

Engenharia Reversa de Padrões de Interação

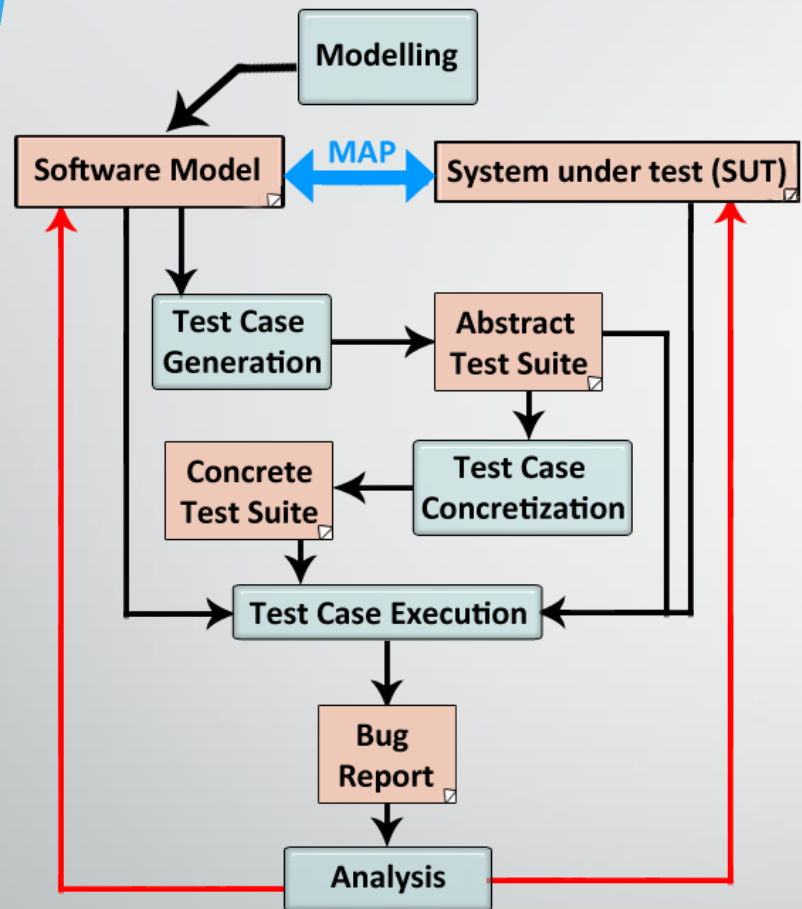
Clara Sacramento / Ana Paiva (orientadora)

