



Especialização em Engenharia de Software

Linhas de Produto de Software Parte 1

Sérgio Soares
DSC - UPE
sergio@dsc.upe.br



©Sérgio Soares 2008

Objetivos

- Introduzir os principais conceitos de Linhas de Produto de Software
- Apresentar exemplos de Linhas de Produto de Software
- Discutir um modelo de variabilidade
- Exercitar a definição de variabilidades com uso de uma ferramenta

Linhas de Produto de Software - Parte 1

2

Conteúdo Programático

- Introdução à Linhas de Produto de Software (LPS)
 - motivação e princípios básicos
 - conceitos de LPS
 - o que não são LPS
 - exemplos de LPS
- Desenvolvimento de Linhas de Produto de Software
 - engenharia de Domínio e de Aplicação
 - diferenças entre desenvolvimento de linhas de produto e de produto único
- Variabilidade
 - conceitos, tipos de variabilidade
 - modelo ortogonal de variabilidade

Linhas de Produto de Software - Parte 1

3

Parte 1

- Introdução à Linhas de Produtos de Software (LPS)
 - motivação
 - conceitos
- O que não são LPS
- Exemplos de LPS

- Exercícios

Linhas de Produto de Software - Parte 1

4

Parte 2

- Desenvolvimento de Linhas de Produto de Software
- Engenharia de Domínio e de Aplicação
- Diferenças entre desenvolvimento de linhas de produto e de produto único

- Exercícios

Linhas de Produto de Software - Parte 1

5

Parte 3

- Variabilidade
 - conceitos de variabilidade
 - tipos de variabilidade
- Modelo ortogonal de variabilidade

- Exercícios

Linhas de Produto de Software - Parte 1

6

Metodologia

- Aulas teóricas para apresentação de conceitos
- Aulas práticas para execução de exercícios e sedimentação de conceitos de linhas de produtos de software
 - uso da ferramenta VARMOD-EDITOR
 - plug-in do Eclipse

Linhas de Produto de Software - Parte 1

7

Software

- JDK 1.5
 - <http://java.sun.com/j2se> (1.5.0_01-b08)
- Eclipse
 - <http://www.eclipse.org> (3.2)
 - AspectJ 1.5
 - <http://www.eclipse.org/aspectj> (1.5.3)
 - AJDT
 - <http://www.eclipse.org/ajdt> (1.4.0)
 - VARMOD-EDITOR
 - <http://www.software-productline.com/SEGOS-VM-Tool>
- <http://sergio.dsc.upe.br/lps>

Linhas de Produto de Software - Parte 1

8

Avaliação de alunos

- Avaliação com base em frequência, participação e entrega de trabalhos
 - exercícios e projeto final
 - trabalho em grupo
 - no máximo 4 integrantes por grupo

Linhas de Produto de Software - Parte 1

9

Normas

- Serão considerados aprovados os alunos que, em cada disciplina, obtiverem 75% de frequência e média de proficiência igual ou superior a 7,0 (sete) nos trabalhos.
- Quando obtiver nota entre 5,0 (cinco) e 6,9 (seis vírgula nove), terá direito a nova avaliação, determinada pelo professor, na qual o aluno deverá obter média final igual ou superior a 7,0 (sete).

Linhas de Produto de Software - Parte 1

10

Referências Bibliográficas

- ALVES, V. **Implementing Software Product Line Adoption Strategies**. Tese de Doutorado, CIN-UFPE, 2007.
- CLEMENTS, P.; NORTHROP, L. **Software Product Lines: Practices and Patterns**. Addison-Wesley, 3a ed, 2001.
- KRUEGER, C. **Variation management for software production lines**. In Proceedings of the 2nd International Software Product Line Conference (SPLC'02), pages 37-48, San Diego, California, August 2002. Lecture Notes in Computer Science (LNCS).
- POHL, K.; BÖCKLE, G.; VAN DER LINDEN, F. **Software Product Line Engineering**. Springer, 2005.

Linhas de Produto de Software - Parte 1

11

Referências Diversas

- SEI
 - <http://www.sei.cmu.edu/productlines/>
- SPG
 - <http://www.cin.ufpe.br/spg>
- Este curso e outros materiais em
 - <http://sergio.dsc.upe.br/lps>

Linhas de Produto de Software - Parte 1

12

Introdução à Linhas de Produto de Software

Motivação

- Produtos desenvolvidos **manualmente** para clientes individuais
 - atendimento as necessidades do cliente
- Produtos desenvolvidos em **massa**
 - + menor custo
 - menor diversificação
 - Linha de produção de carros (Ford)
 - maior velocidade na produção de um carro

Linhas de Produto de Software - Parte 1

14

Motivação (software)

- Produtos desenvolvidos **manualmente** para clientes individuais
 - Software **contratado**
 - alto custo
- Produtos desenvolvidos em **massa**
 - Software de **prateleira**
 - pouca diversificação

Como customizar um produto/software a necessidades específicas de um cliente?

Linhas de Produto de Software - Parte 1

15

Customização em massa

- Fabricar carros em massa
 - diferentes necessidades
 - tamanho
 - cor
 - terreno
 - opcionais ...
- Produção em larga escala de bens moldados à necessidade do cliente

Linhas de Produto de Software - Parte 1

16

Pré-requisito para customização em massa

- Definição de uma **plataforma**
 - partes que podem ser usadas em diferentes carros
 - chassis, suspensão, motor, transmissão
- Criar um novo carro do zero ou a partir de uma plataforma já existente?
 - aumento de 35% nas vendas de carros baseados em plataformas
 - aumento de 7% nas vendas de carros desenvolvidos do zero

Linhas de Produto de Software - Parte 1

17

Outros exemplos de uso de plataformas

- Câmeras fotográficas (Kodak vs. Fuji)
 - De 1989 a 1990 a Kodak criou diferentes modelos de máquinas que utilizavam componentes e forma de produção em comum
- Post-It (3M)
 - Vários produtos criados a partir do original

Plataforma é qualquer base de tecnologias sobre a qual outras tecnologias ou processos são construídos

Linhas de Produto de Software - Parte 1

18

Customização em massa e desenvolvimento baseado em plataforma

- Uso de plataformas
 - mesmo motor em diferentes carros
 - mesma suspensão em diferentes carros
 - mesma transmissão em diferentes carros
 - Com customização em massa
 - diferentes tipos de motor
 - diferentes tipos de suspensão
 - diferentes tipos de transmissão
- todos aplicáveis a diferentes carros

Linhas de Produto de Software - Parte 1

19

Definindo uma plataforma

- Comportar os diferentes produtos desejados
 - partes comuns
 - partes variáveis (diferentes)
 - flexibilidade é a chave
 - acompanha restrições
 - motores mais potentes necessitam de transmissões mais fortes
 - chassis mais pesado necessita de suspensão mais resistente
 - definição das possíveis combinações

Linhas de Produto de Software - Parte 1

20

Primeiros conceitos chaves

- Variabilidade
 - flexibilidade da plataforma
- Família (Linha) de produto
 - diferentes produtos que compartilham uma plataforma

Um dos **objetivo** de uma Linha de Produto é provê produtos **customizados** a um preço **razoável**

Linhas de Produto de Software - Parte 1

21

Qual o menor custo?

