A failure to communicate...

Start Integrated, Stay Integrated
Complexity = more & more brains involved
Doing more, with more constraints, less time

Dealing with very demanding customers

Interacting with more people

Communication & Information Management Problem
Systems Engineering
Doing more, with more constraints, less time

Dealing with very demanding customers

Interacting with more people

Documents don’t scale…
Doing more, with more constraints, less time

Dealing with very demanding customers

Interacting with more people

Models without common methods/grammar can’t communicate meaning
Doing more, with more constraints, less time

Dealing with very demanding customers

Interacting with more people

Continuous communication requires:

- architecture/system modeling with embedded standard process/method
  - Blueprint with requirements for programs
  - integrated with scalable PLM
  - with access to standard services
  - sharable with suppliers without compromising IP

...enables continuous integration/communication
Integrated MBSE Vision
How integrated MBSE enables communication…

Minimum Turn Radius: 24 ft.
Automatic Dry Pavement Braking Distance at 60 MPH: 110 ft. 90 ft

Hydraulic Fluid: SAE 1340 not-compliant

Power Rating: 18 Amps
MBSE Process...
Shift left...

SoS
Scope
Constraints
Interactions

Use Cases & Demands
Solution (wished by Customer)

Product
Concept Design
Selected Product Requirements

Derived System Requirements

System Design

Derived mechanical Requirements

Derived EE Requirements

Derived Software Requirements

Derived By Parts Requirements

Product Definition

Selected Product Requirements

Derived System Requirements

Derived mechanical Requirements

Derived EE Requirements

Derived Software Requirements

Derived By Parts Requirements

Product Implementation

Mechanical Design
Components (virtual)

EE Design
Components (virtual)

Software Design
Components (binary)

Purchase
Buy Parts

Component

Component (virtual)

Component (binary)

Product Build

Manufacturing BOM

As Maintained BOM

Solution (wished by Customer)
Failure to communicate architecture downstream…
Late discovery of integration problems
Late discovery costs money…

…throws off program balance
You plan for this communication failure...

...you plan up to ½ your program schedule for this system integration cycle
A failure to communicate costs money…

- ~47 million automotive recalls in the US last year
- NHTSA est. $100/recall per vehicle; that’s $4.7 billion in direct costs fixing the problem

…many of these are failures to comply with requirements & regulations

March 2, 2020: Ford & Volvo Door latch [1.4m, $140m]
Hidden costs from communication failures…
Solving the same problems over & over

Problem resurface metric: how long does a problem once solved take to come back

- Auto: ~3 years
- High Tech ~6 mo.
- Aero ~15 years

Cross-Domain problems result from:
- Siloed/Disconnected Decisions
- Form follows function, Problems follow functions
- Everyone involved, including purchasing
- Disconnected requirements
- Uncommunicated change
- Happen at domain/organizational boundaries
- Migrates with people (overt or covert)
- Missing/disconnected product architecture

“Water on the knee”
### MBSE Maturity

#### Disconnected Communication with documents
- Integrated Variation rules
- Integrated RAMS, continuous risk assessment/alarms with dashboards

#### Continuous Communication with models
- Functions/logical allocation drives interface definitions
- Logical architecture with allocation with traceability
- Reusable parameter library with traceability

#### Communication Continuum

<table>
<thead>
<tr>
<th>System Modeling/Architecture</th>
<th>Disconnected Communication with documents</th>
<th>Continuous Communication with models</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLE/Configuration (variation)</td>
<td>Integrated variation rules</td>
<td>Integrated RAMS with FMEC dashboards</td>
</tr>
<tr>
<td>Technical Risk (RAMS, cost,…)</td>
<td>Risk documents, spreadsheets</td>
<td>Integrated RAMS, continuous risk assessment/alarms with dashboards</td>
</tr>
</tbody>
</table>

#### Interface Management
- ICD in docs
- Managed interfaces
- Standard-based interface library
- Reused interfaces
- Functions/logical allocation drives interface definitions

#### Logical Modeling
- Logical description documents
- Logical hierarchy models
- Isolated logical behavior models
- Integrated logical behavior models
- Logical architecture with allocation with traceability
- Reusable parameter library with traceability

#### Parameter Management
- Unmanaged spreadsheets
- Managed spreadsheets
- Parameter library
- Integrated with functions
- Reusable parameter library with traceability

#### Change Management
- Document-based change process
- Controlled targets
- Impact analysis & susceptibility mgmt
- Metrics with history for improvement
- Project level reuse, starting point for next project

