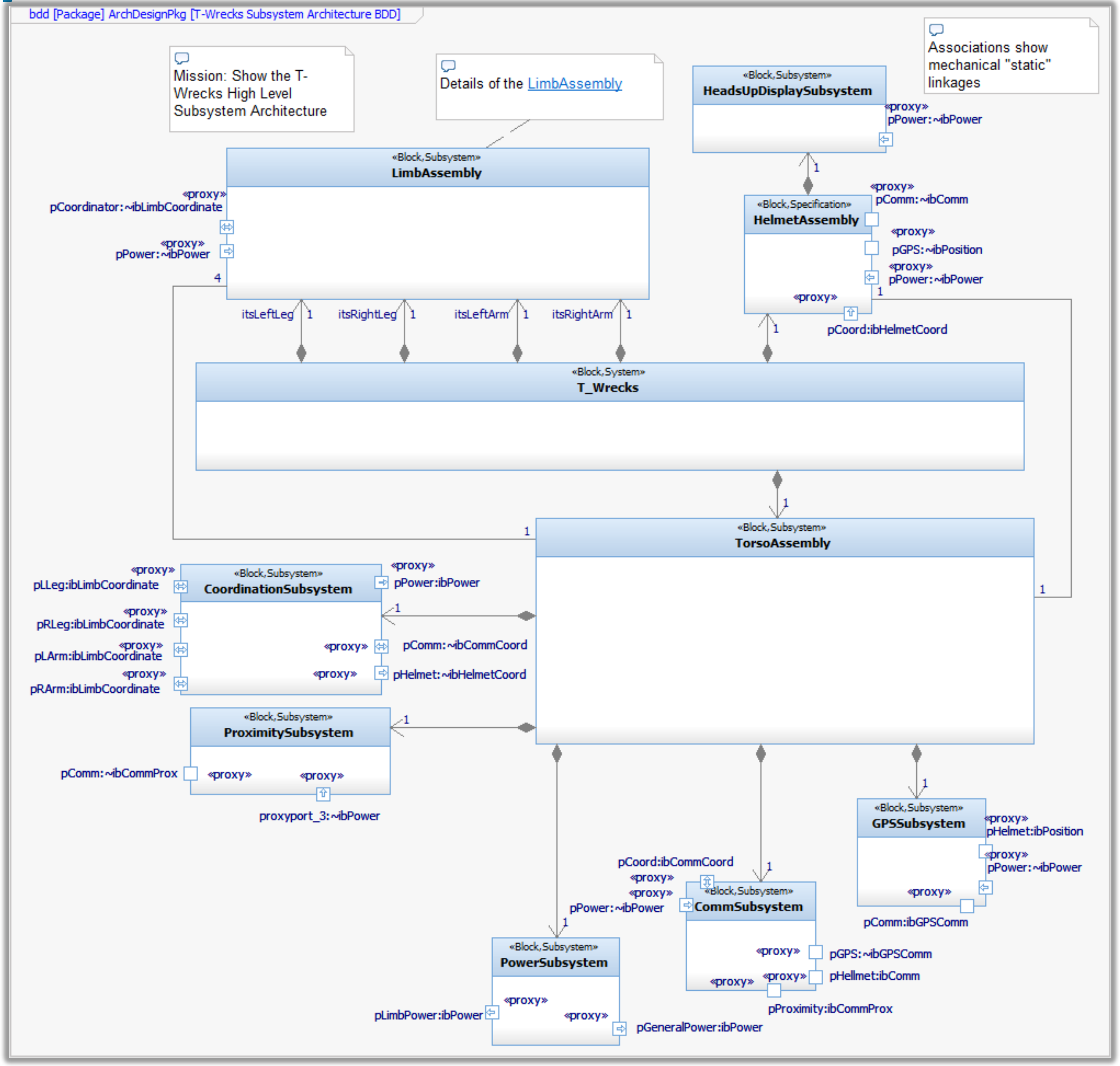
Managing Diagrammatic Complexity

- This diagram has too many level of abstraction and mixes type and containment taxonomies


«Block,System» T_Wrecks
itsLeftLeg:LimbAssembly
itsRightLeg:LimbAssembly
itsLeftArm:LimbAssembly
itsRightArm:LimbAssembly
itsProximalSegment:SingleJointSegmentSubAssembly
itsMiddleSegment:SingleJointSegmentSubAssembly
itsDistalSegment:SingleJointSegmentSubAssembly
«Block,Subsystem» HeadsUpDisplaySubsystem
«Block,Specification» HelmetAssembly
«Block,Subsystem» TorsoAssembly
«Block,Subsystem» CoordinationSubsystem
«Block,Subsystem» ProximitySubsystem
«Block,Subsystem» PowerSubsystem
«Block,Subsystem» CommSubsystem
«Block,Subsystem» GPSSubsystem
«Block,Subsystem» LimbAssembly
«Block» LeftHipSubAssembly
«Block» LeftShoulderSubAssembly
«Block» RightHipSubAssembly
«Block» RightShoulderSubAssembly
«Block» LeftBallJointSubAssembly
«Block» RightBallJointAssembly
«Block» BallJointSubAssembly
«Block» SingleJointSegmentSubAssembly
«Block» PlanarJointSubAssembly
«Block» ElbowJointSubAssembly
«Block» KneeJointSubAssembly
«Block» LeftGripperSubAssembly
«Block» RightGripperSubAssembly
«Block» DistalJointSubAssembly
«Block» AnkleSubAssembly
«Block» LegAssembly
«Block» ArmAssembly
«Block» GripperSubAssembly
«Block,Subsystem» LeftLegAssembly
«Block,Subsystem» RightLegAssembly
«Block,Subsystem» LeftArmAssembly
«Block,Subsystem» RightArmAssembly
itsProximalSegment
itsMiddleSegment
itsDistalSegment
«proxy» pPower:~ibPower
«proxy» pComm:~ibComm
«proxy» pGPS:~ibPositio
pCoord:ibHelmetCoord
pLLeg:ibLimbCoordinate
pRLeg:ibLimbCoordinate
pLArm:ibLimbCoordinate
pRArm:ibLimbCoordinate
pPower:ibPower
pComm:~ibCommCoord
pHelmet:~ibHelmetCoord
pComm:~ibCommProx
proxyport_3:~ibPower
pLimbPower:ibPower
pGeneralPower:ibPower
pCoord:ibCommCoord
pPower:~ibPower
pGPS:~ibGPSComm
pHelmet:ibComm
pProximity:ibCommProx
pHelmet:ibPosition
pComm:ibGPSComm

