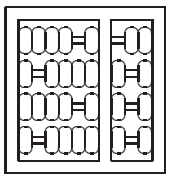


INSTITUTO DE COMPUTAÇÃO
UNIVERSIDADE ESTADUAL DE CAMPINAS

Anais do 4º Workshop de Teses de Doutorado em Andamento do IC-UNICAMP

Claudia Bauzer Medeiros, Alan Massaru Nakai, Carla Geovana do Nascimento Macario, Gilberto Zonta Pastorello Jr, Jorge Lima de Oliveira Filho, Patrick H. S. Brito, Rodrigo Dias Arruda Senra, Nádia Kozievitch (Eds.)

Technical Report - IC-08-025 - Relatório Técnico

September - 2008 - Setembro

Apresentação

Este relatório técnico contém resumos de 24 trabalhos apresentados no 4^o Workshop de Teses de Doutorado do Instituto de Computação da UNICAMP. O workshop, realizado de 22 a 24 de setembro de 2008, permitiu que doutorandos do Instituto apresentassem os principais aspectos de suas pesquisas. Cada capítulo corresponde a uma apresentação, sendo o texto limitado a 4 páginas. A participação foi voluntária e o perfil acadêmico dos participantes foi variado, cobrindo desde alunos recém-admitidos no programa até aqueles que já tinham defesa marcada em setembro de 2008. A publicação dos resumos sob forma de um relatório técnico tem por objetivo uma maior divulgação de trabalhos de doutorado em andamento no IC. Além disso, é um registro sucinto do estado de várias dessas pesquisas. Como coordenadora do Workshop, destaco o trabalho dos sete alunos de doutorado que foram os efetivos organizadores do evento e que são co-editores deste relatório - Alan, Carla, Gilberto, Jorge, Patrick, Rodrigo e Nádia. Ressalto, igualmente, o trabalho de todos os autores de capítulos, que se dispuseram a apresentar trabalhos em andamento e seus orientadores que incentivaram tal participação. Finalmente, agradeço à coordenação de pós-graduação do IC pelo apoio.

Claudia Bauzer Medeiros
Coordenadora do 4^o WTD
Professora - Instituto de Computação - UNICAMP

Sumário

1. Workflow-based Sensor Data Management

Authors: Gilberto Zonta Pastorello Jr and Claudia Bauzer Medeiros

In many domains, scientific models and applications have relied on sensing systems. Initially, the entire data path between the data acquisition by sensors and data consumption by an application was centralised. However, this scenario has evolved into far more complex data paths. This new scenario is depicted in Figure 1. As sensors evolved and data became more widely available, scientific applications started to rely on shared data sets, revealing new sensor data management issues. As shown in the bottom of the figure, the used sensors can vary from small ground-based sensor networks to large satellite embarked multi-spectral electromagnetic sensors. However, different scientific models and applications – at the top of the figure – need these data with specific characteristics. Because of the size and complexity of the data sets, it is undesirable or even unfeasible to transmit all the raw data. Thus, many different pre-processing operations need to be available on the data provider so that the applications can make effective use of the data, as shown in the middle layers of the figure. This scenario is described in more detail in [9].

