Software Product Lines: Today's Impact and Tomorrow's Potential

Linda Northrop

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-2612

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

Today's Talk

SOFTWARE PRODUCT LINES: The Past

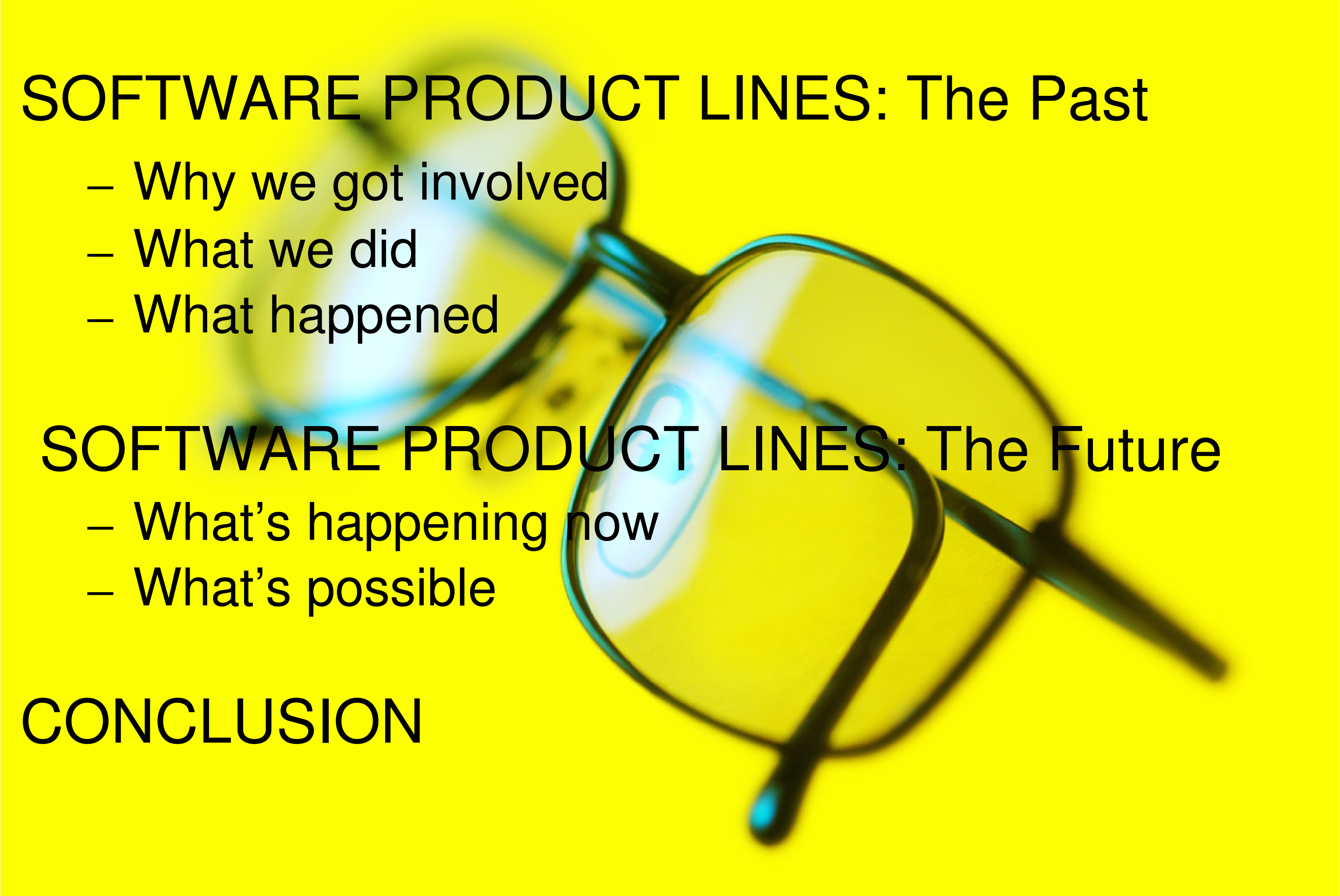
- Why we got involved
- What we did
- What happened

SOFTWARE PRODUCT LINES: The Future

- What's happening now
- What's possible

CONCLUSION

With an Architecture Perspective



SOFTWARE PRODUCT LINES: The Past

- Why we got involved
- What we did
- What happened

SOFTWARE PRODUCT LINES: The Future

- What’s happening now
- What’s possible

CONCLUSION

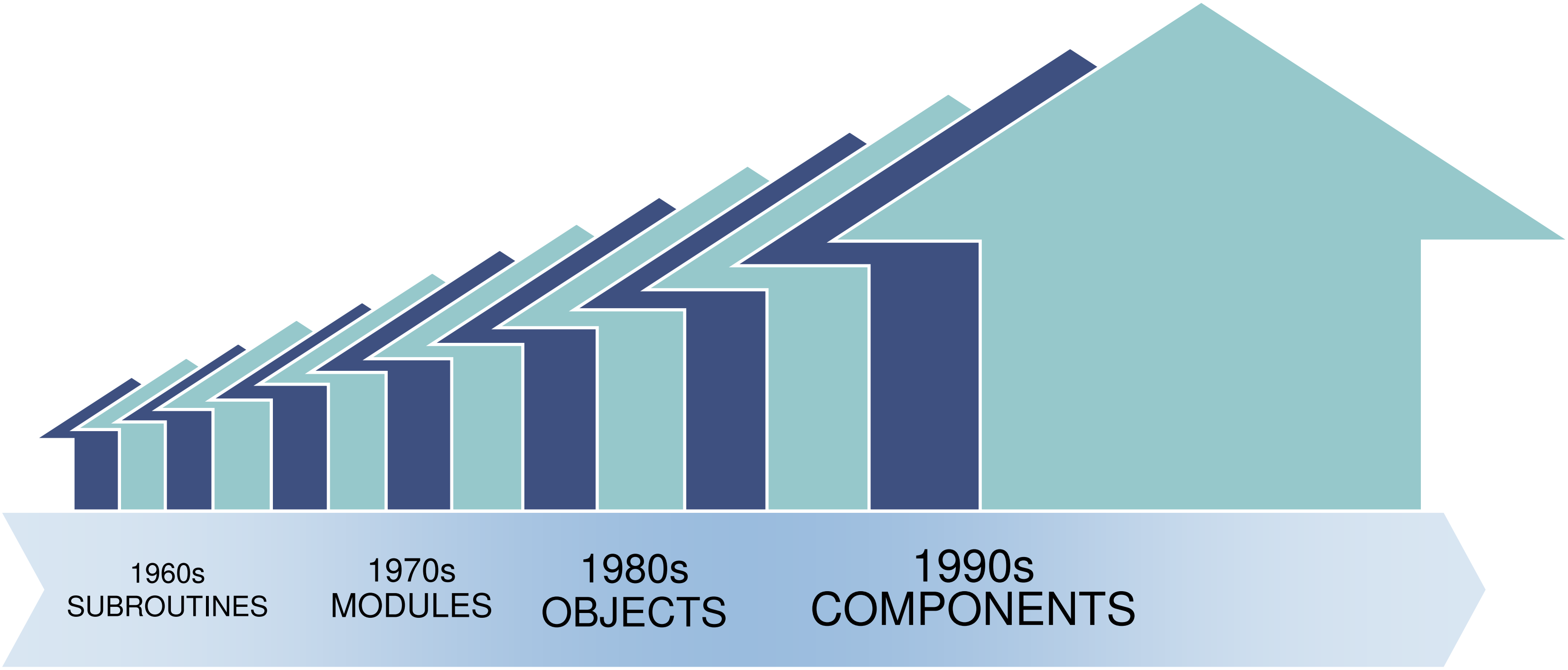
Few Systems Are Unique

Most organizations produce families of similar systems, differentiated by features.

A reuse strategy makes sense.

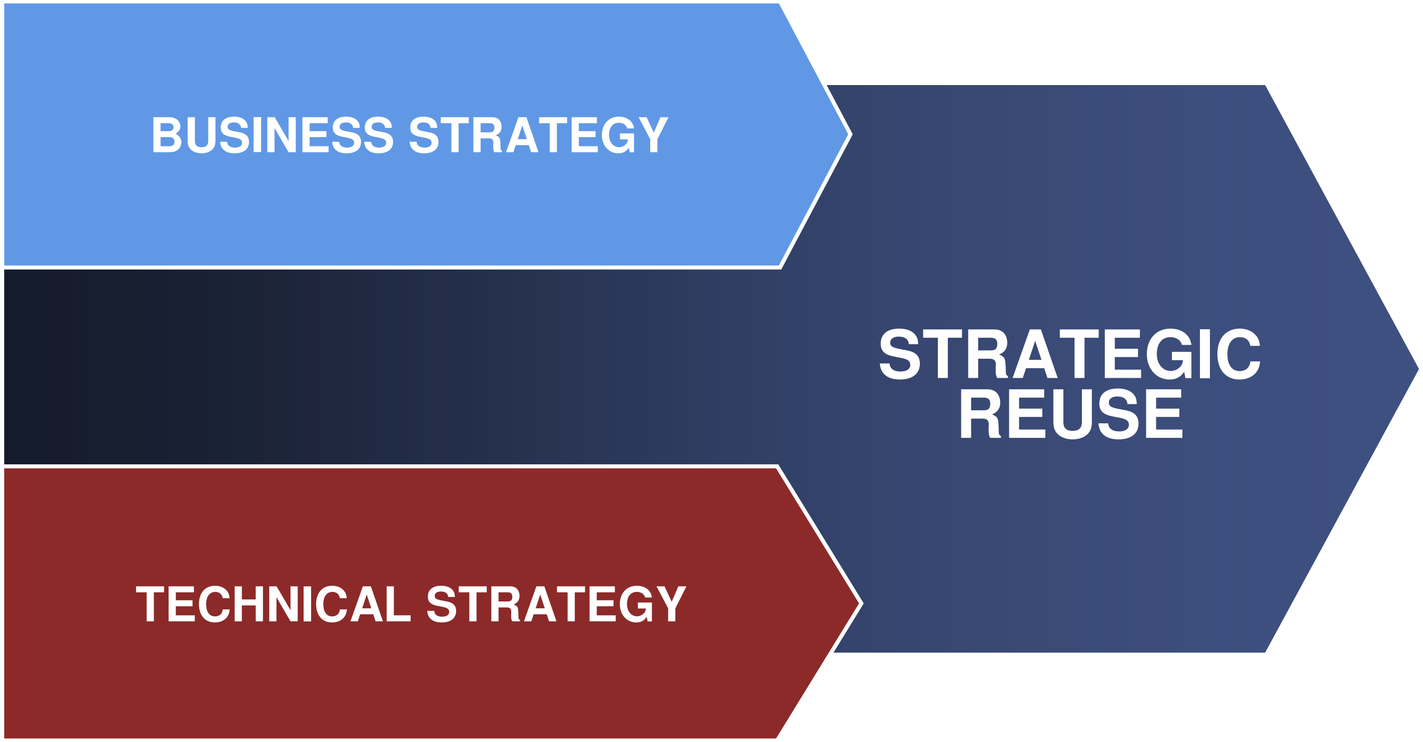
Traditional reuse strategies have had little economic benefit.

Reuse History



Focus was small-grained, opportunistic, and technology-driven. Results did not meet business goals.

Strategic Reuse is Needed for Business Benefits



Product Lines Are a Proven Concept

Product lines – building a family of products from interchangeable parts – have existed for centuries.

- Li Chieh, the state architect of the Chinese emperor Hui-tsung, published a set of building codes for official buildings. This book defined a set of reusable designs: a “product line” of buildings.
- IBM’s System/360 family of computers, introduced in 1964, was a product line of computers.
- The Airbus A318/319/320/321 family of commercial aircraft is a product line, as is the Boeing 757/767 family.
- McDonalds and Burger King each have a product line of restaurant menu items.

Source: www.burgerking.com

CelsiusTech: Ship System 2000

A family of 55 ship systems built using a *software product line* approach

- Need for developers dropped from 210 to roughly 30.
- Time to field decreased from about 9 years to about 3 years.
- Integration test of 1-1.5 million SLOC requires 1-2 people.
- Rehosting to a new platform/OS takes 3 months.
- Cost and schedule targets are predictably met.
- Customer satisfaction is high.

What Is A Software Product Line?

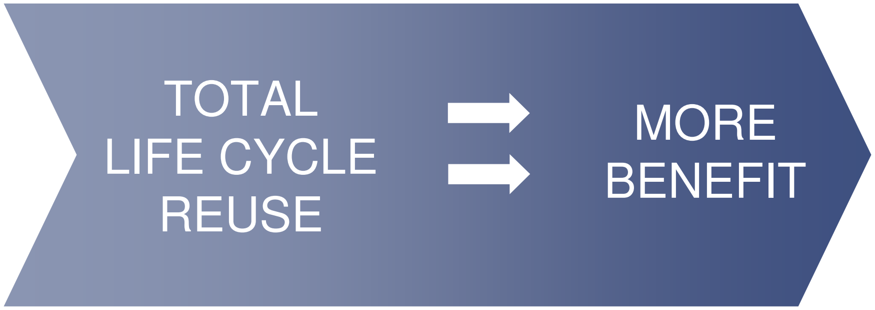
A *software product line* is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way.

- a new application of a proven concept
- an innovative, growing concept in software engineering

How Do Product Lines Help?

