

# More Insights without More Effort

Graham McLeod  
Owner/Chief Architect

inspired!

# EA Fuel Consumption



Ford Pickup, Circa 1960. Approx 23l/100km

# EA Fuel Consumption



Tesla Electric 2012 - Free electricity after purchase

# Value



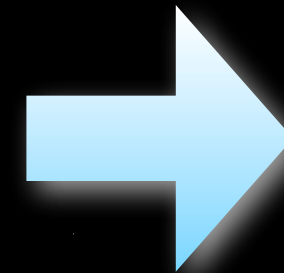
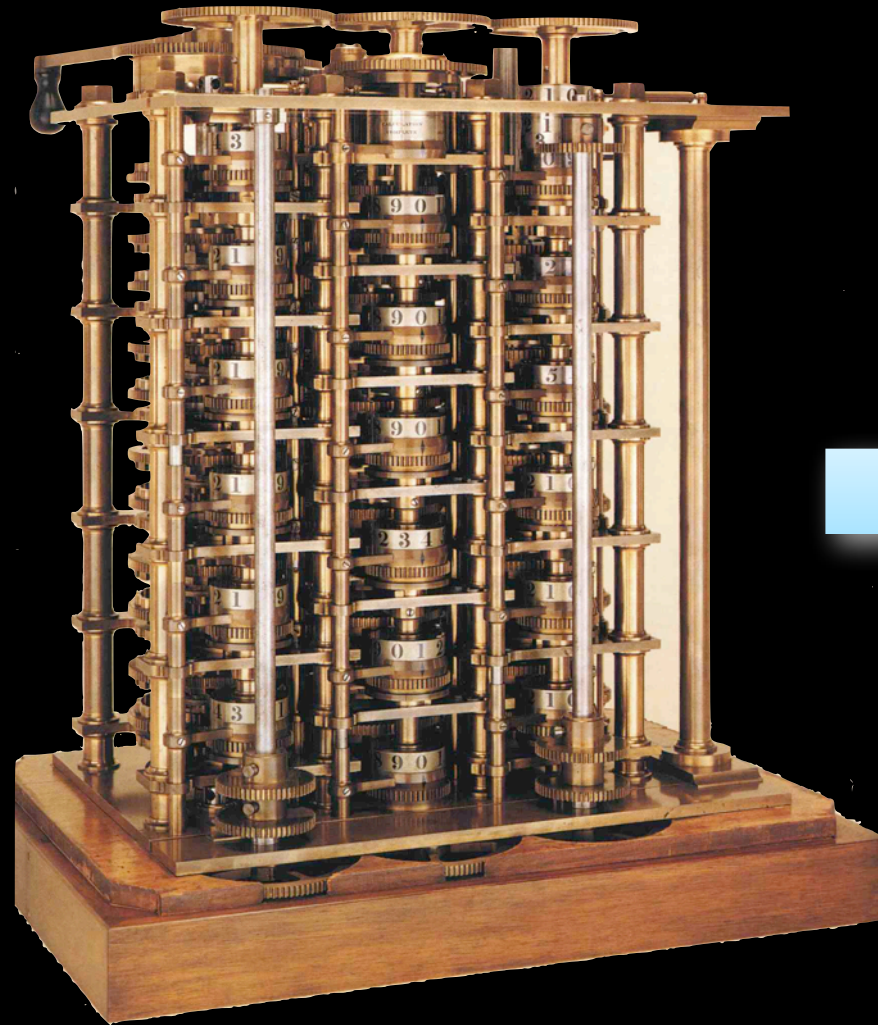
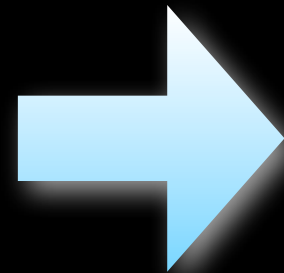
Stakeholder Time



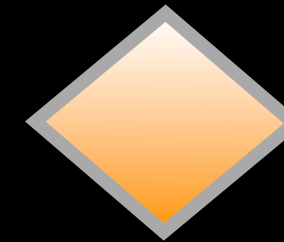
EA Team Effort



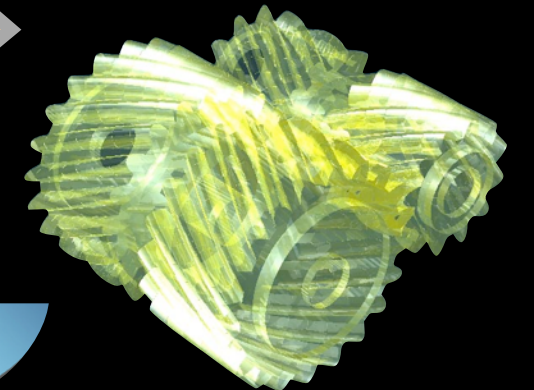
EA Repository



Insights



Decisions



Alignment



Integration



Automation



# Effort to...

Decide upon required information

Identify sources of information / stakeholders

Collect, capture and validate information

Report, analyse, collate, share

Derive insights

Keep up to date and consistent



Effort is a bad thing:  
It takes money,  
resources  
AND time

# Reduce Effort by...



Comprehensive Integrated Meta Model across Domains (of EA) and related Disciplines

(e.g. Programme Mgmt, Requirements Mgmt, Change Mgmt etc.)

Integrated Shared Repository

Reuse of information across domains, models, representations

Consistent information between physical / logical / conceptual layers

Automating analysis, insight, collation into required output formats

# Add Value By..

Higher integration (meta model, collect through any input medium)

Inferencing

$a \Rightarrow b \Rightarrow c \quad a \Rightarrow c$

Hierarchical summary

Auto population/layout

Difference Analysis



Derived Values, Algorithms

Consistent visual language

Principles

Rich pictures from same semantic model as technical models

Polymetric Models







# Human Visual System

Pattern Oriented

Optimised for movement

Optimised to identify the unusual



# Visual Language

Relationships

Line Style →

concept ('kɒnsɛpt )  
concept ('kɒnsɛpt )  
noun  
1. an idea, esp an abstract idea ⇒  
*"the concepts of biology"*