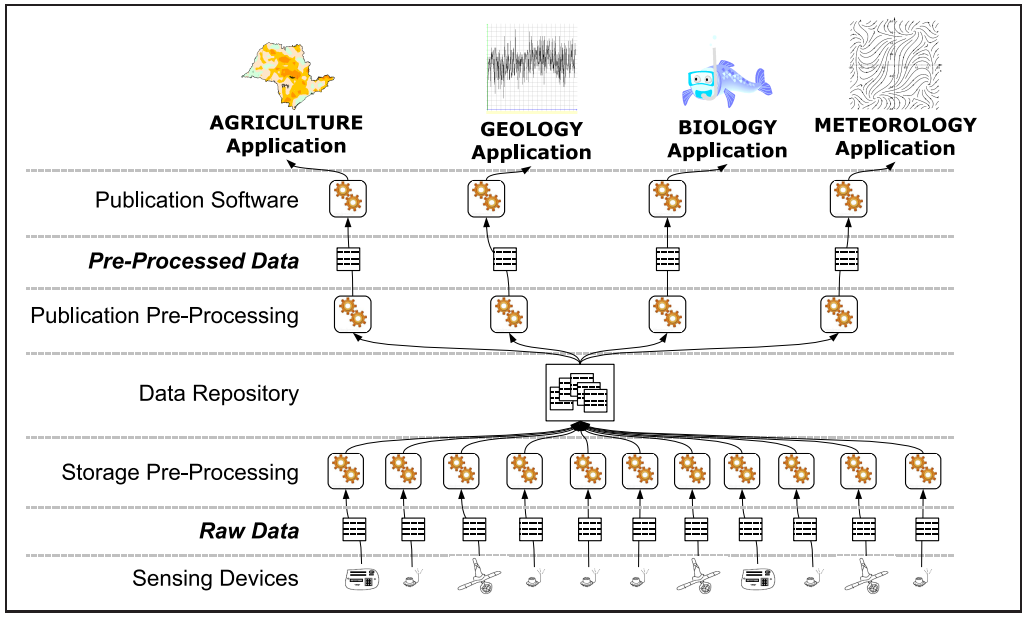


Figura 1: Sensor data usage scenario

The scope of this work encompasses all phases of the sensor data life cycle management, including: (i) acquisition, (ii) pre-processing before storage, (iii) storage mechanisms, (iv) pre-processing of data for publication, (v) publication mechanisms, and (vi) data annotation. The last aspect – annotation – is actually present on all the other phases. Each of these phases involves particular issues and requirements, as will be seen.

Sensors are devices capable of measuring physical phenomena in a given environment, such as heat, light, sound, pressure, magnetism, all subject to signal processing functions. Sensing devices range from satellites (few in quantity, large and expensive), to radio-frequency tags (lots of tiny and cheap units) [4]. In particular, sensor networks lever several new applications and research [3, 11]. Data produced by all these kinds of sensors can be stored as time series, which stay available for later consumption or can be consumed directly by the target applications, skipping the storage phase.

Our proposal is divided in four parts. First, the data acquisition, storage preprocessing and storage mechanisms are treated. Our solution for this part is based on using a component technology called *Digital Content Components* (DCCs) [10]. A DCC is a unity of encapsulation for digital content. It has four parts: (i) the content itself, in its original format (a temperature summarization function on the figure); (ii) a structure for the content, specified in XML; (iii) an interface, describing the operations available in WSDL and OWL-S; and, (iv) a metadata section in OWL, describing the DCC's context information. More details on DCCs are available in [10]. We use DCCs to uniformly encapsulate access to: (a) sensors and other sensor data sources such as databases, (b) the sensor data themselves and (c) the software used to process the data. This allows seamless composition of data, data sources and operations, helping solve interoperability issues. This involves several challenges, such as how to encapsulate data sources with dynamic and/or context-sensitive behavior. Details on this part of the solution are presented in [8].

Second, the management challenges from the pre-processing requirements are addressed by a combination of DCCs to provide homogeneous access to resources and scientific workflows to manage these DCCs. We use scientific workflows to model and manage the operations applied to the sensor data in the pre-processing phases. A *workflow* is a model of a *process*, which is a set of interrelated steps used to achieve a given goal. Each step of a process corresponds to an *activity* in a workflow. A *scientific workflow* [12] is a model of a scientific experiment. Each activity in a scientific workflow is described and executed using the DCCs. Workflows are applied to compose and control the operations available on the DCC interfaces so that the data are tailored to fit applications' needs. Four classes of scientific workflows are used: monitoring (assesses data acquisition), validation (evaluates collected data), management (applies specific operations) and publication (applies general operations). To maintain the seamless integration, scientific workflows can themselves be encapsulated within DCCs, providing more complex executable operations. Details on this part of our solution appear in [7].