Product lines amortize the investment in these and other core assets:

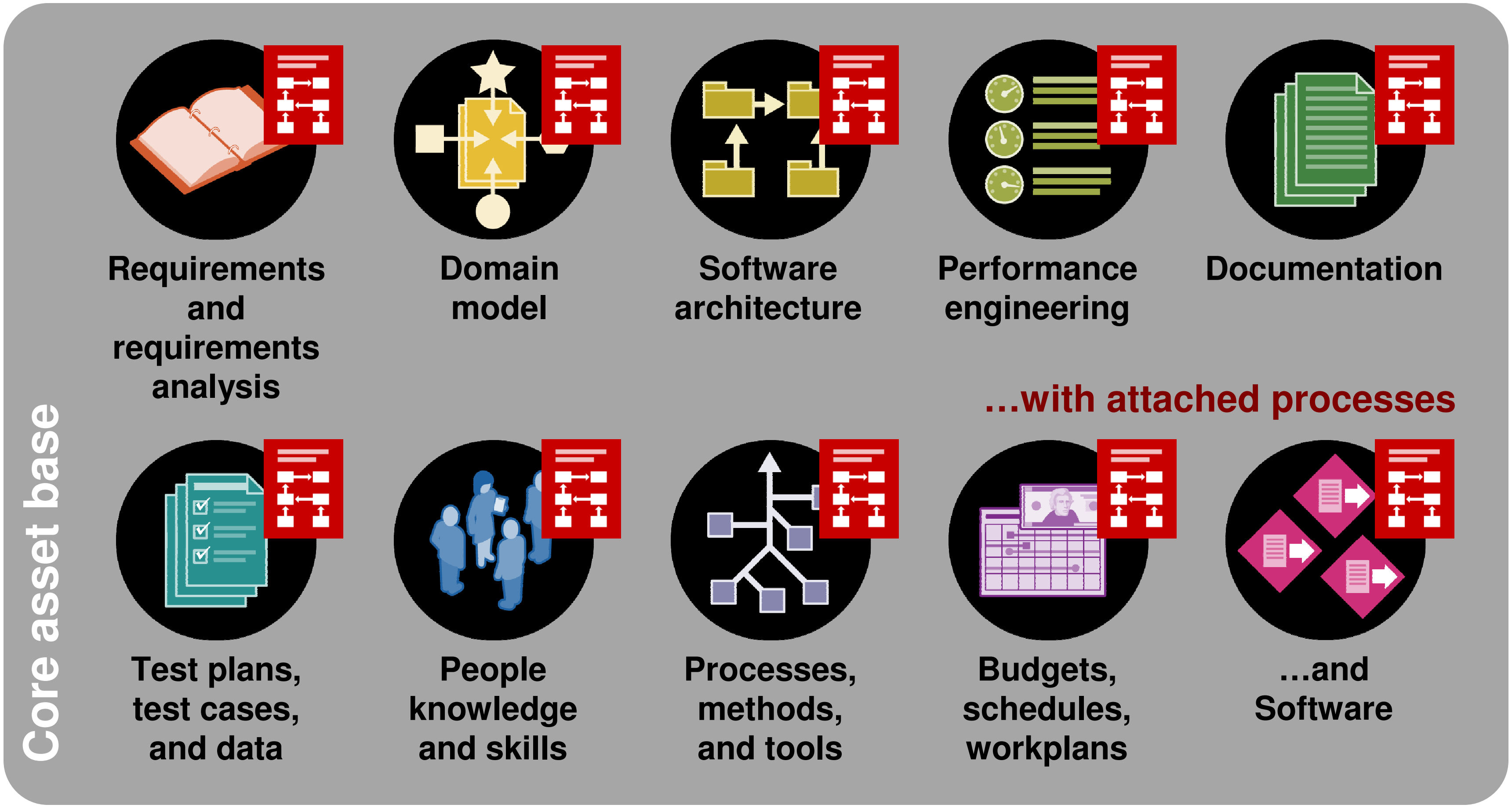
- requirements and requirements analysis
- domain model
- software architecture and design
- performance engineering
- documentation
- test plans, test cases, and test data
- people: their knowledge and skills
- processes, methods, and tools
- budgets, schedules, and work plans
- components and services



TOTAL LIFE CYCLE REUSE ⇒ MORE BENEFIT

Building the Core Asset Base

Core assets include:



Core asset base

Requirements and requirements analysis

Domain model

Software architecture

Performance engineering

Documentation

...with attached processes

Test plans, test cases, and data

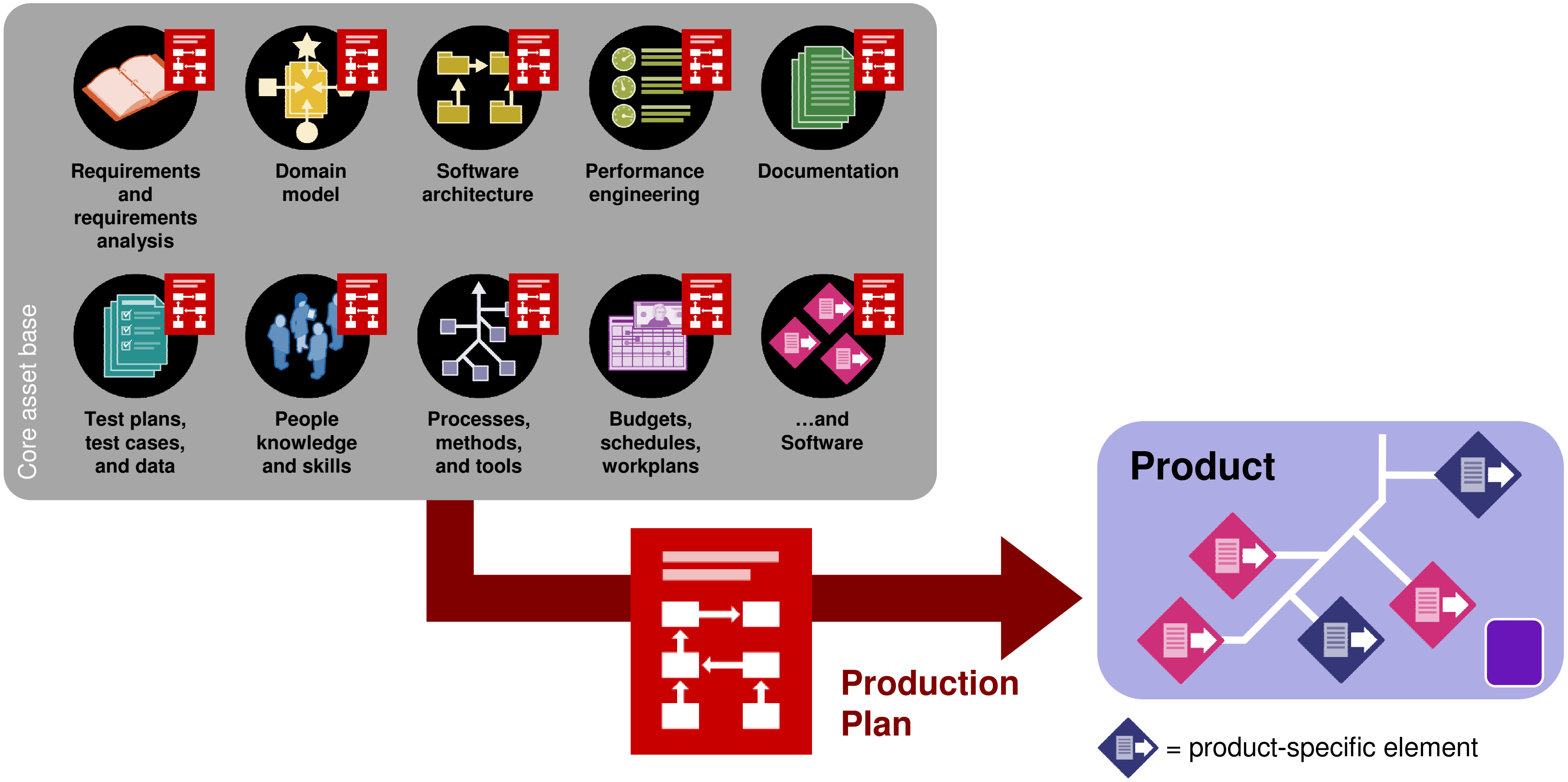
People knowledge and skills

Processes, methods, and tools

Budgets, schedules, workplans

...and Software

Building a Product...



Core asset base

Requirements and requirements analysis

Domain model

Software architecture

Performance engineering

Documentation

Test plans, test cases, and data

People knowledge and skills

Processes, methods, and tools

Budgets, schedules, workplans

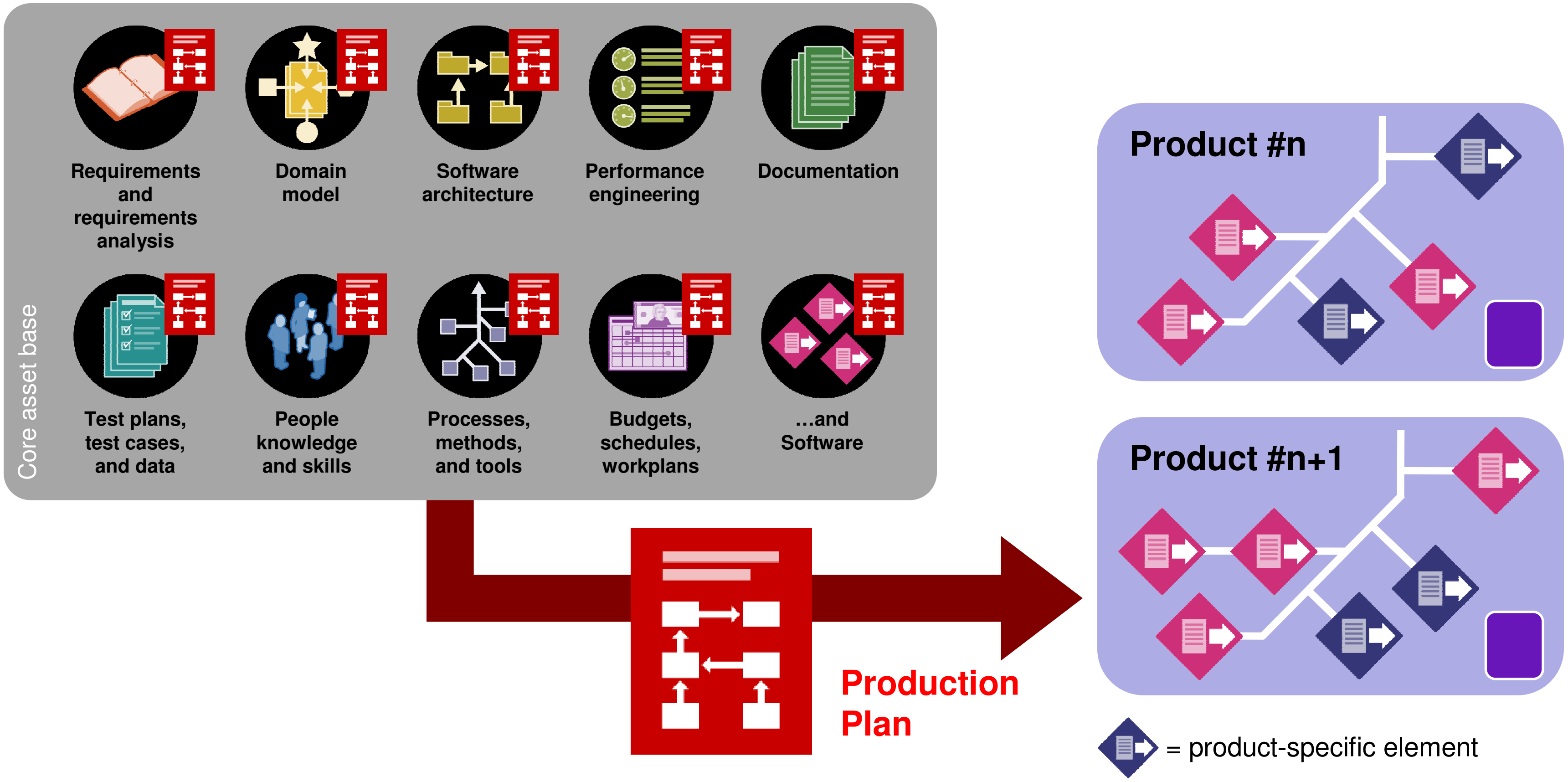
...and Software

Production Plan

Product

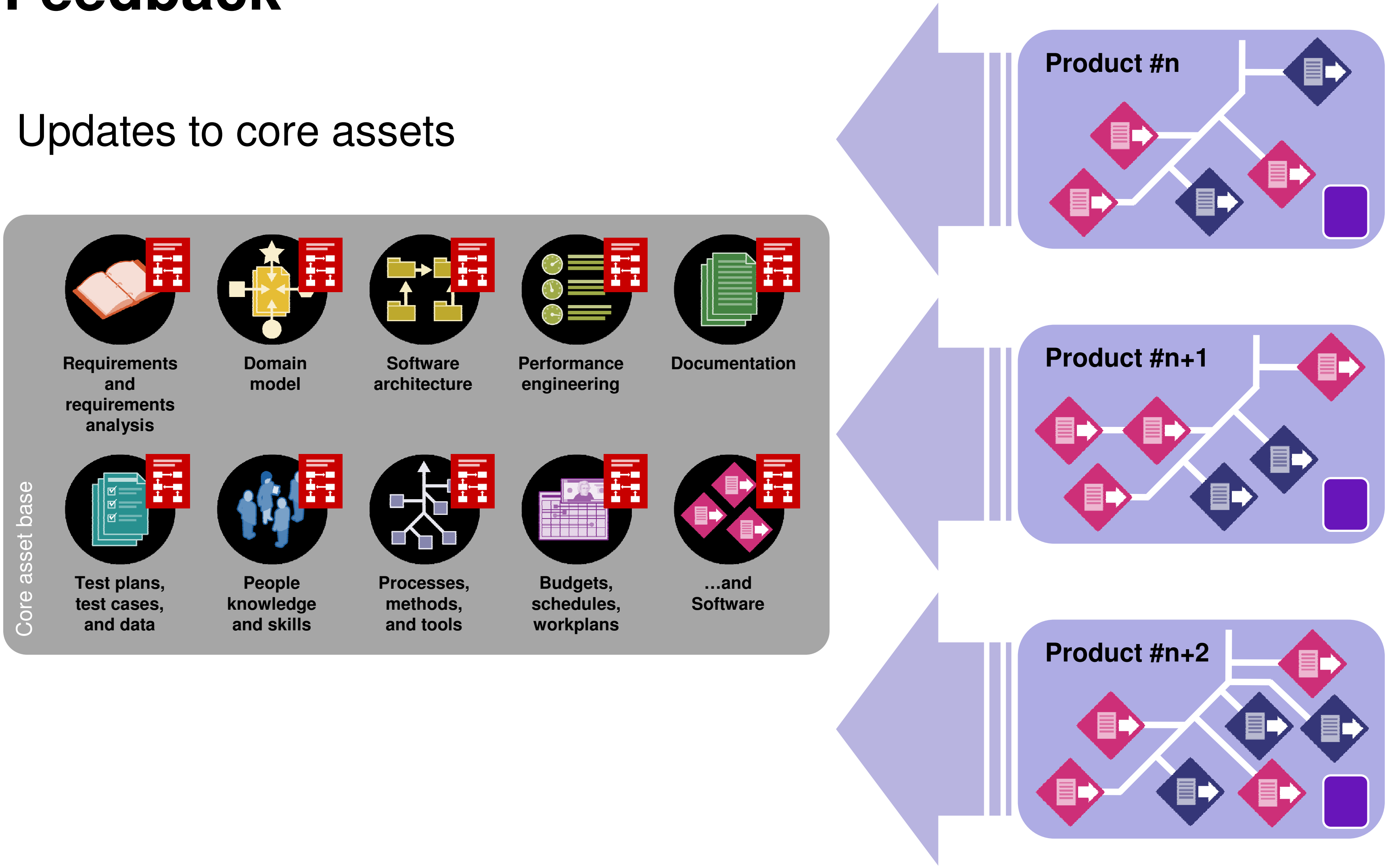
= product-specific element

Building Subsequent Products...


