

Dinâmica da Evolução (Leis de Lehman)

Eduardo Figueiredo

<http://www.dcc.ufmg.br/~figueiredo>

[Leis de Lehman]

- O crescimento e a evolução de vários sistemas de grande porte foram examinados
- Objetivo
 - Definir uma teoria unificada para evolução de software
- Resultados
 - Um conjunto de oito leis que “governam” a evolução de sistemas

[1 - Mudança Contínua]

- Sistemas devem ser continuamente adaptados ou eles se tornam progressivamente menos satisfatórios
 - O ambiente muda, novos requisitos surgem e o sistema deve ser modificado
 - Após o sistema modificado ser re-implantado, ele muda o ambiente

[2 - Complexidade Crescente]

- A medida que um sistema evolui, sua complexidade aumenta, a menos que seja realizado esforço para mantê-la ou diminuí-la
 - Manutenções preventivas são necessárias
 - Precisa-se de recursos extras para implementar as mudanças

[3 - Auto-Regulação]

- A evolução de software é um processo auto-regulável
 - Atributos do sistema como tamanho, tempo entre versões e número de erros reportados é quase invariável em cada versão do sistema
- Uma grande alteração inibe futuras grandes alterações
 - Consequência de fatores estruturais e organizacionais

[4 - Estabilidade Organizacional]

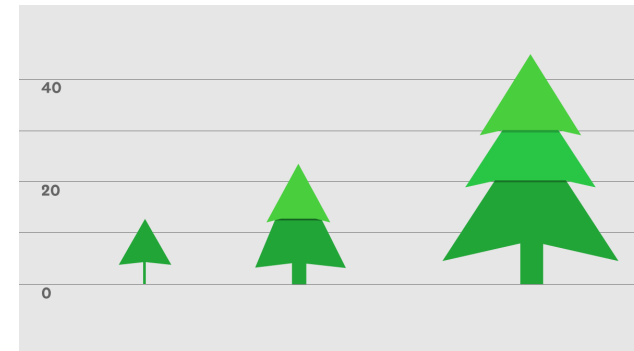
- Durante o ciclo de vida de um programa, sua taxa de desenvolvimento é quase constante
 - Independe de recursos dedicados ao desenvolvimento do sistema
 - Alocação de grandes equipes é improdutivo, pois o *overhead* de comunicação predomina

[5 - Conservação de Familiaridade]

- Durante o ciclo de vida de um sistema, mudanças incrementais em cada versão são quase constantes
 - Um grande incremento em uma release leva a muitos defeitos novos
 - A release posterior será dedicada quase exclusivamente para corrigir os defeitos
- Ao orçar grandes incrementos, deve-se considerar as correções de defeitos

[6 - Crescimento Contínuo]

- O conteúdo funcional de sistemas devem ser continuamente aumentado para manter a satisfação do usuário
 - A cada dia, o usuário fica mais descontente com o sistema
 - Novas funcionalidades são necessárias para manter a satisfação do usuário



[7 - Qualidade Declinante]

- A qualidade de sistemas parecerá estar declinando a menos que eles sejam mantidos e adaptados às modificações do ambiente



[8 - Sistema de Feedback]

- Os processos de evolução incorporam sistemas de *feedback* com vários agentes e *loops*
 - Estes agentes, *loops* e *feedback* devem ser considerados para conseguir melhorias significativas do produto

[Bibliografia da Aula]

- Ian Sommerville. **Engenharia de Software**, 9ª Edição. Pearson Education, 2011.
 - Capítulo 9 Evolução de Software (Seção 9.2)
- Bibliografia adicional
 - M. M. Lehman. **Rules and Tools for Software Evolution Planning and Management**. Annals of Software Engineering, 2001.