How the Modeling and Simulation (M&S) Product Line Approach Supports Concept Exploration

David R. Pratt, PhD
Robert W. Franceschini, PhD
Robert B. Burch
Science Applications International Corporation (SAIC)
prattda@saic.com
Agenda

- What is a Product Line
- How does a Product Line Work
- M&S as a System Engineering Tool
- Applications
- Caveats
Classic Cross-Program Acquisition

Enterprise Level Requirement

...shall train ...

The Same Need...

... with stovepiped acquisitions ...

... resulting in unnecessarily different approaches that usually don’t interoperate.

Program ABC Lifecycle

MNS/ORD/TRD/... Development

Acquire/Develop

Support

Program MNO Lifecycle

MNS/ORD/TRD/... Development

Acquire/Develop

Support

Program XYZ Lifecycle

MNS/ORD/TRD/... Development

Acquire/Develop

Support

Duplicate work, lost money!

Less Bang for the Warfighter Buck!

No Enterprise/System Engineering Across Similar Acquisitions!
Typical Entity Count and Fidelity Characterization

- **Weapon/System Development**
- **Virtual Training**
  - Focused
  - CCTT
  - AVCATT
  - OTB
- **STOW Light**
- **AoA**
- **DBST**
- **Concept Development**
- **Force Design**

**Entity Count**

- **AFAP**
- **Slaved Real-Time**
- **Repeatable**

**Fidelity**

- **High**
- **Low**

**Prohibitive Technology Costs**

**Low Utility**
A software product line (SPL) is a set of software-intensive systems that share a common, managed set of features satisfying the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.

– Software Engineering Institute

A collection of interrelated, and possibly redundant, software components that can be brought together to create instances to suit different needs.

– Me
Product Line Architecture Development Process

<table>
<thead>
<tr>
<th>Virtual Domain</th>
<th>Develop Architecture &amp; Reusable Components</th>
<th>Used By Development Community (Govt./Industry)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harvesting Requirements</td>
<td>Forecasted Requirements</td>
<td>Current Systems</td>
</tr>
<tr>
<td>Requirements/Algorithms</td>
<td>Product Line Baseline</td>
<td>Stakeholders</td>
</tr>
<tr>
<td>Architectures</td>
<td>Development Specifications</td>
<td>Stakeholders</td>
</tr>
<tr>
<td>Stakeholders</td>
<td>Processes &amp; Policies</td>
<td>Stakeholders</td>
</tr>
<tr>
<td>Developers</td>
<td>Sustainment</td>
<td>Stakeholders</td>
</tr>
<tr>
<td>Developers</td>
<td>Harvesting</td>
<td>Stakeholders</td>
</tr>
<tr>
<td>Developers</td>
<td>Development</td>
<td>Stakeholders</td>
</tr>
<tr>
<td>Developers</td>
<td>Support</td>
<td>Stakeholders</td>
</tr>
<tr>
<td>Developers</td>
<td>A&amp;I IPT</td>
<td>Stakeholders</td>
</tr>
</tbody>
</table>

Client Feedback (DRs, Requests, Improvements)
# Product Line Components

<table>
<thead>
<tr>
<th>Architectural Applications (System Compositions)</th>
<th>Leader and Staff Training System Composition</th>
<th>Seamless Training System Composition</th>
<th>Force and Organizations Analysis Tool System Composition</th>
<th>Test and Evaluation System Composition</th>
<th>Other System Compositions</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Composer</td>
<td>Knowledge Eng. Env.</td>
<td>Event Planner</td>
<td>Model Composer</td>
<td>Simulation Generator</td>
<td>Technical Manager</td>
</tr>
</tbody>
</table>

## Component Layer

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Environment Database Generation Environment</td>
<td>Entity Composer</td>
<td>Data Collection Specification Tool</td>
<td>Federation Develop. Tool</td>
<td>Federation Mgmt. Tool</td>
<td>Federation Mgmt. Tool</td>
<td>Translation Services</td>
<td>Connect Services</td>
<td>Stealth Tool</td>
<td>Information Meta-Data Tool</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Icon Tool</td>
<td>Battlefield Enum. Tool</td>
<td>Benchmark Tool</td>
<td>Network Loader Tool</td>
<td>Network Loader Tool</td>
<td>Network Loader Tool</td>
<td>Stealth Tool</td>
<td>Connect Services</td>
<td>Stealth Tool</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Component Support Layer

<table>
<thead>
<tr>
<th>Composition Services</th>
<th>Environment Runtime Services</th>
<th>Environment Reasoning Services</th>
<th>GUI Services</th>
<th>Plan View Display</th>
<th>Data Collection Services</th>
<th>Simulation Services</th>
<th>Simulation Object Runtime Database</th>
<th>Modeling Services</th>
<th>System Repository Services</th>
</tr>
</thead>
<tbody>
<tr>
<td>KA/KE Repository</td>
<td>Environment Repository</td>
<td>Software Repository</td>
<td>System Composition Repository</td>
<td>Military Scenario Repository</td>
<td>Local Exercise Environment Repository</td>
<td>Parametric &amp; Initialization Repository</td>
<td>Simulation Output Repository</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