Core asset base
Requirements and requirements analysis
Domain model
Software architecture
Performance engineering
Documentation
Test plans, test cases, and data
People knowledge and skills
Processes, methods, and tools
Budgets, schedules, workplans
...and Software
Production Plan
Product #n
Product #n+1
= product-specific element

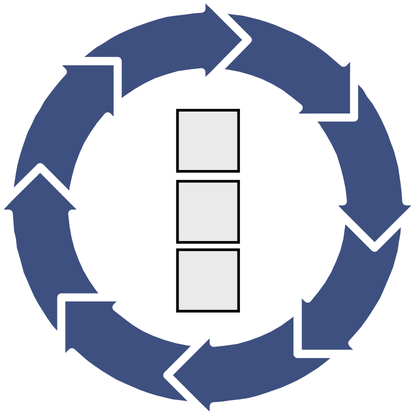
Feedback

Updates to core assets

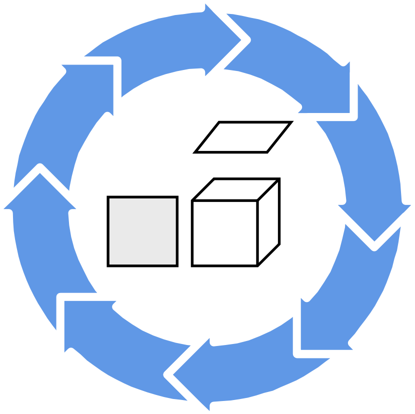

Core asset base
Requirements and requirements analysis
Domain model
Software architecture
Performance engineering
Documentation
Test plans, test cases, and data
People knowledge and skills
Processes, methods, and tools
Budgets, schedules, workplans
...and Software
Product #n
Product #n+1
Product #n+2

The Key Concepts

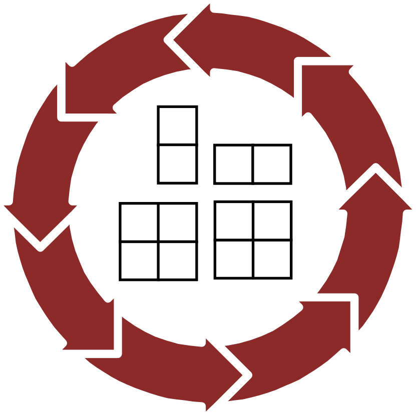
Use of a core
asset base



in production



of a related
set of products

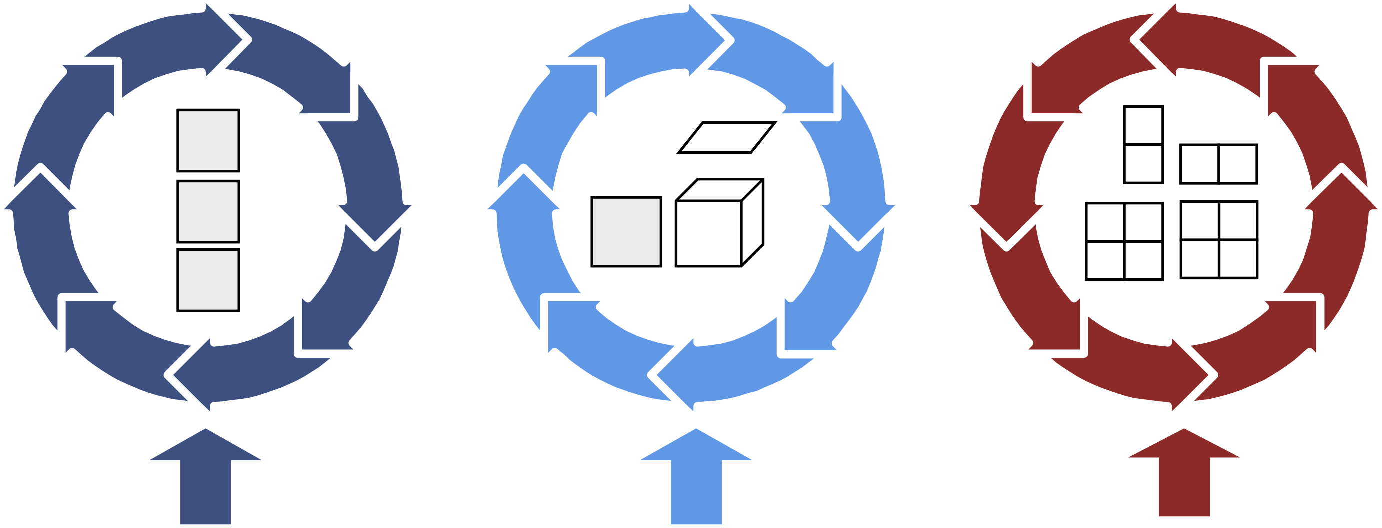


The Key Concepts

Use of a core asset base

in production

of a related set of products



Architecture

Production Plan

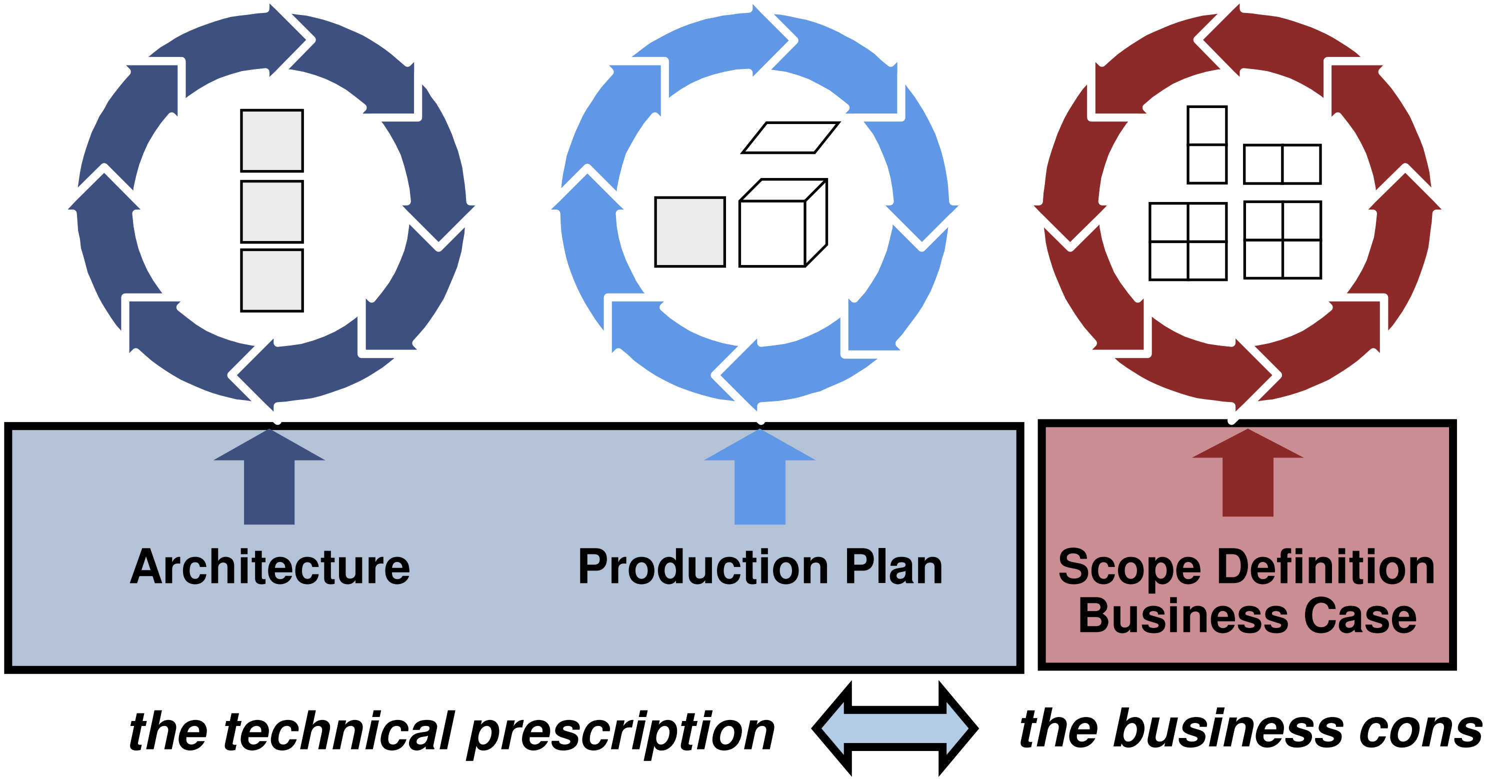
Scope Definition
Business Case

The Key Concepts

Use of a core asset base

in production

of a related set of products



Architecture

Production Plan

Scope Definition
Business Case

the technical prescription *the business constraints*

Real World Motivation

Organizations use product line practices to:

- achieve large scale productivity gains
- improve time to market
- maintain market presence
- sustain unprecedented growth
- achieve greater market agility
- compensate for an inability to hire
- enable mass customization
- get control of diverse product configurations
- improve product quality
- increase customer satisfaction
- increase predictability of cost, schedule, and quality

The actual business goals impact the production strategy and the architecture – both of which manage the variation essential to achieving business goals.

What happened next?

Other companies were trying this approach

- on varying scales
- with varying success

Topics of interest at the time

- Domain analysis
- Software architecture
- Process improvement
- Component-based software development

- In 1996, the SEI published the CelsiusTech case study.
- In 1997, the SEI held its first product line workshop.
- In 2000, the first International Software Product Line Conference was held.

Cummins Inc.: Diesel Control Systems

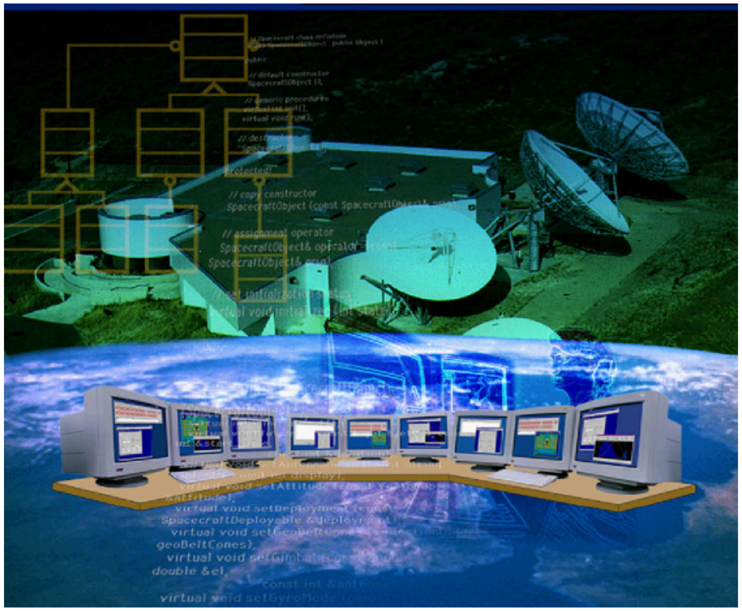
Over 20 product groups with over 1,000 separate engine applications

- Product cycle time was slashed from 250 person-months to a few person-months.
- Build and integration time was reduced from one year to one week.
- Quality goals are exceeded.
- Customer satisfaction is high.
- Product schedules are met.

National Reconnaissance Office/ Raytheon: Control Channel Toolkit

Ground-based spacecraft command and control systems

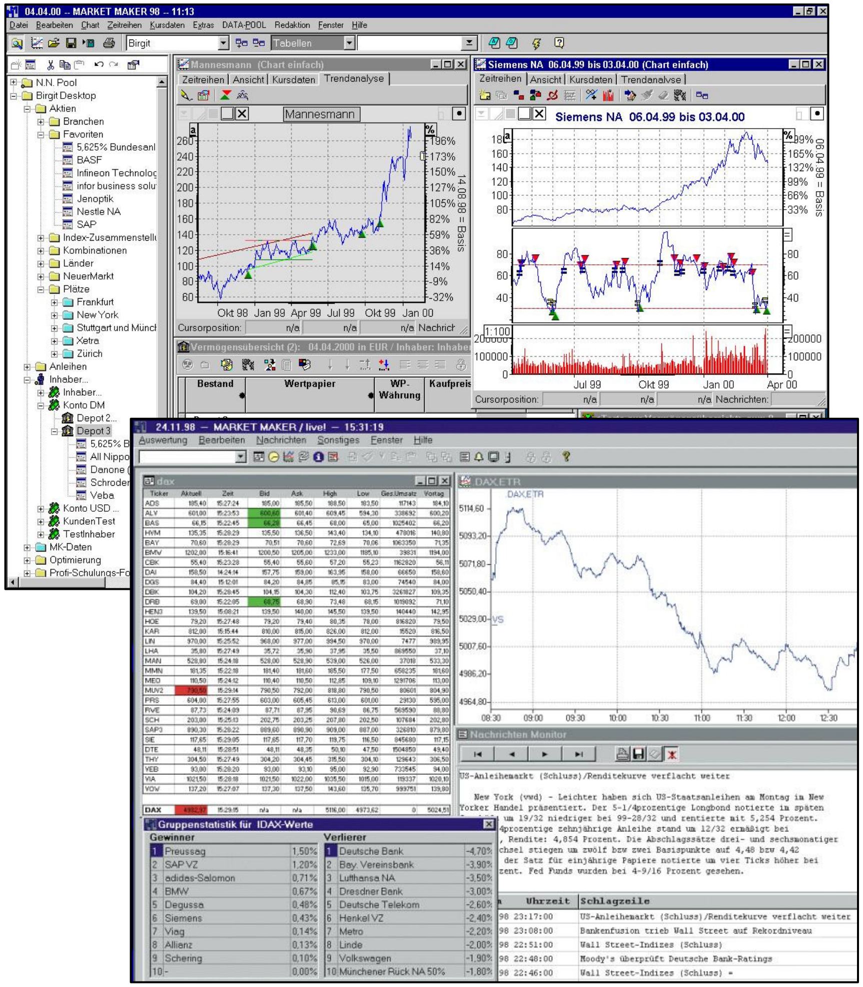
- First system had 10 times fewer defects than usual.
- The incremental build time was reduced from months to weeks.
- The system development time and costs decreased by 50%.
- There was decreased product risk.



Market Maker GMBH: Merger

Internet-based stock market software

- Each product is “uniquely” configured.
- Putting up a customized system takes three days.



Ticker	Aktuell	Zeit	Bid	Ask	High	Low	Ges.Umsatz	Vortag
ADS	185,40	15:27:24	185,00	185,50	188,50	183,50	117143	184,10
ALV	601,00	15:23:53	600,60	601,40	609,45	594,30	338692	600,20
BAS	66,15	15:22:45	66,28	66,45	68,00	65,00	1025402	66,20
HVM	135,35	15:28:29	135,50	136,50	143,40	134,10	478016	140,80
BAY	70,60	15:28:29	70,51	70,60	72,69	70,06	1063350	71,35
BMW	1202,00	15:16:41	1200,50	1205,00	1233,00	1185,10	39831	1194,00
CBK	55,40	15:23:28	55,40	55,60	57,20	55,23	1162820	56,11
DAI	158,50	14:24:14	157,75	159,00	163,95	158,00	66650	158,60
DGS	84,40	15:12:01	84,20	84,85	85,15	83,00	74540	84,00
DBK	104,20	15:28:45	104,15	104,30	112,40	103,75	3261827	109,35
DRB	69,00	15:22:05	68,75	68,90	73,48	68,15	1019092	71,10
HEN3	139,50	15:08:21	139,50	140,00	145,50	139,50	140440	142,95
HOE	79,20	15:27:48	79,20	79,40	80,35	78,00	816820	79,50
KAR	812,00	15:15:44	810,00	815,00	826,00	812,00	15520	816,50
LIN	970,00	15:25:52	968,00	977,00	994,50	970,00	7477	989,95
LHA	35,80	15:27:49	35,72	35,90	37,95	35,50	869550	37,10
MAN	528,80	15:24:18	528,00	528,90	539,00	526,00	37018	533,30
MMN	181,35	15:22:18	181,40	181,60	185,50	177,50	658235	181,60
MEO	110,50	15:24:12	110,40	110,50	112,85	109,10	1291706	113,00
MUV2	790,50	15:29:14	790,50	792,00	818,80	790,50	80601	804,90
PRS	604,00	15:27:55	603,00	605,45	613,00	601,00	29130	595,00
RWE	87,73	15:24:09	87,71	87,95	90,69	86,75	569590	88,80
SCH	203,00	15:25:13	202,75	203,25	207,80	202,50	107684	202,80
SAP3	890,30	15:28:22	889,60	890,90	909,00	887,00	326810	879,80
SIE	117,65	15:29:05	117,65	117,70	119,75	116,50	845680	117,15
DTE	48,11	15:28:51	48,11	48,35	50,10	47,50	1504850	49,40
THY	304,50	15:27:49	304,20	304,45	315,50	304,10	129643	306,50
VEB	93,00	15:28:20	93,00	93,10	95,00	92,90	733545	94,00
VIA	1021,50	15:28:18	1021,50	1022,00	1035,50	1015,00	119337	1020,10
VOW	137,20	15:27:07	137,30	137,50	143,60	135,70	999751	139,80
DAX	4982,97	15:29:15	n/a	n/a	5116,00	4973,62	0	5024,51