Vector  
Symbol

Compose into  
Diagrams

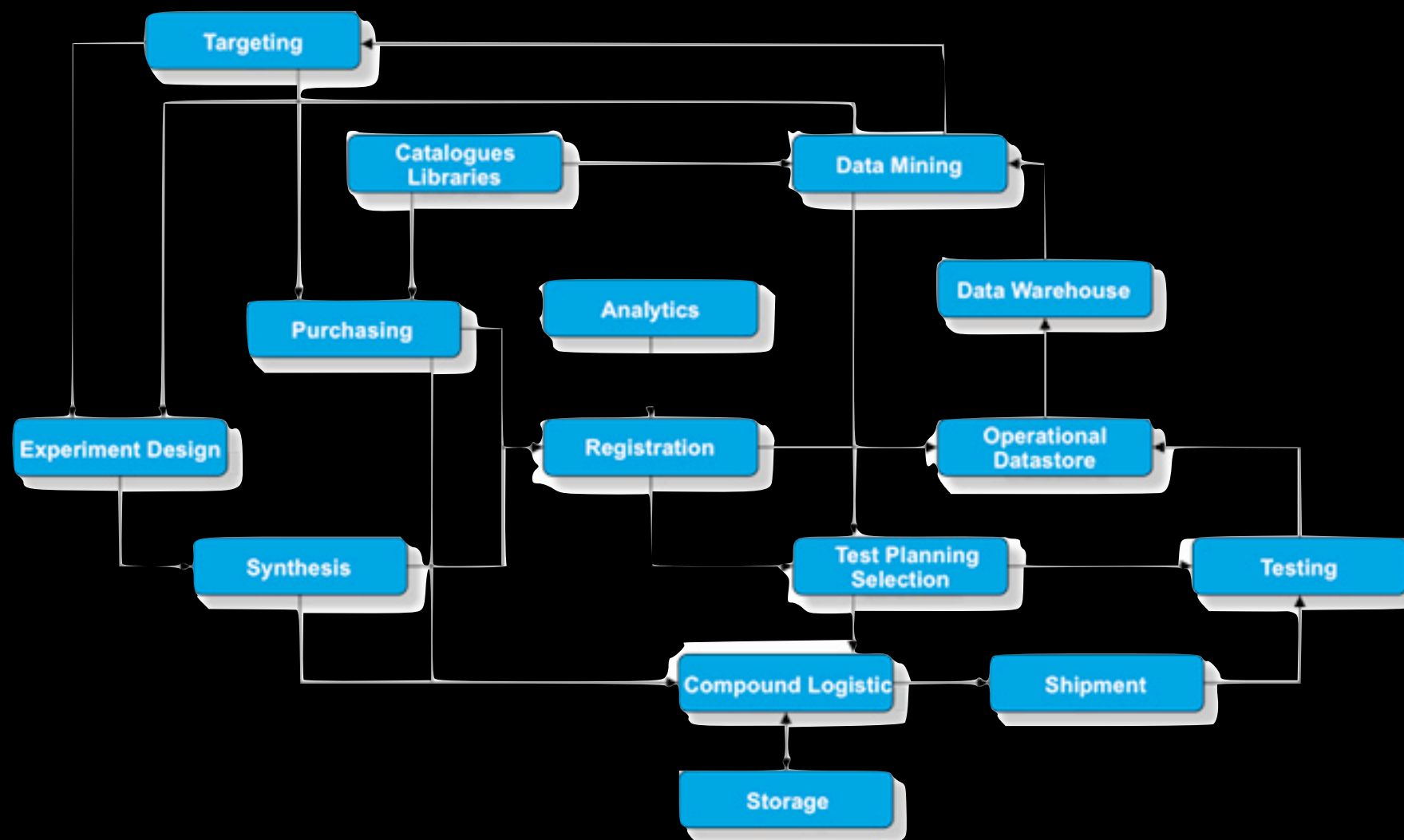


Compose into  
Rich Pictures

Icon / Pictogram



# Visual Language



✓ Standardised =>  
Shared  
Understanding

✗ All nodes of similar  
type look the same

✗ No insight re key  
items of interest or  
requiring attention

✗ “Camouflage”  
Models

e.g. UML, Archimate, BPMN

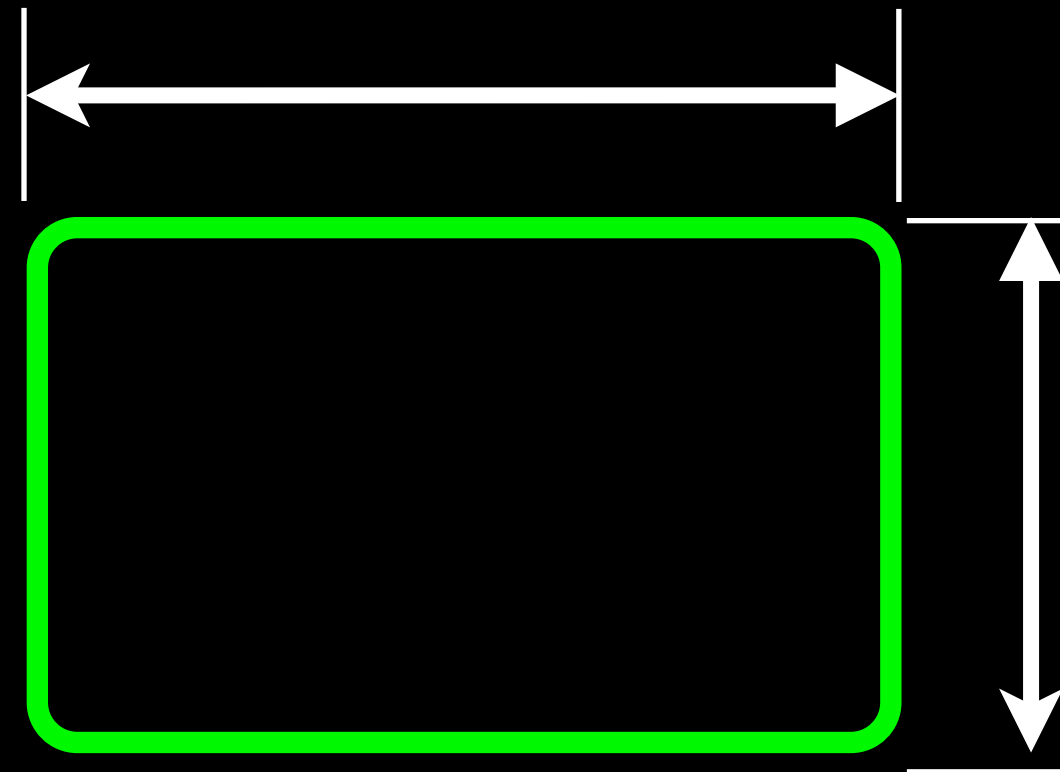
# Polymetric Diagramming

Not an entirely new idea

Defined formally in a Software Re-engineering Thesis in 2003 (Lanza at Univ of Bern)

Modifying symbols in diagrams/models based upon properties of the objects they represent

I am researching application/extension of these techniques to a wide range of models, especially those in EA and Business Architecture





# Polymetric Examples in SW Eng.

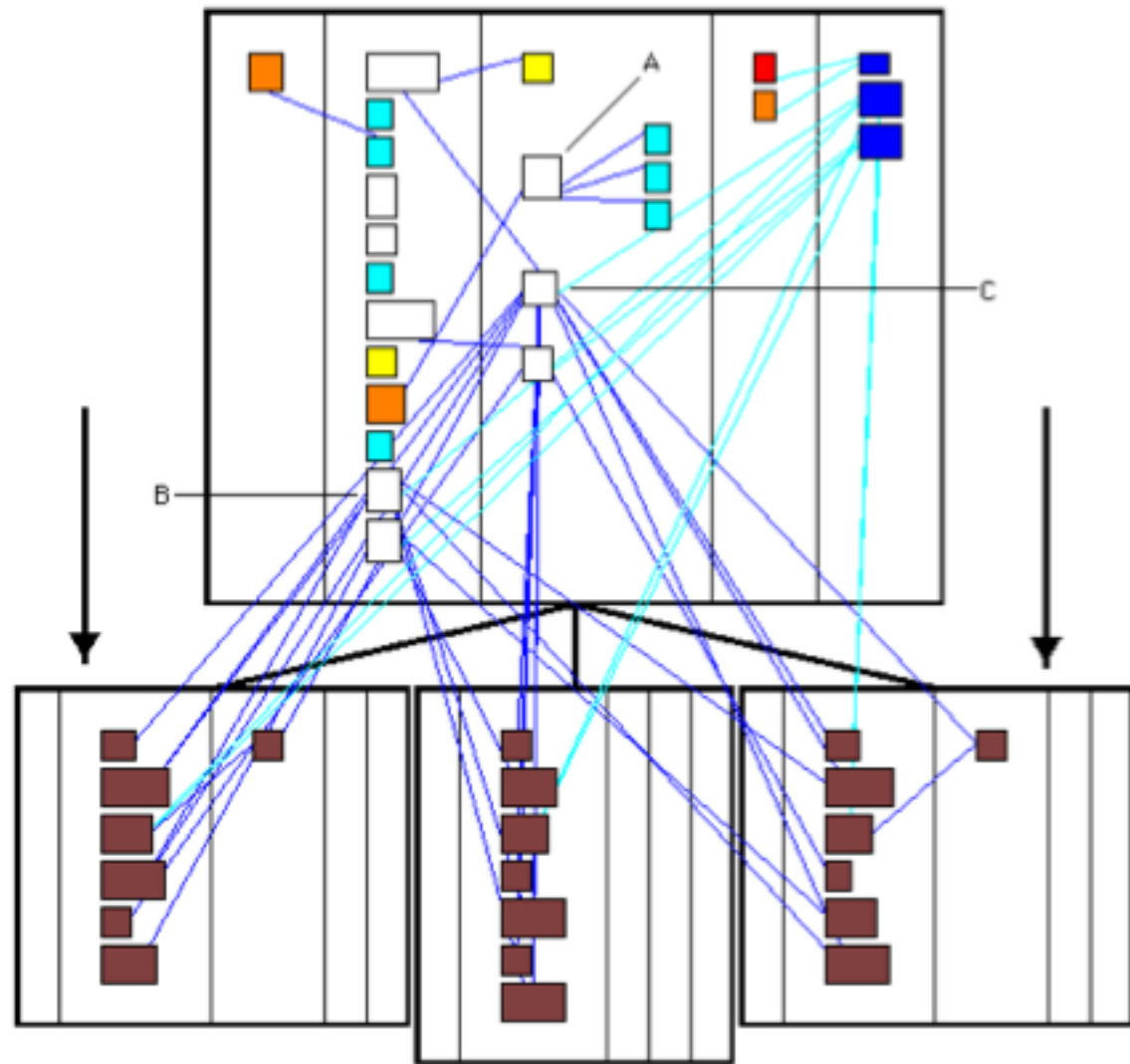


Figure 5.21: A class blueprint visualization of an inheritance hierarchy with the class *JunColorChoice* as root class. Each subclass contains a pure *Siamese Twin* pattern.