## Repository Component Layer

<table>
<thead>
<tr>
<th>Monitor Services</th>
<th>Time Services</th>
<th>Name Directory Services</th>
<th>Messaging Services</th>
<th>Coordinate Services</th>
<th>Interchange Services</th>
<th>Middleware Services</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>RTI DIS COE Services WWW JDBC/ODBC ORB Live Range Adapter</td>
</tr>
</tbody>
</table>

## Common Services Layer

<table>
<thead>
<tr>
<th>Platform Layer</th>
<th>Hardware</th>
<th>Operating System</th>
<th>Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hardware</td>
<td>Operating System</td>
<td>Network</td>
<td></td>
</tr>
</tbody>
</table>
Evolution and Maintenance

Product Line Architecture Specification (PLAS) 1.0

Component Development

Product Line Vision / Definition
Domain Analysis
Define Constraints
Develop Product Line Architecture

ORD
TRD
Standards
Use Cases
Legacy Systems

Initial Development

Requirements Analysis
Design
Implementation
Integration
Testing
Fielding
Reviews
Objective Metrics
PLAS Updates

Instance Spin Out
Multiple Instances from the Same Product Line

Components can be simulators, emulators, or real devices
Computational Allocations Vary Per Instance

Training Stand Alone

Study Support

Distributed

Different Computational Platform
Notional Component Allocation Table

<table>
<thead>
<tr>
<th>Models</th>
<th>% entity contain</th>
<th># per entity</th>
<th># models/ processor</th>
<th># models/ system</th>
<th>Rate (Hz)</th>
<th>Period (sec)</th>
<th>% allocated (rel)</th>
<th>Normalized Allocated (msec)</th>
<th>Total Allocation (msec)</th>
<th>Agent Allocation (msec)</th>
<th>Normalized Allocation (msec)</th>
<th>Number of Events/ Second</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Physical Agents</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DynamicsAgent</td>
<td>95%</td>
<td>1</td>
<td>95</td>
<td>1,425</td>
<td>7.00</td>
<td>0.142</td>
<td>41.95%</td>
<td>18.878</td>
<td>81.230</td>
<td>0.855</td>
<td>0.199</td>
<td>665</td>
</tr>
<tr>
<td>WeaponAgent</td>
<td>80%</td>
<td>1</td>
<td>80</td>
<td>1,200</td>
<td>5.00</td>
<td>0.200</td>
<td>15.00%</td>
<td>6.750</td>
<td>40.909</td>
<td>0.511</td>
<td>0.084</td>
<td>400</td>
</tr>
<tr>
<td>SensorAgent</td>
<td>95%</td>
<td>1</td>
<td>95</td>
<td>1,425</td>
<td>0.33</td>
<td>3.000</td>
<td>10.00%</td>
<td>4.500</td>
<td>409.090</td>
<td>4.306</td>
<td>0.047</td>
<td>31</td>
</tr>
<tr>
<td>TurretAgent</td>
<td>60%</td>
<td>1</td>
<td>60</td>
<td>900</td>
<td>5.00</td>
<td>0.200</td>
<td>5.00%</td>
<td>2.250</td>
<td>13.636</td>
<td>0.227</td>
<td>0.038</td>
<td>300</td>
</tr>
<tr>
<td>CSSAgent</td>
<td>5%</td>
<td>1</td>
<td>5</td>
<td>75</td>
<td>0.05</td>
<td>20.000</td>
<td>0.01%</td>
<td>0.005</td>
<td>2.727</td>
<td>0.545</td>
<td>0.001</td>
<td>-</td>
</tr>
<tr>
<td>CommAgent</td>
<td>95%</td>
<td>1</td>
<td>95</td>
<td>1,425</td>
<td>5.00</td>
<td>0.200</td>
<td>10.00%</td>
<td>4.500</td>
<td>27.272</td>
<td>0.287</td>
<td>0.047</td>
<td>475</td>
</tr>
<tr>
<td>VulnerabilityAgent</td>
<td>95%</td>
<td>1</td>
<td>95</td>
<td>1,425</td>
<td>0.05</td>
<td>20.000</td>
<td>0.04%</td>
<td>0.018</td>
<td>10.909</td>
<td>0.114</td>
<td>0.000</td>
<td>4</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td>7</td>
<td>525</td>
<td>7,875</td>
<td>82.00%</td>
<td>36.900</td>
<td>585.773</td>
<td>6.845</td>
<td>0.146</td>
<td>1,875</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Entity Behaviors</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Behavioral Agents</td>
<td>95%</td>
<td>6</td>
<td>570</td>
<td>8,550</td>
<td>1.00</td>
<td>1.000</td>
<td>10.00%</td>
<td>4.500</td>
<td>136.363</td>
<td>0.239</td>
<td>0.008</td>
<td>570</td>
</tr>
<tr>
<td>PrimaryAgent</td>
<td>100%</td>
<td>1</td>
<td>100</td>
<td>1,500</td>
<td>5.00</td>
<td>0.200</td>
<td>2.00%</td>
<td>0.900</td>
<td>5.454</td>
<td>0.054</td>
<td>0.009</td>
<td>500</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td>7</td>
<td>670</td>
<td>10,050</td>
<td>12.00%</td>
<td>5.400</td>
<td>141.817</td>
<td>0.293</td>
<td>0.017</td>
<td>1,070</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Unit Behaviors</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Behavioral Agents</td>
<td>100%</td>
<td>2</td>
<td>200</td>
<td>3,000</td>
<td>0.33</td>
<td>3.000</td>
<td>4.00%</td>
<td>1.800</td>
<td>163.636</td>
<td>0.818</td>
<td>0.009</td>
<td>66</td>
</tr>
<tr>
<td>PrimaryAgent</td>
<td>100%</td>
<td>1</td>
<td>100</td>
<td>1,500</td>
<td>0.33</td>
<td>3.000</td>
<td>2.00%</td>
<td>0.900</td>
<td>81.818</td>
<td>0.818</td>
<td>0.009</td>
<td>33</td>
</tr>
<tr>
<td><strong>Subtotal</strong></td>
<td>3</td>
<td>300</td>
<td>4,500</td>
<td>6.00%</td>
<td>2.700</td>
<td>245.454</td>
<td>1.636</td>
<td>0.018</td>
<td>99</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Totals</strong></td>
<td>18</td>
<td>1,595</td>
<td>23,925</td>
<td>100.00%</td>
<td>45.000</td>
<td>973.044</td>
<td>8.774</td>
<td>0.451</td>
<td>3,044</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Ability of the system to meet the computational requirements in addition to the functional requirements
- Interrelationships between the components can result in cascading performance issue
Updates to the Model