Gruppenstatistik für IDAX-Werte

Gewinner			Verlierer		
1	Preussag	1,50%	1	Deutsche Bank	-4,70%
2	SAP VZ	1,20%	2	Bay. Vereinsbank	-3,90%
3	adidas-Salomon	0,71%	3	Lufthansa NA	-3,50%
4	BMW	0,67%	4	Dresdner Bank	-3,00%
5	Degussa	0,48%	5	Deutsche Telekom	-2,60%
6	Siemens	0,43%	6	Henkel VZ	-2,40%
7	Viag	0,14%	7	Metro	-2,20%
8	Allianz	0,13%	8	Linde	-2,00%
9	Schering	0,10%	9	Volkswagen	-1,90%
10	-	0,00%	10	Münchener Rück NA 50%	-1,80%

US-Anleihemarkt (Schluss)/Renditekurve verflacht weiter

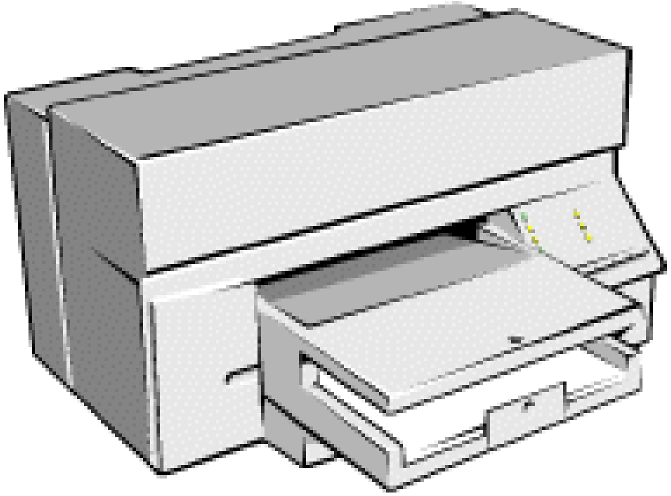
New York (vwd) - Leichter haben sich US-Staatsanleihen am Montag im New Yorker Handel präsentiert. Der 5-1/4prozentige Longbond notierte im späten um 19/32 niedriger bei 99-28/32 und rentierte mit 5,254 Prozent. 4prozentige zehnjährige Anleihe stand um 12/32 ermäßigt bei , Rendite: 4,854 Prozent. Die Abschlagssätze drei- und sechsmonatiger chsel stiegen um zwölf bzw zwei Basispunkte auf 4,48 bzw 4,42 der Satz für einjährige Papiere notierte um vier Ticks höher bei zent. Fed Funds wurden bei 4-9/16 Prozent gesehen.

Uhrzeit	Schlagzeile
98 23:17:00	US-Anleihemarkt (Schluss)/Renditekurve verflacht weiter
98 23:08:00	Bankenfusion trieb Wall Street auf Rekordniveau
98 22:51:00	Wall Street-Indizes (Schluss)
98 22:48:00	Moody's überprüft Deutsche Bank-Ratings
98 22:46:00	Wall Street-Indizes (Schluss) =

Hewlett Packard

Printer systems

- 2-7x cycle time improvement (some 10x)
- 400% productivity improvement
- Sample Project
  - –shipped 5x number of products
  - –that were 4x as complex
  - –and had 3x the number of features
  - –with 4x products shipped/person



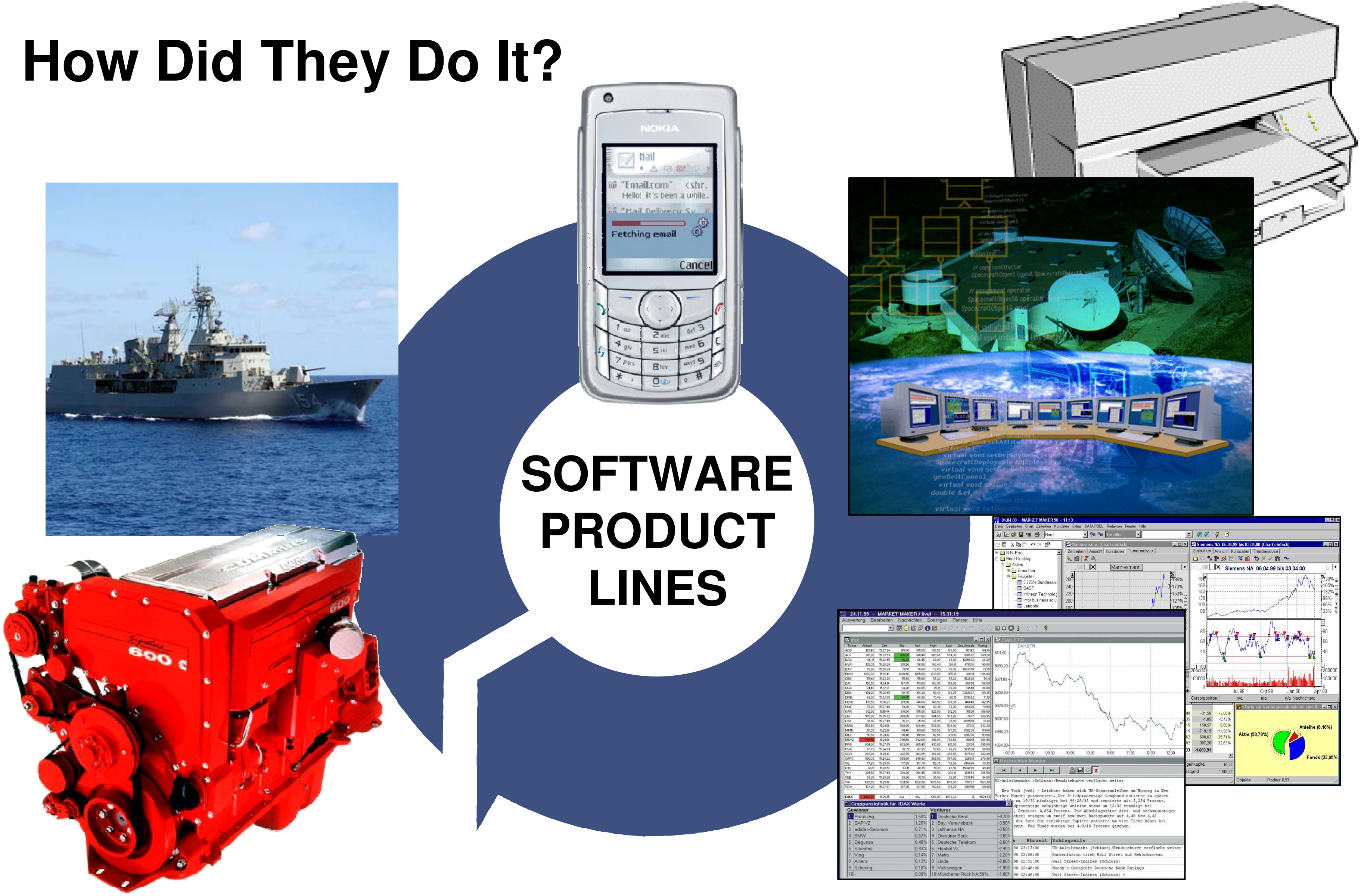
Nokia Mobile Phones

Product lines with 25-30 new products per year versus 5 per year originally.

Across products there are

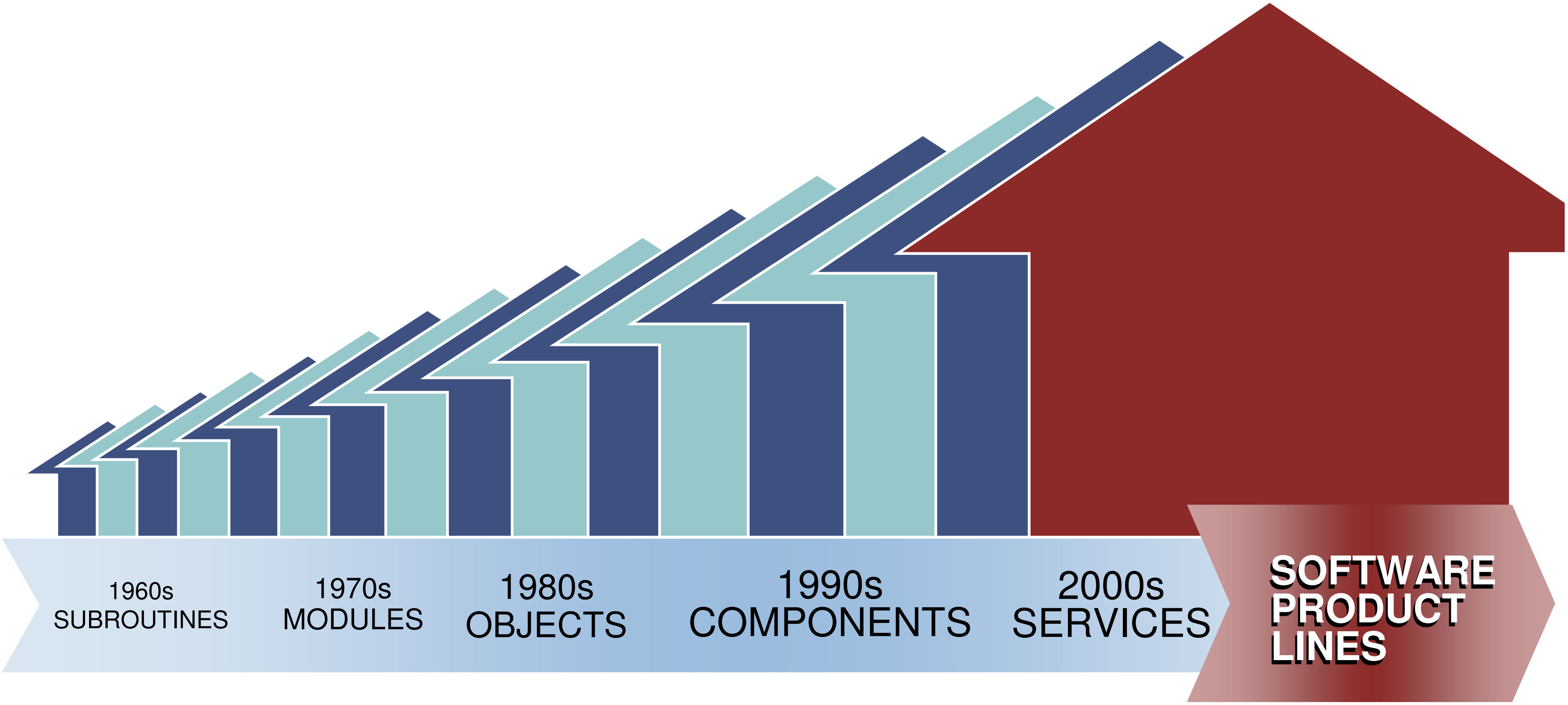
- varying number of keys
- varying display sizes
- varying sets of features
- 58 languages supported
- 130 countries served
- multiple protocols
- needs for backwards compatibility
- configurable features
- needs for product behavior
- change after release

How Did They Do It?



SOFTWARE PRODUCT LINES

Reuse History: From Ad Hoc To Systematic



Software Product Lines Are Not

Clone and own: single-system development with reuse

- modifying code as necessary for the single system only

Fortuitous small-grained reuse

- reuse libraries containing algorithms, modules, objects, or components

Just component-based or service-based development

- selecting components or services from an in-house library, the marketplace, or the Web with no architecture focus

Just versions of a single product

- rather, simultaneous release and support of multiple products

Just a configurable architecture

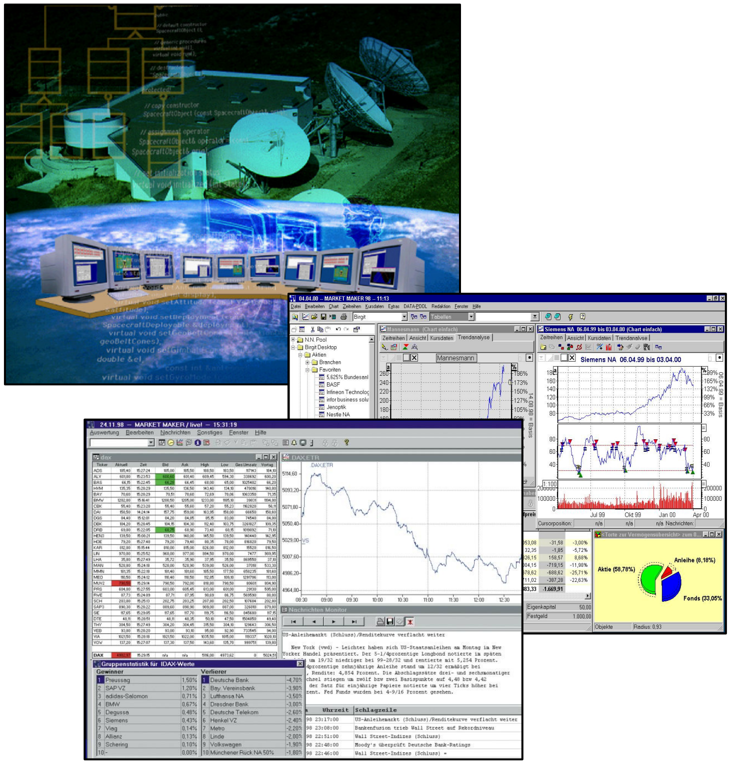
- a good start, but only part of the reuse potential

Just a set of technical standards

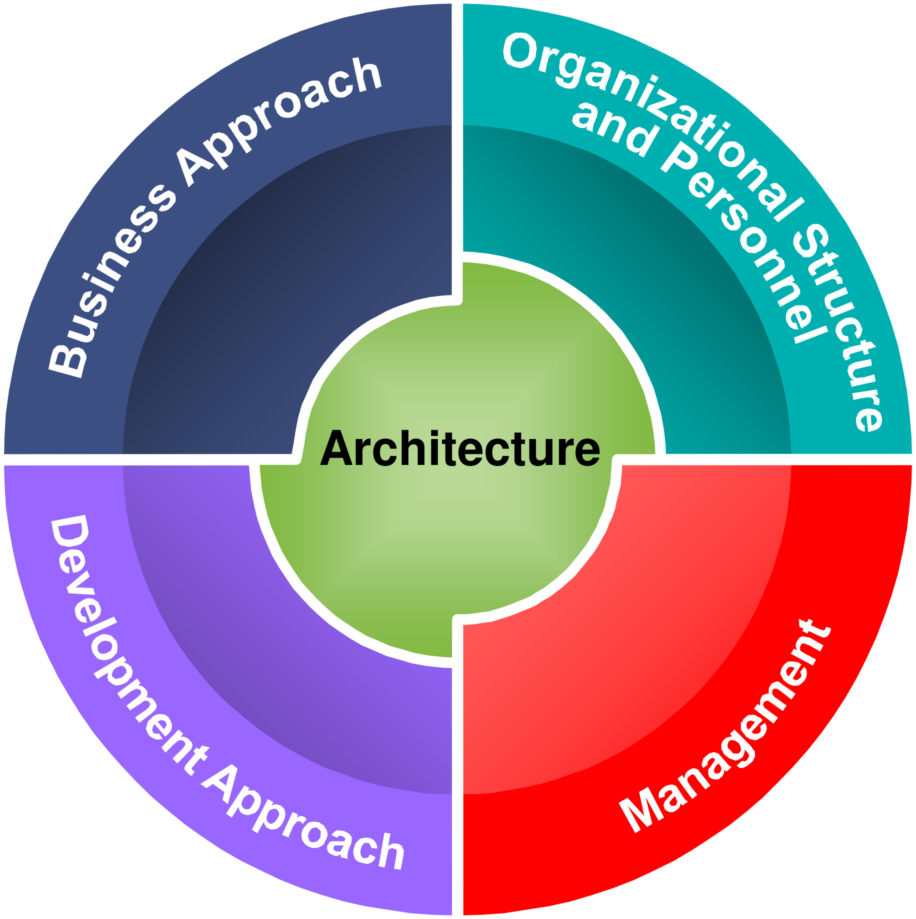
- constraining choices without an architecture-based reuse strategy

Software Product Lines Are

Software product lines involve strategic, planned reuse that yields predictable results.