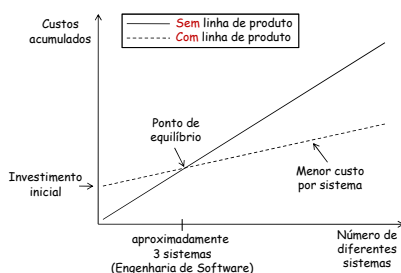
- Desenvolver uma plataforma e seus componentes reusáveis
 - há um custo fixo inicial independente do número dos **diferentes** produtos gerados
 - poucos produtos = alto custo por produto
 - quanto maior o número de **diferentes** produtos mais diluído o investimento inicial
- Desenvolver produtos únicos
 - pode valer a penas se a quantidade de produtos **diferentes** for pequena

Linhas de Produto de Software - Parte 1

22

Custo de desenvolvimento

Sistemas únicos vs Famílias de Produtos



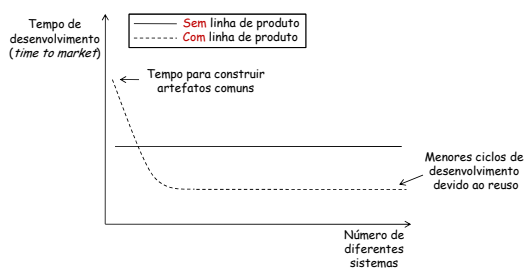
Traduzido e adaptado de [Pohl et al 2005] e [Clemens e Northrop 2001]

Linhas de Produto de Software - Parte 1

23

Time to market

Sistemas únicos vs Famílias de Produtos



Traduzido e adaptado de [Pohl et al 2005]

Linhas de Produto de Software - Parte 1

24

Motivações adicionais

- Aumento da qualidade
 - revisão e testes dos artefatos para vários produtos
 - maiores chances de detecção e correção de erros
- Redução do custo de manutenção
 - mudanças nos artefatos são propagadas para vários produtos
 - baixo custo de aprendizado entre diferentes produtos

Linhas de Produto de Software - Parte 1

25

Motivações adicionais

- Evolução organizada
 - novos artefatos geram oportunidade de evolução para vários produtos
- Menor complexidade
 - reuso de código diminui a quantidade de código a ser mantido
 - separação de funcionalidades em artefatos (*separation of concerns*)
 - artefatos = módulos de funcionalidade

Linhas de Produto de Software - Parte 1

26

Motivações adicionais

- Melhor estimativa de custo/ menor risco
 - extensões na plataforma para adaptar um novo produto (*e não criar do zero*)
- Benefícios para os clientes
 - mudanças entre produtos mais fácil
 - interfaces similares
 - upgrade de funcionalidades
 - melhor qualidade por menores preços
 - diluição de custos entre vários produtos!
 - produtos customizados a sua necessidade

Linhas de Produto de Software - Parte 1

27

O que é uma linha de produto de software?

Conjunto de sistemas compartilhando **um conjunto comum e gerenciado de funcionalidades (features)** que satisfazem necessidades específicas de um segmento, e desenvolvidos a partir de um conjunto comum de **artefatos base** e de uma forma **determinada**.
[Clemens e Northrop 2001]

Conjunto de aplicações desenvolvidas utilizando **plataformas e customização** em massa.
[Pohl et al 2005]

Linhas de Produto de Software - Parte 1

28

Tecnologias que habilitam LPS

- Orientação a Objetos
 - encapsulamento, ocultação de informação
- Desenvolvimento baseado em componentes
 - baixo acoplamento, desenvolvimento, compilação, ligação e carregamento em separado
- Técnicas de ligação
 - ligação tardia (em tempo de instalação ou de execução)

Linhas de Produto de Software - Parte 1

29

E a orientação a aspectos?

- Tudo o que a orientação a objetos tem mais:
 - módulos com menor acoplamento
 - aspectos são estruturas plug-and-play
 - ligação automática em tempo de
 - compilação
 - carregamento
 - execução

DSOA parece promissor para o desenvolvimento de LPS

Linhas de Produto de Software - Parte 1

30

Além de tecnologia...

- ... é fundamental o uso de
 - processos de desenvolvimento bem estruturados
 - com disciplinas de desenvolvimento adequadas
 - engenharia de requisitos
 - análise e projeto
 - implementação
 - testes
 - técnicas de abstração (modelagem)
 - lidam com a complexidade

Linhas de Produto de Software - Parte 1

31

Domínio da aplicação

- *Experts* no domínio
 - fundamental para identificar as *comunalidades* e as variabilidades
 - escolhas erradas são custosas
 - variabilidades além ou aquém do necessário
- Estabilidade do domínio
 - quão dinâmico é o domínio?
 - quão bem entendido está o domínio?
 - fator de decisão para investir ou não na criação de uma LPS

Linhas de Produto de Software - Parte 1

32

O que Linhas de Produto de Software não são (1)

- Reuso de baixa granularidade
 - pequenos pedaços de programa em uma *biblioteca*
 - algoritmos, classes, componentes, módulos
 - *grande* dificuldade em identificar algo necessário e integrar a aplicação
 - o reuso é uma atividade *oportuna*
- Em linhas de produto
 - reuso *planejado, viável, lucrativo* e em *grande escala* (e não apenas software)

Linhas de Produto de Software - Parte 1

33

O que Linhas de Produto de Software não são (2)

- Desenvolvimento de sistema único com reuso
 - sistema muito parecido com um anterior
 - reusa o que pode e modifica/adapta
 - "*clone and own*"
 - dois sistemas diferente a serem mantidos
- Em linhas de produto
 - aplicações (produtos) feitos sobre uma *única base* a ser mantida

Linhas de Produto de Software - Parte 1

34

O que Linhas de Produto de Software não são (3)