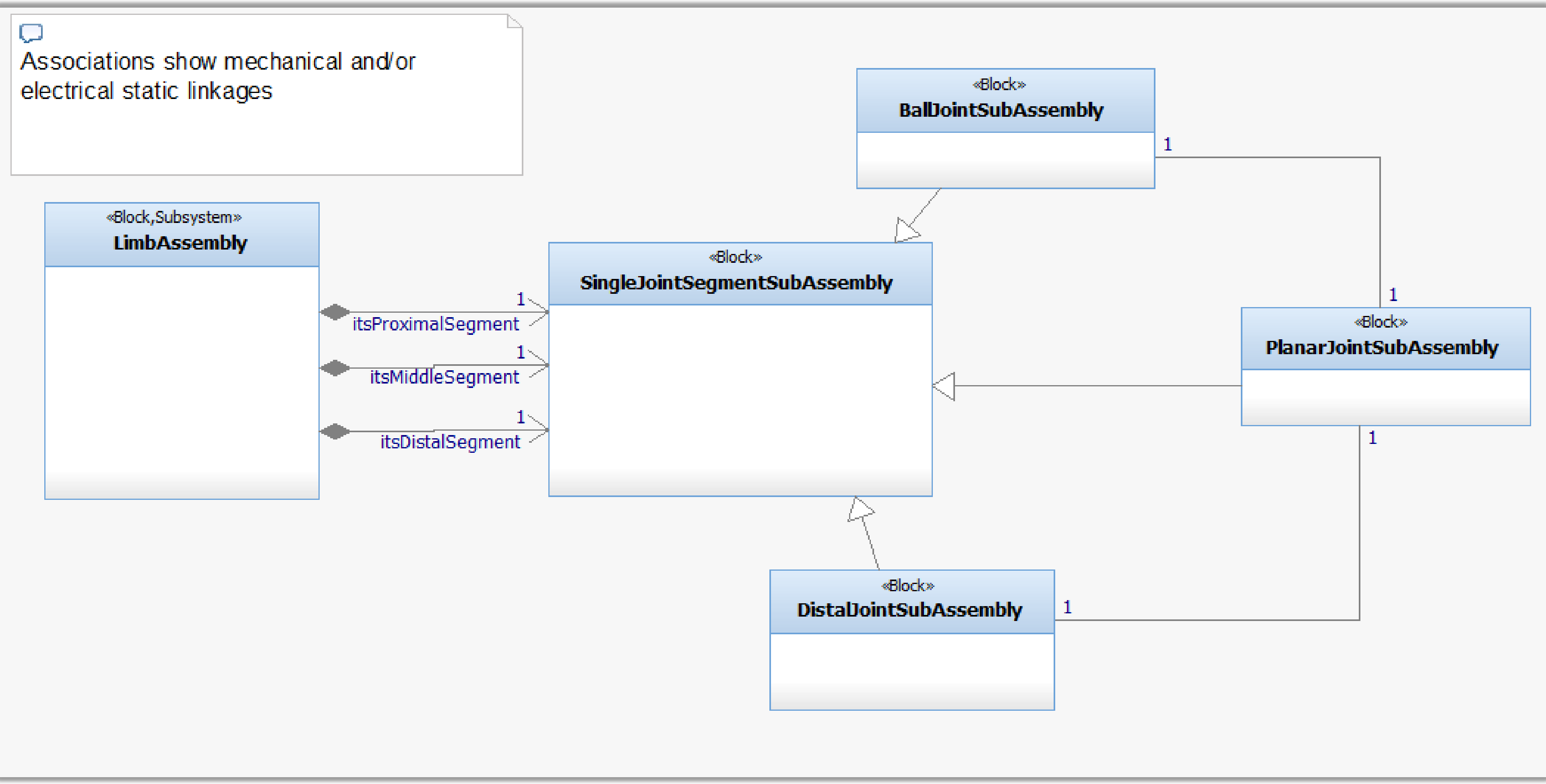
InternetofThings

Mission 1: Overall Subsystem Architecture

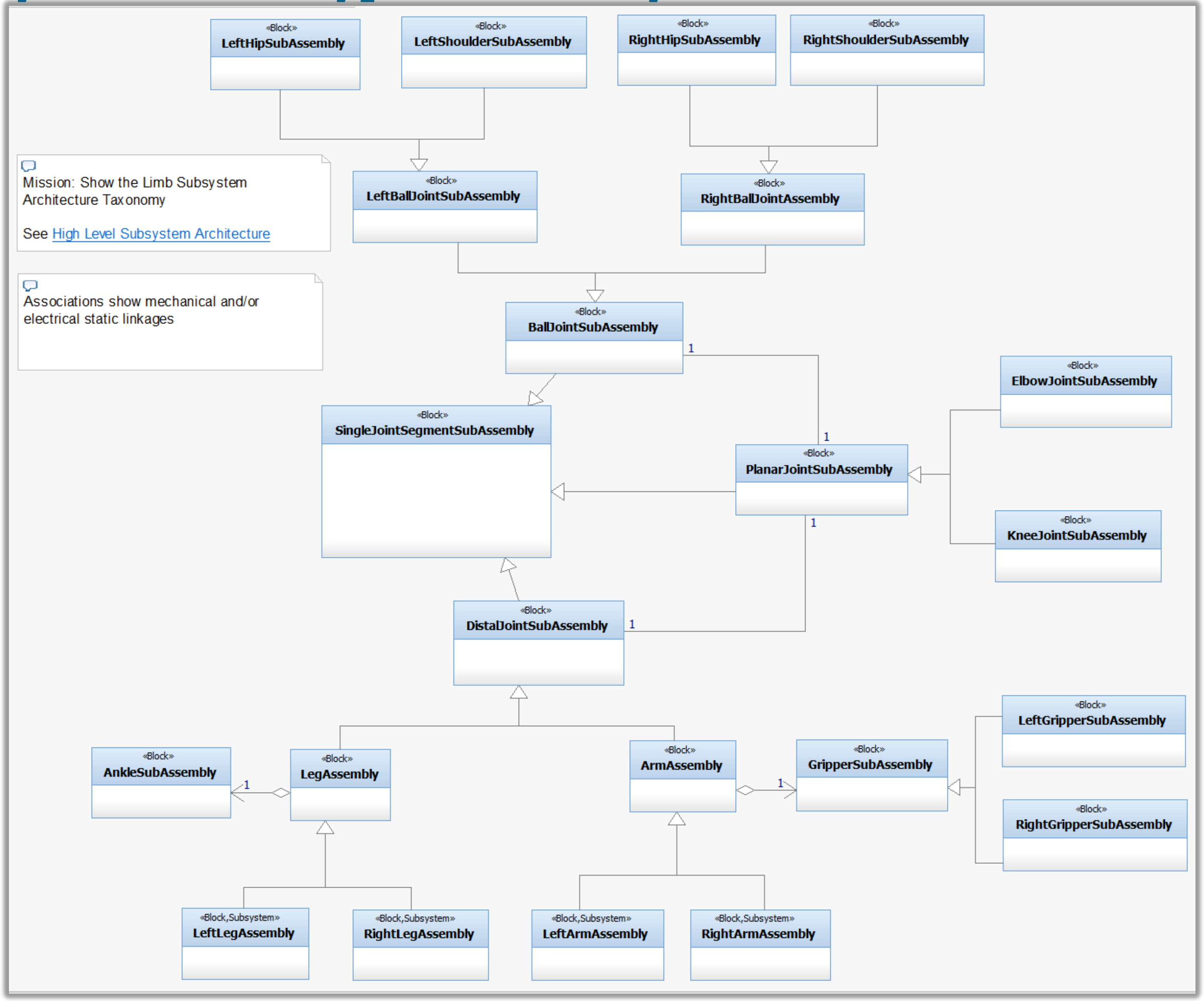

bdd [Package] ArchDesignPkg [T-Wrecks Subsystem Architecture BDD]
Mission: Show the T-Wrecks High Level Subsystem Architecture
Details of the LimbAssembly
Associations show mechanical "static" linkages
«Block,Subsystem» HeadsUpDisplaySubsystem
«proxy» pPower:~ibPower
«Block,Subsystem» LimbAssembly
«proxy» pCoordinator:~ibLimbCoordinate
«proxy» pPower:~ibPower
4
itsLeftLeg 1
itsRightLeg 1
itsLeftArm 1
itsRightArm 1
«Block,Specification» HelmetAssembly
«proxy» pComm:~ibComm
«proxy» pGPS:~ibPosition
«proxy» pPower:~ibPower
«proxy» pCoord:ibHelmetCoord
«Block,System» T_Wrecks
«Block,Subsystem» TorsoAssembly
«Block,Subsystem» CoordinationSubsystem
«proxy» pLLeg:ibLimbCoordinate
«proxy» pRLeg:ibLimbCoordinate
«proxy» pLArm:ibLimbCoordinate
«proxy» pRArm:ibLimbCoordinate
«proxy» pPower:ibPower
«proxy» pComm:~ibCommCoord
«proxy» pHelmet:~ibHelmetCoord
«Block,Subsystem» ProximitySubsystem
pComm:~ibCommProx «proxy»
«proxy» proxyport_3:~ibPower
«Block,Subsystem» GPSSubsystem
«proxy» pHelmet:ibPosition
«proxy» pPower:~ibPower
«proxy» pComm:ibGPSComm
«Block,Subsystem» CommSubsystem
pCoord:ibCommCoord «proxy»
«proxy» pPower:~ibPower
«proxy» pGPS:~ibGPSComm
«proxy» pHellmet:ibComm
«proxy» pProximity:ibCommProx
«Block,Subsystem» PowerSubsystem
«proxy» pLimbPower:ibPower
«proxy» pGeneralPower:ibPower