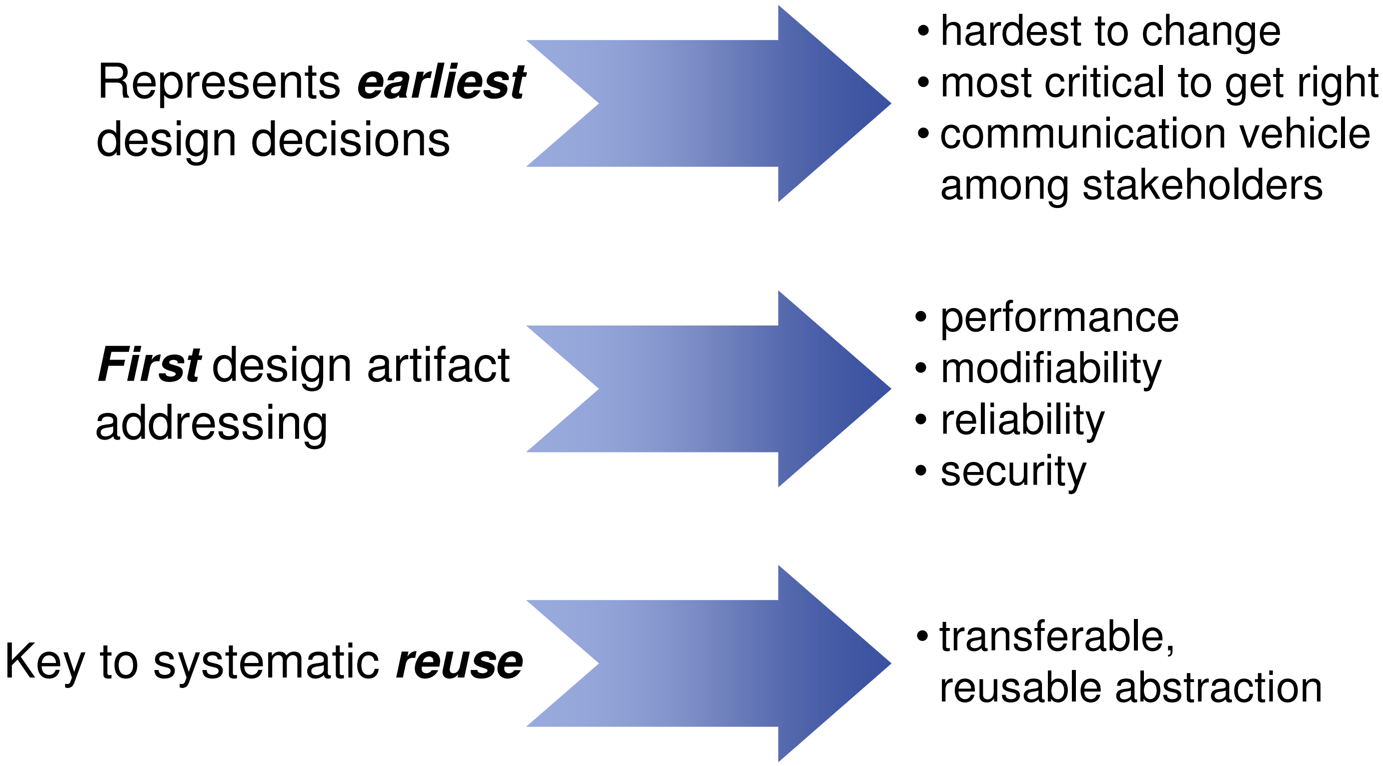


Needed Changes Are Significant



The product line architecture is central to success.

Why Is Software Architecture Important?



Represents ***earliest*** design decisions

- hardest to change
- most critical to get right
- communication vehicle among stakeholders

First design artifact addressing

- performance
- modifiability
- reliability
- security

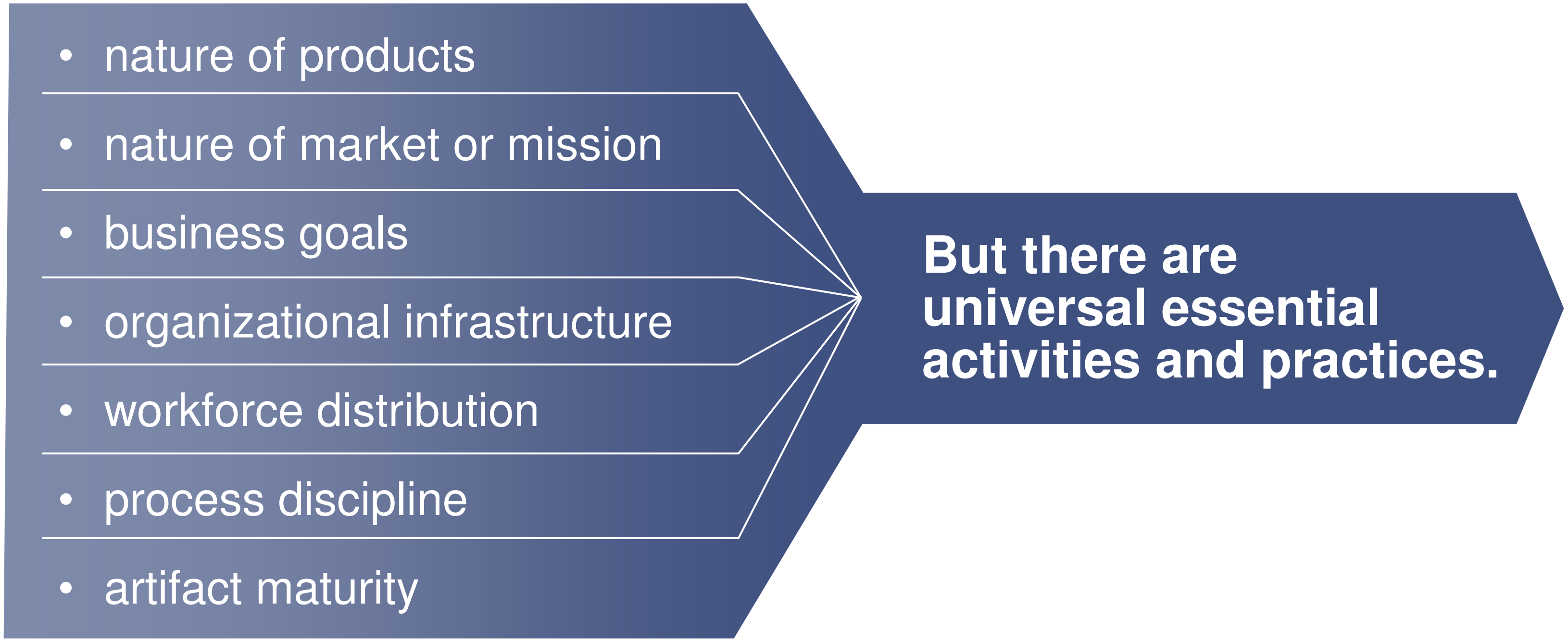
Key to systematic ***reuse***

- transferable, reusable abstraction

The **right architecture** paves the way for system **success.**
The **wrong architecture** usually spells some form of **disaster.**

The Starting Point Makes a Difference

Contexts for product lines vary widely, based on



- nature of products
- nature of market or mission
- business goals
- organizational infrastructure
- workforce distribution
- process discipline
- artifact maturity

But there are universal essential activities and practices.

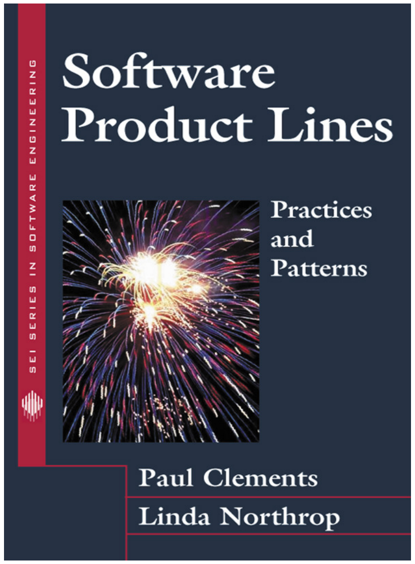
The SEI Framework for Software Product Line Practice^{SM}

The SEI Framework for Software Product Line Practice is a conceptual framework that describes the essential activities and twenty-nine practice areas necessary for successful software product lines.

The Framework, originally conceived in 1998, is evolving based on the experience and information provided by the community.

Version 4.0 –
in *Software Product Lines: Practices and Patterns*

Version 5.0 –
http://www.sei.cmu.edu/productlines/framework.html



[SM] Framework for Software Product Line Practice is a service mark of Carnegie Mellon University.

SEI Information Sources

Case studies, experience reports, and surveys

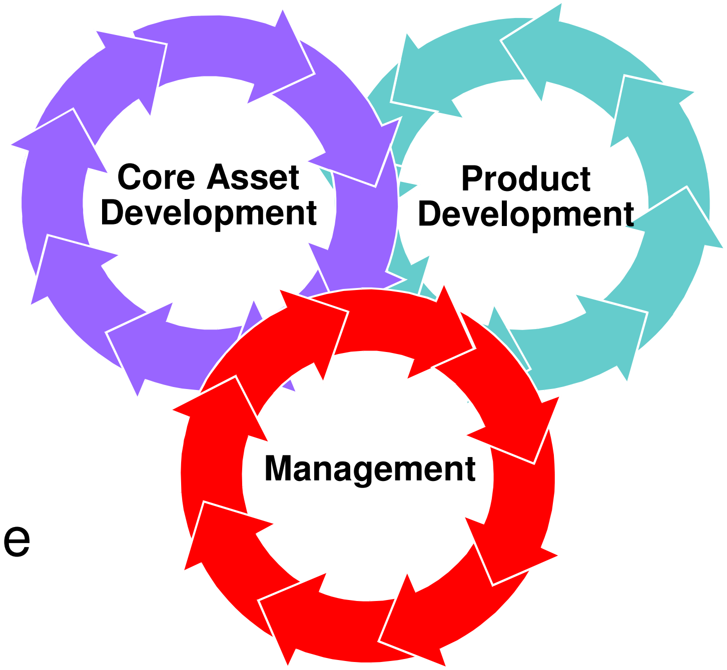
Workshops and conferences

Applied research

Collaborations with customers on actual product lines

Three Essential Activities

All three activities are interrelated and highly iterative.



There is no “first” activity.

- In some contexts, existing products are mined for core assets.
- In others, core assets may be developed or procured for future use.

There is a strong feedback loop between the core assets and the products.

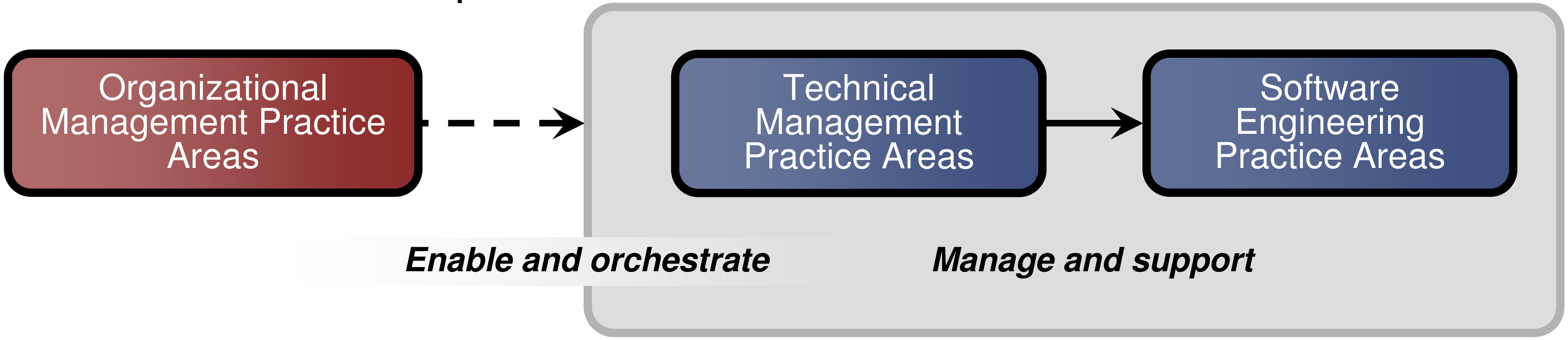
Strong management at multiple levels is needed throughout.

Management oversees core asset and product development.

Management orchestrates all activities and processes needed to make the three essential activities work together.

Driving the Essential Activities

Supporting the essential activities are essential practices that fall into practice areas. A ***practice area*** is a body of work or a collection of activities that an organization must master to successfully carry out the essential work of a product line.

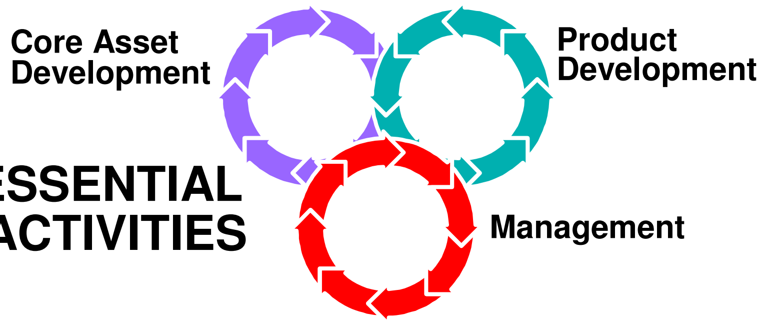

Organizational Management Practice Areas
Technical Management Practice Areas
Software Engineering Practice Areas
Enable and orchestrate
Manage and support

Three Categories Of Practice Areas

The practice areas represent common activities in software development that are adapted to the needs of a product line approach.

Framework Version 4.0

ESSENTIAL ACTIVITIES



Core Asset Development

Product Development

Management

PRACTICE AREAS		
Software Engineering	Technical Management	Organizational Management
Architecture Definition	Configuration Management	Building a Business Case
Architecture Evaluation	Data Collection, Metrics, and Tracking	Customer Interface Management
Component Development	Make/Buy/Mine/Commission Analysis	Developing an Acquisition Strategy
COTS Utilization	Process Definition	Funding
Mining Existing Assets	Scoping	Launching and Institutionalizing
Requirements Engineering	Technical Planning	Market Analysis
Software System Integration	Technical Risk Management	Operations
Testing	Tool Support	Organizational Planning
Understanding Relevant Domains		Organizational Risk Management
		Structuring the Organization
		Technology Forecasting
		Training

The Product Line Architecture

The product line architecture is the single specification that captures the overall architecture of a portfolio of products. It

- applies to all members of the product line (even if their functions and quality attributes differ)
- embodies the commonalities and accommodate the variations needed by the products
- includes specific mechanisms for variation

The variation mechanisms chosen must support business needs.