- Apenas desenvolvimento baseado em componentes
 - aplicação formada por componentes de uma biblioteca
- Em linhas de produto
 - componentes *específicos* para a linha
 - *composição* planejada e facilitada
 - mecanismos de variabilidade
 - arquitetura

Linhas de Produto de Software - Parte 1

35

O que Linhas de Produto de Software não são (4)

- Apenas uma arquitetura reconfigurável
 - aplicações podem/devem reusar estruturas arquiteturais
 - diminuição de custos
- Em linhas de produto
 - arquitetura
 - definida para suportar variabilidade
 - é um artefato (muito importante) da linha

Linhas de Produto de Software - Parte 1

36

O que Linhas de Produto de Software não são (5)

- **Releases e versões de produtos únicos**
 - usam a mesma arquitetura, documentação, componentes, planos de testes e outros
 - *releases* ou versões anteriores não são mantidas
- **Em linhas de produto**
 - vários produtos simultâneos
 - com releases e versões próprias
 - evolução de um produto considera toda a linha
 - versões anteriores de um produto podem ser instâncias válidas da linha

Linhas de Produto de Software - Parte 1

37

Então o que são linhas de produtos?

Vejamos alguns exemplos...

Linhas de Produto de Software - Parte 1

38

Boeing

Exemplos reais de LPS

- **OPF - Operational Flight Program**
 - distribuído, tempo real, embarcado
 - controle de instrumentos e funcionalidades do cockpit de aviões
- **Desafio na definição da plataforma**
 - diferentes subsistemas, instrumentos, requisitos, hardware de computação

Linhas de Produto de Software - Parte 1

39

Boeing

Exemplos reais de LPS

- **Plataforma**
 - hardware, software, padronização e práticas
- **Experiência**
 - redução de dependência entre componentes
 - independência do hardware usando camadas
 - uso com sucesso em diferentes aeronaves e configurações de hardware

Linhas de Produto de Software - Parte 1

40

HP (Hewlett-Packard)

Exemplos reais de LPS

- **Firmware**
 - impressoras, scanners, copiadoras, fax
- **Plataforma**
 - times de produtos formam uma comunidade em forma de cooperativa
 - o que desenvolvem são de sua propriedade/responsabilidade
 - time reduzido de plataforma assegura a robustez e guia os times de produto no uso dos componentes comuns

Linhas de Produto de Software - Parte 1

41

HP (Hewlett-Packard)

Exemplos reais de LPS

- **Novos produtos**
 - 70% - reuso
 - 20% - mudanças em componentes existentes
 - 10% - novos componentes
 - 1/4 da equipe necessária
 - 1/3 do tempo necessário
 - 96% menos defeitos

em comparação com produtos desenvolvidos antes do uso de LPS

Linhas de Produto de Software - Parte 1

42

Outras grandes empresas

- LG
 - controle de elevadores
- Lucent
 - controle de chaveamento telefônico
- Philips
 - componentes presentes em vários tipos de aparelhos eletrônicos
- Bosch
 - assistência de direção
- Siemens
 - controle de radiologia

Linhas de Produto de Software - Parte 1

43

Exercício 1

- O projeto a ser desenvolvido na disciplina de Aspectos pode ser uma linha de produtos? Quais seriam os produtos gerados por essa linha?

- Envie a resposta por e-mail
 - sergio@dsc.upe.br
 - colocar no assunto: "[LPS] Exercício 1"
 - acrescente o nome dos integrantes do grupo

Linhas de Produto de Software - Parte 1

44



Especialização em Engenharia de Software

Linhas de Produto de Software Parte 1

Sérgio Soares
DSC - UPE
sergio@dsc.upe.br



©Sérgio Soares 2008

 Especialização em Engenharia de Software

Linhas de Produto de Software
Parte 2

Sérgio Soares
DSC - UPE
sergio@dsc.upe.br

©Sérgio Soares 2008

Um framework para o desenvolvimento de Linhas de Produto de Software

Linhas de Produto de Software - Parte 2

2

Reverendo conceitos

- Plataforma
 - conjunto de artefatos reusáveis
 - modelos de requisito e arquitetura, componentes, testes, etc.
- Customização em massa
 - satisfazer requisitos de diferentes *stakeholders*
 - variabilidade

Linhas de Produto de Software - Parte 2

3

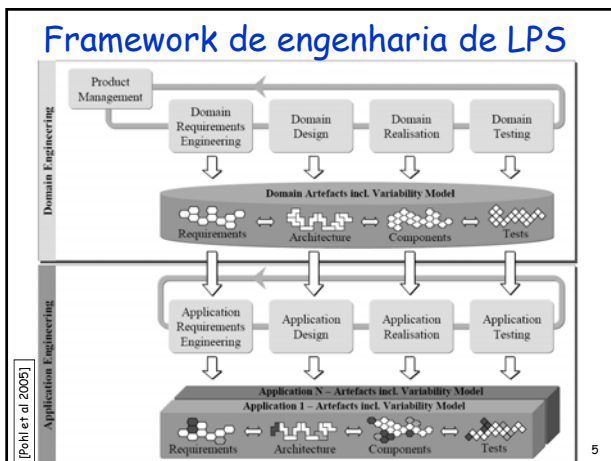
Processos para o desenvolvimento de LPS

- Engenharia de Domínio
 - estabelece a plataforma, define e implementa as comunicações e variabilidades
- Engenharia de Aplicação
 - constrói as aplicações da linha a partir da plataforma definida (reuso de artefatos)
 - compõe comunicações e variabilidades de acordo com os requisitos da aplicação

Lembram da HP? Times para a plataforma e os produtos

Linhas de Produto de Software - Parte 2

4



Sobre o framework

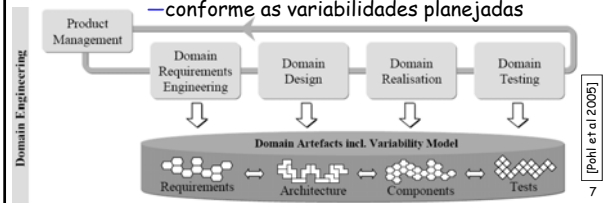
- É um framework
 - pode ser combinado com outros métodos/processos (RUP, espiral, ...)
- Os sub-processos não têm de ser executados sequencialmente
 - desenvolvimento de software é iterativo

Linhas de Produto de Software - Parte 2

6

Engenharia de Domínio

- Objetivos principais
 - definir comunalidades e variabilidades
 - definir o escopo da linha de produtos
 - conjunto de aplicações suportadas
 - definir e construir artefatos reusáveis
 - conforme as variabilidades planejadas