#### Functional Modeling
- Functional description docs
- Function hierarchy models
- Isolated functional behavior models
- Integrated functional modeling
- Functional arch with allocations & traceability

#### Requirement Management
- Uncontrolled spreadsheets & docs
- Managed Docs
- Standalone solutions (disconnected)
- RM/traceability exchange
- Connected, configured, cross-domain traceability with reuse

#### Model Management
- Uncontrolled, rules-of-thumb, hierarchies
- Uncontrolled, behavior models
- Shared model repository
- Integrated, component library
- Model reuse with controlled parameters

#### Verification & Validation
- Minimum to no planning
- Manually testing everything
- Isolated validation simulations
- Integrated simulation (HiL, SiL)
- Focused testing, reuse results, swap out models

#### Design Management
- Unmanaged CAX/SW models
- Locally Managed CAX/SW
- Enterprise repositories
- Integrated models (HiL, SiL,...)
- Cross-domain design/optimization

#### CMMI Staged Levels:
1. Initial
2. Managed
3. Defined
4. Qualitative
5. Optimizing

---

**How bad is your communication problem?**
## Where are we?

Everyone has a communication problem.
PLM Required to solve your communication problem…

- Different tools speak different languages
- Tool specific integrations are not scalable (NxN problem)
- Digital thread between different tools carried by PLM with integrated systems methodology
- Thru infrastructure defined by Product Architecture that is part of PLM

<table>
<thead>
<tr>
<th>MBSE Process Maturity Level</th>
<th>Requirement Management</th>
<th>Model Management</th>
<th>Verification &amp; Validation</th>
<th>Design Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 1</td>
<td>Uncontrolled spreadsheets &amp; docs</td>
<td>Uncontrolled, rules-of-thumb, hierarchies</td>
<td>Minimum to no planning</td>
<td>Unmanaged CAX/SW models</td>
</tr>
<tr>
<td>Level 2</td>
<td>Managed Docs</td>
<td>Shared model repository</td>
<td>Manually testing everything</td>
<td>Locally managed CAX/SW models</td>
</tr>
<tr>
<td>Level 3</td>
<td>Integrated, component library</td>
<td>Integrated simulation (HiL, SiL)</td>
<td>Integrated models (HiL, SiL,...)</td>
<td>Integrated model reuse with controlled parameters</td>
</tr>
<tr>
<td>Level 4</td>
<td>Connected, configured, cross-domain traceability with reuse</td>
<td>Model reuse with controlled parameters</td>
<td>Cross-domain design/optimization</td>
<td>Focused testing, reuse results, swap out models</td>
</tr>
</tbody>
</table>

**Integrated systems methodology**

- Digital thread between different tools carried by PLM
- Thru infrastructure defined by Product Architecture that is part of PLM

**Tool specific integrations**

- Different tools speak different languages
- Not scalable (NxN problem)

**Solution**

- PLM integrates different tools
- Uses a standardized approach

**Exemplary company**

- Best case scenario
- Integrated systems methodology

**Example**

- RAMS (Reliability, Availability, Maintainability, and Safety)
- SysML (Systems Modeling Language)
- Excel
- RM Tools
Discover early vs late…
Shift left…

- **Use Cases & Demands**
- **Concept Design**
- **System Design**
- **Derived System Requirements**
- **Selected Product Requirements**

**Shift discovery to the left**
Fail fast, early, once
Start integrated, stay integrated
PLM Integrated system model
- Drives out integrated requirements
- Validated by system simulation
- Maps into discipline tools
- Shared with suppliers
- Driven by standard integrated process
Enabling a continuous agile communication process
Summary…
You don’t have an engineering problem, you have a knowledge communication/management problem

Today’s products are built by everyone/everywhere…
• Documents aren’t scalable
• Disconnected models provide knotholes
• *SysML v1 language is missing methods to go with the language & no ability to share IP with suppliers*

Symptoms:
• Half your program schedule is spent on system integration (supplier collaboration?)
• Tedious communication via meetings (inter-team and intra-team)
• Uncommunicated change
• Innocent impact understanding
• …

An integrated product architecture/blueprint with requirements is mandatory
• Delivered thru PLM
• Allocated through suppliers for continuous feedback

…to start integrated, stay integrated
How to start solving your communication problem...

Possible starting spots…
To solving your communication problem
...enabling continuous communication
Thank you