

**Eduardo Henrique de Almeida Izidorio - 2020000315**

**Gabriel Peixoto Menezes da Costa - 2020022626**

**Marcia Gabrielle Bonifácio de Oliveira - 2020011319**

**Shelly da Costa Leal - 2020001671**

### **Resenha artigo Métricas e Leis da Evolução do software - A visão dos Anos Noventa**

Pode-se observar que o artigo busca trazer métricas a partir da análise da Evolução do Software baseado no tipo-E. Onde através de uma primeira análise constatou-se que os processos de evolução de software são e devem ser tratados como sistemas restritos e orientados por feedback. Essa análise foi possível graças a estudos prévios e com isso pode se rotular as leis da evolução do Software.

Espera-se que o FEAST/ 1 forneça uma base para dominar os aspectos de feedback do processo de evolução de software, abrindo novos caminhos para modelagem e melhoria de processos. O tema feedback também foi aplicado por Abdel-Hamid e Madnick em seu trabalho sobre o uso da dinâmica de sistemas na modelagem de problemas de gerenciamento de projetos de software. Do ponto de vista da engenharia de software, tais declarações devem, portanto, ser aceitas como uma força reguladora e restritiva externa. Se conseguida, tal generalização forneceria uma base e uma estrutura teórica e prática para a evolução dos programas do tipo-E, ou seja, software resolvendo um problema ou abordando uma aplicação no mundo real. Assim, elas foram posteriormente referidas como leis da evolução do software .

Nos últimos anos, muitas empresas, e a indústria de software em particular, desenvolveram um forte interesse e compromisso com a melhoria disciplinada de processos. A melhoria de processos de negócios e de software está fortemente ligada e é interdependente.

O presente artigo tem como foco a melhoria de processos de software. Quanto a outros sistemas complexos de feedback, a dinâmica dos processos de desenvolvimento e evolução de software do mundo real possuirá um grau de autonomia e estabilidade global.

Além disso, o sistema que está sendo modelado inclui elementos que contêm modelos implícitos de si mesmos. Outros critérios incluíram a disponibilidade de dados históricos sobre a evolução do sistema para permitir a análise inicial da caixa preta para detectar a presença ou ausência de comportamento semelhante ao feedback. Os primeiros dados de evolução do sistema a serem disponibilizados para o projeto FEAST/1 foram no sistema de transações financeiras Logica plc Fastwire. Este sistema de 8 anos está agora instalado em cerca de cem sites.

À medida que mais familiaridade com o histórico do sistema foi alcançada, os critérios e, portanto, o subconjunto selecionado, tiveram que ser alterados para produzir o conjunto. Por definição, um sistema do tipo E opera em um domínio sempre sujeito a mudanças em uma taxa acelerada pelo desenvolvimento, instalação e operação do sistema.

Crescimento do sistema razões. Estes também foram incluídos na análise se envolverem crescimento do sistema em, pelo menos, um módulo.

Com o custo de armazenamento diminuindo em todos os níveis, o tamanho do sistema não é, em si, uma grande preocupação. Pode, portanto, ser visto como um monitor independente e composto da evolução do sistema que, dentro de certos limites, não é planejado nem gerenciado. Os determinantes de tamanho incluem design do sistema, estilo e experiência do programador, cronogramas e restrições de desenvolvimento, intensidade do desejo de obter compacidade ou clareza<sup>6</sup>. A grande maioria do trabalho de métricas de software relatado tende a usar locs (linhas de código) como a medida do tamanho do sistema.

Na ausência de uma medida melhor, a contagem de módulos também serve como um estimador inicial da funcionalidade e potência do sistema. Foi mostrado que estes eram essencialmente semelhantes, mas com a medida locs fornecendo uma imagem menos consistente do comportamento evolutivo do sistema do que a contagem de módulos. O número de módulos em um sistema específico também não depende, em geral, da prática individual do programador. Os números dos módulos podem, portanto, fornecer uma medida mais consistente do tamanho do sistema e, portanto, um indicador melhor, embora reconhecidamente grosseiro, da funcionalidade do sistema.

A tendência de longo prazo da média é usada como uma medida de tamanho e, por implicação, como um móvel do crescimento incremental dos sistemas do tipo E conforme eles estimulam a potência do sistema, na investigação do FEAST. A análise mostra claramente a tendência ascendente do crescimento do sistema, fornecendo também informações adicionais para o FW e de identificar o modelo de tendência de crescimento que, pelo menos para esse sistema, deve ser o preferido. Assim, as influências que tendem a aumentar a funcionalidade do sistema, ou seja, o crescimento em direção aos picos, podem ter sua origem no feedback positivo.

Para os modelos de tendência estimados a partir do conjunto de saída identificada na quarta lei como sendo necessária para levar completo de 21 pontos de dados, as medidas estatísticas da o sistema de um lançamento para o outro. Tal crescimento se deve, em parte, ao aumento da complexidade da aplicação, por exemplo, à medida que recursos não incluídos na definição original do sistema, e muitas vezes ortogonais a ela, são adicionados. Nenhum sistema pode crescer para sempre. No geral, no entanto, parece indicar a força da dinâmica do sistema.

Esse fenômeno apóia a observação feita por um dos autores há muitos anos com relação à evolução do OS/360 de que em vez de os gerentes gerenciarem o sistema, o sistema gerencia os gerentes. É claro que deve ser entendido que a referência aqui é a evolução de longo prazo, não às especificidades de decisões individuais, muitas vezes localizadas no tempo, espaço do sistema e espaço de implementação.

A alternativa, de que os resultados da investigação demonstram que as leis e a hipótese não são de relevância geral, embora satisfeitas para instâncias particulares de sistemas do tipo E e seus processos de evolução, não pode, neste estágio, ser descartada. O estudo FEAST/1 já fez um progresso visível ao ilustrar como os conceitos de medição podem ser aplicados ao estudo da evolução do software.