32nd Annual INCOSE international symposium

hybrid event

Detroit, MI, USA
June 25 - 30, 2022

Practical Experience Applying Feature-based Product Line Engineering in a DevOps Environment

Achieving the Best of Both Worlds

www.incose.org/symp2022

Your Authors

- Chris Pedone
  - VT Group (VTG)
- David Hartley
  - General Dynamics Mission Systems
- Rowland Darbin
  - General Dynamics Mission Systems
- Paul Clements
  - BigLever Software, Inc.

Feature-based PLE Overview

Product Line Engineering (PLE) Defined

ISO 26580 Methods and Tools for Feature-based PLE

Product Line:

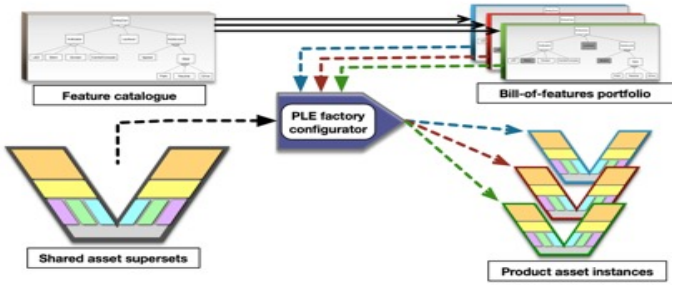
A family of similar products or systems with variations in features.

Product lines are ubiquitous — virtually all products and systems are built in the context of a family.

International Organization for Standardization

Product Line Engineering:

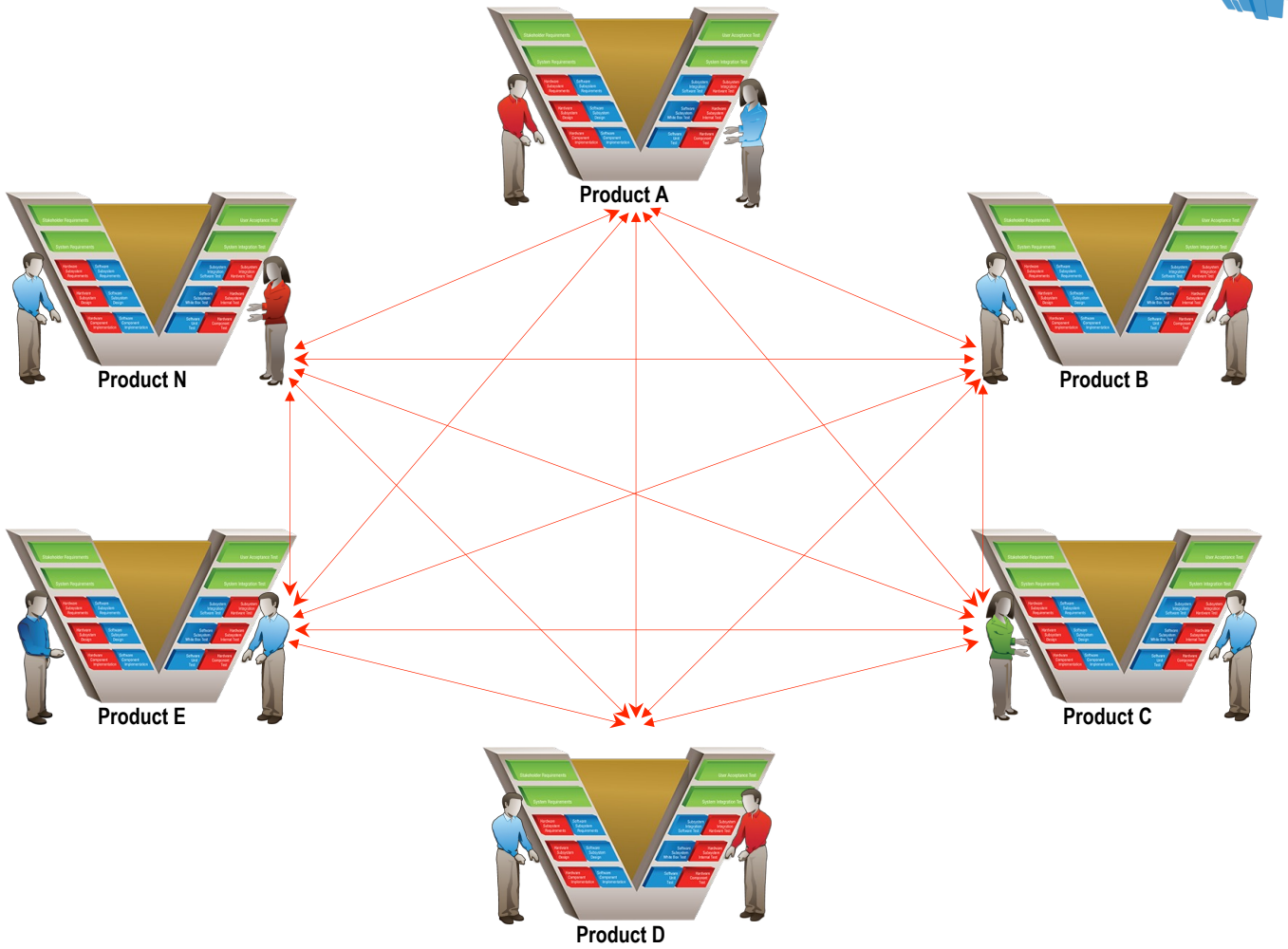
the engineering of a product line using
shared engineering assets,
a *managed catalog of features*, and
an *automated means of production*…



