# Uso da Ferramenta de Análise Estática Klocwork na Motorola – Relato de experiência

**Rachel C. D. Reis, Denise P. Prado, Maria G. S. M. Fernandes**

Departamento de Software Básico - Instituto de Pesquisas Eldorado
Rod. SP340 KM 118,5 s/n – Bairro Fazenda Pau D'Alho CEP 13086-902 – Campinas – SP – Brasil

{rachel.pereira,denise.prado,maria.fernandes}@eldorado.org.br

***Abstract.*** *This paper describes the relevance and benefits due to the use of static analysis technique as an important helper to guarantee software quality. It presents the experience gained in working with a static analysis tool usage, called Klocwork, integrated into Motorola development environment.*

***Resumo.*** *Este artigo discorre sobre a relevância e benefícios decorrentes do uso da técnica de análise estática como importante aliada para garantir a qualidade de software. Apresenta a experiência adquirida com o uso da ferramenta de análise estática, chamada Klocwork, integrada no ambiente de desenvolvimento da Motorola.*

## Introdução

Nos últimos anos, as empresas cujos produtos são compostos por hardware e software têm buscado atingir a qualidade em seus produtos de forma que isso as torne mais competitivas em um mercado cada vez mais exigente. Os métodos e técnicas para garantir a qualidade de hardware estão num patamar de estabilidade delineados por suas características físicas quase que imutáveis. Já não se pode dizer o mesmo com relação à estabilidade do software, pois este possui uma flexibilidade similar à mente humana, em que não há limites para variações de suas criações [Pemmaraju 1998].

Neste contexto, as organizações maduras estão gastando esforços para aprimorar seus conhecimentos em técnicas, métodos e processos de qualidade de software que garantam a distribuição de um produto e/ou serviço com rapidez, qualidade e custo reduzido.

A empresa Motorola [Motorola 1947], considerada uma das gigantes do mercado de telefonia celular, adquiriu ao longo de sua existência, conhecimentos de que durante o ciclo de desenvolvimento de software, um pequeno erro de código poderia resultar em uma vulnerabilidade crítica, que comprometeria o funcionamento de todo um sistema, no caso, o telefone celular. Para evitar tais ocorrências, foi fundamental a escolha e a implementação de mecanismos de revisão do código fonte, que facilitassem a detecção dos erros em fases iniciais de desenvolvimento. Como um dos mecanismos mais recomendados pelos especialistas de segurança e qualidade de código é a análise estática [Graff and Van 2003], a Motorola selecionou a ferramenta Klocwork [Klocwork 1996].

Este artigo tem como objetivo discorrer sobre a experiência da empresa Motorola na utilização da Klocwork, mostrando os benefícios encontrados com a implementação e uso da ferramenta na divisão de telefonia celular.

## Análise Estática de Código

Análise estática em software é uma técnica que varre o código fonte de um programa, percorrendo-o linha a linha, à procura de erros, e produz um relatório que descreve os defeitos que potencialmente podem existir. Em sistemas com milhões de linhas de código e que são desenvolvidos em ambientes *multisite*, onde o desenvolvimento é realizado paralelamente por times de projeto geograficamente distribuídos, apenas a utilização de técnicas tradicionais como a revisão manual [Teixeira et al. 2007] para verificar um código desta dimensão seria custoso e ineficiente. A aplicação desta técnica em grandes empresas globalizadas se tornou possível devido ao desenvolvimento de ferramentas de análise estática de código.

O emprego das ferramentas de análise estática nas fases iniciais de desenvolvimento, onde há mais oportunidades de injeção de erros, é um dos modos mais efetivos para reduzir os custos de correção e conseqüentemente elevar as taxas de ROI (*Return of Investment*) das empresas que as utilizam. Segundo [Gordon 2006] um defeito encontrado em fases posteriores ao ciclo de desenvolvimento, ou seja, quando o produto está em fase final de teste ou mesmo já disposto ao mercado, é entre 50 a 1000 vezes mais caro de ser corrigido, do que se fosse detectado em fases iniciais de desenvolvimento, Figura 1.