- **Network**
  - Network throughput, rates, and utilization
  - Interaction verses object update characterizations

- **Disk**
  - More accurate quantification taking XML into account
  - Sizes of repositories

- **Timings**
  - Times for specific capabilities
  - An event oriented model
  - Better approach for multi-node

- **Structure**
  - Must adapt the model to any architectural changes in the system

- **Verification**
  - Sample real times and compare them to the allocation model
VSA - Product Line Application

Many funding sources...

- USMC
- AF
- UK
- $ $ $
- TEMO, TRAC, AMSAA, FCS, ...
- $
- $

SE Core A&I

OOS

SE Core Common Virtual Environment Repository

SE Core A&I/DVED

... with system engineered interoperation...

... contributing to a common repository of reusable items.

Your Collective Trainer

Your Driver Trainer

Their SOF CAVE

Product Lines give the Warfighter more capability for the same cost!

Because Other People’s Money Becomes Your Money Too!
C4ISR Example Using SE Core

Army Enterprise Level Training Need

...shall train C4ISR...

OOS

SE Core CVE Repository

SE Core A&I/DVED

Cost of Systematic Reuse, Money leveraged!

Program ABC Lifecycle

Program MNO Lifecycle

Program XYZ Lifecycle

Less Overall Cost, Common System Engineering, Common Support!

Same initial investment...

...with reuse repository support and reduced incorporation costs...

...gives common, interoperable approach.

Less Overall Cost, Common System Engineering, Common Support!
VSA Products

- **Product Line Architecture Framework (PLAF)**
- **Product Line Architecture Specification (PLAS)**
  - Defines the VSA including components
  - Organizes architectural artifacts and documentation
- **Classical Analysis/Design Artifacts**
  - **DoDAF Product Set**
    - AV-1, AV-2, OV-1, OV-2, OV-5, SV-1, SV-4, TV-1, OV-3, OV-4, SV-2, SV-6, SV-8, SV-9, TV-2
  - Domain System/Subsystem Specification (SSS)
  - Technical Use Cases
    - High-level trainer operational
    - Identify key system level to drive VSA/CVCs
- **Evolution Plan**
  - How/when to migrate current programs to VSA
The Ugly Truth

- The **Product Line** is only as good as its inputs
  - The architecture depends on the structure of the system and how thread of control is portioned out
  - The model depends on subjective allocations of relative times (percents) based on legacy systems
  - The longevity is only as good as the configuration control board
- The **Product Line** will change!
  - The components allocated today to a capability may be different in the future
  - It won’t change that much, more like +/- 10% not +/- 50%
  - Usually it changes because of structure changes or misallocation of relative time and new requirements
    - Usually because of something the user doesn’t know
- The **Product Line** is only viable when it is accessible
  - The updated component listing should be maintained on the repository web page
  - The updated performance model should be maintained on the performance web page
- The **Product Line** will need to manage the change
  - Repository updates
  - Performance Model updates
  - Structure changes or additions
  - Rate changes
  - Data rates and sizes