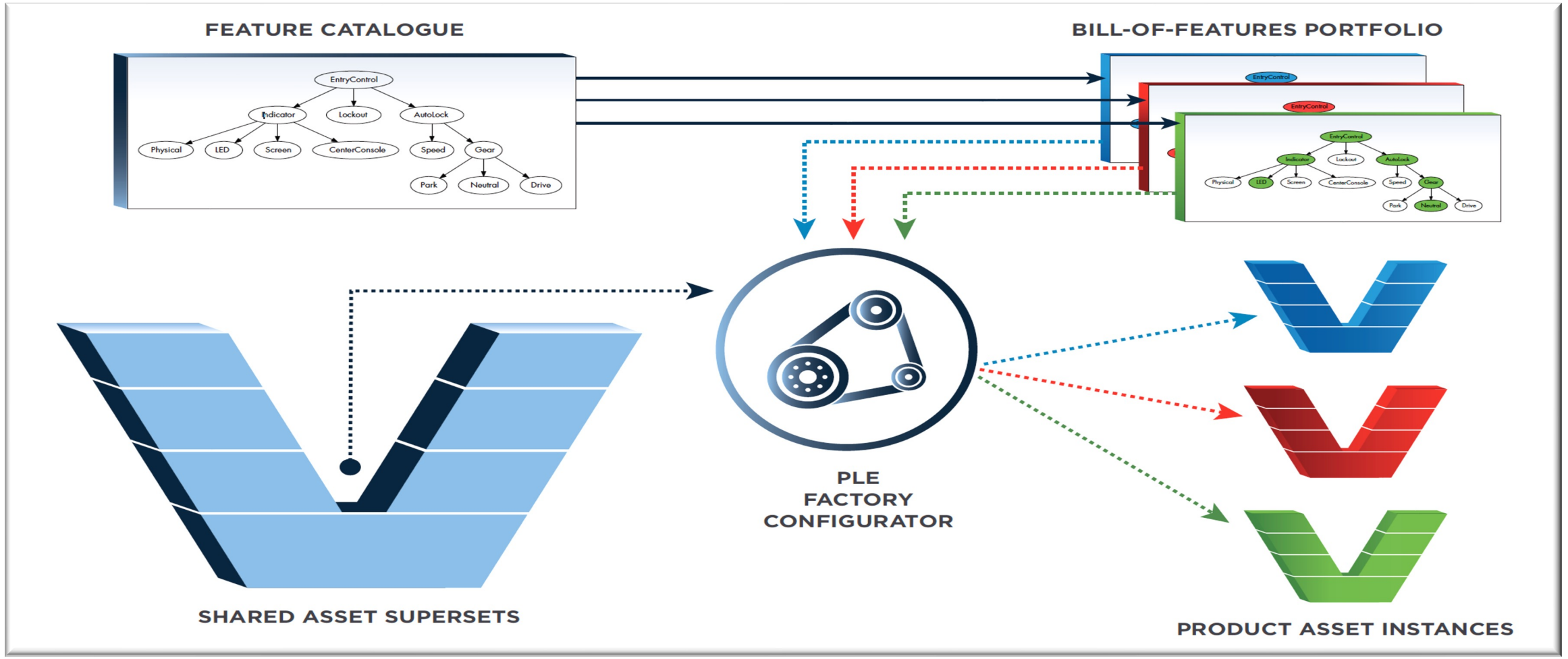
- taking advantage of the **commonality** shared across the family
- efficiently and systematically managing the **variation** among the products or systems