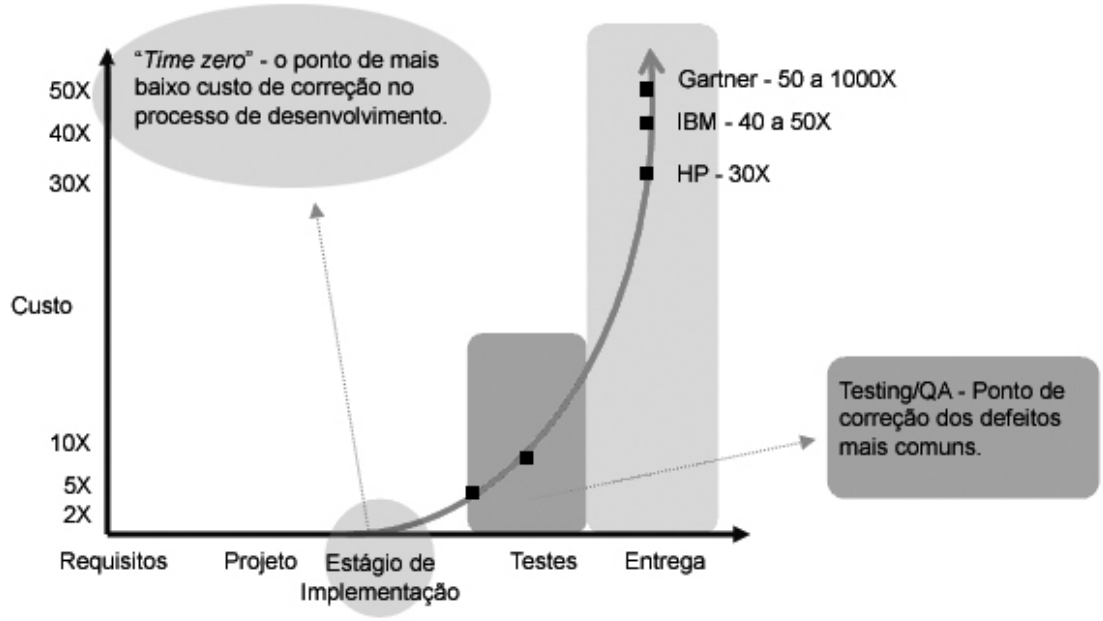


**Figura 1. Custo para correção de defeitos no ciclo de desenvolvimento de código.**

Dentre as diversas ferramentas comerciais existentes [Feiman and MacDonald 2007a] [Feiman and MacDonald 2007b] [MacDonald and Feiman 2007], a Motorola selecionou a ferramenta Klocwork, pois ela provê os tipos de erros que a empresa

estava interessada em analisar, e não havia necessidade de modificar o código fonte. Além disso, a Klocwork é capaz de entender o código através de uma base de conhecimento que é alimentada recursivamente com informações do sistema analisado, não se restringindo apenas à procura de padrões.

A ferramenta Klocwork foi projetada para análise estática de códigos C, C++ e Java visando atender empresas que produzem sistemas de software em larga escala, tipicamente sistemas com 500.000 ou mais linhas de código. Ela possibilita a identificação precoce de erros potenciais que poderiam provocar a indisponibilidade da aplicação, bem como graves falhas de segurança que poderiam ser exploradas por agentes maliciosos.

A ferramenta disponibiliza uma interface gráfica, chamada *Insight*, para visualização do código facilitando a compreensão da arquitetura do sistema (Figura 2), a navegação e análise dos defeitos e a coleta de dados para a criação de gráficos de tendências de ocorrências dos tipos de erros (Figura 3).