7

Gerenciamento de produto

Product Management

- Aspectos econômicos, estratégia de mercado
 - definição do escopo da linha de produto
- Entrada: objetivos da empresa/cliente
- Saída: planejamento de produtos
 - *features* comuns e variáveis dos produtos e cronograma de entrega dos mesmos
 - lista de produtos ou artefatos existentes que possam ser reusados para definir a plataforma

Linhas de Produto de Software - Parte 2

8

Gerenciamento de produto

Product Management

- Diferenças do desenvolvimento de um **único** produto
 - plataforma com um significado estratégico
 - foco no desenvolvimento de **vários** produtos a um custo razoável
 - produtos fortemente relacionados
 - plataforma comum
 - considerar **antecipação** de mudanças em
 - *features*, restrições legais e padronizações
 - evoluções (mercado, tecnologia, restrições) levadas em conta

Linhas de Produto de Software - Parte 2

9

Engenharia de Requisitos de Domínio

Domain Requirements Engineering

- Elicitação e documentação dos requisitos
 - comuns e variáveis
- Entrada: planejamento de produtos
- Saída: modelo de variabilidade
 - requisitos textuais e em modelos
 - reusáveis (para o conjunto de produtos da linha)
 - **não** uma especificação de requisitos para uma aplicação específica

Linhas de Produto de Software - Parte 2

10

Engenharia de Requisitos de Domínio

Domain Requirements Engineering

- Diferenças do desenvolvimento de um **único** produto
 - **identificação** de requisitos comuns para todas aplicações e dos específicos para uma aplicação em particular
 - possíveis escolhas documentadas no modelo de **variabilidade**
 - **antecipação** das possíveis mudanças
 - recebidos do gerenciamento

Linhas de Produto de Software - Parte 2

11

Projeto de Domínio

Domain Design

- Atividades para definição da arquitetura de referência
 - estruturas de alto nível comuns para os produtos da linha
- Entrada: requisitos e variabilidades
- Saída: arquitetura e modelo de variabilidade refinado
 - variabilidades internas (questões técnicas)

Linhas de Produto de Software - Parte 2

12

Projeto de Domínio

Domain Design

- Diferenças do desenvolvimento de um **único** produto
 - mecanismos de configuração
 - flexibilidade (produtos futuros)
 - regras de desenvolvimento comuns entre várias aplicações
 - definição das partes reusáveis e das partes específicas das aplicações

Afetam ou são afetados pela arquitetura de referência

Linhas de Produto de Software - Parte 2

13

Realização de Domínio

Domain Realisation

- Projeto detalhado e implementação de componentes reusáveis
- Entrada:
 - arquitetura de referência e lista de artefatos de software reusáveis
- Saída:
 - projeto detalhado e componentes reusáveis implementados

Linhas de Produto de Software - Parte 2

14

Realização de Domínio

Domain Realisation

- Diferenças do desenvolvimento de um **único** produto
 - componentes configuráveis e fracamente acoplados, não um programa executável
 - componentes planejados, projetados e implementados para reuso em diferentes contextos (produtos da linha)
 - mecanismos de configuração para compor as variabilidades

Linhas de Produto de Software - Parte 2

15

Testes de Domínio

Domain Testing

- Validação e verificação de componentes reusáveis, produção de componentes de teste reusáveis
- Entrada:
 - requisitos do domínio, arquitetura de referência e componentes reusáveis
- Saída:
 - resultados dos testes e componentes de testes reusáveis

Linhas de Produto de Software - Parte 2

16

Testes de Domínio

Domain Testing

- Diferenças do desenvolvimento de um **único** produto
 - não há aplicação executável a ser testada
 - testes de partes montadas a partir de variabilidades
 - várias possíveis combinações
 - definição de artefatos de testes a serem aplicados nos testes de aplicação

Linhas de Produto de Software - Parte 2

17

Assegurando qualidade

- Além de testes
 - inspeções
 - revisões
 - *walkthroughs*
- Carência de propostas especializadas para linhas de produto
 - têm de ser adaptados de desenvolvimento único

Linhas de Produto de Software - Parte 2

18

Artefatos de Domínio

- Planejamento de produto (*product roadmap*)
- Modelo de variabilidade do domínio
- Requisitos do domínio
- Arquitetura do domínio
- Artefatos de realização do domínio
- Artefatos de teste do domínio

Linhas de Produto de Software - Parte 2

19

Planejamento de produto

Product roadmap

- Todas as funcionalidades (features) de todas as aplicações
 - classificadas em funcionalidades comuns e variáveis
- Cronograma para introdução no mercado
 - o cronograma pode ser extenso
 - planejamento estratégico
- Evolução durante o ciclo de vida da linha (evoluções do mercado)

Linhas de Produto de Software - Parte 2

20

Planejamento de produto

Product roadmap

- Template (simples)
 1. Introdução sobre a linha de produtos
 - introdução geral
 2. Funcionalidades comuns da linha
 - listagem com explicação das funcionalidades
 3. Funcionalidades variáveis da linha
 - listagem com explicação das funcionalidades
 4. Cronograma
 - em meses ou anos (alguns produtos podem ser lançados muito tempo depois do primeiro)

Linhas de Produto de Software - Parte 2

21

Exercício 2

- Imagine uma linha de produtos
- Crie uma primeira versão do product roadmap
 - esta primeira versão deve ter pelo menos 5 (cinco) produtos com funcionalidades comuns e variáveis que justifiquem uma linha de produtos
 - não iremos implementar a linha, portanto, não tenha receio de usar a criatividade
- Envie a resposta por e-mail
 - sergio@dsc.upe.br
 - colocar no assunto: "[LPS] Exercício 2"
 - acrescente o nome dos integrantes do grupo

Linhas de Produto de Software - Parte 2

22

Engenharia da Aplicação

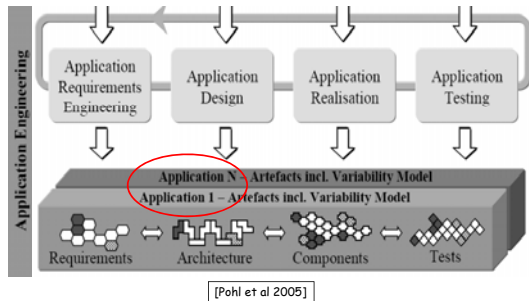
Engenharia da Aplicação

- Objetivos principais ao definir e desenvolver uma aplicação (produto)
 - atingir o máximo de **reuso** dos artefatos de domínio
 - explorar a **comunalidade** e **variabilidade** da linha
 - documentar os artefatos de aplicação e relacioná-los com os de domínio
 - **compor** as variabilidades de acordo com as necessidades das aplicações
 - estimar o impacto das diferenças entre requisitos de domínio e aplicação sobre arquitetura, componentes e testes