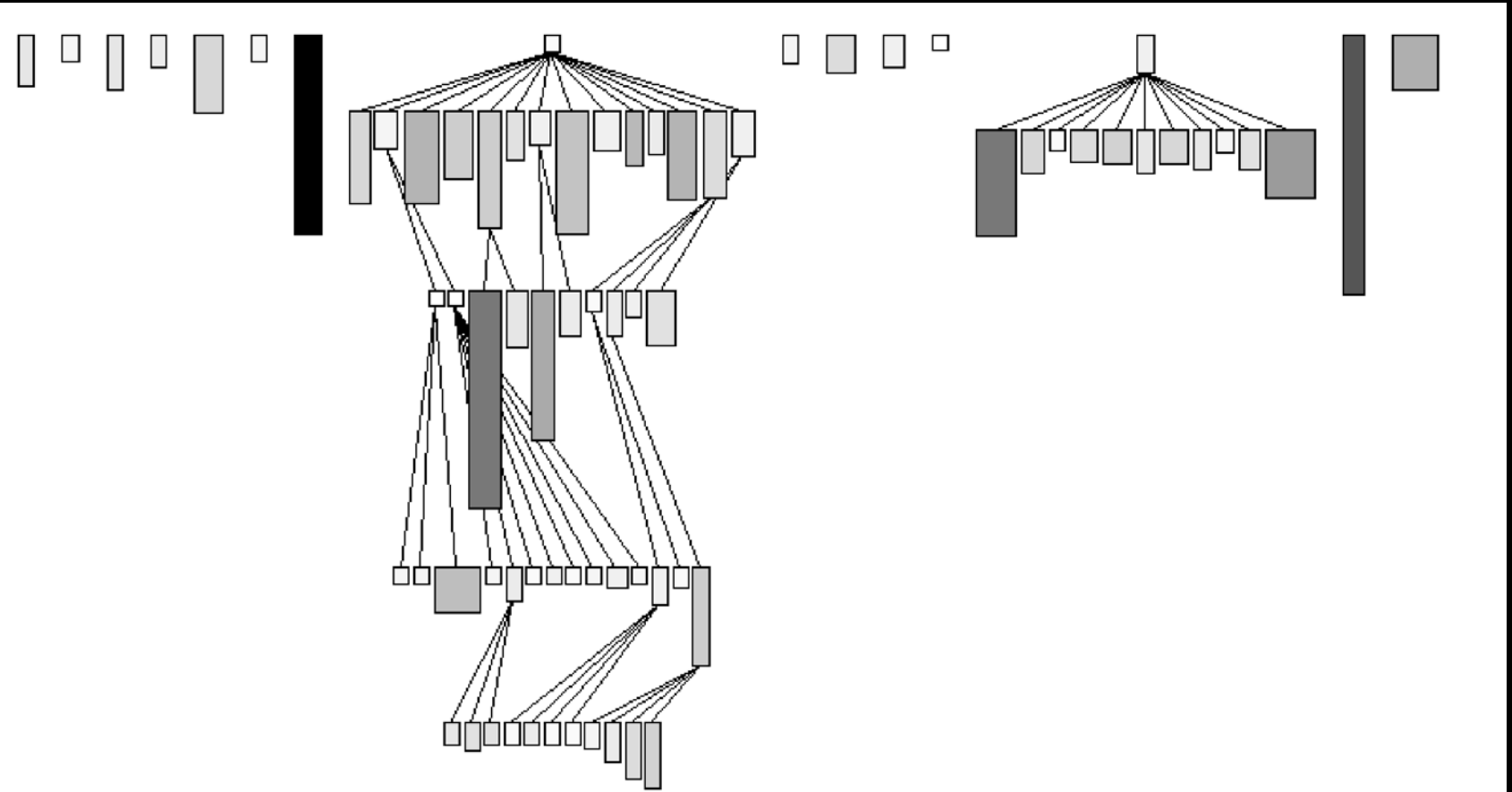
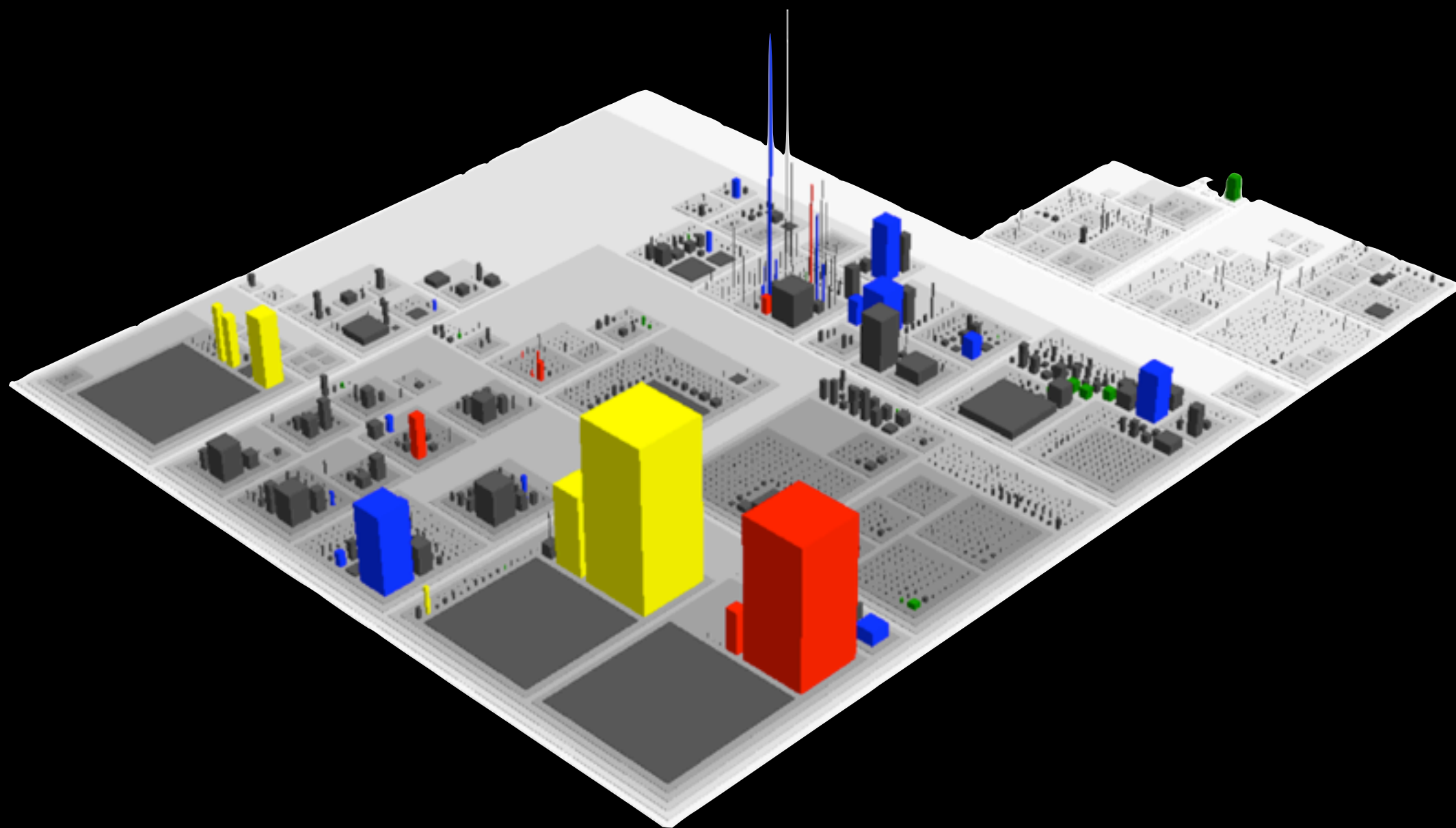


Figure 3.2: The SYSTEM COMPLEXITY view. This visualization of classes uses a tree layout. The edges represent inheritance relationships. The metrics we use to enrich the view are NOA (the number of attributes of a class) for the width and NOM (the number of methods of a class) for the height. The color shade represents WLOC (the number of lines of code of a class).