- the variations reflected in the products
- the production strategy and production constraints
- efficient integration

Each product may have its own architecture, which is an instance of the product line architecture achieved by exercising the variation mechanisms.

The attached process for the product line architecture specifies how to exercise the variation mechanisms.

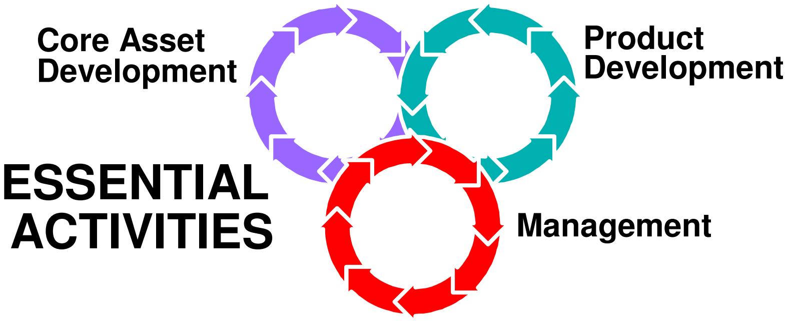
Architecture Variation Mechanisms

- component omission or replication
- component substitution where components may be
  - aspects
  - code components
  - plug-ins
  - services
- parameterization (including macros, templates)
- compile-time selection of different implementations (e.g., *#ifdef*)
- inheritance, specialization, and delegation
- configuration and module interconnection languages
- generation and generators
- application or service component frameworks

Variation mechanisms must support business goals.

Consider the cost to build the core assets and the cost to use them.

Dilemma: How Do You Apply The 29 Practice Areas?



Core Asset Development

Product Development

ESSENTIAL ACTIVITIES

Management

PRACTICE AREAS

Software Engineering | Technical Management | Organizational Management

GUIDANCE

Case Studies

Patterns

Probe

Curriculum

Case Studies

CelsiusTech – CMU/SEI-96-TR-016
http://www.sei.cmu.edu/publications/documents/01.reports/96.tr.016.html

Cummins, Inc. *Software Product Lines: Practices and Patterns*

Market Maker *Software Product Lines: Practices and Patterns*

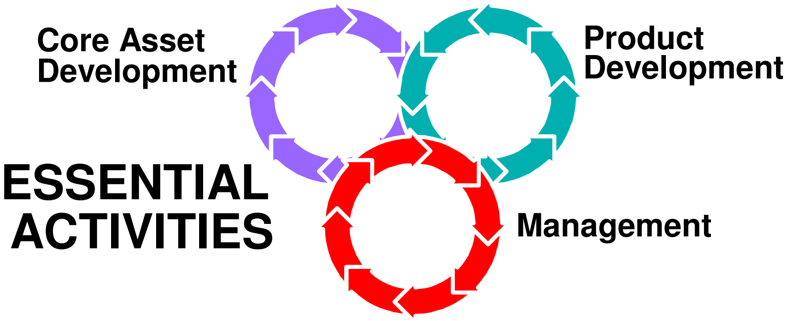
NRO/Raytheon – CMU/SEI-2001-TR-030
http://www.sei.cmu.edu/publications/documents/01.reports/02tr030.html

NUWC – CMU/SEI-2002-TN-018
http://www.sei.cmu.edu/publications/documents/02.reports/02tn018.html

Salion, Inc. – CMU/SEI-2002-TR-038
http://www.sei.cmu.edu/publications/documents/02.reports/02tr038.html

U.S. Army – CMU/SEI-2005-TR-019
http://www.sei.cmu.edu/publications/documents/05.reports/05tr019.html

Help To Make It Happen



PRACTICE AREAS

Software Engineering	Technical Management	Organizational Management

GUIDANCE

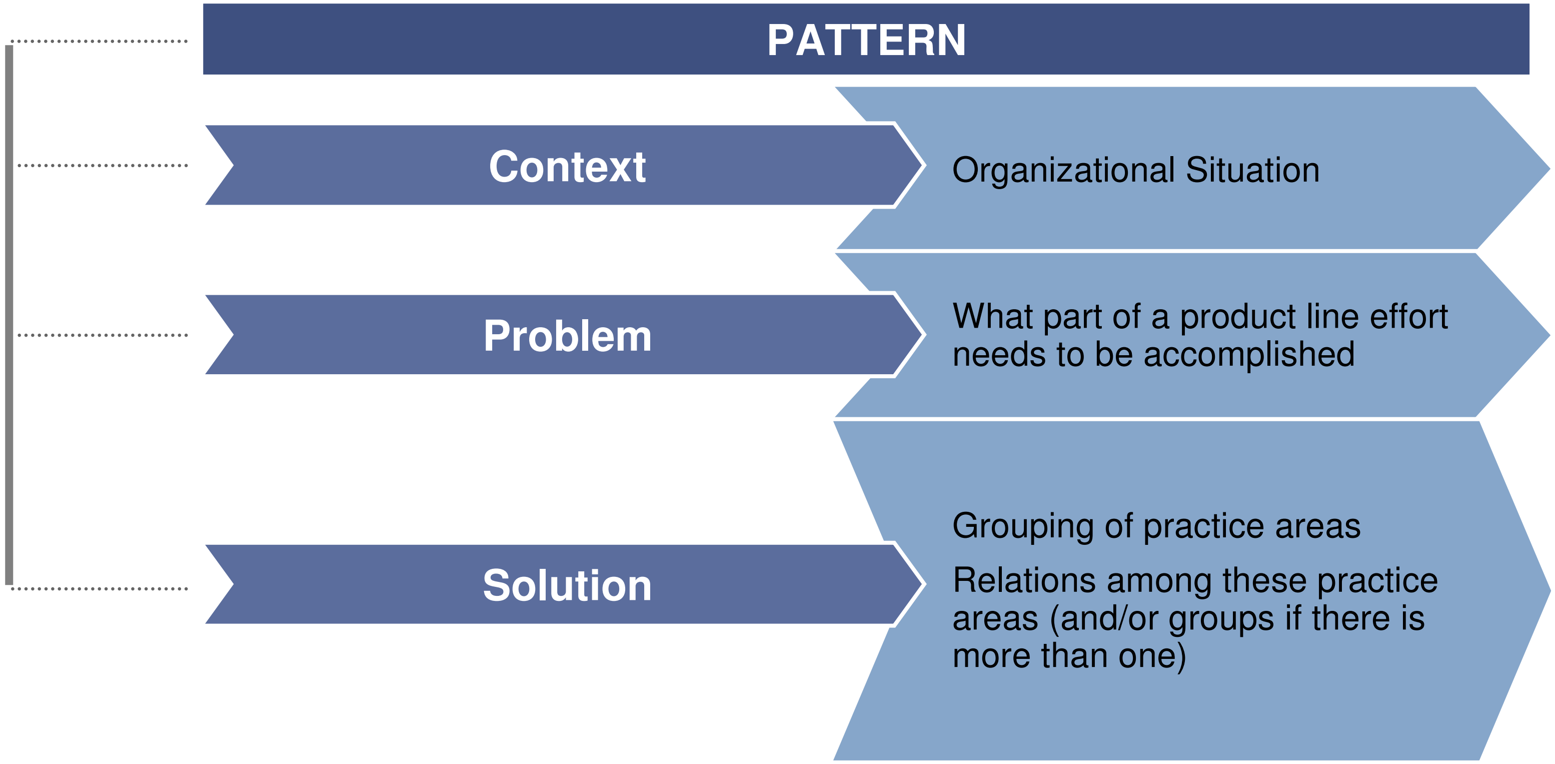
Case Studies

Patterns

Probe

Curriculum

Software Product Line Practice Patterns


PATTERN
Context
Organizational Situation
Problem
What part of a product line effort needs to be accomplished
Solution
Grouping of practice areas
Relations among these practice areas (and/or groups if there is more than one)

What To Build Pattern - 1

Name:

The ***What to Build*** pattern helps an organization determine what products ought to be in its software product line – what products to build.

Context:

An organization has decided to field a software product line and knows the general product area for the set of products.

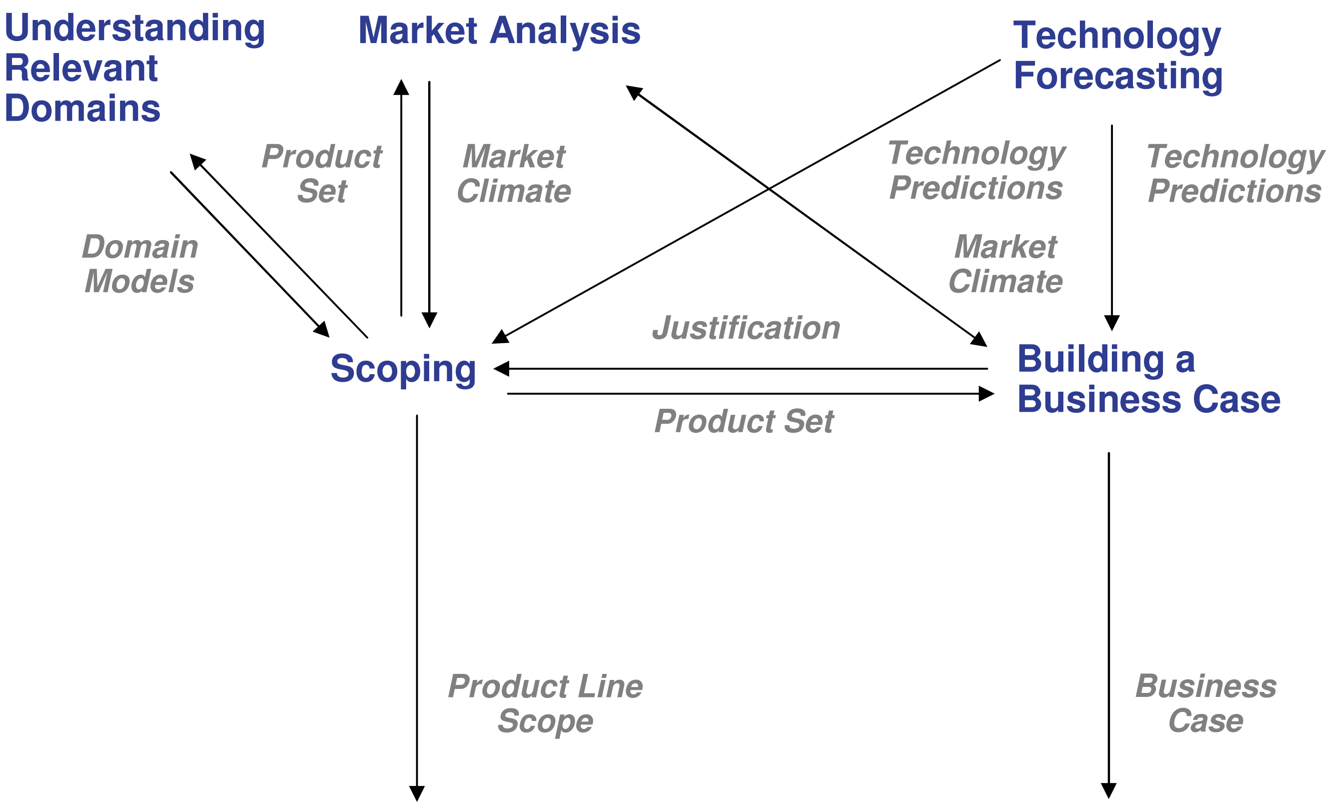
Problem:

To determine what products should be included in the product line

Solution:

Determining what to build requires information related to the product area, technology, and market; the business justification; and the process for describing the set of products to be included in the product line.

What To Build Pattern - 2

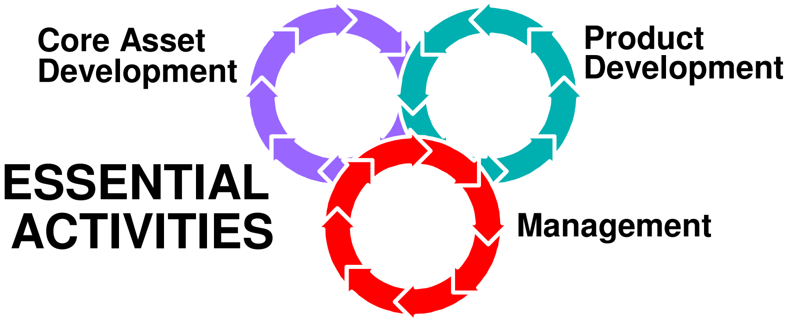


Dynamic Structure

Current Set Of Patterns

Pattern	Variants
Assembly Line	
Cold Start	Warm Start
Curriculum	
Each Asset	Each Asset Apprentice Evolve Each Asset
Essentials Coverage	
Factory	Adoption Factory
In Motion	
Monitor	
Process	Process Improvement
Product Builder	Product Gen
Product Parts	Green Field Barren Field Plowed Field
What to Build	Analysis Forced March

Help To Make It Happen



Core Asset Development

Product Development

ESSENTIAL ACTIVITIES

Management

PRACTICE AREAS

Software Engineering	Technical Management	Organizational Management

GUIDANCE

Case Studies

Patterns

Probe

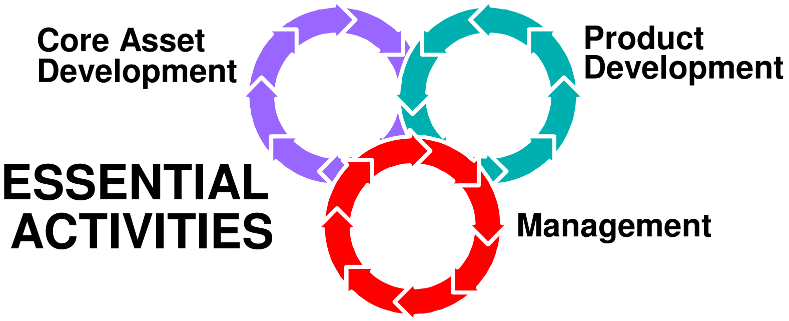
Curriculum

What Is An SEI Product Line Technical Probe (PLTP)?

The SEI PLTP is a method for examining an organization’s readiness to adopt or ability to succeed with a software product line approach.

- It is a diagnostic tool based on the SEI Framework for Software Product Line Practice.
- The 29 practice areas are the basis of data collection and analysis.