Linhas de Produto de Software - Parte 3

24

Engenharia da Aplicação



Linhas de Produto de Software - Parte 3

25

Engenharia de Requisitos da Aplicação

Application Requirements Engineering

- Especificação de requisitos de uma aplicação (produto) da linha
 - reuso de artefatos de domínio
- Entrada:
 - planejamento de produtos
 - requisitos de domínio
- Saída:
 - especificação de requisitos para uma aplicação específica

Cliente pode requerer novos requisitos específicos de uma aplicação

Linhas de Produto de Software - Parte 3

26

Engenharia de Requisitos da Aplicação

Application Requirements Engineering

- Diferenças do desenvolvimento de um **único** produto
 - maioria dos requisitos já identificada a partir dos requisitos do domínio
 - comunalidades e variabilidades da linha
 - praticamente não há elicitação
 - **eventuais** deltas entre requisitos de domínio e aplicação
 - documentar** e avaliar o esforço de adaptação

Linhas de Produto de Software - Parte 3

27

Projeto da Aplicação

Application Design

- Criação da arquitetura da aplicação
 - instanciação da arquitetura de referência (configuração e adaptação)
- Entrada:
 - especificação de requisitos e arquitetura de referência
- Saída:
 - arquitetura da aplicação

Linhas de Produto de Software - Parte 3

28

Projeto da Aplicação

Application Design

- Diferenças do desenvolvimento de um **único** produto
 - não define a arquitetura do zero
 - escolhas específicas nos pontos de variação
 - eventuais adaptações
 - o projeto se restringe a regras/decisões definidas na arquitetura de referência
 - avaliar eventuais adaptações e rejeitá-las dependendo do esforço necessário

Linhas de Produto de Software - Parte 3

29

Realização da Aplicação

Application Realisation

- Seleção e configuração de componentes reusáveis
- Implementação de software específico da aplicação
- Entrada:
 - arquitetura e componentes reusáveis
- Saída:
 - aplicação executável e projeto detalhado

Linhas de Produto de Software - Parte 3

30

Realização da Aplicação

Application Realisation

- Diferenças do desenvolvimento de um **único** produto
 - vários componentes, interfaces e outros recursos de software não são criados, mas configurados a partir da plataforma
 - implementações devem seguir as interfaces reusáveis
 - componentes da aplicação implementados como variações de componentes da plataforma

Linhas de Produto de Software - Parte 3

31

Testes da Aplicação

Application Testing

- Validação e verificação da aplicação
- Entrada:
 - documentos e modelos utilizados como referência para verificar os testes
 - a aplicação e artefatos de teste reusáveis
- Saída:
 - relatório de testes com o resultado dos testes
 - documentação de defeitos

Linhas de Produto de Software - Parte 3

32

Testes da Aplicação

Application Testing

- Diferenças do desenvolvimento de um **único** produto
 - vários artefatos de teste não são criados, mas derivados da plataforma
 - testes para checar defeitos de configuração
 - variações corretamente selecionadas/compostas
 - cobertura de testes
 - considerar partes reusadas (comuns e variáveis) e partes especificamente implementadas para a aplicação

Linhas de Produto de Software - Parte 3

33

Artefatos da Aplicação

- Modelo de variabilidade da aplicação
- Requisitos da aplicação
- Arquitetura da aplicação
- Artefatos de realização da aplicação
- Artefatos de teste da aplicação

Linhas de Produto de Software - Parte 3

34



Especialização em Engenharia de Software

Linhas de Produto de Software Parte 2

Sérgio Soares
DSC - UPE
sergio@dsc.upe.br



©Sérgio Soares 2008

 Especialização em Engenharia de Software

Linhas de Produto de Software
Parte 3

Sérgio Soares
DSC - UPE
sergio@dsc.upe.br

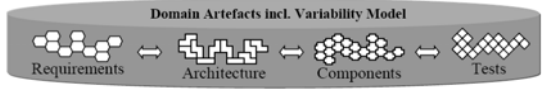
  **d.s.c. UPE**
Departamento de Sistemas e Computação

©Sérgio Soares 2008

Variabilidade

- Suporte ao desenvolvimento e reuso de artefatos variáveis
 - requisitos
 - arquitetura
 - componentes
 - testes

Definição durante a Engenharia de Domínio



[Pohl et al 2005]

Linhas de Produto de Software - Parte 3 4

Variabilidade em linhas de produto de software

Variabilidade

- Na Engenharia da Aplicação
 - sub-processos compõe/ligam variabilidades introduzidas pelo sub-processo equivalente da Engenharia de Domínio
 - ligação consistente
 - assegurar a construção correta do(s) variante(s) requerido(s)

Linhas de Produto de Software - Parte 3 5

Variabilidade

- Habilidade ou tendência a mudar
- Em LPS tais mudanças são propositais (conscientes) e não eventuais
 - aplicação com suporte a diferentes
 - línguas
 - aparelhos celulares
 - protocolos de distribuição
 - meios de armazenamento de dados

Linhas de Produto de Software - Parte 3 3

Identificando variabilidades

- O que varia?
 - identifica item ou propriedade variável
 - **sujeito** da variabilidade
- Por que varia?
 - necessidades dos stakeholders, leis diferentes, questões técnicas, outros itens, etc.
- Como varia?
 - quais as possíveis formas/variantes da variação
 - **objeto** da variabilidade (instância do sujeito)

Linhas de Produto de Software - Parte 3 6

Sujeito e objeto da variabilidade

- **Sujeito: Cor**
 - Objetos: verde, vermelho, azul ...
- **Sujeito: Forma de pagamento**
 - Objetos: cartão, dinheiro ou cheque
- **Sujeito: Mecanismo de identificação**
 - Objetos: teclado, scanner de digitais
 - Razão para variabilidade: padrão de segurança, necessidade do cliente
 - mudanças no mecanismo implicam em mudanças em outros componentes

Linhas de Produto de Software - Parte 3

7

Identificando pontos de variação e variantes

1. Identificar o sujeito da variabilidade (mundo real)
2. Identificar o ponto de variação no contexto da linha de produto de sw
 - dificilmente muda com o tempo
3. Definir os conjunto de variantes
 - selecionar objetos da variabilidade e defini-los como variantes do ponto de variação
 - pode variar com o tempo

