Mega-scale Product Line Engineering for the Automotive Industry

Advanced Variant & Complexity Management across the full Product Lifecycle

Method Park, Product Lines in the Park
Rochester, MI
April 26, 2016

Dr. Charles W. Krueger, CEO
BigLever Software

Product Line Engineering (PLE) Defined

- **Product Line**: a family of similar products with variations in features and functions

- **Product Line Engineering**: the engineering of a product line using a *shared set of engineering assets*, a *managed set of features*, and an *efficient means of production*...

  - taking advantage of the **commonality** shared across the family
  - efficiently and systematically managing the **variation** among the products
Automotive Problem Statement for PLE

Complexity is making it intractable to engineer product family Features in terms of an interacting Bill-of-Materials from mechanical, electrical, and software.

A new approach is required...

Source of BigLever's Perspective on PLE

- **Industry leader** in Product Line Engineering solutions
  - 15 years of commercial practice with PLE tools, methods & organizational change
- **Industry standard** PLE framework, ecosystem & methodology
  - Gears PLE Lifecycle Framework™ and PLE Ecosystem of third party integrations
    - IBM, Aras, Microsoft, ANSYS, Open Source, MadCap, Sparx, No Magic...
  - BigLever 3-Tiered PLE Methodology™
- **Industry’s only** service provider with **proven success** in organizational change management for transition to PLE practice
## Common Motivation in PLE Success Stories: Overwhelming Competitive Advantage

<table>
<thead>
<tr>
<th>Who are they?</th>
<th>What is their product line?</th>
<th>Driving problem</th>
<th>PLE results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Worlds #1 defense contractor</td>
<td>AEGIS Weapon System</td>
<td>High cost of old approach threatened loss of entire contract</td>
<td>Over 100 ship deployments: $47 million saved per year[^1][^2][^3]</td>
</tr>
<tr>
<td>World’s #4 defense contractor</td>
<td>Live Training Transformation, family of large-scale training systems for US Army, Air Force, and Marines</td>
<td>Innovative low-cost solution required to win and keep major contract</td>
<td>Over 300 training range deployments: $750 million saved over 12 years[^4][^5]</td>
</tr>
<tr>
<td>World’s #1 auto-maker</td>
<td>Largest, most complex product line comprising over 10,000,000 instances</td>
<td>Vehicles taking too long to bring to market; expensive and error-prone processes</td>
<td>Will save “hundreds to thousands of man/years per year, worth tens to hundreds of millions of dollars per year” for one asset type alone[^6][^7][^8]</td>
</tr>
<tr>
<td>NetApp</td>
<td>High-end server storage systems</td>
<td>Unable to accommodate growth in market</td>
<td>2x-5x improvements in scalability, productivity, time-to-market, and product quality[^9]</td>
</tr>
<tr>
<td>World leader in on-line vacation property rental</td>
<td>Product line of e-commerce web sites hosted in over 200 countries worldwide</td>
<td>Broad variation in sites around the world; needed to go live ASAP</td>
<td>First product went live in 60 days[^10]</td>
</tr>
<tr>
<td>Leading aviation supplier</td>
<td>Whole-aircraft avionics product line</td>
<td>High cost of product certification</td>
<td>8:1 improvement in time to produce certification documents</td>
</tr>
</tbody>
</table>
An Efficient Means of Production for Product Lines

Shared assets are like the factory’s supply chain.
Features describe capabilities that vary among products.

**Assets** are configured according to the feature profiles of the products you want build.
**Features** come in.

A **product** comes out.

**Assets** are configured.

Just like a factory.

---

**The Gears PLE Lifecycle Framework**
A Shift in Perspective

Rather than use a *Bill-of-Materials* to determine *Features*, use a *Bill-of-Features™* to determine *Materials*.

*M.C. Escher, “Metamorphosis,” 1937*

*Materials == mechanical, electrical, wiring, software, calibrations, diagnostics…*
The Power of Abstraction in PLE
A Single Source of the Feature Truth

Consolidated ALM/PLM/PLE Lifecycle

The PLE Workflow

Push a button to generate Requirements, Supplier part specs, Test cases, Software builds, Calibration sets, Serial data sets, Diagnostics, BOMs, Manufacturing part selection logic, Proliferations, Wiring harnesses, Owners Manuals, …
The Evolution and Convergence of Product Line Engineering & Operations

Broadening the Scope of Abstraction with PLE&O

Single Source of the Feature Truth for Product Line Engineering and Operations
Automotive Technical Challenge — Astronomical Combinatorics

• Challenge
  - How can you have a uniform approach that bridges the gap between 1,000,000 variation points in the assets to 10-20 customer decisions?
  - $2^{1,000,000} : 2^{10}$ combinatoric reduction
  - (Note there are $2^{280}$ atoms in the universe and $2^{33}$ humans on the Earth)

• Engineering Solution
  - Abstraction
  - Automation

Feature Ontology for PLE
Six Layers of Feature Abstractions for PLE

- Feature Bundling
- Multistage Product Family Trees
- Hierarchical product-lines of product-lines
- Feature Profiles
- Feature Models
- Feature-based Asset Variation Points

If the benefits of PLE are so large, why hasn’t everyone adopted it?

- It’s more than just variation management, tools and technology
- It’s more than just methodology
- PLE requires a well orchestrated combination of tools, methodology and organizational change across the enterprise
- Organizational Change is hard, but is often overlooked in PLE, variation management, and complexity management initiatives
Organizational Challenge — Enterprise Alignment

Change is good. You go first.

BoM Engineers
Requirements Engineers
Calibration Engineers
Diagnostic Engineers
Network Engineers
Test Engineers
Domain Architects
System Data Modelers
Subsystem Leads
Product Marketing Team
Vendor Ecosystem

Executive leadership
Management
Supply Chain Engineers
Product Calibration Engineers
Business Process Engineers

3-Tiered PLE Methodology
Spiral Organizational Change for PLE

Applying Organizational Change to the 3-Tiered PLE Methodology
PLE ConOps example in MethodPark Stages

Product Change Management

<table>
<thead>
<tr>
<th></th>
<th>Review Customer Product Request</th>
<th>Identify a New Product Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Manager</td>
<td>Request a New Product Configuration</td>
<td></td>
</tr>
<tr>
<td>Anyone</td>
<td>Identify a Defect</td>
<td></td>
</tr>
<tr>
<td>CCB Change Control Board</td>
<td>Triage and Prioritize Change Request</td>
<td>Approve change request (Ch)</td>
</tr>
<tr>
<td>Product Lead Engineer</td>
<td>Analyze Change and Product Line Impact</td>
<td>Plan/Schedule Change Requests</td>
</tr>
<tr>
<td>PLCB Product Line Control Board</td>
<td>Add a new feature</td>
<td></td>
</tr>
</tbody>
</table>
BigLever’s Solution for PLE

**onePLE**

A complete packaged solution for your transition to PLE practice and the competitive advantage that it enables

- **onePLE™** provides
  - PLE technology and training
  - PLE methodology and mentoring
  - PLE organizational change management and guidance

www.biglever.com