The third part, sensor data publication, involves mechanisms for accessing the data, means of describing the data and ways of selecting pre-processing options before actually transmitting the data. We apply both web standards and semantic annotations to deal with publication aspects. Web standards such as the ones proposed by OGC provide means to uniformly describe, publish and access data produced by sensor devices. Semantic annotations provide similar functionality when combi-

ned with software for manipulating the ontologies which are used to describe the operations and the data. More details on this part are in [6], where compatibility between web standards and semantic annotations is also explored.

The fourth and last part of our solution regards data description. We consider two basic approaches to describe data sets: metadata and annotations. Metadata have a well defined structure, and annotations are notes added as comments, or explanations, without any defined structure or value range. The very flexibility of annotations hampers automated processing, whereas metadata are less flexible, but support functions like indexing or searching. We combine both approaches into what we call *semantic annotations*, in which metadata structures and part of their contents are defined by means of references to ontologies tailored to user needs. The latter can be extended as needed, providing contextual information.

Concerning sensor data annotation, we consider the problem of how to keep the annotations valid during the transformation steps which the data goes through. When an operation is performed on a data set, generating a new data set, the annotations on the original data are often not accurate for the new data. After a series of operations, this problem is even more serious. This problem is known as *annotation propagation* [1, 2]. Sensor data annotation (materialised as metadata) is present on all the previous parts. It is common to annotate the data in the storage pre-processing phase. However, in the publication pre-processing phase, often many transformation steps are applied to the data before it is finally made available. This usually renders the previously made annotations inaccurate or even incorrect. In this work we propose a mechanism to modify the annotations along with the modifications made on the data sets, updating the annotations according to the these modifications. The mechanism uses semantic annotations on the operation interfaces and on the data to create new annotations. This is achieved by using operations on ontologies. The solution is very flexible since the mechanism applies to any semantic annotations and ontologies are extensible by nature. Data annotation and annotation propagation are addressed in [5].

The ideas presented to solve the presented sensor data management challenges were tested in small, controlled and/or simulated environments, thus lacking more thorough validation. The next steps of our work involve testing the ideas with real sensor-based systems and including more complex data manipulation operations.

Acknowledgements. This work is supported by FAPESP (grant no. 04/14052-3).

Referências

[1] D. Bhagwat, L. Chiticariu, W. Tan, and G. Vijayvargiya. An annotation management system for relational databases. *The VLDB Journal*, 14(4):373–396, 2005.

[2] S. Bowers and B. Ludäscher. A Calculus for Propagating Semantic Annotations Through Scientific Workflow Queries. In *EDBT Workshops*, pages 712–723, 2006.

[3] D. Culler, D. Estrin, and M. Srivastava. Overview of Sensor Networks. *Computer*, 37(8):41–49, 2004.

[4] J. M. Hellerstein, W. Hong, and S. Madden. The Sensor Spectrum: Technology, Trends and Requirements. *SIGMOD Record*, 32(4):22–27, 2003.

[5] G. Z. Pastorello Jr, J. Daltio, and C. B. Medeiros. Multimedia Semantic Annotation Propagation. In *1st IEEE Int. Workshop on Data Semantics for Multimedia Systems and Applications*, 2008. Accepted for presentation.

[6] G. Z. Pastorello Jr, L. C. Gomes Jr, C. B. Medeiros, and A. Santanchè. Sensor Data Publication on the Web for Scientific Applications. In *Proc. 4th Int. Conf. on Web Information Systems and Technologies*, pages 137–142, 2008.

[7] G. Z. Pastorello Jr, C. B. Medeiros, and A. Santanchè. Applying Scientific Workflows to Manage Sensor Data. In *Proc. 1st e-Science Workshop – 22nd Brazilian Symposium on Databases*, pages 09–18, 2007.

[8] G. Z. Pastorello Jr, C. B. Medeiros, and A. Santanchè. Accessing and Processing Sensing Data. In *Proc. 11th IEEE Int. Conf. on Computational Science and Engineering*, pages 353–360, 2008.