Internet of Things

Mission 2: Subsystem Internal Architecture

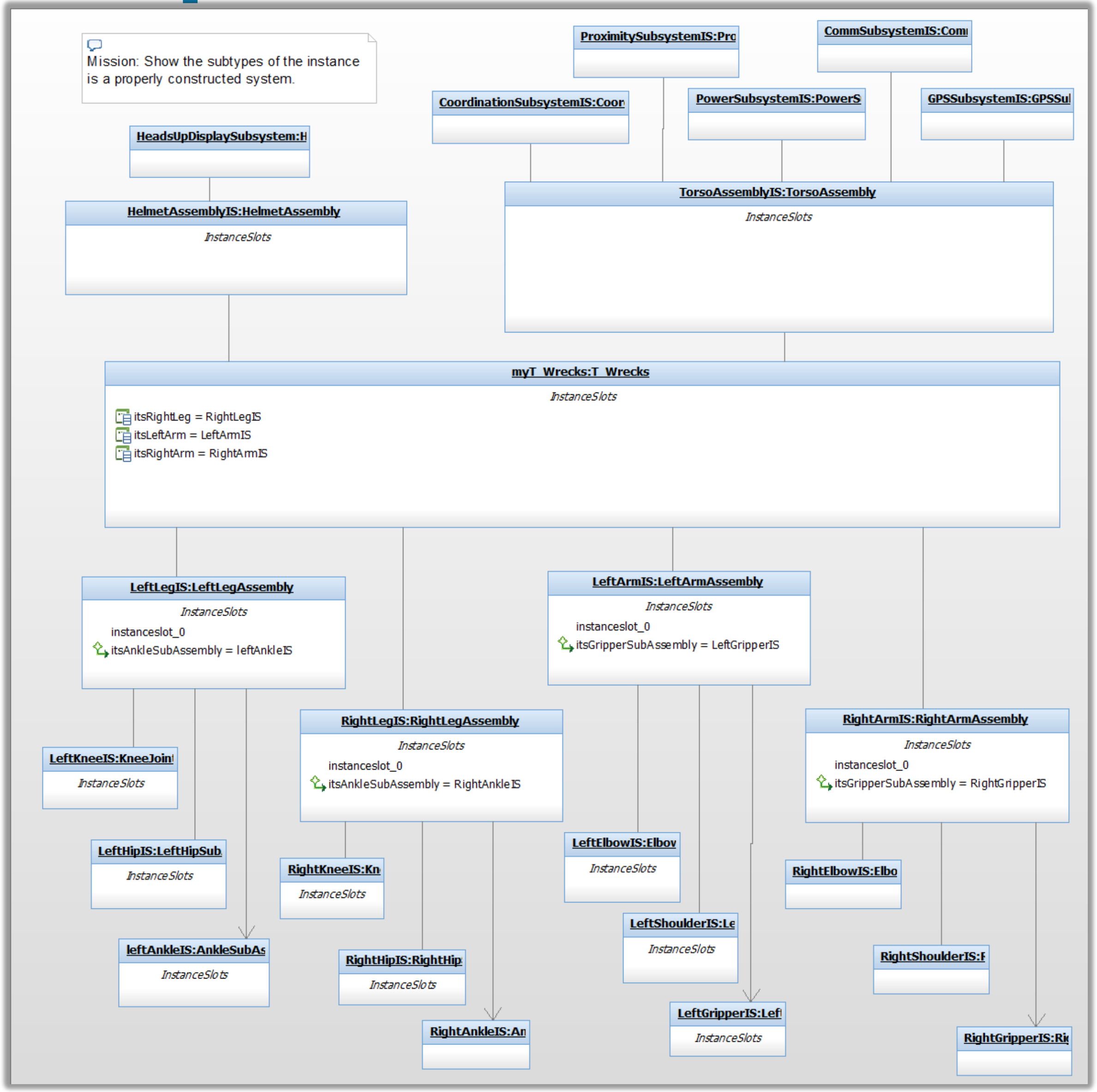


Mission 3: Subsystem Block Type Taxonomy


«Block» LeftHipSubAssembly
«Block» LeftShoulderSubAssembly
«Block» RightHipSubAssembly
«Block» RightShoulderSubAssembly
Mission: Show the Limb Subsystem Architecture Taxonomy
See High Level Subsystem Architecture
«Block» LeftBallJointSubAssembly
«Block» RightBallJointAssembly
Associations show mechanical and/or electrical static linkages
«Block» BallJointSubAssembly
1
«Block» ElbowJointSubAssembly
«Block» SingleJointSegmentSubAssembly
1
«Block» PlanarJointSubAssembly
1
«Block» KneeJointSubAssembly
«Block» DistalJointSubAssembly
1
«Block» LeftGripperSubAssembly
«Block» AnkleSubAssembly
1
«Block» LegAssembly
«Block» ArmAssembly
1
«Block» GripperSubAssembly
«Block» RightGripperSubAssembly
«Block,Subsystem» LeftLegAssembly
«Block,Subsystem» RightLegAssembly
«Block,Subsystem» LeftArmAssembly
«Block,Subsystem» RightArmAssembly

Note that I can link together relevant diagrams with navigation hyperlinks

Mission 4: Instance Specifications of Architecture


Mission: Show the subtypes of the instance is a properly constructed system.

