

---

# Princípios de Linhas de Produtos de Software

---

Prof. Alberto Costa Neto

[alberto@ufs.br](mailto:alberto@ufs.br)

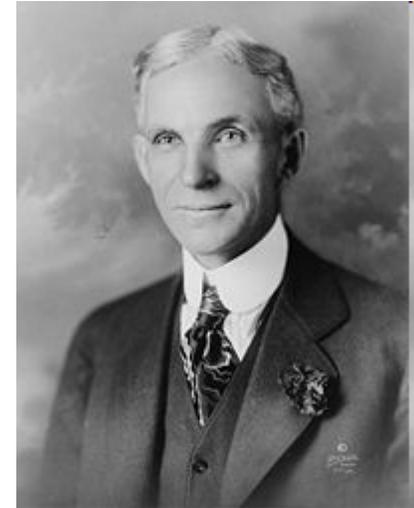
# Surgimento das Linhas de Produtos

- Inicialmente produtos eram feitos artesanalmente
- Mas... N° de pessoas que poderiam comprá-los aumentou!
- Como atender a esta demanda?



# Henry Ford

- No área automotiva, Henry Ford inventou a linha de produtos
  - Produção em massa
  - Preço mais baixo que produtos artesanais
- O primeiro automóvel produzido desta forma foi o Ford T (1908)



---

# Customização x Custo de Produção

- Até 1914, o Ford T era produzido em várias cores (ao gosto do consumidor)
- Mas a partir daí, para cortar custos, passou a ser produzido somente na cor preta, o que durou até 1926.
  - “O carro é disponível em qualquer cor, contanto que seja preto.” [Henry Ford]

---

# Software individual x Software padrão

- **Software individuais** são muito caros
  - Produzidos para satisfazer requisitos específicos do comprador
- **Software padrões** são mais baratos
  - Sofrem de falta de diversificação para atender os vários clientes

---

# Motivação para Customização

- Pessoas têm necessidades diferentes
  - Carro simples para 5 pessoas
  - Carro completo para 7 pessoas
  - Carro com capacidade de carga e para 2 pessoas
  - Carro com tração 4x4
- Algumas podem pagar a mais por certas características do produto
- Outras querem **mostrar que podem pagar a mais por produtos diferenciados**

# Customização de veículos



NOVO

**UNO**  
2012



**MILLE**  
ECONOMY 2012



**PALIO FIRE**  
ECONOMY 2012



**PALIO**  
2011



**SIENA FIRE**  
2012



**SIENA**  
2012



**PALIO WEEKEND**  
2012



**PALIO ADVENTURE**  
2012



**PUNTO**  
2011



NOVO

**IDEA**  
2012



NOVO

**BRAVO**  
2012



**FIAT 500**  
2010



NOVO

**LINEA**  
2011



**DOBLÒ**  
2012



**DOBLÒ ADVENTURE**  
2012



**STRADA FIRE**  
2012



**STRADA WORKING**  
2012



**STRADA TREKKING**  
2012



NOVO

**STRADA SPORTING**  
2012



**STRADA ADVENTURE**  
2012

# Customização de telefones celulares



W880i



M600i



W700i



K320i



Z550a



W710i



Z610i



K790i



Z550i



W910i ☆



Z320i ☆



P1i



W580i



S500i



W200i



K550i



W610i



K310i



W300i



J100i



W810i



J100a



Z530i



Z300i



Z300a



W600i



W800i



K750i



K300i



P900

Source:  
<http://sonyericsson.com.br>

---

# Customização em Massa

- *“Customização em massa é a produção em larga escala de bens voltados às necessidades individuais dos consumidores.”*  
[Davis 1987]
  
- Mas como lidar com o custo elevado da customização?