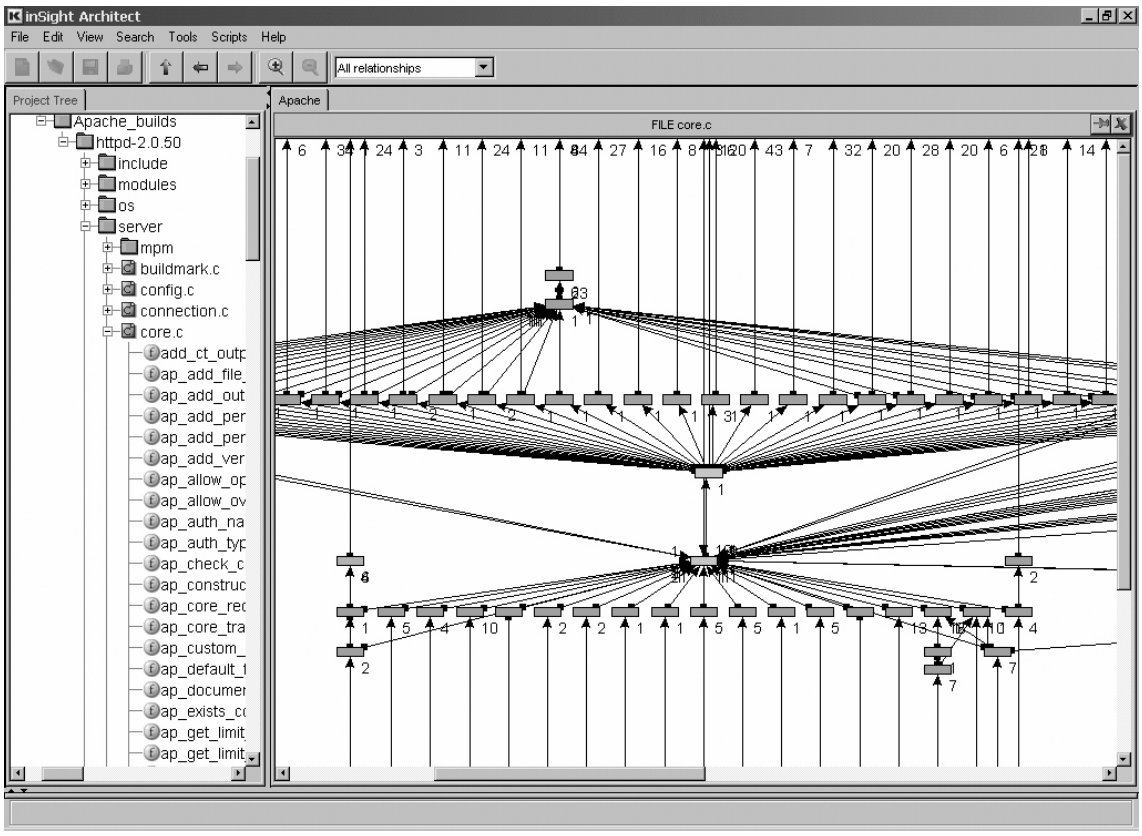


**Figura 2. Insight - Visão da Arquitetura do Sistema.**

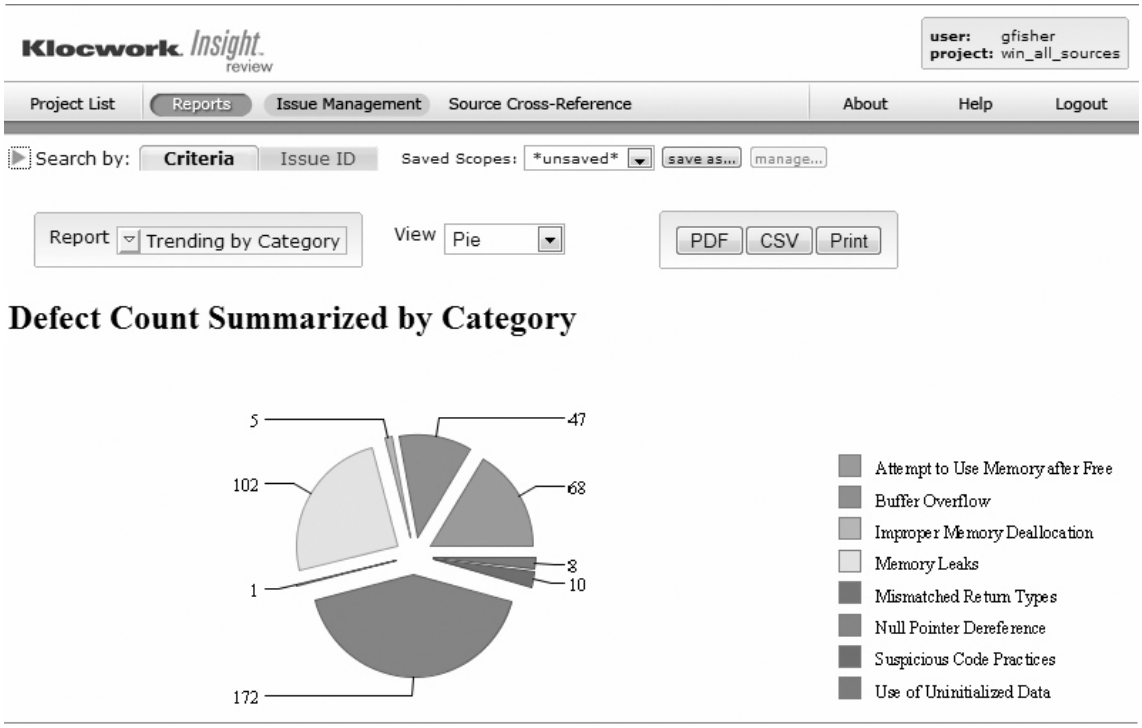


**Figura 3. Insight - Categorias de defeitos encontrados em um sistema.**

## Klocwork na Motorola

### O Projeto

A Motorola iniciou a utilização da ferramenta Klocwork como uma iniciativa de prover uma ferramenta de análise estática durante o ciclo de desenvolvimento de código na divisão Mobile Devices. Tinha como perspectiva identificar e corrigir erros encontrados no "*time zero*", ou seja, na fase inicial de desenvolvimento.