Janeiro 2014

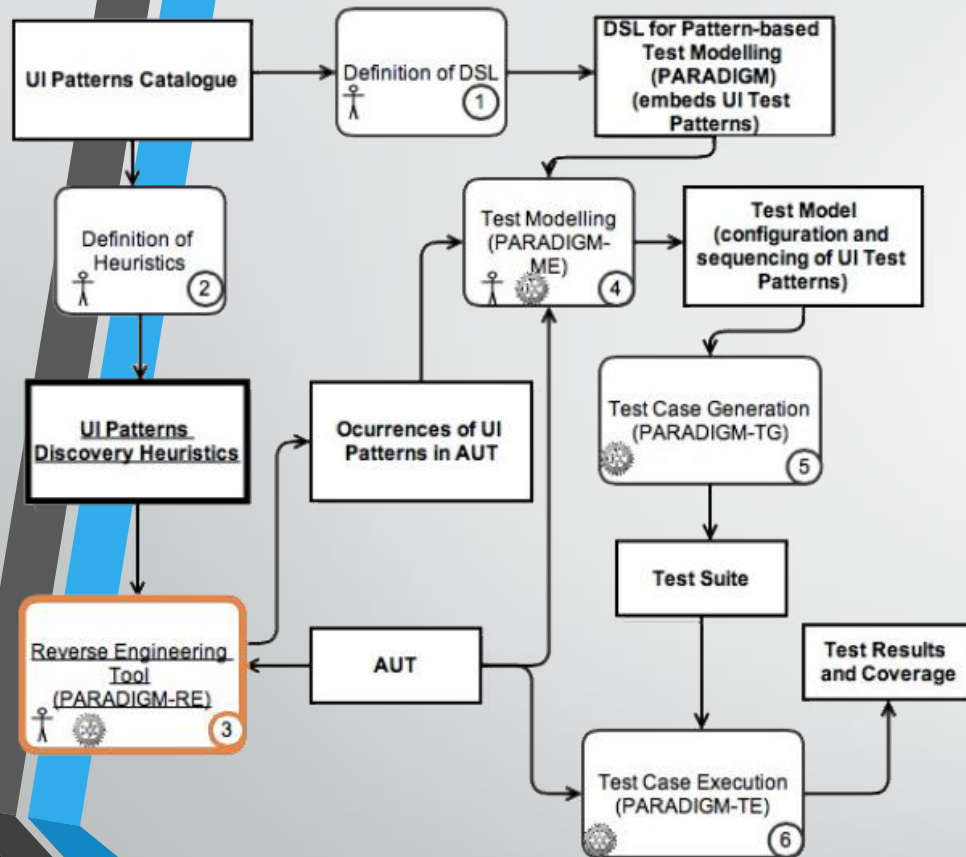


Contexto

- Aplicações Web são muito usadas hoje em dia
- É necessário haver métodos adequados de teste de GUIs – e maneira de os automatizar
- Um dos métodos de testar GUIs é **testes baseados em modelo** (*model-based testing*)



- Testes baseados em modelo: geração automática de testes unitários a partir de um **modelo** que reflecte os requisitos do sistema
- Criar modelos manualmente é penoso e pode conter erros
- Uma boa solução é gerar o modelo parcialmente recorrendo a **engenharia reversa** da GUI
- Como uma GUI é desenvolvida baseadas em **padrões de interação**, são o foco das estratégias de teste genéricas que se pretende gerar



Contexto

- Esta tese encontra-se inserida no projecto de investigação PBGT (*Pattern Based GUI Testing* – teste de GUIs baseado em padrões)
- Esta tese pretende continuar o trabalho realizado na componente PARADIGM-RE

Objetivos

- Aplicar um **algoritmo de aprendizagem** para refinar as heurísticas de identificação dos padrões já existentes
- Verificar se o novo conjunto de heurísticas obtidas permite melhor identificação de padrões
- Implementar **identificação** de mais padrões
- Produção de **modelo XMI** para PARADIGM-ME

Estado de Arte - tópicos

- Engenharia reversa - estática, dinâmica, híbrida
 - **Híbrida/Dinâmica:** Abordagens que extraem informação de traços de execução [TraceServer, jRapture, ReGUI, EvoTrace, e outros]
 - **Estática:** Abordagens que extraem informação de aplicações Web [ReWeb, WARE, WANDA, Crawljax, e outros]
 - **Híbrida:** Abordagens que inferem padrões de aplicações Web [não se encontraram resultados fora do trabalho realizado no projecto]
- Algoritmos de *data mining* - pontos fortes e fracos [Clustering, ILP]
- Padrões de interação com utilizador

Plano de trabalho



Referências

- **TraceServer:** Igor Andjelkovic and Cyrille Artho. *Trace server: A tool for storing, querying and analyzing execution traces*. In JPJF Workshop, Lawrence, USA, 2011
- **jRapture:** John Steven, Pravir Chandra, Bob Fleck, and Andy Podgurski. *jRapture: A capture/replay tool for observation-based testing*, volume 25. ACM, 2000
- **ReGUI:** Inês Coimbra Morgado, Ana Paiva, and João Pascoal Faria. *Reverse engineering of graphical user interfaces*. In ICSEA 2011, The Sixth International Conference on Software Engineering Advances, pages 293–298, 2011.
- **EvoTrace:** Michael Fischer, Johann Oberleitner, Harald Gall, and Thomas Gschwind. *System evolution tracking through execution trace analysis*. In Program Comprehension, 2005. IWPC 2005. Proceedings. 13th International Workshop on, pages 237–246. IEEE, 2005.
- **ReWeb:** Filippo Ricca and Paolo Tonella. *Understanding and restructuring web sites with ReWeb*. Multimedia, IEEE, 8(2):40–51, 2001
- **WARE:** Giuseppe Antonio Di Lucca, Anna Rita Fasolino, and Porfirio Tramontana. *Reverse engineering web applications: the WARE approach*. Journal of Software Maintenance and Evolution: Research and Practice, 16(1-2):71–101, 2004.
- **WANDA:** Giuliano Antoniol, Massimiliano Di Penta, and Michele Zazzara. *Understanding web applications through dynamic analysis*. In Program Comprehension, 2004. Proceedings. 12th IEEE International Workshop on, pages 120–129. IEEE, 2004
- **Crawljax:** Danny Roest. *Automated regression testing of ajax web applications*. Master's thesis, Delft University of Technology, February 2010.