# Code City Example





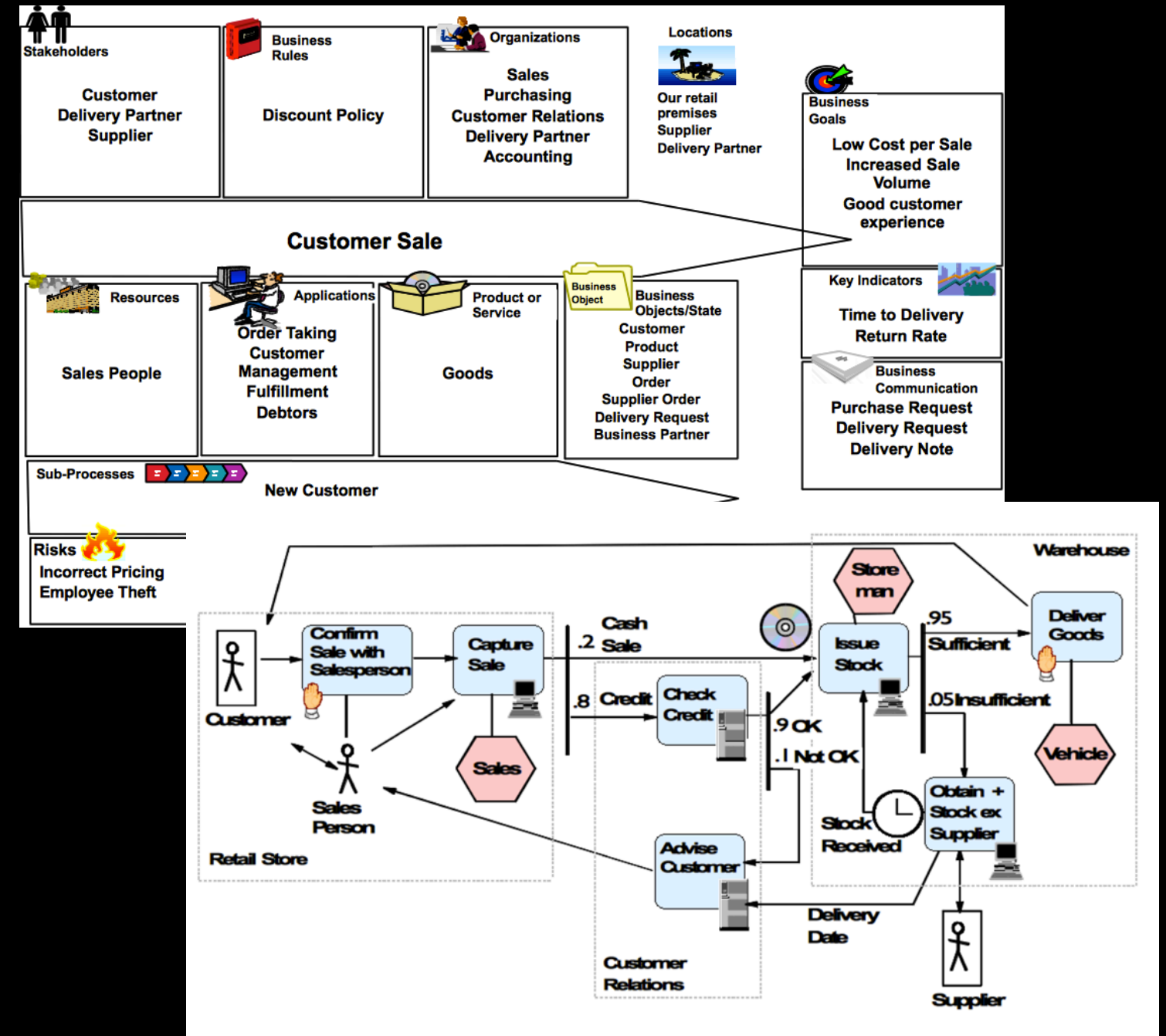
# Model Types

Associate concepts, relationships, visual language

Definition of interesting properties, relationships, statistics to modify symbols

Definition of algorithms to modify symbols sensibly

Layout Specs



# Modifications

Dimensions (Width, Height, Depth)

Shape

Colour (but there are limitations)

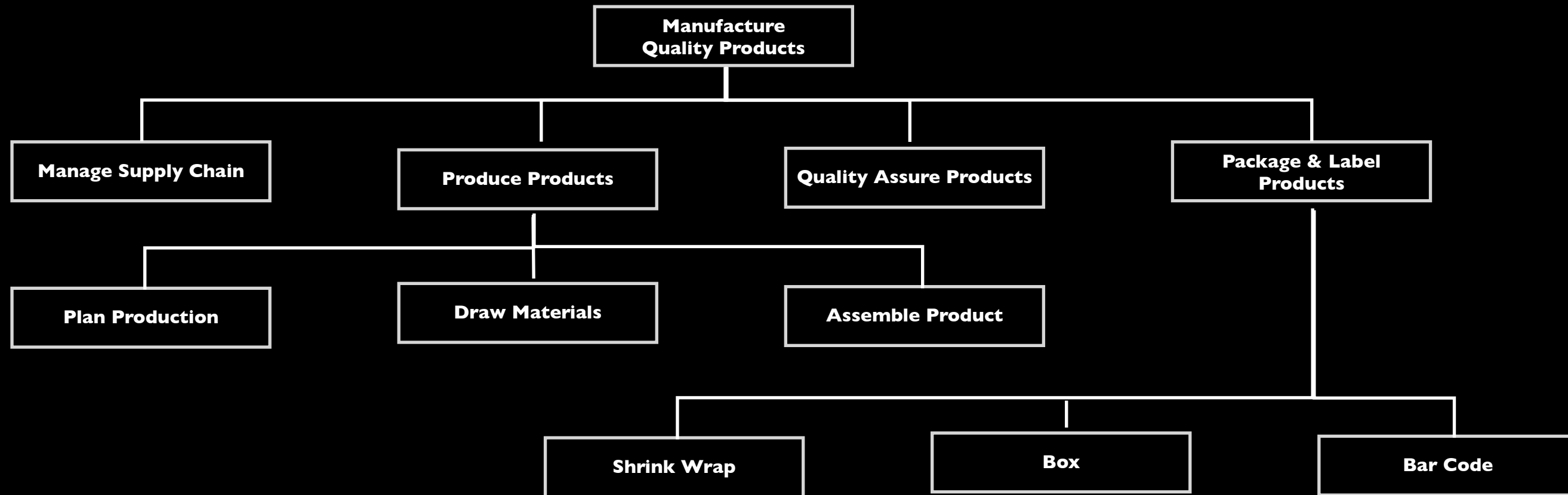
Annotation

Position

Style (e.g. Border Width, Shading, Line Style)