itsRightLeg = RightLegIS
itsLeftArm = LeftArmIS
itsRightArm = RightArmIS

instanceslot_0
itsAnkleSubAssembly = leftAnkleIS

instanceslot_0
itsGripperSubAssembly = LeftGripperIS

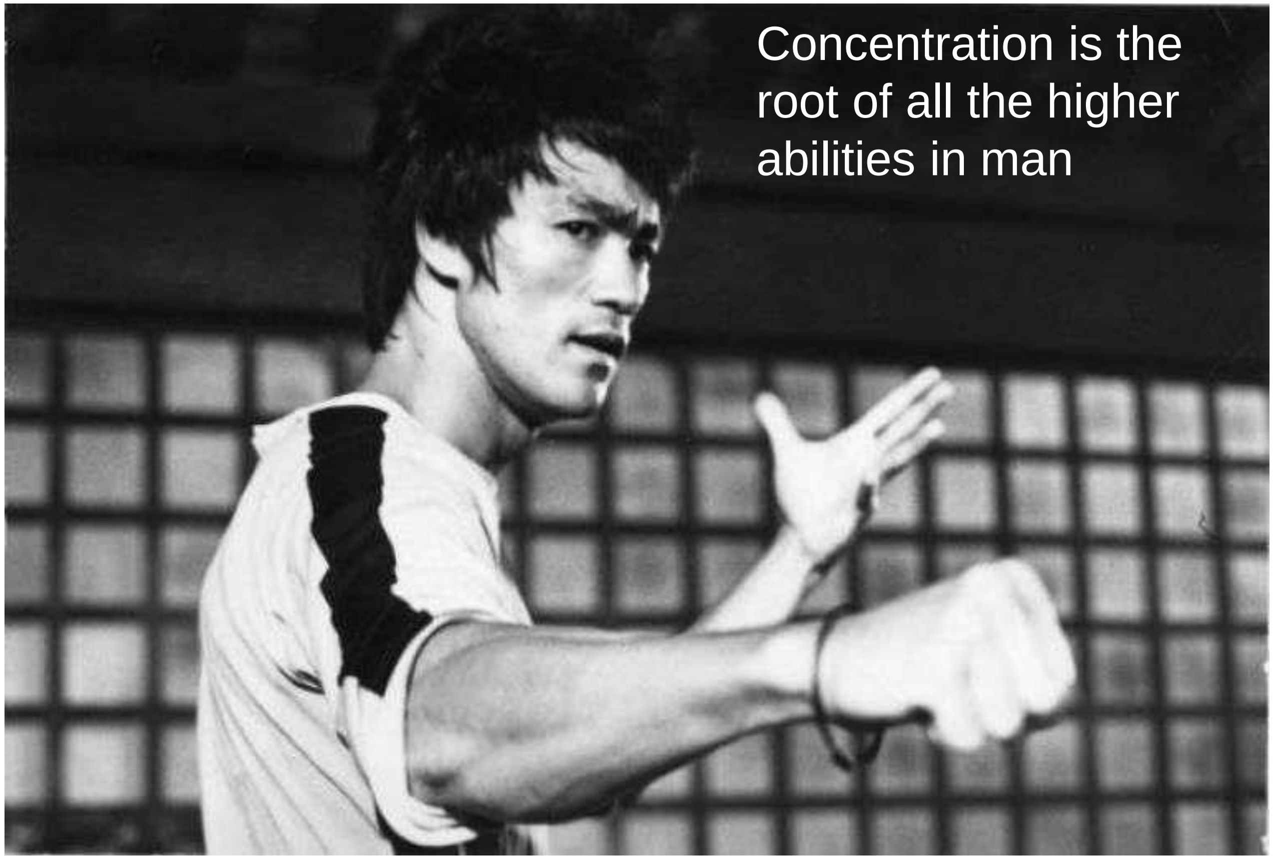
instanceslot_0
itsAnkleSubAssembly = RightAnkleIS