A divisão Mobile Devices é a responsável pela criação e desenvolvimento de software básico e aplicações para os telefones celulares da Motorola. Os fatores chaves que levaram à sua escolha para a utilização de uma ferramenta de análise estática foram:

- apesar da existência de um processo de qualidade já implementado e incorporado no ciclo de desenvolvimento de software, a grande dimensão e complexidade do sistema dão margem à maior possibilidade de injeção de erros durante o ciclo de desenvolvimento, notadamente na fase de codificação;
- a percepção da alta gerência da divisão e do time de qualidade, de que muitos erros eram encontrados em fases finais, próximas às datas de entrega dos produtos, aumentam consideravelmente o custo de correção;
- os tipos de erros mais comuns poderiam ser facilmente detectados por ferramentas de análise estática e, se corrigidos no momento próximo de onde são injetados, são de fácil correção, minimizando o impacto no planejamento de esforço e prazo de entrega do produto.

Durante o processo de implantação da ferramenta Klocwork algumas dificuldades foram identificadas e contornadas para tornar viável o sucesso do projeto. Dentre as principais podemos destacar:

- uniformizar o uso da ferramenta para os grupos de desenvolvimento geograficamente separados. Devido ao grande número de usuários espalhados em vários sites da divisão de Mobile Devices era importante que fosse estabelecido um processo bem definido, centralizado e claro de utilização da ferramenta pelos usuários e administração pelo time de suporte;
- customizar a ferramenta para atendimento das necessidades da Motorola;
- disseminar o conhecimento do uso da ferramenta sem treinamentos específicos para cada desenvolvedor.

O real sucesso dessa iniciativa foi alcançado nos últimos 2 anos, quando o uso da ferramenta Klocwork foi integrado oficialmente no processo de desenvolvimento de software e um time de especialistas no uso da ferramenta foi criado para dar todo o suporte aos usuários.

O time foi formado com a proposta de disseminar conhecimentos sobre as melhores práticas de programação e sobre a melhor forma de usar a ferramenta Klocwork: como executá-la, como interpretar os erros apontados, indicar a melhor forma de corrigí-los, etc. Além disso, os membros do time deveriam conhecer profundamente o ambiente de desenvolvimento e manter a infraestrutura necessária para o uso transparente da ferramenta pelos desenvolvedores.

A proposta original foi amplamente atendida e extendida para explorar as funcionalidades complementares oferecidas pela Klocwork. Além da análise estática, esta também oferece análise de uso de memória local (*stack analysis*) [Regehr 2004a] [Regehr 2004b], análise de complexidade do código [IEEE 1990] [McCabe and Watson 1994], análise de arquitetura e métricas sobre a densidade de defeitos [Klocwork 1996]. Portanto, atualmente as atividades do time são:

- Analisar os defeitos encontrados e reportá-los aos times de desenvolvedores, de forma a ajudá-los a entender a melhor maneira de corrigí-los;
- Manter a infraestrutura da ferramenta atualizada;
- Participar de fóruns de discussão técnica e prover suporte aos usuários orientando-os no uso da ferramenta;
- Analisar a complexidade do código e sugerir melhorias;
- Analisar o gerenciamento de memória local das aplicações dos celulares, de forma a sugerir otimizações, visto ser esse um dos recursos mais críticos, pela sua escassez;
- Fornecer à alta gerência informações sobre a quantidade e tipos dos defeitos encontrados no código de cada produto.

**Os Ganhos**

Como participantes do time de suporte à ferramenta Klocwork, percebemos que os ganhos obtidos através da experiência no uso de uma ferramenta de análise estática em código de celulares, são vários.

O mais evidente, e amplamente ilustrado na literatura especializada [Gordon 2006], diz respeito à redução no custo do produto quando a correção de erros é realizada nas fases iniciais de desenvolvimento de código, conforme ilustrado na Figura 1.

O desenvolvedor que faz uma parte de um componente de software de um telefone celular só tem conhecimento da pequena porção sob a sua responsabilidade. Muitas vezes ele não tem a noção de que uma alteração na sua parte pode afetar o sistema como um todo. Usando a ferramenta Klocwork este problema é minimizado, porque a ferramenta oferece a possibilidade de analisar uma porção de código ou uma aplicação inteira, onde as relações entre as diversas partes também são analisadas.