# O que estes carros têm em comum?



# Plataforma

- “Uma plataforma é qualquer base de tecnologias sobre a qual outras tecnologias ou processos são construídos.”  
[TechTarget 2004]



Fonte: <http://www.1302super.com>

---

# Software Product Line Engineering

Plataforma comum para o des. de software

+

Customização em massa



Engenharia de Linha de Produtos de Software

---

# Desenvolvimento de Produtos Customizados

- Combinar plataforma e customização em massa traz implicações sobre:
  - Processo de desenvolvimento
    - Desenvolver a plataforma e depois as aplicações (ao invés de cada aplicação independentemente)
    - “Commonality first, differences later”
  - Organização da empresa
    - Desenvolvimento e manutenção da plataforma pode requerer um grupo especial
    - Produtos individuais podem ser responsabilidade de outros grupos

---

# Flexibilidade é um fator chave

- Um componente deveria ser **adaptável para vários produtos**
- Componentes devem poder **interagir com componentes diferentes**
- Exemplo: retrovisores de veículos
  - Citroën C3/Picasso e Peugeot 206/307
  - Renault Logan
- No contexto de LPS, flexibilidade é chamada de **Variabilidade**

---

# Compartilhamento de Plataformas

- É possível compartilhar Plataformas entre Linhas de Produtos
  - Quebra a relação direta entre Plataforma e LP
- Necessidade de manter um rastreamento entre uma plataforma e os produtos gerados por ela