Help To Make It Happen



Core Asset Development

Product Development

ESSENTIAL ACTIVITIES

Management

PRACTICE AREAS

Software Engineering | Technical Management | Organizational Management

GUIDANCE

Case Studies

Patterns

Probe

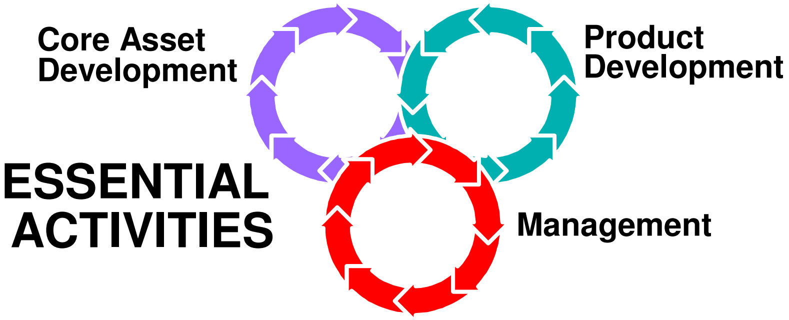
Curriculum

The SEI Software Product Line Curriculum

	Three Certificate Programs		
Five Courses	**Software Product Line Professional**	**PLTP Team Member**	**PLTP Leader**
Software Product Lines	✓	✓	✓
Adopting Software Product Lines	✓	✓	✓
Developing Software Product Lines	✓	✓	✓
PLTP Team Training		✓	✓
PLTP Leader Training			✓
PLTP Lead Observation			✓

✓ : course required to receive certificate

Adding An Adoption Roadmap



PRACTICE AREAS

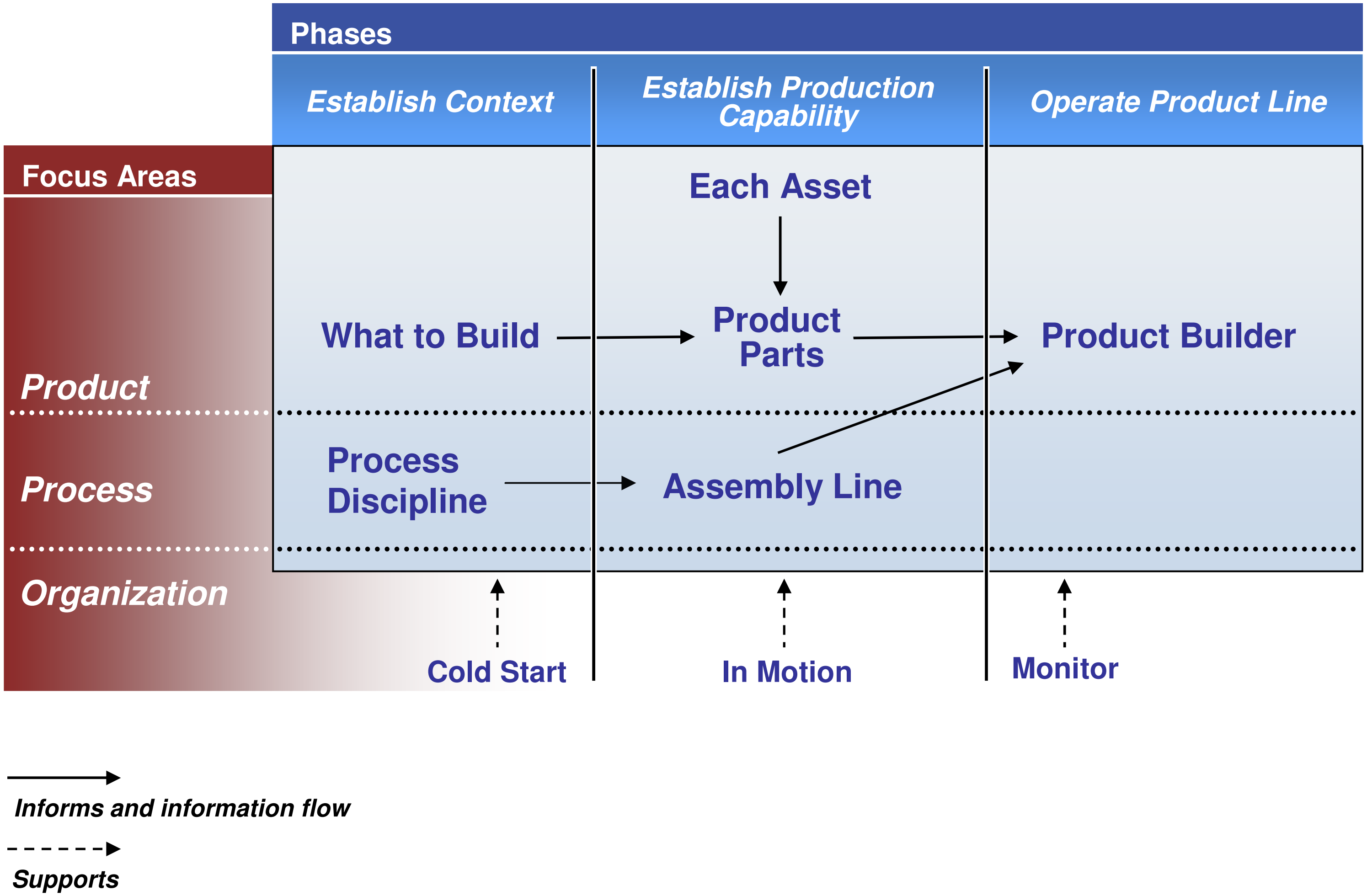
Software Engineering	Technical Management	Organizational Management

GUIDANCE

Case Studies	Patterns	Probe	Curriculum

ADOPTION FACTORY

The SEI Adoption Factory Pattern



Associated Practice Areas

	Establish Context	Establish Production Capability	Operate Product Line
Product	• Marketing Analysis • Understanding Relevant Domains • Technology Forecasting • Building a Business Case • Scoping	• Requirements Engineering • Architecture Definition • Architecture Evaluation • Mining Existing Assets • Component Development • Using Externally Available Software • Software System Integration • Testing	• Requirements Engineering • Architecture Definition • Architecture Evaluation • Mining Existing Assets • Component Development • Using Externally Available Software • Software System Integration • Testing
Process	• Process Discipline	• Make/Buy/Mine/Commission • Configuration Management • Tool Support • Measurement and Tracking • Technical Planning • Technical Risk Management	
Organization	• Launching and Institutionalizing • Funding • Structuring the Organization • Operations • Organizational Planning • Customer Interface Management • Organizational Risk Management • Developing an Acquisition Strategy • Training	• Launching and Institutionalizing • Funding • Structuring the Organization • Operations • Organizational Planning • Customer Interface Management • Organizational Risk Management • Developing an Acquisition Strategy • Training	• Measurement and Tracking • Technical Risk Management • Organizational Risk Management • Customer Interface Management • Organizational Planning

Typical Engineering Organization

You are here

	Establish Context	Establish Production Capability	Operate Product Line
Product	• Marketing Analysis • Understanding Relevant Domains • Technology Forecasting • Building a Business Case • Scoping	• Requirements Engineering • Architecture Definition • Architecture Evaluation • Mining Existing Assets • Component Development • Using Externally Available Software • Software System Integration • Testing	• Requirements Engineering • Architecture Definition • Architecture Evaluation • Mining Existing Assets • Component Development • Using Externally Available Software • Software System Integration • Testing
Process	• Process Discipline	• Make/Buy/Mine/Commission • Configuration Management • Tool Support • Measurement and Tracking • Technical Planning • Technical Risk Management	
Organization	• Launching and Institutionalizing • Funding • Structuring the Organization • Operations • Organizational Planning • Customer Interface Management • Organizational Risk Management • Developing an Acquisition Strategy • Training	• Launching and Institutionalizing • Funding • Structuring the Organization • Operations • Organizational Planning • Customer Interface Management • Organizational Risk Management • Developing an Acquisition Strategy • Training	• Measurement and Tracking • Technical Risk Management • Organizational Risk Management • Customer Interface Management • Organizational Planning

Typical Challenges (in Bold Italics)

You are here

	Establish Context	Establish Production Capability	Operate Product Line
Product	• Marketing Analysis • Understanding Relevant Domains • Technology Forecasting • ***Building a Business Case*** • ***Scoping***	• Requirements Engineering • ***Architecture Definition*** • Architecture Evaluation • Mining Existing Assets • ***Component Development*** • Using Externally Available Software • Software System Integration • Testing	• Requirements Engineering • Architecture Definition • Architecture Evaluation • Mining Existing Assets • Component Development • Using Externally Available Software • Software System Integration • Testing
Process	• Process Discipline	• Make/Buy/Mine/Commission • ***Configuration Management*** • Tool Support • ***Measurement and Tracking*** • Technical Planning • Technical Risk Management	
Organization	• ***Launching and Institutionalizing*** • ***Funding*** • ***Structuring the Organization*** • ***Operations*** • Organizational Planning • Customer Interface Management • Organizational Risk Management • Developing an Acquisition Strategy • Training	• Launching and Institutionalizing • Funding • Structuring the Organization • Operations • Organizational Planning • Customer Interface Management • Organizational Risk Management • Developing an Acquisition Strategy • Training	• Measurement and Tracking • Technical Risk Management • Organizational Risk Management • Customer Interface Management • Organizational Planning

What Resulted

Successful Product Lines

GM

LSI LOGIC®

BOSCH

SALiON™
TARGET. WIN. DELIVER.™

U.S.ARMY®

PHILIPS

Rockwell Collins

Lucent Technologies
Bell Labs Innovations

BOEING

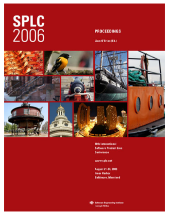
Institutionalized Software Product Lines

Cummins®

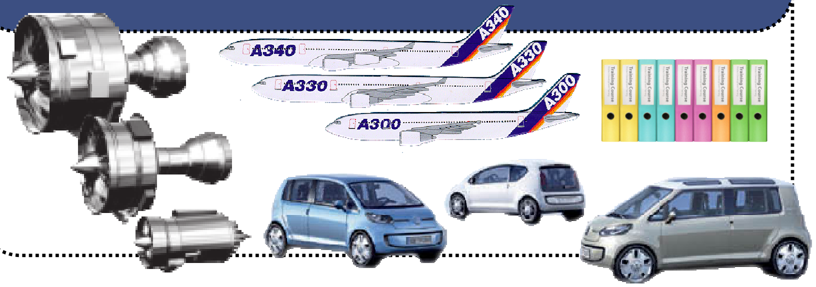
hp
invent

International Conference Series: SPLC

SPLC



Software Product Line Concepts Applied in Non-Software Areas



Other Research and Books

Software Product-Line Engineering
A Family-Based Software Development Process
David M. Weiss
Chi Tau Robert Lai
Foreword by David Lorge Parnas
Jan Bosch
Design & Use of Software Architectures
Adopting and evolving a product-line approach
Klaus Pohl
Günter Böckle
Frank van der Linden
Software Product Line Engineering
Foundations, Principles, and Techniques
Springer
Frank van der Linden
Klaus Schmid
Eelco Rommes
Software Product Lines in Action
The Best Industrial Practice in Product Line Engineering
Springer

Widespread Use of Software Product Lines

Successful software product lines have been built for families of among other things

- mobile phones
- command and control ship systems
- satellite ground station systems
- avionics systems
- command and control/situation awareness systems
- pagers
- engine control systems
- mass storage devices
- billing systems
- web-based retail systems
- printers
- consumer electronic products
- acquisition management enterprise systems
- financial and tax systems
- medical devices
- farm fish management software

2009: Catalog of Over 45 Published Successes

Company	Software Product Line(s)	Company	Software Product Line(s)
AKVAsmart ASA	Feed control and fish farm mgt. software	*Nokia*	Mobile phones
Argon Engineering	Various	*Nokia*	Mobile Browsers
Asea Brown Boveri (ABB)	Gas turbines	*Nokia Networks*	Telecommunication network products for public, private, and cellular networks
Axis Communications AB	Train control; Computer printer servers, storage servers, network camera and scanner servers.	*Nortel*	Digital loop carriers for telecommunications
Boeing	Avionics	*Philips*	High-end televisions
CelsiusTech	Naval command and control	*Philips*	PKI telecommunications switching system
Cummins	Diesel engine controls	*Philips Medical*	Diagnostic imaging equipment
Deutsche Bank	Financial global transaction/settlement	*Raytheon/U.S. NRO*	Satellite ground control station software
Dialect Solutions	Internet payment gateway infrastructures	*Ricoh*	Office appliances
DNV Software	Software products and customized solutions for transportation industries	*Robert Bosch Corp.*	Automotive gasoline systems
E-COM Technology Ltd.	Medical imaging workstations	*Rockwell Collins*	Commercial flight control system avionics
Enea	Real-time OS for telecom and automotive applications and middleware	*Rockwell Collins*	Avionics for U.S. Army helicopters
Ericsson	Telecommunications Switches	*Salion, Inc.*	Revenue acquisition management systems
Ericsson Mobile Data	Packet based mobile communication	*Securitas Larm AB*	Safety and security systems
General Motors P'train	Engines/transmission/controllers s/w	*Siemens*	Software for viewing and quantifying radiological images
Hewlett Packard	Firmware for computer peripherals	*Symbian*	EPOC operating system
LG Industrial Systems	Elevator control systems	*Telvent*	Industrial supervisory control and business process management systems
LSI Logic - Engenio Group	RAID controller firmware for disk storage	*Testo*	Climate and flue gas measurement devices
Lucent Technologies	5ESS telecommunications switch	*U. S. Navy*	Test range facilities
Market Maker Software	Stock market data / financial news mgt.	*U.S. Army*	Command and control simulator for fire support
NASA Jet Propulsion Lab	Interferometer product line	*Unnamed*	Legal expert systems

www.sei.cmu.edu/productlines

Specific Examples - 1

akvasmart

Feed control and farm management software

BOEING

Bold Stroke Avionics

E-COM Technology Ltd.

Medical imaging workstations

hp invent

Firmware for computer peripherals

Lucent Technologies
Bell Labs Innovations

5ESS telecommunications switch

ABB Asea Brown Boveri

Gas turbines, train control, semantic graphics framework

Dialect

Internet payment gateway infrastructure products

ERICSSON

AXE family of telecommunications switches

LG

Elevator control systems

NOKIA

Mobile phones, mobile browsers, telecom products for public, private and cellular networks

AXIS COMMUNICATIONS

Computer printer servers, storage servers, network camera and scanner servers

DNV

Customized solutions for transportation industries

GM

Software for engines, transmissions and controllers

LSI LOGIC

RAID controller firmware for disk storage units

NASA

Interferometer product line

Specific Examples - 2

PHILIPS

High-end televisions,
PKI telecommunications switching
system, diagnostic imaging equipment

Rockwell Collins

Commercial flight control system avionics,
Common Army Avionics System (CAAS),
U.S. Army helicopters

symbian

EPOC operating system

NAVSEA WARFARE CENTERS

Test range facilities

RICOH

Office appliances

SALION
TARGET. WIN. DELIVER.

