



ciência desenvolvimento sociedade

## XXVI SALÃO DE INICIAÇÃO CIENTÍFICA

20 a 24 de outubro - Campus do Vale - UFRGS



<b>Evento</b>	Salão UFRGS 2014: SIC - XXVI SALÃO DE INICIAÇÃO CIENTÍFICA DA UFRGS
<b>Ano</b>	2014
<b>Local</b>	Porto Alegre
<b>Título</b>	Verigraph
<b>Autor</b>	THIAGO RAFAEL BECKER
<b>Orientador</b>	RODRIGO MACHADO

O projeto Verites busca desenvolver técnicas de verificação, validação e testes para o desenvolvimento de sistemas computacionais, e desenvolver ferramentas de apoio para as técnicas desenvolvidas. Gramática de grafos é um modelo de computação, no qual estados são descritos por grafos e computações são descritas por um conjunto de regras de reescrita. O modelo envolve a intuitividade da descrição visual com grafos, e uma semântica de execução precisa. Para representar a evolução de software (como o software é modificado durante o seu desenvolvimento e manutenção), foi proposto o modelo de gramáticas de grafos de segunda ordem, em que estados são representados por gramáticas de grafos, e o comportamento é descrito por regras de reescrita desta gramáticas.

As técnicas de verificação podem ser divididas em duas classes: estáticas, que verificam propriedades estruturais do sistema, e dinâmicas, que verificam propriedades do sistema durante a execução. Existem ferramentas capazes de verificar uma ou outra destas classes, mas não ambas, e estas são baseados em fundamentos teóricos diferentes, sendo difícil de transportar o modelo entre os elas.

O objetivo deste trabalho é construir uma ferramenta, VeriGraph, que unifique análises estáticas e dinâmicas de gramáticas de grafos usando uma base teórica, e servir como uma plataforma para experimentação com novas técnicas de verificação. Para isso, foi necessário estudar a literatura de reescrita de grafos, e desenvolver a arquitetura para tal sistema de acordo com a teoria escolhida.

O sistema foi construído desde o início, e hoje é capaz de realizar a reescrita de grafos e gerar o espaço de estados para executar a verificação dinâmica de sistemas. Como passos futuros, planejamos implementar um verificador de modelos, e implementar a reescrita de segunda ordem, sendo o primeiro sistema a implementar técnicas de verificação para este modelo teórico.

O aluno esteve envolvido desde a concepção do sistema, ajudando na definição de partes da arquitetura, como a representação de grafos e regras, e o algoritmo para reescrita de grafos, de acordo com boas práticas de engenharia de software. Foi responsável pela implementação do algoritmo de reescrita, e o algoritmo de geração de espaço de estados. O software foi estruturado de forma a ser reusável, com estruturas e algoritmos genéricos, valendo-se das capacidades da linguagem escolhida (Haskell) para criar tais estruturas.