# Compartilhamento de Plataformas entre LP

## Plataforma GM4200



Opel Corsa A 1983



Opel Corsa B 1994



Chevrolet Agile 2009

Chevrolet Corsa 1994



Chevrolet Corsa 2002

## Plataforma GM4300



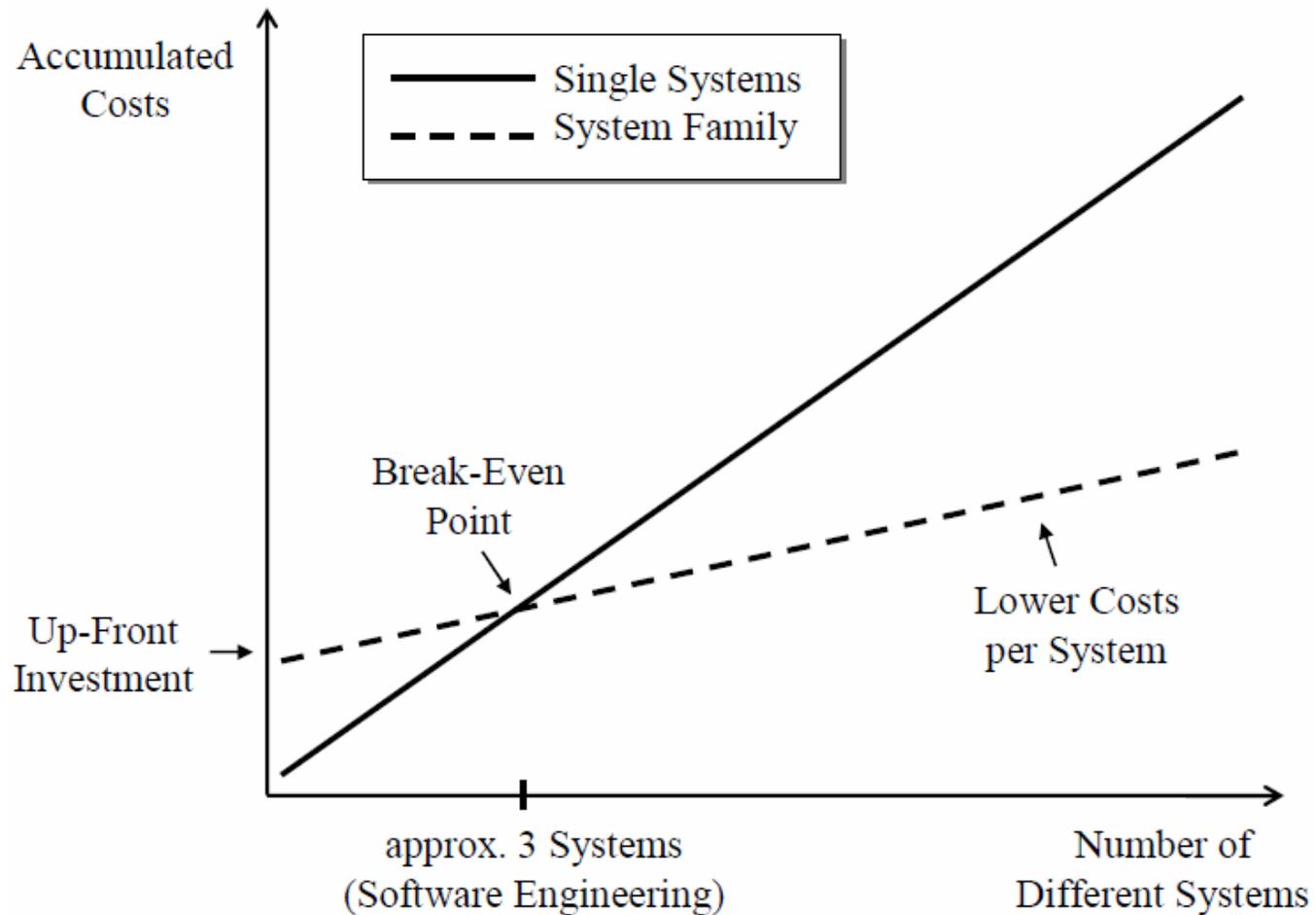
Chevrolet Meriva 2002

---

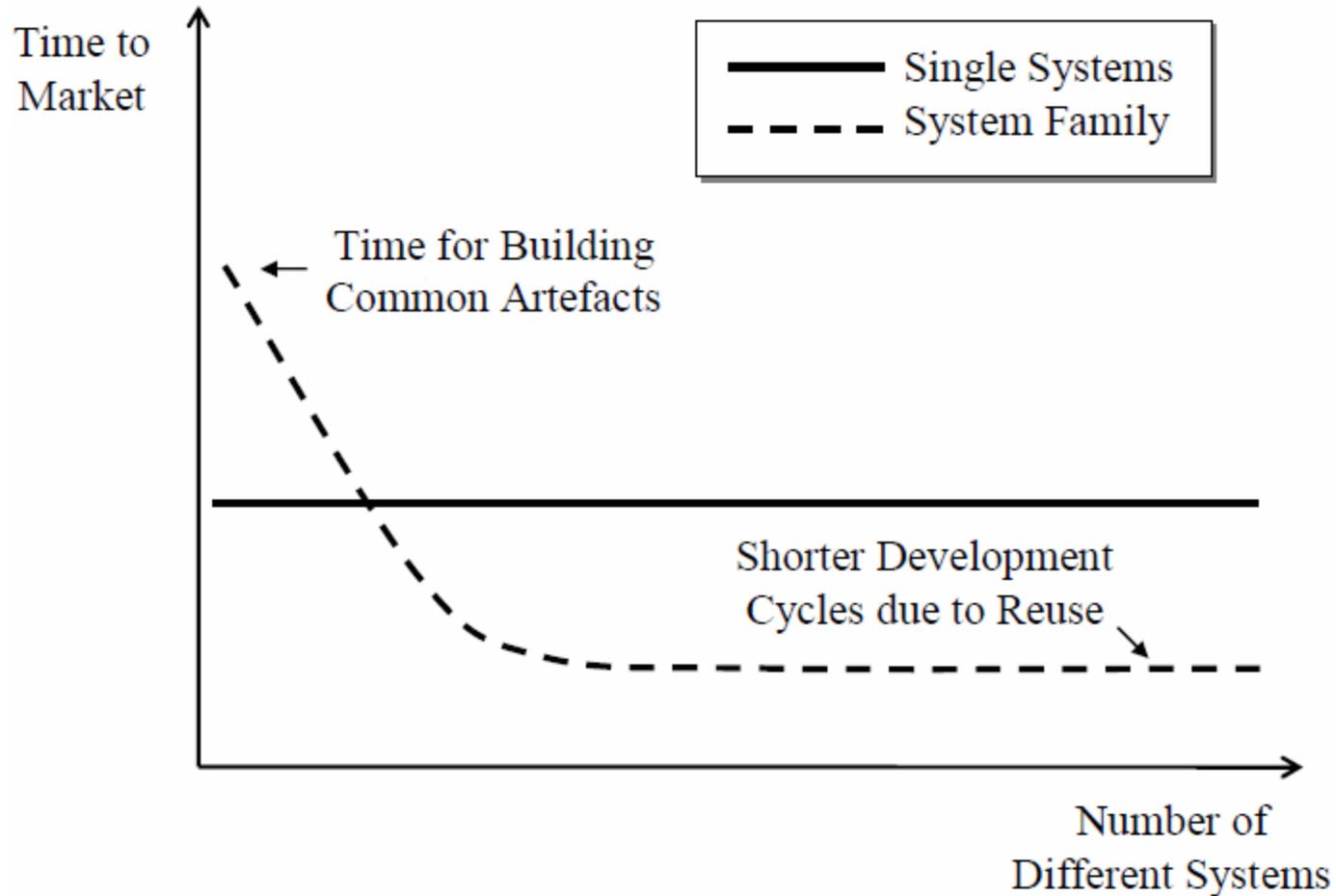
# Motivações para LPS

- Redução do custo de desenvolvimento
- Redução do “time to market”
- Melhoria de qualidade
- Outras motivações:
  - Redução do esforço de manutenção
  - Facilidade para lidar com evolução
  - Capacidade de lidar com complexidade
  - Melhoria nas estimativas de custo
  - Benefícios para os consumidores

# Redução do custo de desenvolvimento



# Redução do “Time to Market”



---

# Demanda pela Engenharia de LPS

- Aumento da quantidade de **dispositivos com software embarcado**
- **Variabilidade** no Software também tem crescido muito
- É preciso tratar estas variabilidades “**antes**” no **processo de desenvolvimento** (não deixar para a implementação)

“Engenharia de LPS é um **paradigma** de desenvolvimento de aplicações usando **plataformas e customização em massa**”

---

# Controle de Variabilidade

- Desenvolver aplicações usando plataformas significa:
  - Planejar proativamente para reuso
  - Construir partes reusáveis
  - Reusar o que foi construído para reuso
- Desenvolver aplicações para customização em massa significa:
  - Empregar o conceito de Controle de Variabilidade
  - *Commonalities* e diferenças nas aplicações da LPS têm que ser modeladas rotineiramente
    - Requisitos, arquitetura, componentes, e artefatos de teste

---

# Restrição sobre adaptações

- Há várias formas de adaptar um software
  - Fácil de mudar e adaptar, mas...
  - Pode corromper sua estrutura, comprometendo aspectos de qualidade:
    - Facilidade de entendimento
    - Manutenibilidade
- Em LPS, é preciso que as adaptações:
  - Possam ser reproduzidas de forma controlada
  - Ocorram em lugares que façam sentido