[9] G. Z. Pastorello Jr, R. D. A. Senra, and C. B. Medeiros. Bridging the gap between geospatial resource providers and model developers. In *Proc. 16th ACM SIGSPATIAL Int. Conf. on Advances in Geographic Information Systems*, 2008. Accepted for presentation.

[10] A. Santanchè, C. B. Medeiros, and G. Z. Pastorello Jr. User-centered Multimedia Building Blocks. *Multimedia Systems Journal*, 12(4):403–421, 2007.

[11] R. Szewczyk, E. Osterweil, J. Polastre, M. Hamilton, A. Mainwaring, and D. Estrin. Habitat Monitoring with Sensor Networks. *Communications of the ACM*, 47(6):34–40, 2004.

[12] J. Wainer, M. Weske, G. Vossen, and C. B. Medeiros. Scientific Workflow Systems. In *Proc. of the NSF Workshop on Workflow and Process Automation Information Systems*, 1996.

2. Um Framework para Tratamento de Incertezas e Flutuações em Grades

Autores: Daniel Macêdo Batista e Nelson Luis Saldanha da Fonseca (Orientador)

Em meados dos anos 90, Ian Foster e Carl Kesselman empregaram o termo "grade" como referência para uma nova abordagem que vinha sendo utilizada na solução de problemas computacionais com alta demanda por recursos. Diferente de *clusters* e supercomputadores, que são ambientes confinados em redes locais e voltados para fornecer capacidade de processamento, as grades são ambientes computacionais voltados para fornecer diversos tipos de serviços através da interligação de recursos distribuídos por organizações ao redor do planeta. Com os avanços nas tecnologias de comunicação, a utilização de grades tem crescido e tornado possível o avanço em diversas áreas do conhecimento, levando inclusive à criação do termo "*e-Science*" [8] para agregar todas as áreas que passaram a se beneficiar da infraestrutura computacional de alto desempenho criada. Apesar do crescimento na utilização de grades, vários pontos de pesquisa permanecem em aberto, com destaque para o tratamento das flutuações no estado dos recursos e das incertezas na descrição das aplicações e do estado da grade.

Como os recursos compartilhados nas grades não são dedicados às aplicações, há entrada e saída constante de recursos, além de variações freqüentes na capacidade que cada recurso disponibiliza. A falta de mecanismos que reajam às flutuações através de migrações das aplicações em execução podem levar a um baixo aproveitamento da capacidade disponibilizada pela grade, como demonstrado em [4]. Em se tratando de aplicações formadas por tarefas dependentes deve-se destacar a importância de monitorar a rede da mesma forma que os demais recursos compartilhados pela grade, visto que a correta execução da aplicação depende da transferência confiável dos dados entre as tarefas [3].

Muitos trabalhos na literatura propõem *frameworks* voltados para o tratamento das flutuações e incertezas em grades. Entretanto, esses trabalhos falham por não tratar todas as fontes de incertezas, por ignorar a possibilidade de múltiplas aplicações estarem em execução simultaneamente e por não considerar a importância da rede. Além disso, as métricas e cenários utilizados para comprovar a eficácia das propostas, na maioria das vezes, não é representativo quando se considera os requisitos de grades reais.

Nesta tese propõe-se a construção de um *framework* para a gerência e o autoajuste das grades que lide com as incertezas e com as flutuações, além de buscar um equilíbrio para as várias aplicações em execução. A metodologia adotada para a implementação do *framework* será a de tratar cada ponto em separado mas de forma a permitir a combinação das várias soluções adotadas. Pretende-se propor e integrar soluções que tratem as incertezas no escalonamento de aplicações formadas por tarefas dependentes, que garantam o equilíbrio na alocação de várias aplicações simultâneas na grade frente às flutuações e que estabeleçam *benchmarks* representa-

tivos para a análise de desempenho de *frameworks* que aloquem recursos em grades. A integração das soluções seguirá as recomendações de trabalhos relacionados com a estabilidade e a otimização de ambientes que implementam arquiteturas em camadas, como são as grades [9].