Problem Space: Early Generation Product-centric Engineering

- Duplication, branch-and-merge, clone-and-own, self-inflicted N^2 complexity, …
- Informality introduces significant risks in the form of defects, errors, and omissions
- Leads to delays, budget overruns, recalls, system failures, and opportunity losses



Feature-based PLE Factory



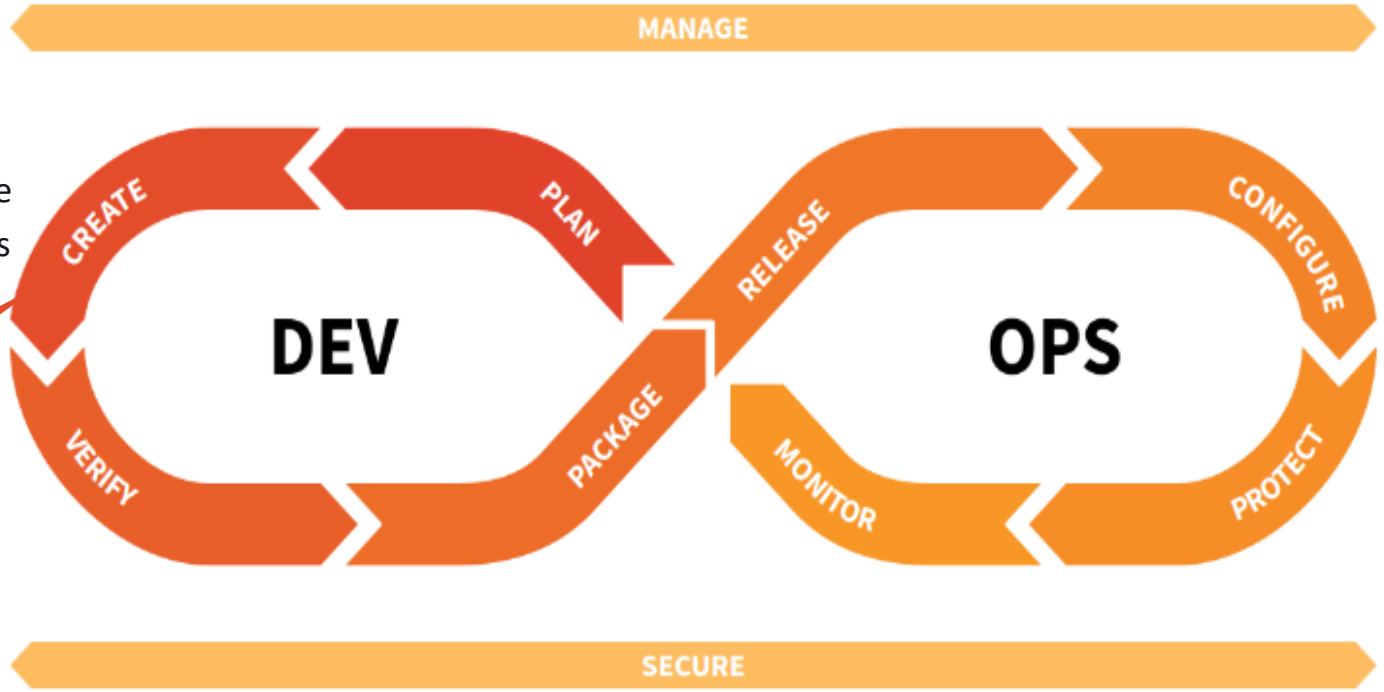
DevOps

- "a set of practices intended to reduce the time between committing a change to a system and the change being placed into normal production, while ensuring high quality" (Bass 2015)

Core DevOps principles

1. Automation of the software development lifecycle
2. Collaboration and communication
3. Continuous improvement and minimization of waste
4. Hyperfocus on user needs with short feedback loops