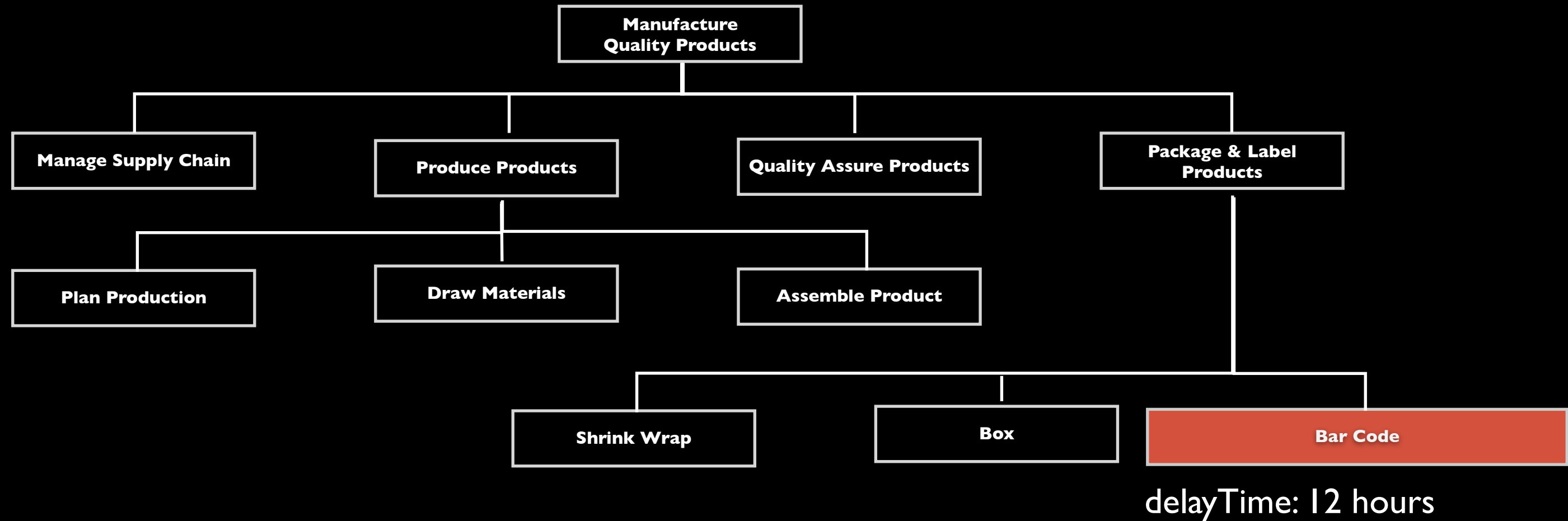
# Function Model



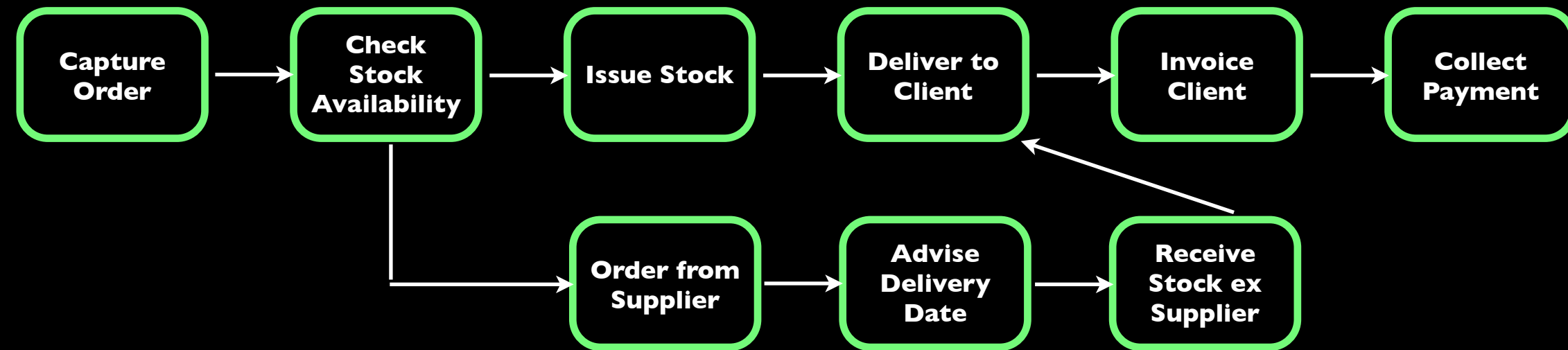
Add: delayTime to: width show: delayTime  
Colour: red where: [ (actualStaff / budgetedStaff) < .7]



# Function Model

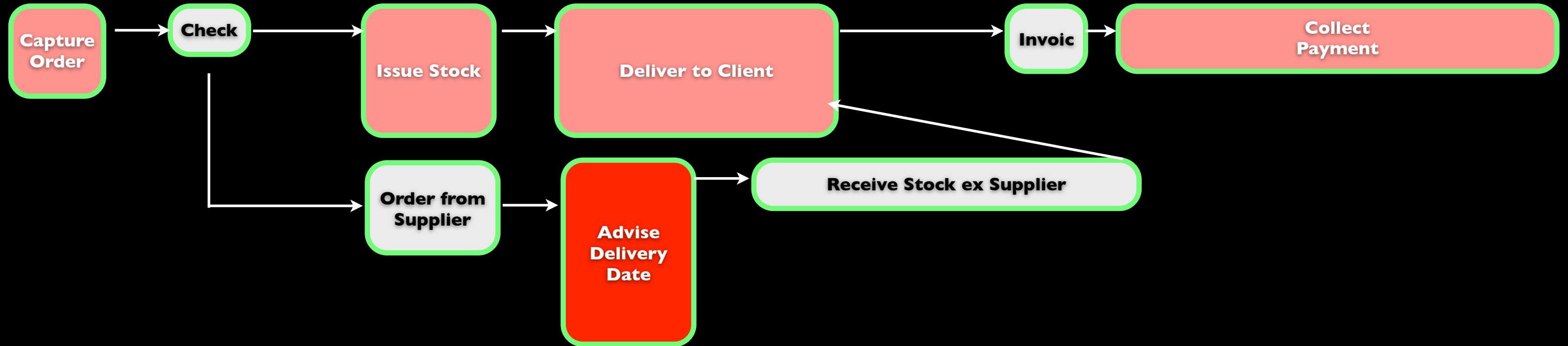


# Process Model



Map: duration to: width. Map: cost to: height.  
Map: resourceConsumption to: redness shades: 4

# Process Model



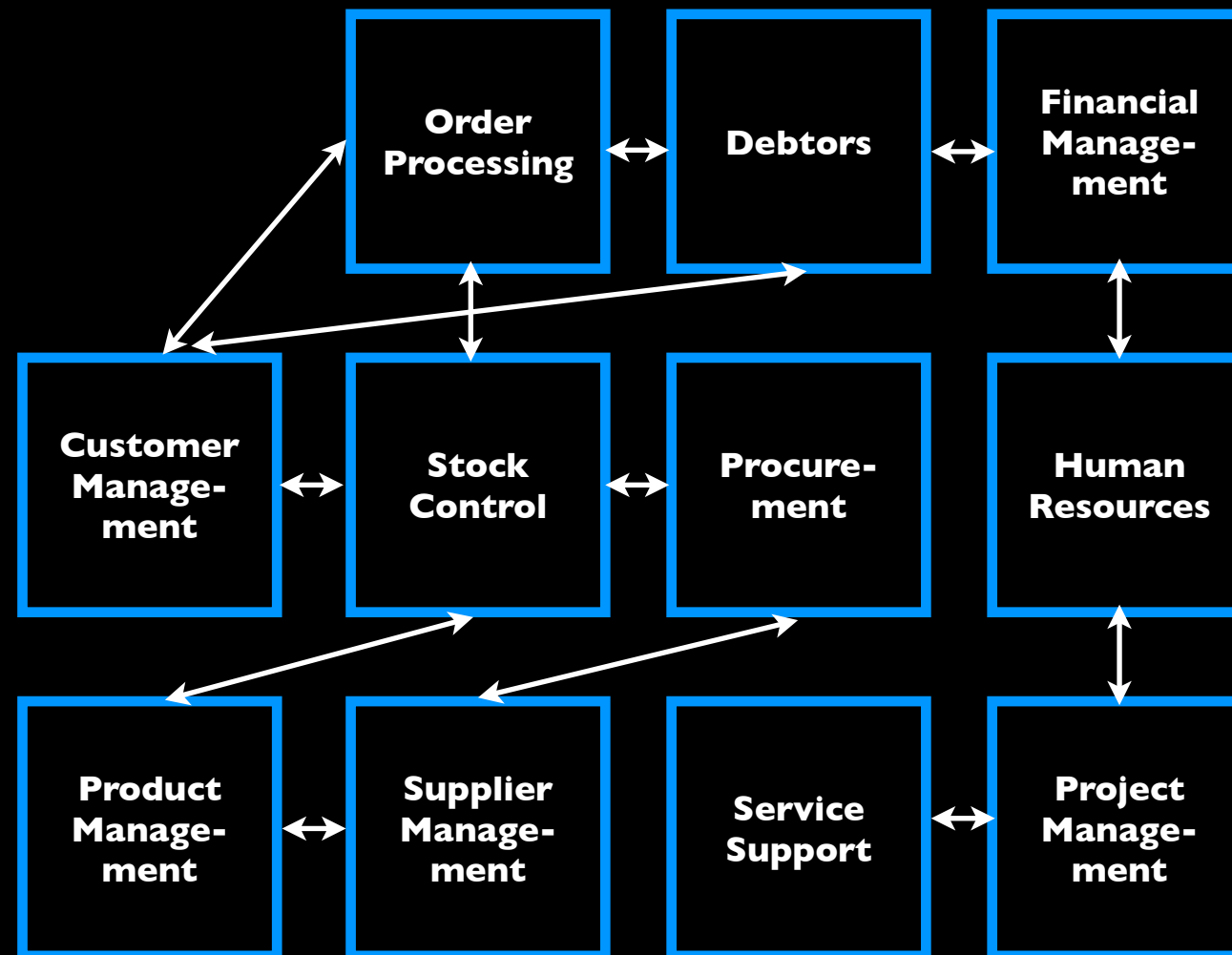
What is the longest activity?