A descoberta das várias falhas dos *frameworks* encontrados na literatura foi alcançada através da comparação de 12 propostas. As propostas foram comparadas levando-se em consideração 8 critérios diretamente relacionados com o suporte que as grades devem implementar para tratar as flutuações e as incertezas. Os resultados alcançados com a comparação realizada foram publicados em [1], onde foca-se nos requisitos que os sistemas de grades devem fornecer para se alcançar grades 100% orientadas a serviço, e em [2], onde foca-se nos mecanismos que devem ser implementados para garantir o correto funcionamento das grades independente dos níveis de incertezas presentes no ambiente.

Para escalonar as aplicações formadas por tarefas dependentes e ao mesmo tempo lidar com as incertezas, propõe-se um escalonador modelado como um problema de programação inteira e que utiliza técnicas de otimização fuzzy. O escalonador recebe como entrada grafos que descrevem a aplicação e a grade, além das estimativas de erros, em termos de porcentagem, com relação às incertezas nas demandas da aplicação e nas capacidades disponíveis dos recursos da grade. O escalonador é implementado de modo a minimizar o tempo de execução da aplicação, objetivo este que está de acordo com as necessidade dos usuários de aplicações de e-Science. Os resultados alcançados até agora com relação ao tratamento de incertezas das aplicações nos escalonadores estão publicados em [5] e [6]. Em [5] avalia-se o escalonador e, a partir de resultados de simulações, mostra-se que o mesmo alcança ganhos de até 26% no *speedup* e consegue fornecer os escalonamentos em um tempo até 32% menor do que sua contra-parte sem suporte a incertezas. No trabalho publicado em [6] o escalonador é estendido para lidar com uma quantidade maior de recursos e de tarefas das aplicações e obtém-se um ganho no tempo de execução de até 85%. Em ambos os trabalhos concluí-se que os ganhos do escalonador são representativos a partir de incertezas acima de 200% nas demandas das aplicações.

Os trabalhos que encontram-se em desenvolvimento envolvem a ampliação do escalonador, de modo que ele tenha suporte às incertezas nas descrições dos recursos, e a proposta de uma solução para encontrar o equilíbrio quando há competição de múltiplas aplicações frente às flutuações das grades. A ampliação do escalonador já foi realizada também através de técnicas de otimização fuzzy. Os resultados preliminares apresentaram ganhos de até 24% e sem um crescimento significativo na complexidade do escalonador frente à versão anterior que só lidava com incertezas nas aplicações. Novos experimentos estão em execução para que se possa tirar conclusões mais genéricas a respeito dos ganhos do escalonador. A modelagem do problema de competição de múltiplas aplicações está sendo realizada utilizando técnicas derivadas de teoria dos jogos. O jogo considera cada aplicação como um jogador que visa agir de forma egoísta no sentido de migrar sempre para o melhor recurso quando há a detecção de variações na capacidade disponível da grade. O

nosso objetivo é derivar o equilíbrio para tal situação e compará-lo com o caso ótimo para diversas configurações de grades e aplicações.

Os próximos passos do nosso trabalho incluem a proposta de benchmarks e a interligação de todas as soluções dentro de um *framework*. Para se avaliar o *framework* proposto serão coletadas informações de grades reais que mantém *traces* das aplicações e do estado dos recursos em diversos instantes de tempo. De posse dessas informações será realizado um estudo estatístico a fim de se construir aplicações e cenários de simulação sintéticos que representem ambientes com características próximas às encontradas em grades reais. Já a interligação dos vários mecanismos propostos utilizará resultados recentes na área de teoria de decomposição de otimização [7]. O objetivo será encontrar as localizações ideais para implementar cada um dos mecanismos do *framework* de modo a garantir que o *overhead* gerado não invalidará a solução como um todo.