Linhas de Produto de Software - Parte 3

10

Conceitos sobre variabilidade

- **Ponto de variação (*variation point*)**
 - representação do sujeito da variabilidade em artefatos do domínio enriquecidos com informação de contexto
 - meio de armazenamento dos dados da aplicação
- **Variante (*variant*)**
 - representação de um objeto da variabilidade em artefatos do domínio
 - uma opção de um ponto de variação
 - persistência com SGBD relacional

Linhas de Produto de Software - Parte 3

8

Exemplo

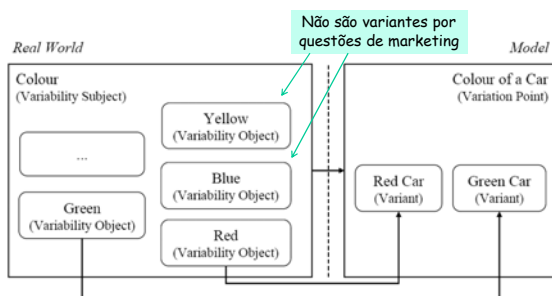
Engenheiros de uma linha de produtos de automação doméstica sugerem diferentes formas de comunicação entre componentes do sistema. Exemplos são: rede com fio, rede sem fio, bluetooth ou linha de força

- **Qual o sujeito da variabilidade?**
 - Tipo de rede de comunicação
- **Qual o ponto de variação?**
 - Rede de comunicação do sistema de automação doméstico
- **Quais os variantes?**
 - Rede com e sem fio (selecionados pelos engenheiros)

Linhas de Produto de Software - Parte 3

11

De sujeitos para pontos De objetos para variantes



Linhas de Produto de Software - Parte 3

[Pohl et al 2005]

9

Variabilidade vs. comunalidade

- **Variabilidade**
 - funcionalidades (*features*) que distinguem as diferentes aplicações de uma LPS
- **Comunalidade**
 - funcionalidades (*features*) que estão presentes em todas as aplicações da LPS
- **Como decidir se uma funcionalidade (*feature*) é variável ou é comum?**

Linhas de Produto de Software - Parte 3

12

Exemplo

Interface com usuário

Variabilidade

Clientes do sistema de automação doméstico devem escolher a língua do sistema que será instalado ou pagar mais por uma versão multilíngüe do sistema.

Comunalidade

A interface do sistema de automação doméstico oferece ao usuário a escolha da língua preferida a qualquer momento. **Todas** as versões do sistema são multilíngüe.

Linhas de Produto de Software - Parte 3

13

Variabilidade no espaço

- Um artefato com diferentes formas que são válidas ao mesmo tempo
 - variantes de um ponto de variação
- Não se aplica a sistemas únicos
- É o foco da engenharia de LPS
- Uso simultâneo de **artefatos variáveis** em diferentes formas (variantes) por produtos diferentes

Linhas de Produto de Software - Parte 3

16

Variabilidade no tempo

- Diferentes versões de um artefato válidas em tempo diferentes
 - evolução
- Se aplica a sistemas únicos e a LPS
 - naturalmente suportado por LPS
 - se as mudanças forem nos pontos de variação
- Gerenciamento de configuração
- **Artefatos** variam com o tempo

Linhas de Produto de Software - Parte 3

14

Exemplo

Variabilidade no espaço

- Ponto de variação (sistema de automação doméstico)
 - "Acesso ao sistema por"
- Variantes
 - web browser, celular (SMS), chamada telefônica (atendimento automatizado), cliente SSH.
 - são associadas a diferentes artefatos da LPS (requisitos, arquitetura, ...)

Linhas de Produto de Software - Parte 3

17

Exemplo

Variabilidade no tempo

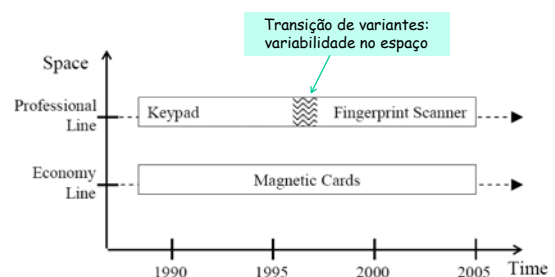
- Mecanismo de identificação (sistema de automação doméstico)
 - ponto de variação com um variante
 - cartão magnético
 - surgimento de scanners confiáveis
 - novo variante: scanner de digitais
 - o variante anterior permanece ou não na linha
 - analistas, engenheiros, clientes, mercado ...

E se o mecanismo de identificação não fosse um ponto de variação desde o início da LPS?

Linhas de Produto de Software - Parte 3

15

Variabilidade no tempo e no espaço



Linhas de Produto de Software - Parte 3

[Pohl et al 2005]

18

Variabilidade Externa

- Variabilidade de artefatos de domínio visível ao cliente
- Clientes escolhem as variantes
 - diretamente
 - selecionando as variantes desejadas
 - indiretamente
 - selecionando uma das aplicações disponíveis na linha

Linhas de Produto de Software - Parte 3

19

Variabilidade Interna

- Causas:
 - variabilidades externas (e refinamentos)
 - oferecer uma opção para o cliente implica em várias opções de baixo nível (internas)
 - refinamento de variabilidades internas
 - variabilidades em um nível de abstração mais baixo
 - questões técnicas não consideradas pelo cliente
 - armazenar imagens das digitais sem compressão durante o desenvolvimento (testes)

Linhas de Produto de Software - Parte 3

22

Variabilidade Externa

- Exemplo:
 - teclado, cartão magnético ou scanner de digitais
- Causas:
 - diferentes necessidades de *stakeholders*
 - diferentes leis que se aplicam ao domínio da LPS
 - domínio médico ou de automação doméstica
 - diferentes padrões a serem atendidos
 - níveis de segurança impostos por uma associação

Linhas de Produto de Software - Parte 3

20

Interna ou Externa?