instanceslot_0
itsGripperSubAssembly = RightGripperIS

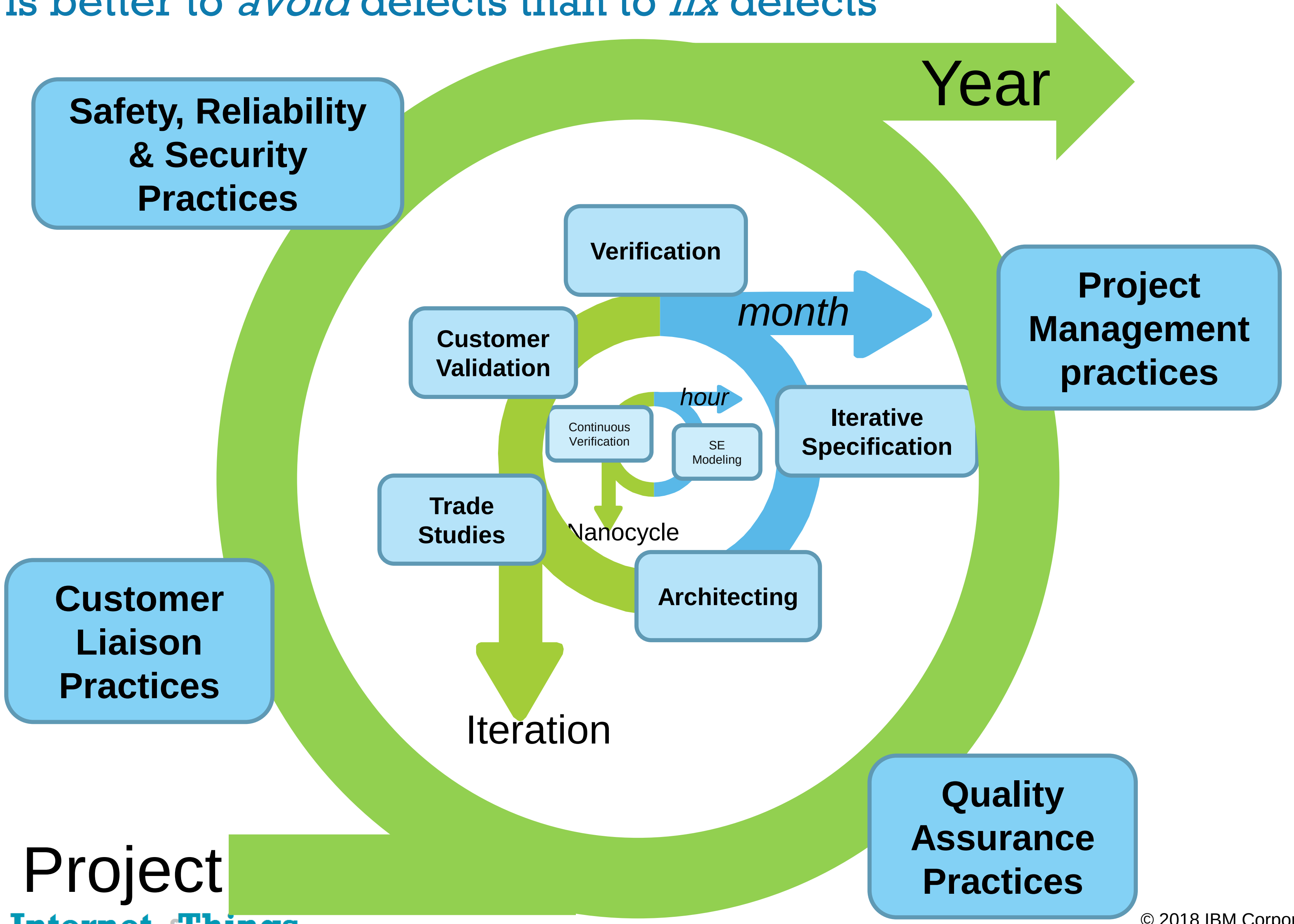
Internet of Things

Wisdom: Right Mindfulness

- Avoid Defects
- Maintain External Consistency
- Manage Traceability
- Focus on Models Goals and Objectives