---

# Plataforma de Computador Clássica

- Definição: Sistema de computador básico no qual aplicações podem rodar
  - Pode ser o hardware (ex: processador)
  - Ou mais comumente: hardware + software
- Exemplos:
  - **Symbian + processadores ARM** da Nokia (Nokia, Motorola e Siemens)
  - **Android** (Smartphones e Tablets)
  - Microsoft **Windows Phone**

---

# Plataformas na Arquitetura de Software

- A arquitetura de um software normalmente consiste de **múltiplas camadas**
- Do ponto de vista de uma camada, sua **camada base** é chamada de **plataforma**
- Exemplos:
  - Java(ME, SE, J2EE) e .NET
  - Eclipse
  - iWorkplace (Infonet)
  - WebIntegrator (Infox)

---

# Por que a demora de LPS?

- Parnas já falava disso em 1975!
- **Motivo:** faltavam alguns pré-requisitos, como:
  - ❑ **Suporte tecnológico** para implementar os princípios de LP
  - ❑ **Processos de desenvolvimento adequados** para lidar com LP
  - ❑ **Conhecimento e experiência no domínio da aplicação** para projetar a LP

---

# Mas aí surgiram algumas tecnologias...

- POO
- Desenvolvimento baseado em Componentes
- Vinculação Tardia
- Middleware
- Gerenciamento de Configuração

---

# E amadurecimento dos processos...

- CMM
- Engenharia de requisitos para lidar com:
  - Commonality
  - Variabilidade
- Técnicas de Modelagem (UML) e Ferramentas de suporte
- RUP

---

# E a experiência no Domínio...

- Foi aumentando e sendo documentada através dos processos e modelos criados
- Mesmo assim, é preciso:
  - Conhecer bem o domínio
    - Sua falta pode levar a introduzir variações desnecessárias e não incluir as necessárias
  - O domínio precisa ser estável
    - Ao mudar muito, o investimento na linha pode ser perdido

---

# Referências

- Software Product Line Engineering (Klaus Pohl) – Capítulo 1
- Henry Ford ([Wikipedia](#))
- Plataformas GM (Wikipedia)
- [iWorkplace](#)
- [WebIntegrator](#)
- On the Design and Development of Program Families (Parnas 75)