- Considerar os interesses do cliente
 - Int. • menor complexidade na decisão do cliente
- Estratégia de negócio
 - Int. • proteger segredos da empresa
 - LAN versus EIB (*European Installation Bus*)
- Marketing
 - Ext. • percepção de valor de um produto pelo cliente
 - Int. • variabilidade interfere com aplicações antigas, porém lucrativas, da linha não oferecer a variabilidade

Linhas de Produto de Software - Parte 3

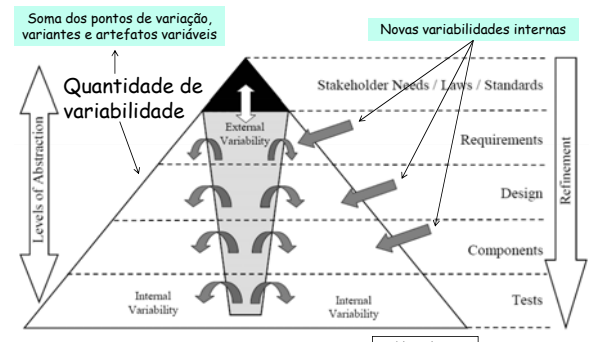
Variabilidade Interna

- Variabilidade de artefatos de domínio escondida do cliente
 - clientes não levam tais variantes em conta ao escolher uma aplicação
 - decisões a cargo dos provedores (desenvolvedores, técnicos, ...) da LPS
- Exemplo:
 - protocolo de comunicação otimizado ou com correção de erros
 - decisão depende do meio de transmissão disponível

Linhas de Produto de Software - Parte 3

21

Pirâmide da variabilidade



Linhas de Produto de Software - Parte 3

24

Pirâmide da variabilidade

- Variabilidade de requisitos leva a uma quantidade maior de variabilidades na arquitetura
 - um requisito normalmente mapeado em vários elementos de projeto
 - uma variabilidade no projeto refinada em variabilidade de componentes (variantes)
 - testes precisam lidar com variabilidades de requisitos, projeto e componentes
- Novas variabilidades internas aumentam a complexidade

Linhas de Produto de Software - Parte 3

25

Modelo ortogonal de variabilidade

- Define a variabilidade de uma LPS
- Relaciona a variabilidade da LPS com outros modelos de desenvolvimento
 - feature
 - caso de uso
 - projeto
 - componente
 - teste

Linhas de Produto de Software - Parte 3

28

Modelo de variabilidade

- Adaptações de modelos e diagramas tradicionais de desenvolvimento
 - modelos de *feature*
 - modelos de caso de uso
 - diagramas de seqüência
 - diagramas de classe

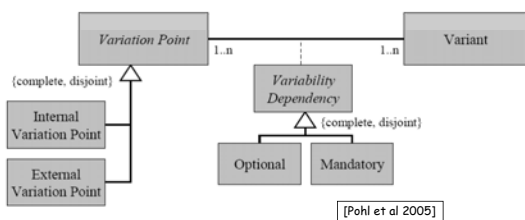
Linhas de Produto de Software - Parte 3

26

Meta modelo

Facilita o entendimento

- Pontos de variação, variantes e dependências de variabilidades



Linhas de Produto de Software - Parte 3

29

Problemas com as adaptações

- Inconsistência
 - informações da variabilidade espalhada entre modelos
- Falta de rastreabilidade
 - qual informação de variabilidade nos requisitos influenciou o que no projeto, na realização,....?
- Modelos tradicionais já são (muito) complexos
- Dificuldade de integração
 - diferentes conceitos (entre os modelos) para definir variabilidade
- Ambigüidade
 - variabilidade nestes modelos levam a ambigüidades em artefatos de desenvolvimento

Linhas de Produto de Software - Parte 3

27

Dependência de variabilidade opcional

- Variante **pode** ser selecionado **se** o ponto de variação fizer parte do produto
- Exemplo:
 - mecanismos de identificação para AD: teclado, cartão magnético e scanner de digitais
 - se definidos como variantes opcionais
 - qualquer combinação dos mesmos caso o ponto de variação seja parte da aplicação
 - nenhum, qualquer um, qualquer dois ou todos os três

Linhas de Produto de Software - Parte 3

30

Dependência de variabilidade mandatória

- Variante **deve** ser selecionado **se** o ponto de variação fizer parte do produto
- Exemplo:
 - tamanhos de chave de criptografia para comunicação remota: 128, 256, 512 e 1024
 - 128 definido como variante obrigatória
 - os demais como opcionais
 - só está presente nas aplicações da linha que têm comunicação remota

Linhas de Produto de Software - Parte 3

31

Restrições de variabilidades

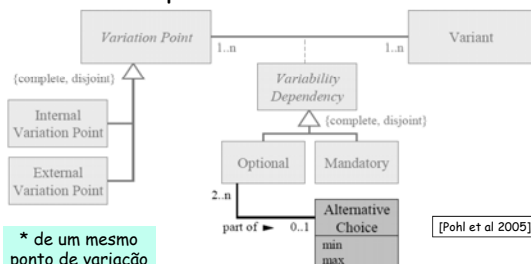
- Definir relações entre variantes de pontos de variação distintos
 - um variante X **requer/exclui** um variante Y para funcionar corretamente
 - um variante X **requer/exclui** um ponto de variação PZ para funcionar corretamente
 - um ponto de variação PZ **requer/exclui** um ponto de variação PW para funcionar corretamente

Linhas de Produto de Software - Parte 3

34

Escolha alternativa

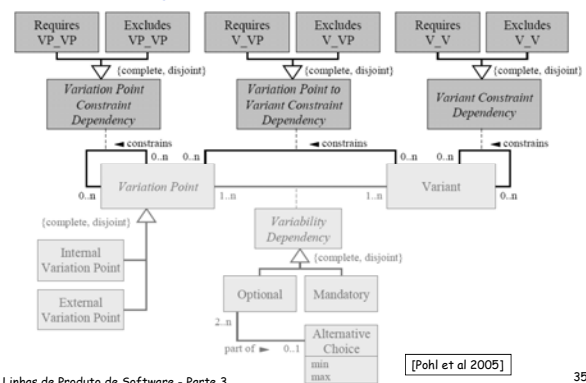
- Define número mínimo e máximo de variantes opcionais* a serem selecionados



Linhas de Produto de Software - Parte 3

32

Restrições



Linhas de Produto de Software - Parte 3

35

Exemplo

- Mecanismos de identificação para AD: teclado, cartão magnético e scanner de digitais
 - definir uma escolha alternativa com os três variantes opcionais com valores
 - mínimo = 1
 - máximo = 2

Linhas de Produto de Software - Parte 3

33

Exemplo

- Pontos de variação (variantes)
 - comunicação sem fio (WLAN, Bluetooth)
 - padrão de comunicação (IEEE 802.11a, b, g)
 - conexão segura (VPN, SSH)
 - detecção de movimento (óptico, radar)
- Restrições
 - WLAN **exclui** radar
 - WLAN **requer** padrão de comunicação
 - comunicação sem fio **requer** comunicação segura