O trabalho resumido nesta seção tem o objetivo de apresentar soluções que mantenham os serviços fornecidos pelas grades estáveis independente das flutuações e das incertezas presentes no ambiente durante todo o tempo de vida das aplicações. Propõe-se a construção de um *framework* que gerenciará os mecanismos independente da quantidade de aplicações que estejam em execução simultânea nas grades. Os resultados apresentados nos trabalhos publicados têm comprovado a eficácia em utilizar os mecanismos propostos.

Referências

[1] Daniel M. Batista and Nelson L. S. da Fonseca. A Brief Survey on Resource Allocation in Service Oriented Grids. In *Globecom Workshops – 1st IEEE Workshop on Enabling the Future Service-Oriented Internet*, pages 1–5. IEEE, 2007.

[2] Daniel M. Batista and Nelson L. S. da Fonseca. Empowering Grids with Flexibility to Cope with Uncertainties. In *ICC Workshops – 13th IEEE International Workshop on Computer-Aided Modeling, Analysis and Design of Communication Links and Networks (CAMAD 2008)*, pages 227–231. IEEE, 2008.

[3] Daniel M. Batista, Nelson L. S. da Fonseca, and Flavio K. Miyazawa. A Set of Schedulers for Grid Networks. In *SAC '07: Proceedings of the 2007 ACM symposium on Applied computing*, pages 209–213, New York, NY, USA, 2007. ACM.

[4] Daniel M. Batista, Nelson L. S. da Fonseca, Flavio K. Miyazawa, and Fabrizio Granelli. Self-Adjustment of Resource Allocation for Grid Applications. *Computer Networks*, 52(9):1762–1781, 2008.

[5] Daniel M. Batista, André C. Drummond, and Nelson L. S. da Fonseca. Scheduling Grid Tasks under Uncertain Demands. In *SAC '08: Proceedings of the*

2008 ACM Symposium on Applied Computing, pages 2041–2045, New York, NY, USA, 2008. ACM.

[6] Daniel M. Batista, André C. Drummond, and Nelson L. S. da Fonseca. Um Escalonador de Tarefas Dependentes Robusto às Incertezas nas Descrições das Aplicações em Grades. In *VII Workshop em Desempenho de Sistemas Computacionais (WPerformance)*, pages 161–179. SBC, 2008.

[7] Mung Chiang, Steven H. Low, A. Robert Calderbank, and John C. Doyle. Layering as Optimization Decomposition: A Mathematical Theory of Network Architectures. *Proceedings of the IEEE*, 95(1):255–312, Jan 2007.

[8] 4th IEEE International Conference on e-Science, 2008. `http://escience2008.iu.edu/`. Último acesso em 12 set 2008.

[9] I. Foster, C. Kesselman, and S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal Supercomputer Applications*, 15(3):200–222, 2001.

3. Balanceamento de Carga para Serviços Web Altamente Disponíveis

Autores: Alan Massaru Nakai, Edmundo Madeira, Luiz Eduardo Buzato

Com a proliferação de companhias que realizam porções significativas de seus negócios *online*, cresce o interesse por alta disponibilidade de serviços Web. A indisponibilidade dos serviços pode gerar descontentamento de clientes, influenciando negativamente na imagem da companhia e, conseqüentemente, gerando perdas de receita. Além disso, a falta de disponibilidade do sistema em aplicações críticas, como aplicações militares e da medicina, pode ter conseqüências catastróficas, levando risco ao bem estar humano.

Muitas razões podem levar à indisponibilidade de um serviço Web, como: falha ou sobrecarga do servidor, problemas de roteamento e congestionamento de vias de comunicação. Uma solução comum para o aumento da disponibilidade de um serviço Web é a adição de redundância ao sistema, via replicação dos servidores. Esta solução aumenta a probabilidade de que um cliente possa conectar-se a um servidor mesmo na presença de falhas parciais. Além disso, a replicação potencializa a escalabilidade do serviço Web, diminuindo a chance de sobrecargas. O sucesso da replicação depende do êxito obtido na solução de dois sub-problemas [1]: (i) a gerência da consistência dos dados replicados entre os múltiplos servidores [2, 3] e (ii) a distribuição da carga de trabalho entre eles.