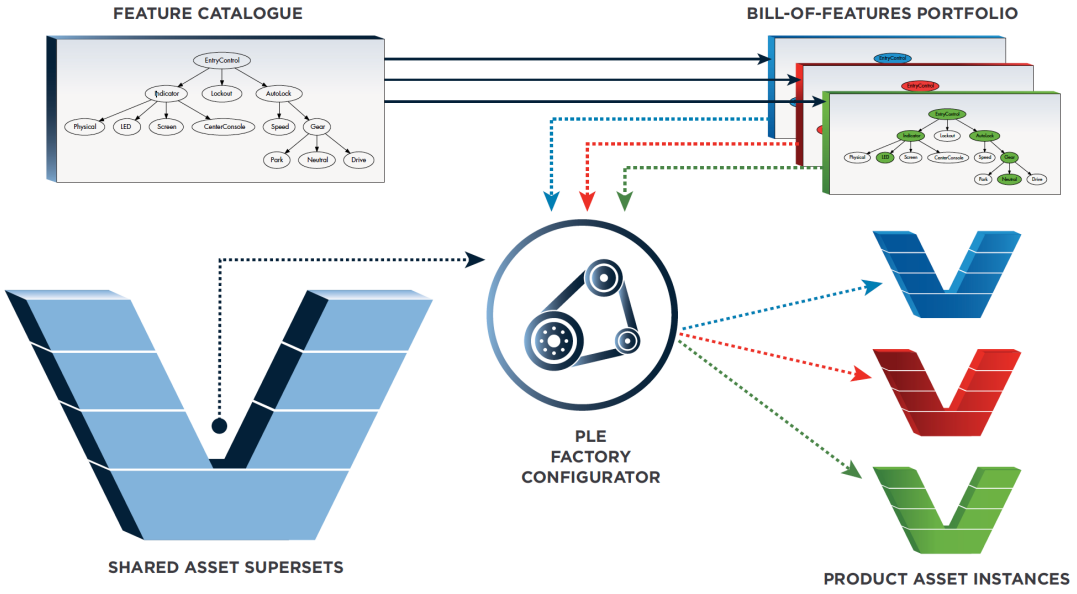
PLE Happens Here!