Which is the most resource intensive activity?

Which is the most expensive activity?

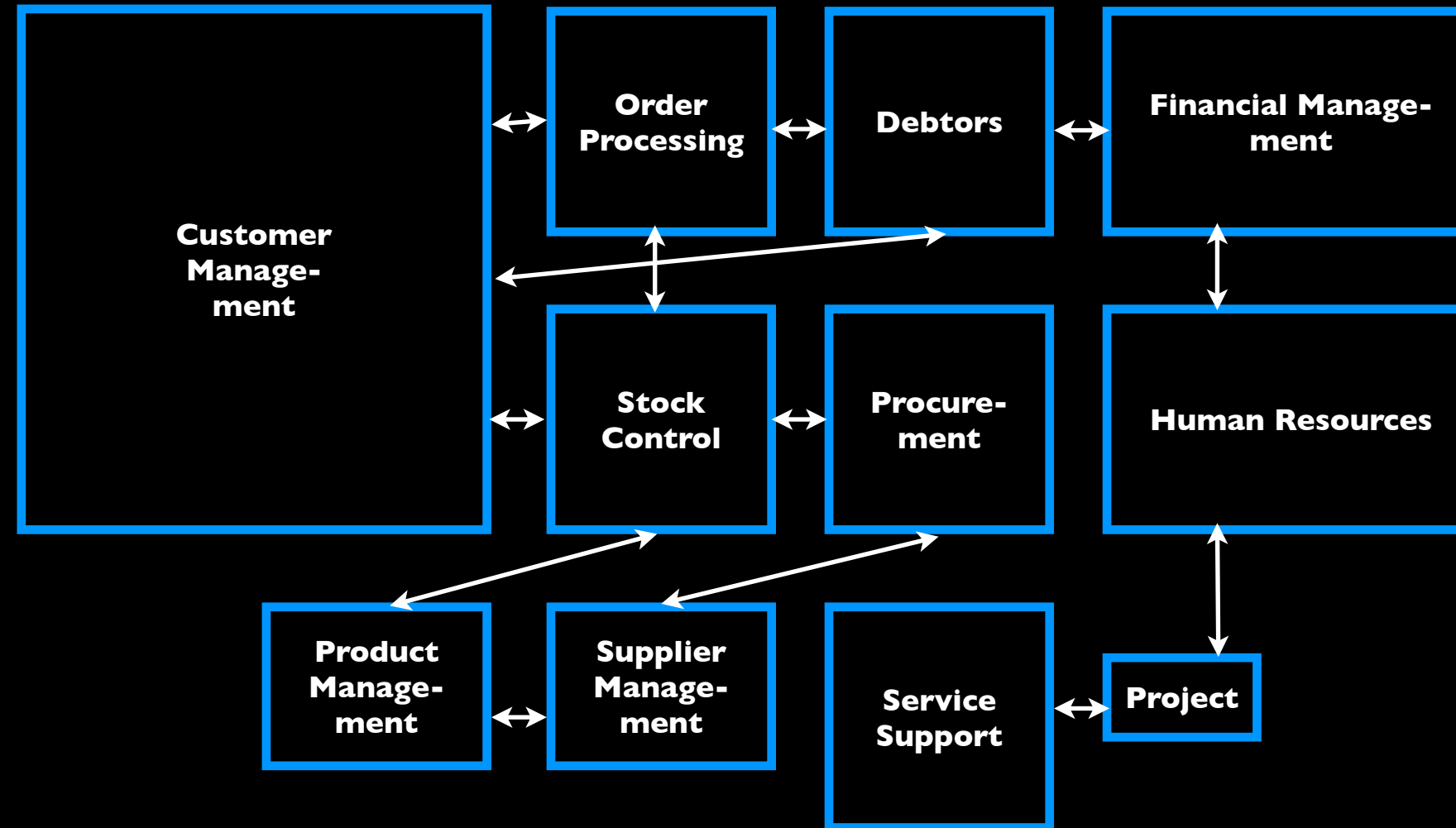


# Application Map



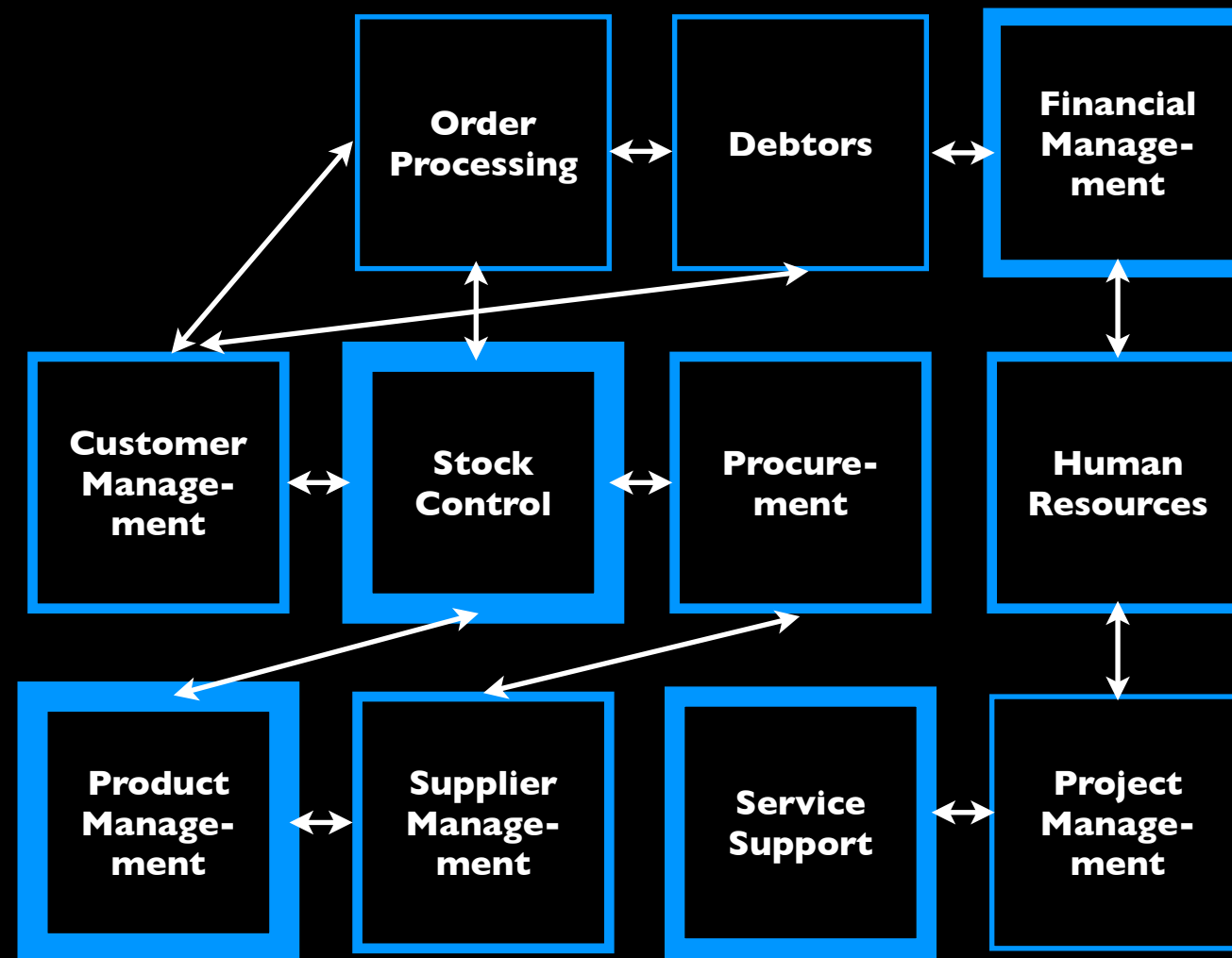
Map: investment to: size

# Application Map



Map: nonStdInterfaces to: border range: 1 to: 12

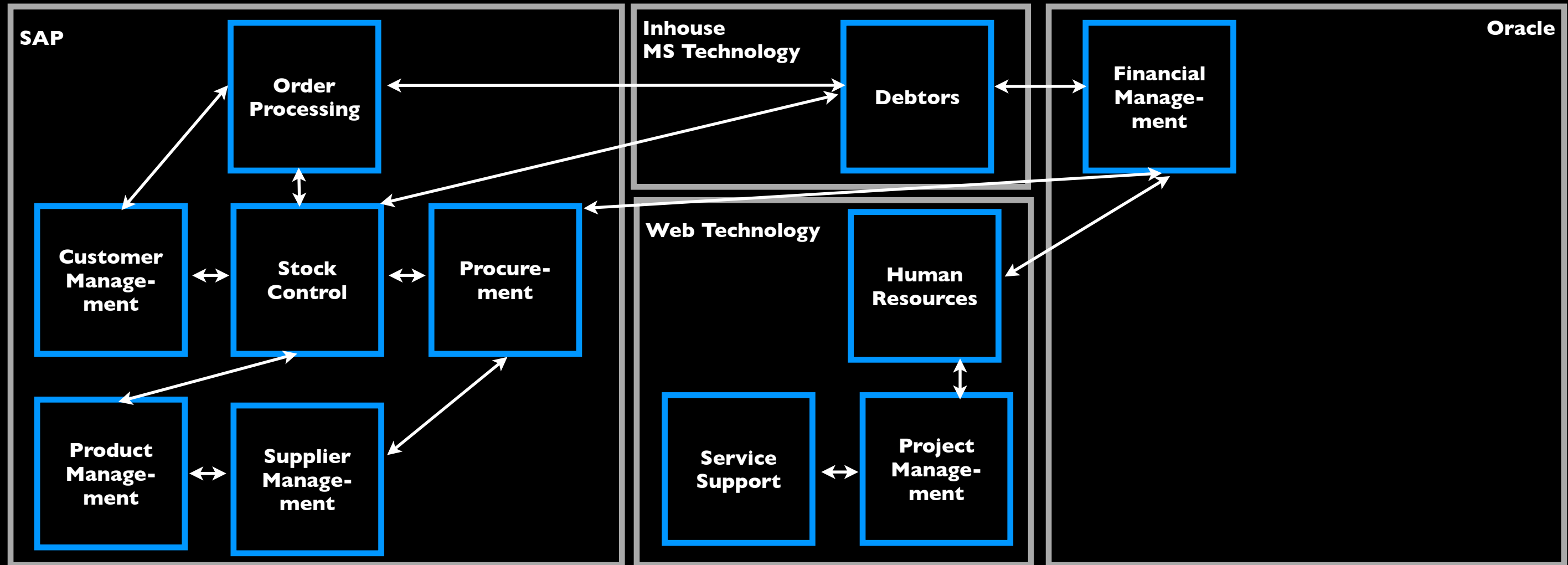
# Application Map



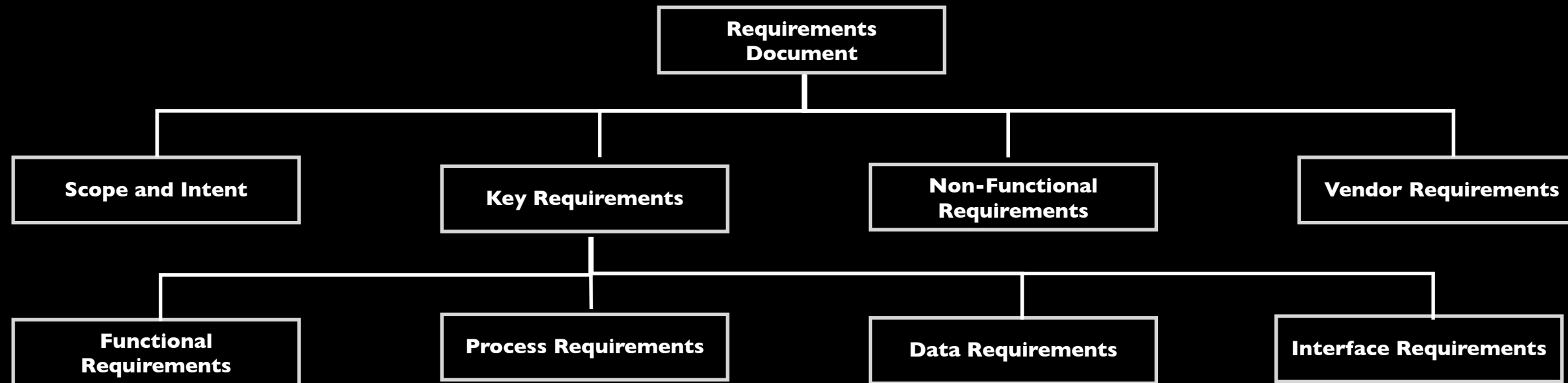
ClusterBy: primaryTechnology



# Application Map

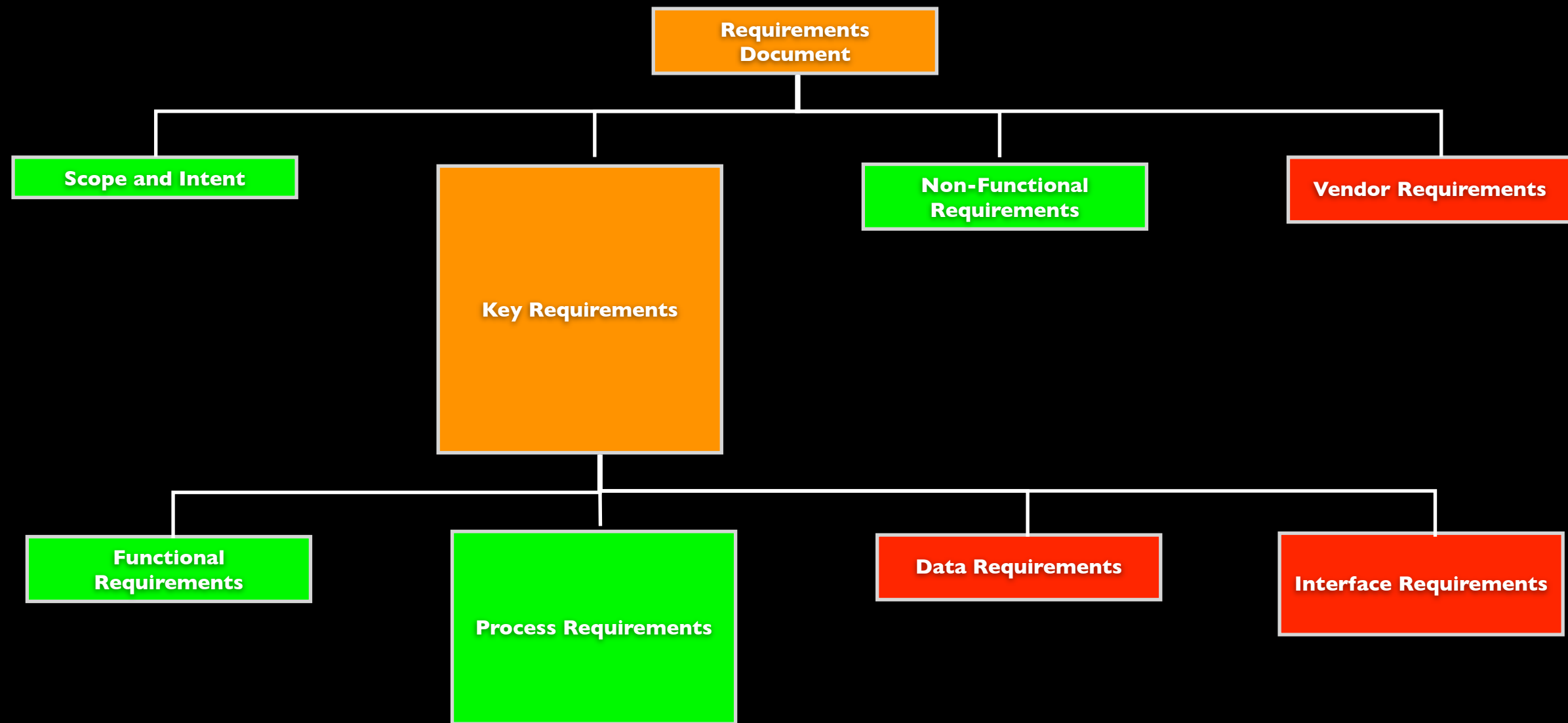


# Product Breakdown



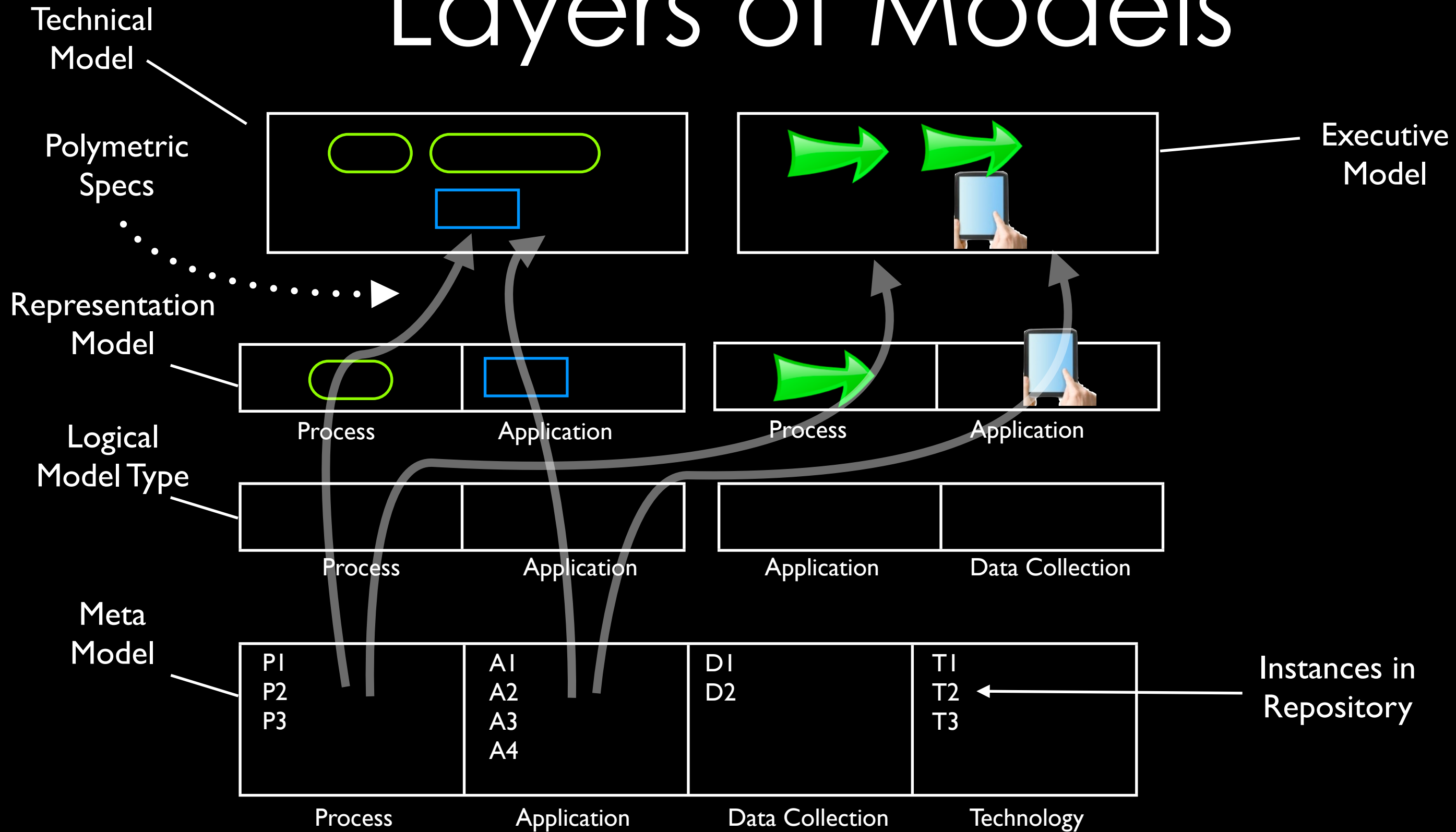
Map: effort to: height.  
Map: completion% to: redGreen.

# Product Breakdown



Map: effort to: height.  
Map: completion% to: redGreen.

# Layers of Models





# Prototype Environments

Pharo - an Open Source Smalltalk

Mondrian and Roassal - Graphics Libraries

Moose - Meta Modeling and Software Analysis / Reengineering toolkit

EVA Graphical Modeler - Inspired Graphical Modeling Environment for  
EVA Netmodeler