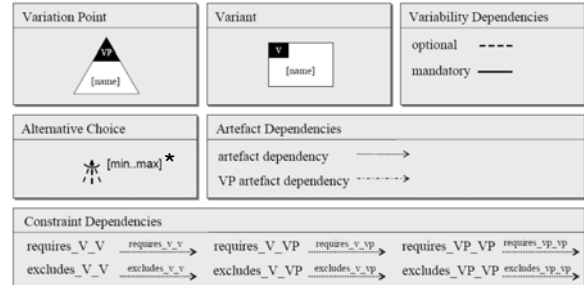
Linhas de Produto de Software - Parte 3

36

Rastreabilidade

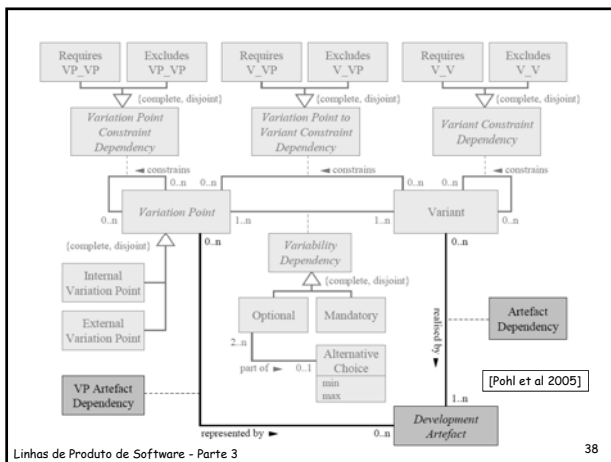
- Relacionar variabilidade e variantes com artefatos do desenvolvimento
 - requisitos
 - projeto
 - realização
 - testes

Notação gráfica para definir o modelo de variabilidade



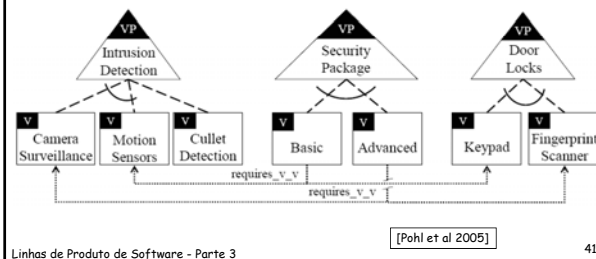
* O default é [1..1]

[Pohl et al 2005]



Exemplo de modelo de variabilidade

- Sistema de automação doméstica



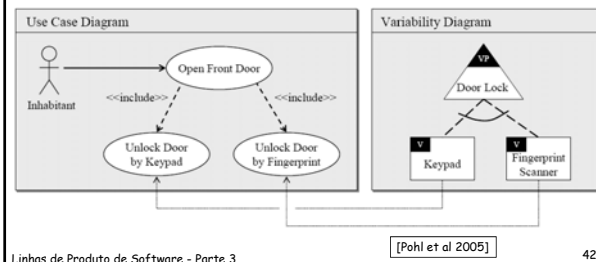
[Pohl et al 2005]

Chega de meta modelo

Como criar modelos com informações sobre pontos de variação e variantes?

Exemplo de modelo ortogonal de variabilidade

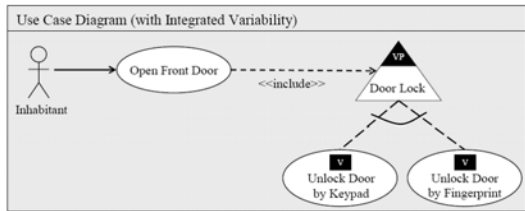
- Sistema de automação doméstica



[Pohl et al 2005]

Outro exemplo

- Modelo de variabilidade integrado com artefato de desenvolvimento



Linhas de Produto de Software - Parte 3

[Pohl et al 2005]

43

Exercício 3

- No diagrama variabilidades o ponto de variação `Operating System` não possui variantes
 - defina variantes para este ponto de variação
 - pelo menos três
 - implemente restrições que podem/devem ser impostas entre essas variantes e o restante do modelo de variabilidades
- Salve o projeto compacte e envie por e-mail
 - sergio@dsc.upe.br
 - colocar no assunto: "[LPS] Exercício 3"
 - acrescente o nome dos integrantes do grupo

Linhas de Produto de Software - Parte 3

46

Suporte de ferramentas é obrigatório!

Linhas de Produto de Software - Parte 3

44

Exercício 4

- Crie o modelo de variabilidades do slide 41 na ferramenta VARMOD-EDITOR
 - crie um novo projeto (vide guia de uso da ferramenta na página 48 deste material)
 - observe em que pasta o projeto será criado
- Salve o projeto compacte e envie por e-mail
 - sergio@dsc.upe.br
 - colocar no assunto: "[LPS] Exercício 4"
 - acrescente o nome dos integrantes do grupo

Linhas de Produto de Software - Parte 3

47

VARMOD-EDITOR

- Execute o Eclipse
- Acesse a página <http://sergio.dsc.upe.br/lps>
- Baixe o "exemplo de projeto do VARMOD-EDITOR" e extraia o mesmo no desktop
- No Eclipse importe o projeto extraído
 - File → Import... → General → Existing projects into workspace → Next → Browse...
 - Selecione a pasta "Exemplo1" no desktop → OK
 - Na lista de projetos aparecerá o projeto "Exemplo 1", clique em Finish
- Abra o arquivo `notebook.vedit_diagram` no projeto

Linhas de Produto de Software - Parte 3

45

Exercício 5

- Crie o modelo de variabilidades da LPS que foi definido no Exercício 2
 - crie um novo projeto (vide guia de uso da ferramenta na página 48 deste material)
 - observe em que pasta o projeto será criado
- Salve o projeto compacte e envie por e-mail
 - sergio@dsc.upe.br
 - colocar no assunto: "[LPS] Exercício 5"
 - acrescente o nome dos integrantes do grupo

Linhas de Produto de Software - Parte 3

48

Projeto

- Crie um documento da linha de produto com:
 - o *product roadmap* revisado
 - o modelo de variabilidade da linha de produto
 - uma lista com os produtos gerados a partir da linha
 - inclua os variantes presentes em cada produto
- Envie por e-mail
 - sergio@dsc.upe.br
 - colocar no assunto: "[LPS] Projeto"
 - acrescente o nome dos integrantes do grupo

Linhas de Produto de Software - Parte 3

49



Especialização em Engenharia de Software

Linhas de Produto de Software Parte 3

Sérgio Soares
DSC - UPE
sergio@dsc.upe.br



©Sérgio Soares 2008