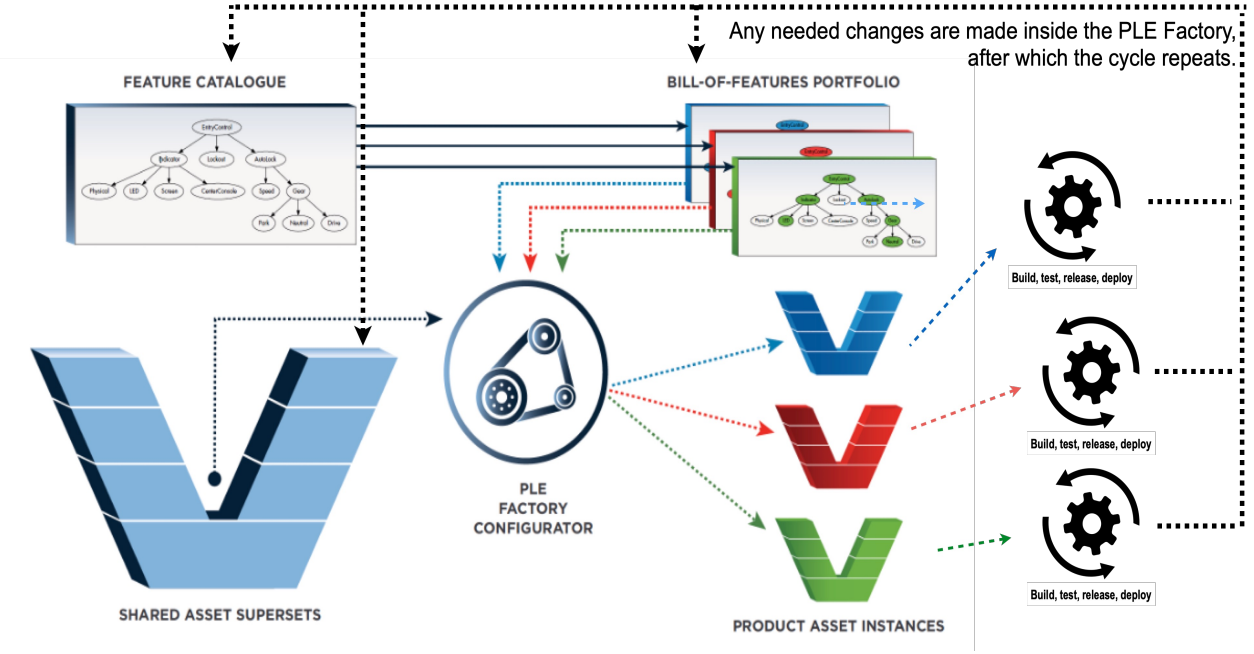
https://about.gitlab.com/topics/devops

Feature-based PLE + DevOps

Feature-based PLE Factory



PLE/DevOps Workflow



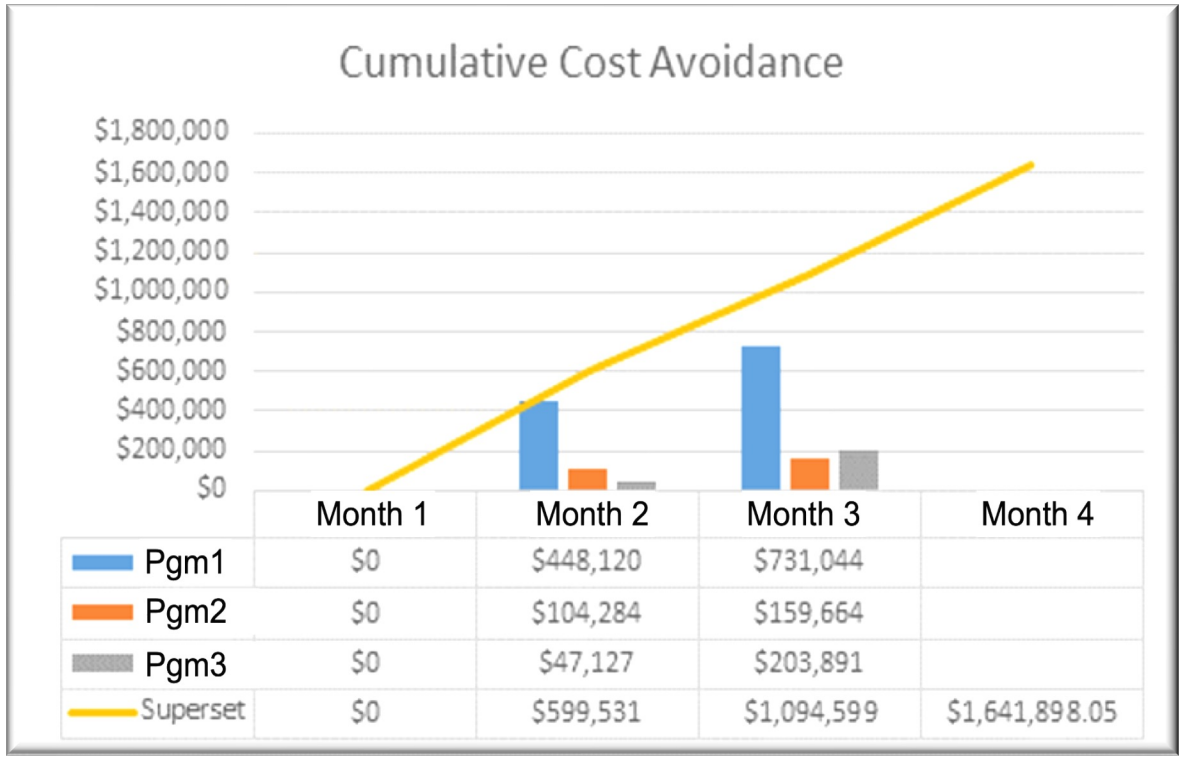
Key principles of combination

- All changes occur inside the PLE Factory
  - No changes are made to the product asset instances
- Merge all changes back to their respective main branches as often as possible
- Validate changes
  - Actuate, build, test and release for each instance impacted by change
- Use Feature-based PLE to manage the variation in the DevOps pipeline itself!