# Demo, Scripting

The screenshot displays the Mondrian Easel application interface. At the top, there is a menu bar with options: "Mondrian", "Export", "Previous scripts", and "Examples". Below the menu bar is a large area showing a complex class hierarchy diagram with nodes and edges. The diagram is rendered in a light gray color on a white background. Below the diagram, there is a scripting interface with a text area containing the following code:

```
view shape rectangle  
  height: #numberOfMethods|  
view nodes: classGroup.  
view edgesFrom: #superclass.  
view treeLayout
```

To the right of the scripting area, there is a panel titled "classGroup->All model classes (2713 FAMI)". Below the scripting area, there is a "Generate View" button.

# Conclusions

Comprehensive Conceptual Integrative Meta Model

Separation of Logical and Physical Model Types

Representation in Vector Symbols and Rich Pictures

Visual Summaries & Difference Models

Polymetric Representations and Flexible Scripting DSL Provide lots of Power and new Insights without more effort

# More Work

Refinement of ideas through prototyping and application

Field testing and validation

Integration into production strength tools

Want to get involved? Help?



# References

Lanza, M. (2003). Object-Oriented Reverse Engineering. PhD Thesis Univ of Bern, Switzerland

Camo Lion Photo: [https://fbcdn-sphotos-c-a.akamaihd.net/hphotos-ak-frc3/971650\\_533671470011763\\_1499037240\\_n.jpg](https://fbcdn-sphotos-c-a.akamaihd.net/hphotos-ak-frc3/971650_533671470011763_1499037240_n.jpg)

Code City example: <http://www.moosetechnology.org/docs/visualhall>

Mondrian Scripting Example: <http://www.moosetechnology.org/docs/demos>

# Graham McLeod

 [graham@inspired.org](mailto:graham@inspired.org)

 [www.inspired.org](http://www.inspired.org)

  <http://grahammcleod.typepad.com/>  
(blog)

 LinkedIn: Graham McLeod



# inspired!

“The best way to predict the future is to invent it”  
*Alan Kay*

Active since 1991, Inspired works closely with clients across many industries and regions to bring greater benefits to organizations and society through the creative use of technology

We are thought leaders in Business Architecture, Process Architecture, Methods Engineering, Business Application of Semantic Technologies, Advanced Systems Delivery and Agile Project Management

We provide high quality consulting, training, research and EA tooling