Revenue acquisition
management systems

TELVENT

Industrial supervisory control
and business process
management systems

U.S. ARMY

Command and
control simulator for
Army fire support

BOSCH

Automotive gasoline
systems

SIEMENS

Software for viewing and
quantifying radiological images

testo Climate and flue gas
measurement devices

alltel FIDELITY NATIONAL FINANCIAL

Support software

MOTOROLA

Pagers product line

Software Product Lines Value Proposition

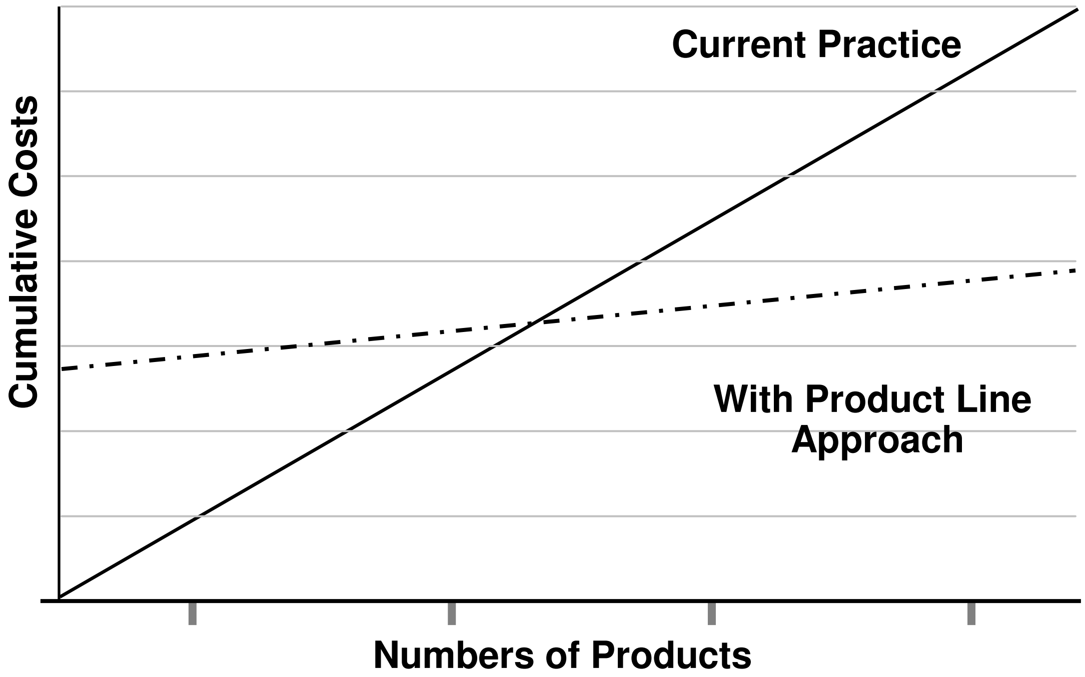
The systematic use of software product line practices results in significant organizational benefits including

- increased quality
  - by as much as 10x
- decreased cost
  - by as much as 60%
- decreased labor needs
  - by as much as 87%
- decreased time to market (to field, to launch...)
  - by as much as 98%
- ability to move into new markets
  - in months, not years

Costs Of A Software Product Line

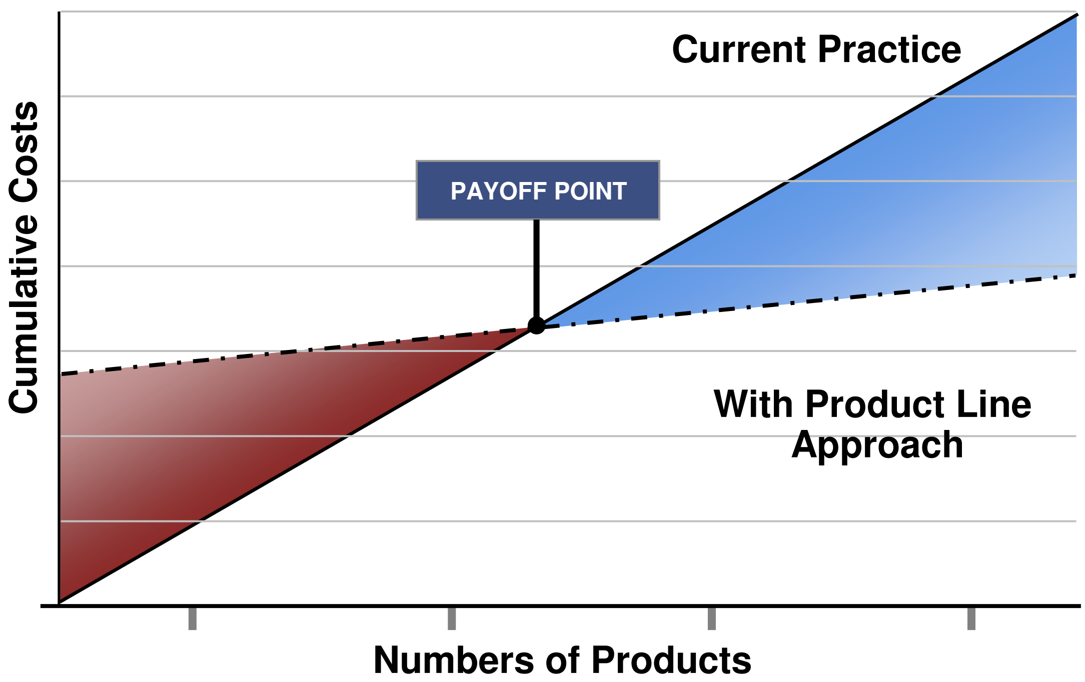
Core Assets	Costs
Architecture	Must support variation inherent in the product line
Software Components	Must be designed to be general without a loss of performance; must build in support for variation points
Test Plans, Test Cases, Test Data	Must consider variation points and multiple instances of the product line
Business Case and Market Analysis	Must address a family of software products, not just one product
Project Plans	Must be generic or be made extensible to accommodate product variations
Tools and Processes	Must be more robust
People, Skills, Training	Must involve training and expertise centered around the assets and procedures associated with the product line

Economics Of Product Lines



Weiss. D.M. & and Lai, C.T.R..
Software Product-Line Engineering: A Family-Based Software Development Process
Reading, MA: Addison-Wesley, 1999.

Economics Of Product Lines



Weiss. D.M. & and Lai, C.T.R..
Software Product-Line Engineering: A Family-Based Software Development Process
Reading, MA: Addison-Wesley, 1999.

The Value of Options

A software product line approach provides options to future market opportunities.

- The exact opportunities and their certainty are impossible to predict.
- Organizations need a way to conduct product experiments in low-cost, low-risk ways.
- Software product lines permit those kind of experiments through predefined variation points that can be exercised to meet new needs.

Options depend on the ability of the product line architecture and the production plan to support variation.

What’s Different About Reuse With Software Product Lines?

- Business dimension
- Iteration
- Architecture focus
- Preplanning
- Process and product connection

Second Generation Product Lines

Cummins launched a Core II product line

- not from emergency business needs, as was Core I
- from a mature realization that the organization could do better

Core II includes

- a new core asset base
- an improved architecture
- newly derived products
- a new product line process
- a new production method, strategy, and plan
- a new organizational structure
- a new operational concept
- a powerful, new toolset

Core II is meeting its goals.

It is a much fuller and more mature product line capability.

Software Product Lines Today at Cummins

The overall impact of a software product line approach on Cummins as measured by Core II results includes

- freed up resources (time, money, and people) to invest in new technologies and state-of-the-art tools and simulation capabilities
- an all-time high in product quality
- continuously shrinking time to market
- an ability to handle increased breadth and complexity of products
- an ability to outpace its market rivals

Product lines have now become institutionalized at Cummins.

Remaining Challenges

Variation mechanisms and variation management

Automating all or part of the production process

Lowering adoption cost

Distributed development and evolution

Scaling to systems of systems and ultra-large-scale systems

Maturing Our Model

Influence of contextual factors

Production plan is really a production process with added project details.

- the production process is the amalgamated set of attached processes
- the “glue” is non-trivial and reflects a production method (overall approach to handling variation) that supports a production strategy

All practice areas don’t “play at the same volume” during product line adoption.

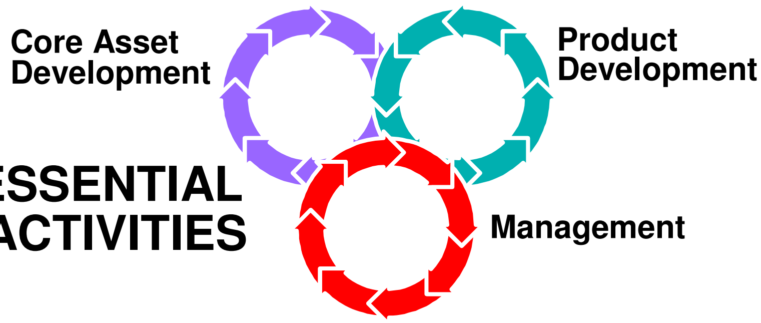
Process discipline is important not just process definition.

Institutionalizing a product line approach usually requires changes in many of the practice areas.

Open source and open source philosophies offer new opportunities for product line efforts.

Framework Version 5.0

ESSENTIAL ACTIVITIES



PRACTICE AREAS		
Software Engineering	Technical Management	Organizational Management
Architecture Definition	Configuration Management	**Building a Business Case**
Architecture Evaluation	Make/Buy/Mine/Commission Analysis	**Customer Interface Management**
Component Development	***Measurement and Tracking***	**Developing an Acquisition Strategy**
Mining Existing Assets	***Process Discipline***	Funding
Requirements Engineering	Scoping	Launching and Institutionalizing
Software System Integration	**Technical Planning**	Market Analysis
Testing	Technical Risk Management	**Operations**
Understanding Relevant Domains	**Tool Support**	Organizational Planning
Using Externally Available Software		Organizational Risk Management
		Structuring the Organization
		Technology Forecasting
		Training

Key:

New Name and Substantial Change

Substantial Change

Today's Talk

INTRODUCTION

SOFTWARE PRODUCT LINES: The Past

- Why we got involved
- What we did
- What happened

SOFTWARE PRODUCT LINES: The Future

- What's happening now
- What's possible

CONCLUSION

What Do We Have Now?

Software product lines have emerged as an important, viable paradigm for software development.

- There is convincing evidence that software product line practice can bring about significant improvements in the development of portfolios of products.
- There is a body of knowledge and a set of standard models for software product lines.
- There is a growing and energetic community of software product line practitioners.

A *Community* of Software Product Line Practitioners

We seek to

- understand the principles and practices behind software product engineering.
- help others successfully adopt the paradigm
- share experience

Major forums, workshops, and conferences

- Software Product Line Conferences (SPLC 1-4)
- Product Family Engineering (PFE 1-5)
- **Merged in 2004 to become SPLC / SPLC-Europe / SPLC-Asia**
- **Next SPLC: September 13-17, 2010 on Jeju Island, South Korea (www.splc.net)**

Community web sites

- www.sei.cmu.edu/productlines
- www.softwareproductlines.com

Challenges - Emerging Solutions

Variation mechanisms and variation management

AOP/AOSD | SOA | End-User Programming

Automating all or part of the production process

MDA | DSL | DDD | Generative Programming

Lowering adoption cost

Agile, Phased Approaches | Tool Support

Distributed development and evolution

Open Source Models | Collaborative Environments | Virtual Worlds

Scaling to systems of systems and ultra-large-scale systems

Product lines reduce interoperability issues

All Are Tightly Linked to the Architecture

Variation mechanisms and variation management

AOP/AOSD | SOA | End-User Programming

Automating all or part of the production process

MDA | DSL | DDD | Generative Programming

Lowering adoption cost

Agile, Phased Approaches | Tool Support

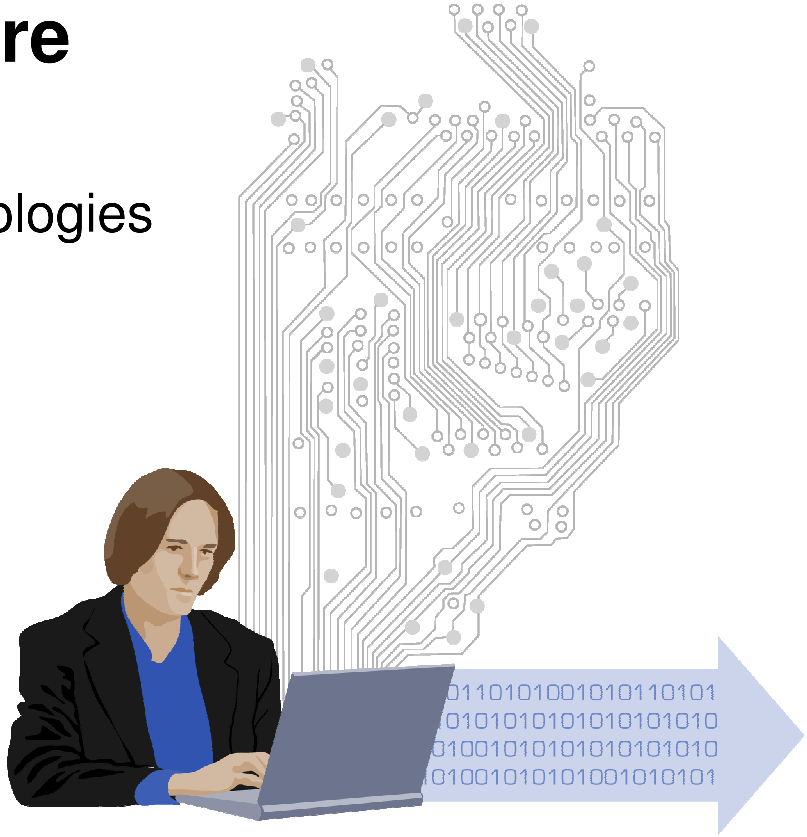
Distributed development and evolution

Open Source Models | Collaborative Environments | Virtual Worlds

Scaling to systems of systems and ultra-large-scale systems

Product lines reduce interoperability issues

Product Lines of the Future



Will harness new and emerging technologies

- metadata
- automated derivation
- SOA
- end-user programming

and new forms of collaboration

- cooperative models
- globalization
- virtual worlds
- collaborative environments

to make product lines more doable, pliable, and dynamic.

Tomorrow’s product lines will accrue even greater benefits than those already demonstrated.

Today's Talk

INTRODUCTION

SOFTWARE PRODUCT LINES: The Past

- Why we got involved
- What we did
- What happened

SOFTWARE PRODUCT LINES: The Future

- What's happening now
- What's possible

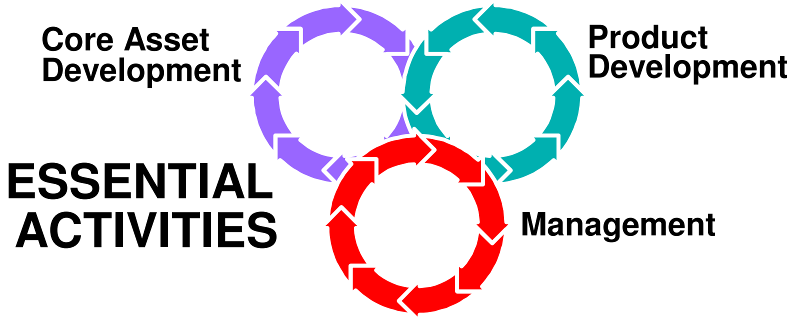
CONCLUSION

In A Nutshell

Software product lines epitomize the concept of strategic, planned reuse.

The product line concept is about more than a new technology. It is a new way of doing one’s software business.

There are essential product line activities and practices areas as well as product line patterns to make the move to product lines more manageable.



Core Asset Development

Product Development

ESSENTIAL ACTIVITIES

Management

PRACTICE AREAS		
Software Engineering	Technical Management	Organizational Management

Final Notes

Research in software product lines was inspired by the proven benefits of product line approaches in manufacturing, and was buoyed by the advent of object and component technology.

The role of architecture is central to a successful product line approach.

The SEI has been a leader in developing a body of knowledge and a set of standard models for software product lines.

Early product line adopters, like Cummins, Inc., are now on second generation product lines that have resulted in even far greater benefits.

Service-oriented and model-driven approaches, as well as developments in collaborative philosophies and environments, are extending the power of product line practice in exciting new ways.

Future product lines will make much greater use of dynamic variation and enable mass customization in ways not achievable today.

Questions – Now Or Later

Linda Northrop
Director, Research, Technology, and System Solutions Program
Telephone: 412-268-7638
Email: lmn@sei.cmu.edu

U.S. Mail:
Software Engineering Institute
Carnegie Mellon University
4500 Fifth Avenue
Pittsburgh, PA 15213-3890

World Wide Web:
http://www.sei.cmu.edu/productlines
SEI Fax: 412-268-5758