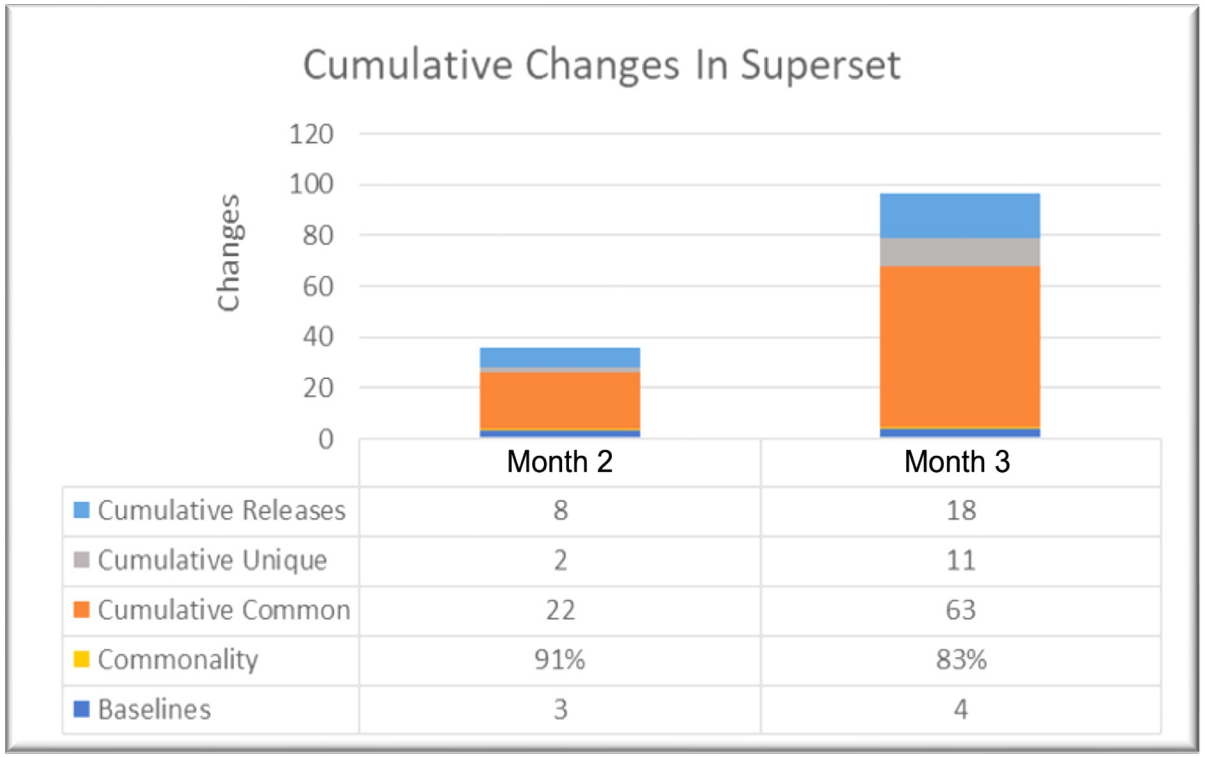
US Navy Program Results

- By eliminating duplicative code copies, Feature-based PLE allowed us to shrink the code base from 4.77M SLOC to 795K SLOC.
  - Less software to manage and sustain, increased capacity to the team to innovate and exceed the customer's expectation
- Increased ability to define new products and capabilities
  - We can leverage existing capabilities in the product line and know exactly what needs to be estimated and built to create new products
- Enhancements made on one system can quickly be shared to other members of the product line
  - Increasing the capabilities of other members of the product line without increasing cost of development for the customer
- Doubling of active program support, growing new capabilities like unmanned operations without extending the size of the development team

Cost Avoidance and Change



	Month 1	Month 2	Month 3	Month 4
Pgm1	$0	$448,120	$731,044	
Pgm2	$0	$104,284	$159,664	
Pgm3	$0	$47,127	$203,891	
Superset	$0	$599,531	$1,094,599	$1,641,898.05



	Month 2	Month 3
Cumulative Releases	8	18
Cumulative Unique	2	11
Cumulative Common	22	63
Commonality	91%	83%
Baselines	3	4

Observations and Lessons Learned

- Break the bonds of “product” ownership
  - Organizational maximization is more important than local product maximization
- Decomposition helps achieve commonality
- Culture matters
- Agile is key
- Governance increases stakeholder communication

32nd Annual INCOSE international symposium

hybrid event

Detroit, MI, USA

June 25 - 30, 2022

www.incose.org/symp2022