Este trabalho enfoca o problema da distribuição de carga para servidores Web distribuídos, tratando o problema da gerência da consistência com soluções já existentes. O problema da distribuição de carga consiste em prover a ligação dinâmica entre o cliente e um dos nós replicados, visando minimizar a carga máxima de um nó servidor, dentre todos os nós, em um dado instante. A atribuição de um nó servidor a um cliente pode levar em consideração informação dinâmica de ambas as partes, por exemplo: capacidade dos servidores, carga dos servidores, proveniência das requisições, conteúdo das requisições, etc.

De forma geral, as técnicas para balanceamento de carga para servidores Web replicados encontradas na literatura podem ser divididas em duas categorias: técnicas para a servidores Web geograficamente distribuídos (ex. [4–6]) e técnicas para a servidores Web replicados e hospedados em aglomerados (ex. [7]). O objetivo deste trabalho é especificar um mecanismo para o balanceamento de carga para servidores Web compostos por múltiplos aglomerados distribuídos geograficamente, visando manter alta disponibilidade dos serviços providos. A metodologia para o desenvolvimento do trabalho inclui quatro grandes tarefas: (i) análise das técnicas existentes e da possibilidade de sua integração, (ii) desenvolvimento de uma arquitetura de software para implantação do mecanismo proposto e (iii) avaliação experimental do mecanismo.

Para realização da primeira tarefa, está sendo desenvolvida uma infra-estrutura que permite que diferentes políticas de balanceamento de carga sejam testadas e

comparadas. A Figura 2 mostra uma visão geral desta infra-estrutura.

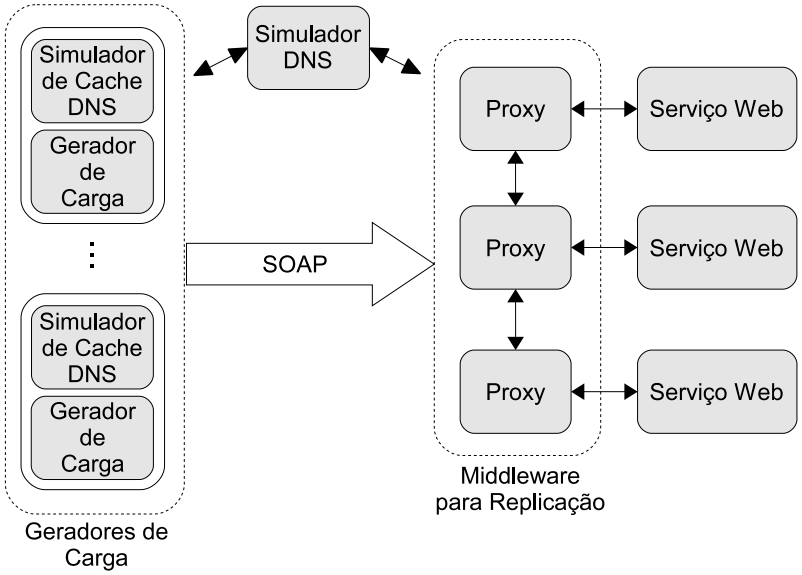


Figura 2: Infra-estrutura de testes

Seus principais componentes são:

- **Simulador DNS:** Permite testar técnicas de balanceamento de carga baseadas em DNS. Nestas técnicas, os clientes do serviço enviam requisições para o servidor DNS que retorna o endereço IP do servidor Web adequado, de acordo com uma política de balanceamento de carga.

- **Simulador de Cache DNS:** Simula uma *cache* do serviço de nomes, permitindo a configuração da probabilidade com que uma requisição ao DNS é atendida localmente ou encaminhada para o simulador DNS. O simulador de *cache* DNS permite testar a sensibilidade das técnicas baseadas em DNS ao sistema de *cache*.