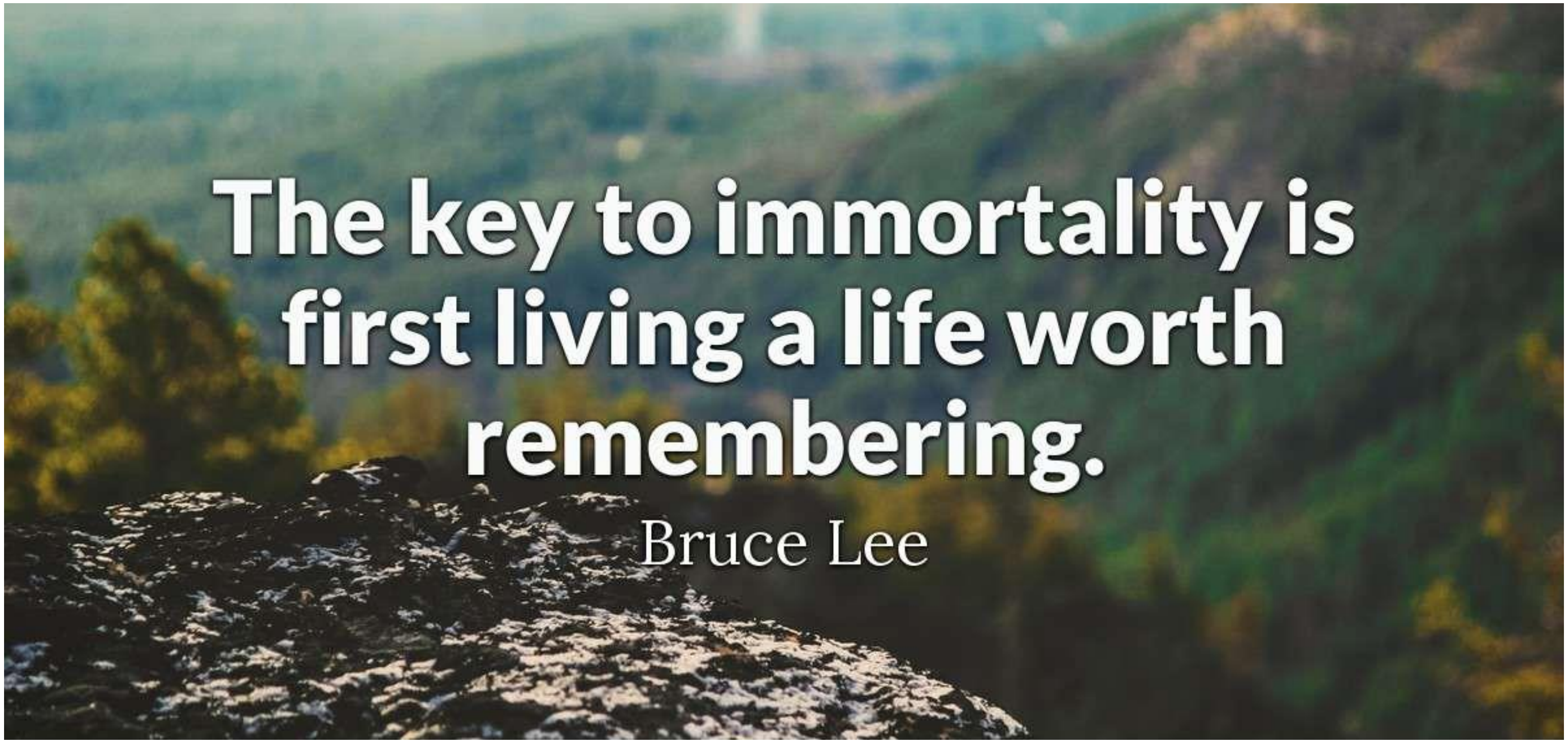

Concentration is the root of all the higher abilities in man

It is better to *avoid* defects than to *fix* defects

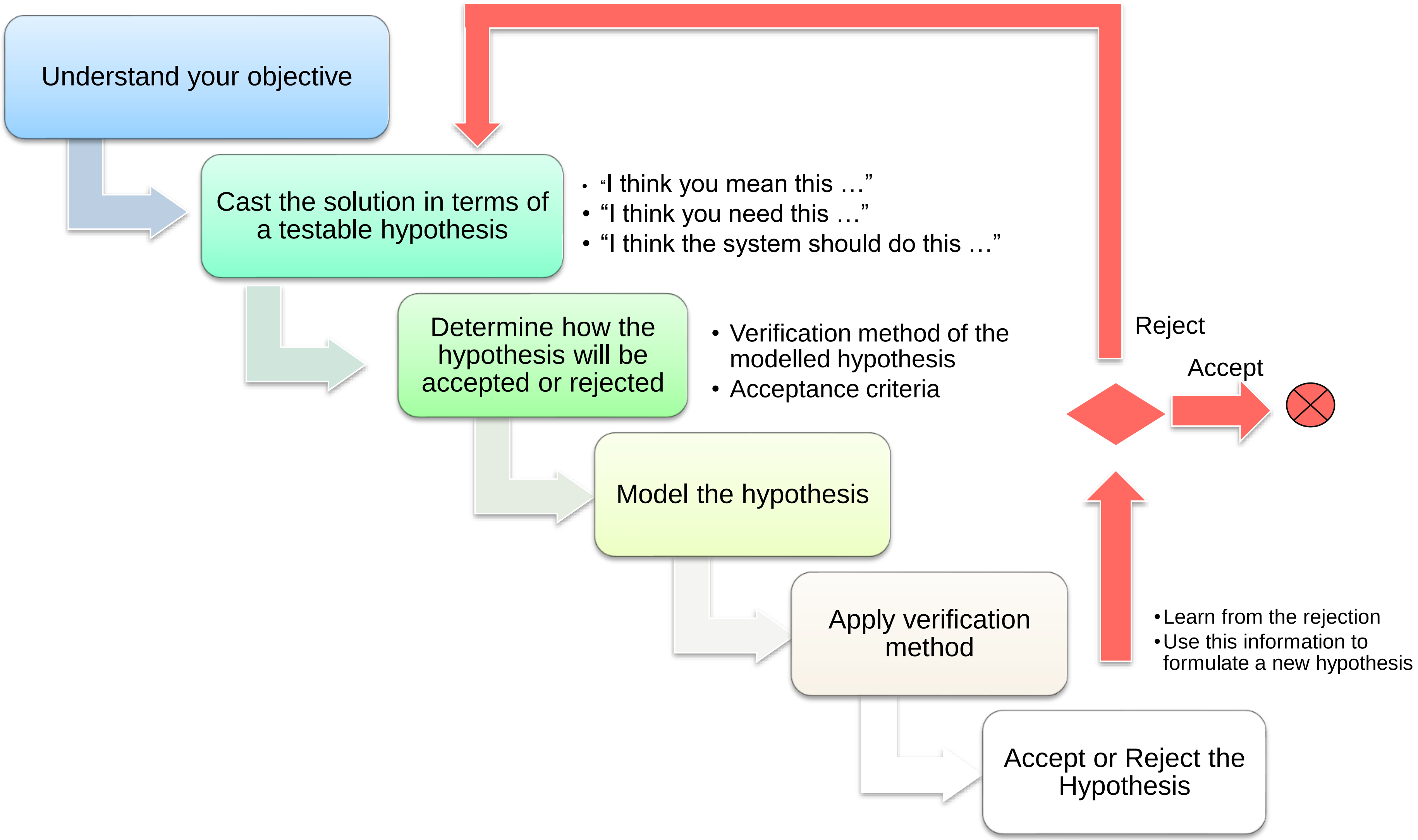


Mental Discipline: Right Effort

- Identify and Remove Model Defects
- Hypothesis-Driven Modeling


The key to immortality is first living a life worth remembering.
Bruce Lee

Hypothesis-Driven Modeling


Understand your objective
Cast the solution in terms of a testable hypothesis
• "I think you mean this …"
• "I think you need this …"
• "I think the system should do this …"
Determine how the hypothesis will be accepted or rejected
• Verification method of the modelled hypothesis
• Acceptance criteria
Model the hypothesis
Apply verification method
Accept or Reject the Hypothesis
Reject
Accept
•Learn from the rejection
•Use this information to formulate a new hypothesis

