**1. Título:**

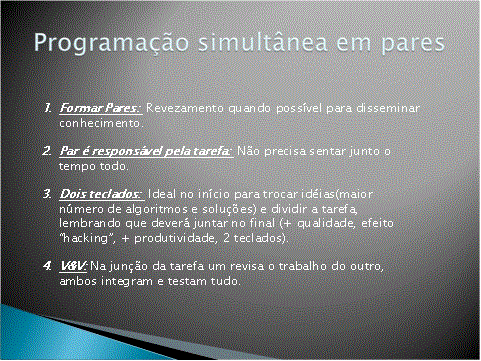
**Programação simultânea em pares, evolução da técnica de programação em pares melhorada através da engenharia simultânea.**

**2. Resumo**

Programação em pares melhora a qualidade, reduz a quantidade de defeitos, porém aumenta o esforço (mensurado em horas de trabalho) do projeto.

A nova técnica de programação simultânea em pares preserva a melhora na qualidade e redução da quantidade de defeitos, além de obter êxito no intento de diminuir o esforço do projeto.

Foram realizados três experimentos preliminares visando validar a viabilidade da pesquisa e comprovaram a efetividade da nova técnica de programação simultânea em pares, tanto em melhorar o impacto negativo ao projeto em relação ao custo e prazo causado pelo aumento do esforço, quanto em preservar a melhora na qualidade e diminuição dos defeitos.

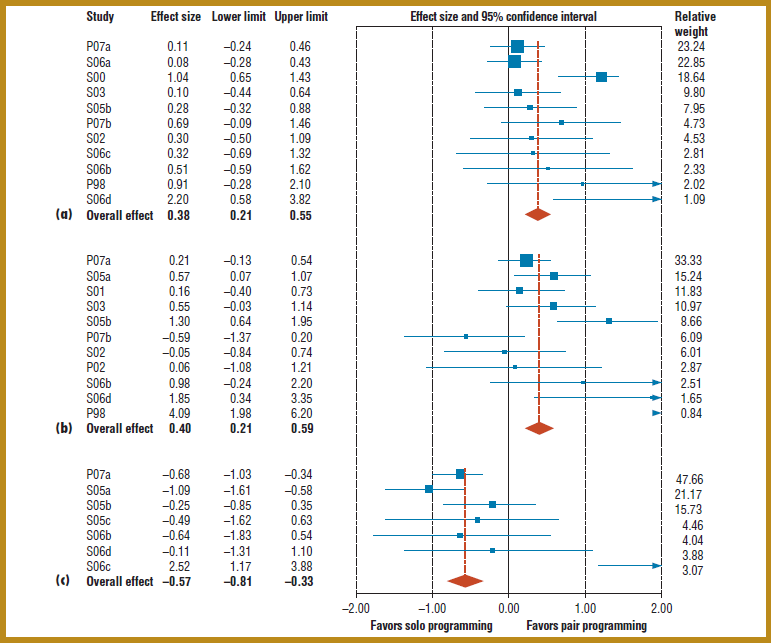


**3. Referências**

**Are Two Heads Better than One? On the Effectiveness of Pair Programming**

Tore Dybå, Erik Arisholm, Dag I.K. Sjøberg, Jo E. Hannay, and Forrest Shull

IEEE SOFTWARE P u b l i s h e d b y t h e I E E E C o m p u t e r S o c i e t y 0 7 4 0 - 7 4 5 9 / 0 7 / $ 2 5 . 0 0 © 2 0 0 7 I E E E



“item c é o esforço mensurado em homem-hora. PP aumentou o esforço.”

**A Field Study of Developer Pairs: Productivity Impacts and Implications**

Allen Parrish, Randy Smith, David Hale, and Joanne Hale, University of Alabama

IEEE SOFTWARE P u b l i s h e d b y t h e I E E E C o m p u t e r S o c i e t y 0 7 4 0 - 7 4 5 9 / 0 4 / $ 2 0 . 0 0 © 2 0 0 4 I E E E

“Em um estudo de campo industrial, foram examinados: a produtividade do programador e o tamanho da equipe. Mostrou que, independentemente do tamanho ou experiência, quanto mais membros da equipe de tempo passar trabalhando simultaneamente nos mesmos módulos de código, menos produtivo torna-se. Em outras palavras, equipes são mais produtivas quando seus membros trabalham de forma independente.”

“A programação em pares entrega supostamente código de qualidade com pouco

perda de produtividade. Estudo de campo dos autores, fora o ambiente de programação, mostra que as equipes de duas pessoas que trabalham independentemente são mais produtivos do que aqueles que trabalham simultaneamente.”

**Should NASA Embrace Agile Processes?**

Jefferey Smith, Tim Menzies

Lane Department of Computer Science

West Virginia University

PO Box 6109, Morgantown

WV, 26506-6109, USA;

This research was conducted atWest Virginia University

under NASA contract NCC2-0979.