O uso da ferramenta além de facilitar a identificação e correção de erros em uma fase precoce, tem ainda um caráter educativo, porque os desenvolvedores aprendem com os erros e tendem a adotar boas práticas de programação, conseqüentemente errando menos no futuro.

A alta gerência recebe informações freqüentes sobre o real estado da qualidade de seus produtos durante o período de construção, o que permite a tomada de decisões rápidas para sanar possíveis pontos fracos. Por exemplo, é possível verificar quais são os tipos de erros mais freqüentes e, com base nesta informação, organizar treinamentos específicos para que os desenvolvedores aprendam a evitá-los.

## Conclusão

A ferramenta Klocwork oferece um conjunto de funcionalidades muito útil para auxiliar na construção de software com mais qualidade, por atuar num momento crucial onde ocorre a maior possibilidade de injeção de erros, durante a fase de codificação.

Uma das constatações mais importante da experiência adquirida, na divisão Mobile Device da Motorola, foi a de que é imprescindível ter um grupo ou um profissional com profundo conhecimento sobre a ferramenta e sobre o ambiente onde ela será utilizada. Por possuir várias funcionalidades, além da análise estática, e com a característica de reconhecimento de código e não apenas padrões, ela é complexa e exige uma boa administração para ter os seus benefícios maximizados.

É importante que o uso da ferramenta seja um item inserido no processo de qualidade de software. A experiência mostra que os usuários, quando pressionados por prazos e demandas, tendem a evitar passos adicionais que sejam opcionais.

Além disso, é importante que ela seja oferecida ao desenvolvedor para ser incorporada ao seu cotidiano como um instrumento que vai ajudá-lo e não como um obstáculo a mais a ser superado. Isso ocorreria se ele não tivesse suporte imediato sempre que encontrasse dificuldades no uso da ferramenta. Portanto, o uso da ferramenta deve ser transparente para o desenvolvedor, ou seja, executar a análise estática deve ser tão simples quanto executar uma compilação.

A atuação do time de Klocwork tornou-se tão efetiva na divisão Mobile Devices que outras divisões estão requisitando a ajuda para implementação do uso da ferramenta como parte de seu processo. Isso demonstra além da boa aceitação dos resultados

obtidos até agora, o desejo de utilizar as melhores práticas para construir produtos com mais qualidade.

## Referências

Feiman, J. and MacDonald, N. (2007a), “Static Application Security Testing: Vendors and Products, Part1”, Doc ID: G00149354, January, available at: http://www.gartner.com. (último acesso: 10 de março de 2008)

Feiman, J. and MacDonald, N. (2007b), “Static Application Security Testing: Vendors and Products, Part3”, Doc ID: G00150687, August, available at: http://www.gartner.com. (último acesso: 10 de março de 2008)

Gordon, I. (2006) “Automated Source Code Analysis: Reduce Customer and QA Defects to Save Time and Money!”, September, available at: http://www.nohau.se/images/pdf/Test-roadshow-nohau-klocwork.pdf. (último acesso: 10 de março de 2008)

Graff, M. G. and Van Wyk, K. R. (2003) “Secure Coding: Principles and Practices”. Cambridge, MA: O'Reilly.

Institute of Electrical and Electronics Engineers. (1990) “IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries”. New York, NY.

Klocwork, Inc. (1996) http://www.klocwork.com.

MacDonald, N. and Feiman, J. (2007), “Static Application Security Testing: Vendors and Products, Part2”, Doc ID: G00150657, August, available at: http://www.gartner.com. (último acesso: 10 de março de 2008)

McCabe, T. J. and Watson, A. H. (1994) “Software Complexity.” Crosstalk, Journal of Defense Software Engineering 7, p. 5-9.

Motorola, Inc. (1947) http://www.motorola.com.

Pemmaraju, K (1998) “The Quest for Software Quality” http://www.cigital.com/papers/download/sil-india-dec98-kp.doc (último acesso: 10 de março de 2008).

Regehr, J.(2004a) “Using Static Analysis to Bound Stack Depth”, October, available at: http://www.gartner.com. (último acesso: 10 de março de 2008)

Regehr J.(2004b) “Say no to stack overflow.” Embedded Systems Programming, 17(10), October.

Teixeira, M. et al. (2007) “Avaliação de Ferramentas de Análise Estática de Código para Detecção de Vulnerabilidades.”, disponível em: http://www.di.fc.ul.pt/sobre/documentos/tech-reports/07-29.pdf. (último acesso: 10 de março de 2008)