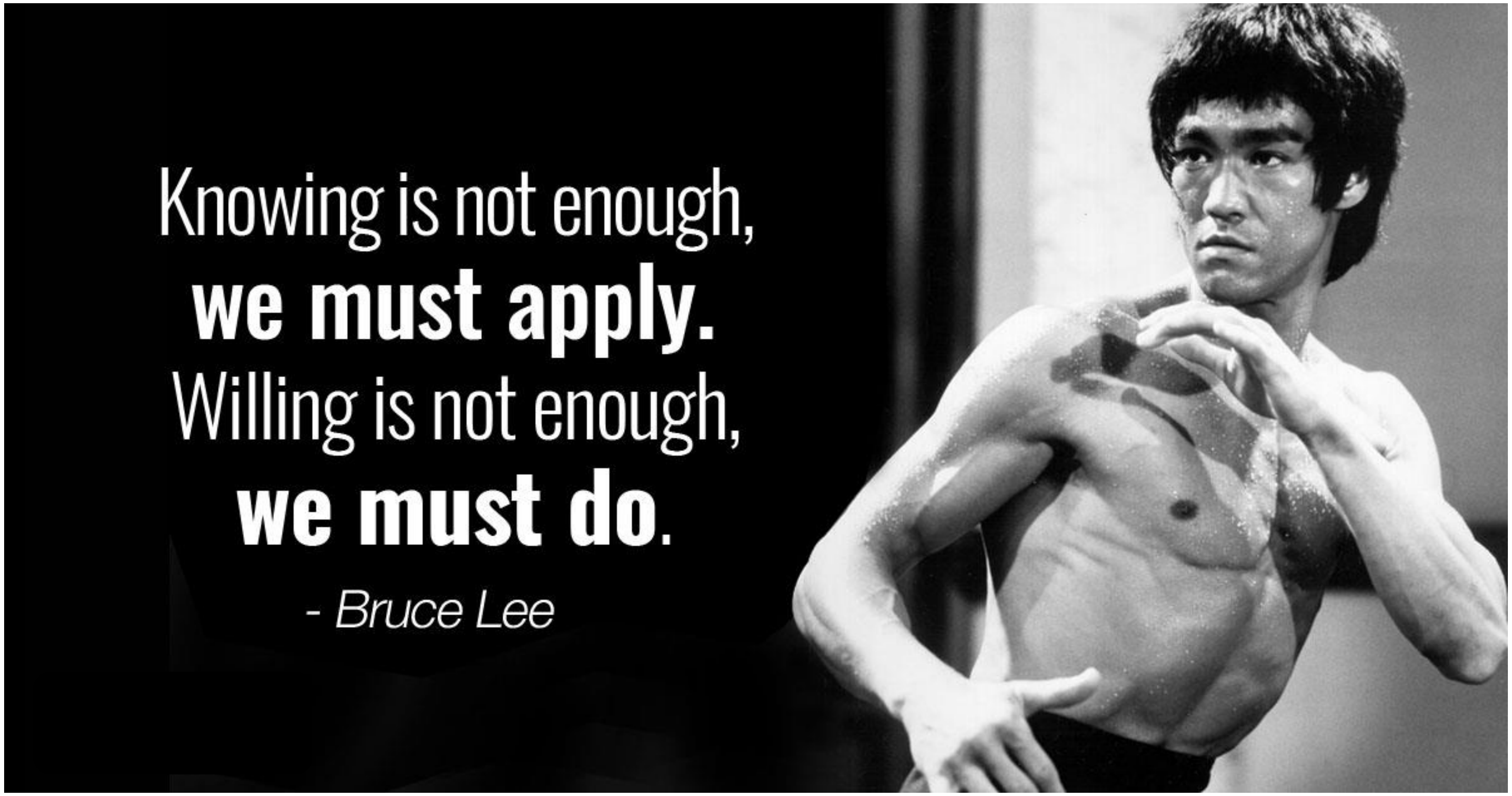
Mental Discipline: Right Views

- Each diagram should have a mission
- Specification vs Design Models

It's like a finger pointing away to the moon.
Don't concentrate on the finger, or you'll miss
all that heavenly glory.
Bruce Lee

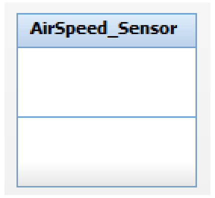
Wisdom: Right Intentions

- Type-Role-Instance Dichotomy
- Useful Descriptions
- Right Conceptualization

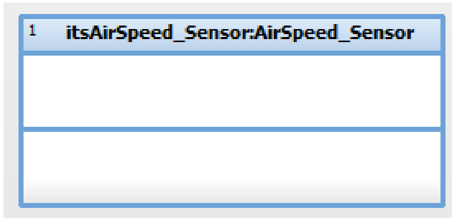


What's a role?

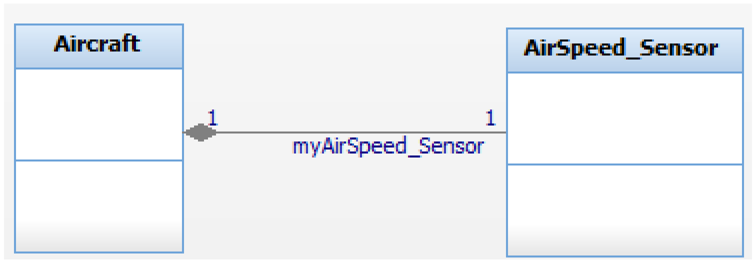
- A **role** is a usage of an **instance** of a **type** in a **context**
- A **type** is a specification of a thing. The type only exists at design-time.



- An **instance** is something that exists at run-time



- A **role** is a usage of an instance of a type in a context. A role exists at design time but is fulfilled at run-time by an instance (part). A part is a role where the context is the owning classifier.



Real-Time Agile Systems and Software Development

Welcome to **www.bruce-douglass.com**

You've found yourself on **www.bruce-douglass.com,** my web site on all things real-time and embedded.

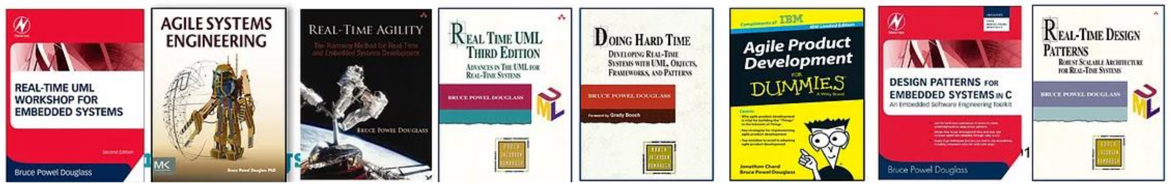
On this site you will find papers, presentations, models, forums for questions / discussions, and links (lots of links) to areas of interest, such as

- Developing Embedded Software
- Model-Driven Development for Real-Time Systems
- Model-Based Systems Engineering
- Safety Analysis and Design
- Agile Methods for Embedded Software
- Agile Methods for Systems Engineering
- The Harmony agile Model-Based Systems Engineering process
- The Harmony agile Embedded Software Development process
- Models and profiles I've developed and authored
- List and links to many of my books.

The menu at the top of each page either takes you to the relevant page or to a list of relevant pages.

There is even a members only site for those who want to access to even more stuff. I teach and consult on all these topics - see my About page.

Policy on Using These Materials
All materials on this web site are free for reuse and distribution, provided their source (me or this web site) is appropriately attributed. I retain sole copyright.


REAL-TIME UML WORKSHOP FOR EMBEDDED SYSTEMS
AGILE SYSTEMS ENGINEERING
REAL-TIME AGILITY
REAL TIME UML THIRD EDITION
DOING HARD TIME
Agile Product Development FOR DUMMIES
DESIGN PATTERNS FOR EMBEDDED SYSTEMS IN C
REAL-TIME DESIGN PATTERNS