“Para os demais projetos, o nosso modelo apenas endossou AP / XP sobre abordagens convencionais em um relativamente pequeno e conjunto especializado de casos, isto é, quando:

* o projeto é relativamente pequeno
* uma abundância de desenvolvedores existe
* é necessário um tempo de desenvolvimento rápido”

“Como há um aumento do esforço é necessário abundância de desenvolvedores.”

**4. Experimento**

**Podemos obter dados (medidas) objetivas do experimento?**

Sim, podemos por exemplo:LOC por hora por desenvolvedor ou pontos de função por hora, bugs por ponto de função, custo financeiro do ponto de função, senioridade/ produtividade, senioridade/custo financeiro, entre outros, abaixo segue uma versão preliminar ainda em evolução das métricas dos experimentos:

A métrica que fornece informações sobre a **produtividade**, como a evolução do par do programador à medida que o andamento do projeto avança e também permite comparar a produtividade de estilos de programação diferentes. Neste experimento, a produtividade pode ser calculada para estilos de programação solo, pares e pares simultâneos, para cada iteração como uma relação de linhas de código lógicas produzidas e esforço despendido.

Assim, a produtividade do estilo de programação N (N é par, par simultâneo ou solo) é

definido como: **Produtividade N = Linhas de código / Esforço**

**Densidade de desvios padrão de codificação:** A primeira métrica usada para descrever os efeitos da programação simultânea em par, solo e em pares na qualidade está relacionada com a aderência aos padrões de codificação.

De acordo com a filosofia ágil: A equipe do projeto é responsável por definir os padrões de codificação no início de cada projeto, e o código é comparado contra esses mesmos padrões quando decorrentes dessa métrica é medida através do número de desvios encontrados a partir dos padrões de codificação relativamente à quantidade de código feita com cada estilo de programação (todas as linhas físicas). Assim, a densidade de codificação de desvios padrão para o estilo de programação N (N é par, par simultâneo ou solo) por cem linhas de código é definido como:

**Densidade de desvios padrão de codificação= Número de falhas de aderência ao padrão de códificação / Número total de linhas de código X 100**

**Taxa de Comentário:** Outra métrica de qualidade utilizada neste experimento é relação de comentário, que é calculada como a relação entre as linhas de comentário e totais (isto é, físicas) de linhas de código. Quanto maior a taxa, mais legível e de fácil manutenção é o código. Assim, a razão de comentário para o estilo de programação N é definido como:

**Taxa de Comentário = 1 - Número de linhas de comentários / Número de linhas de programação**

O que foi feito?

Foram realizados três experimentos preliminares visando avaliar a viabilidade da pesquisa, o primeiro foi em um projeto de homebroker para uma corretora de valores, foi um projeto profissional, com desenvolvedores extremamente qualificados e profissionais.

Definições:

Homebroker – Permite operar no mercado de ações, derivativos.

Facebroker – Homebroker dentro do facebook.

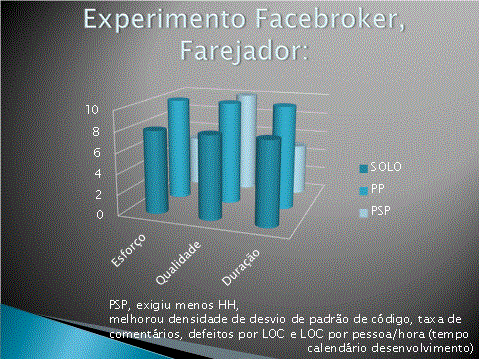
Farejador – baseado em análise gráfica para buscar as melhores oportunidades tanto de compra quanto de venda.

Detalhes:

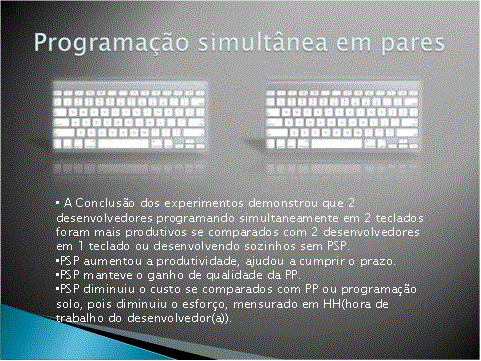
Diversas funcionalidades desenvolvidas usando: Programação em pares, programação simultânea em pares e solo.

Desenvolvedores experientes dividiram as tarefas no início, mas houve problema na hora de juntar, houve retrabalho neste experimento preliminar.

**Exemplo de analise dos resultados do experimento preliminar:**



**Conclusão dos resultados dos experimentos preliminares:**



**5. Prazos disponíveis:**

Início do curso: 10/09/2012.

Término do curso: 10/09/2015.