- ***Middleware* para Replicação de Serviços:** Permite a replicação de serviços Web por meio de ordenação total. O *middleware* realiza o multicast totalmente ordenado das operações de escrita, mantendo as réplicas do serviço mutualmente consistentes. As operações de leitura são distribuídas entre as réplicas de acordo com a política de balanceamento de carga que está sendo testada. Além disso, o *middleware* permite a aplicação de técnicas de balanceamento de carga baseadas em redirecionamento, nas quais servidores sobrecarregados redirecionam requisições de clientes para outros servidores.

A aplicação utilizada para os testes é baseada em uma implementação do *benchmark* TPC-W[1] desenvolvida na universidade de Wisconsin[2]. O TPC-W é um

[1]http://www.tpc.org/tpcw/default.asp
[2]http://mitglied.lycos.de/jankiefer/tpcw/index.html

benchmark transacional que simula a carga de trabalho de um site de comércio eletrônico. Na infra-estrutura de testes, a implementação do TPC-W foi encapsulada como serviços Web utilizando-se o *framework* Axis2[3].

Até o momento a infra-estrutura de testes, que ainda está em desenvolvimento, tem sido testada em uma rede local. No futuro, pretende-se implantá-la na Internet, utilizando aglomerados no Brasil (Unicamp), Suíça (EPFL - Ecole Polytechnique Fédérale de Lausanne - Suíça) e Estados Unidos (Emulab[4]), além de nós geradores de carga da rede PlanetLab[5], que possui mais de 800 nós espalhados pelo mundo. Desta forma, será possível realizar os testes em condições reais da Internet.

Referências

[1] David B. Ingham, Santosh K. Shrivastava, and Fabio Panzieri. Constructing dependable web services. *IEEE Internet Computing*, 4(1):25–33, 2000.

[2] Jim Gray, Pat Helland, Patrick O'Neil, and Dennis Shasha. The dangers of replication and a solution. pages 372–381, 1998.

[3] Rachid Guerraoui and Andre Schiper. Software-based replication for fault tolerance. *Computer*, 30(4):68–74, 1997.

[4] M. Colajanni and P. S. Yu. A performance study of robust load sharing strategies for distributed heterogeneous web server systems. *IEEE Transactions on Knowledge and Data Engineering*, 14(2):398–414, 2002.

[5] Devarshi Chatterjee, Zahir Tari, and Albert Y. Zomaya. A task-based adaptive ttl approach for web server load balancing. In *Proceedings. 10th IEEE Symposium on Computers and Communications, 2005. ISCC 2005.*, pages 877–884. IEEE Computer Society, 2005.

[6] Liming Liu and Yumao Lu. Dynamic traffic controls for web-server networks. *Comput. Networks*, 45(4):523–536, 2004.

[7] Emiliano Casalicchio, Valeria Cardellini, and Michele Colajanni. Content-aware dispatching algorithms for cluster-based web servers. *Cluster Computing*, 5(1):65–74, 2002.

[3] http://ws.apache.org/axis2/

[4] www.emulab.net

[5] www.planet-lab.org

4. Especificação de uma abordagem sistemática para gerenciar e implementar Linhas de Produtos Dinâmicas e baseadas em serviços Web

Autores: Amanda S. Nascimento e Silva, Cecília M.F. Rubira (Orientadora)

Nos dias atuais, a atividade de desenvolvimento de sistemas computacionais enfrenta vários desafios. Há um consenso, na comunidade de Engenharia de Software, que todo sistema de software evolui [1]. Logo, se por um lado há a demanda de software de qualidade que atenda as necessidades individuais, por outro, existe a necessidade crítica de redução de custo, esforço e tempo de chegada ao mercado dos produtos de software. Neste sentido, nas últimas décadas foram criadas novas estratégias de reutilização de software que possibilitam o uso de artefatos já existentes na construção de um novo sistema. Exemplos atuais dessas tecnologias são Linhas de Produtos de Software e Computação Orientada a Serviços.

Conforme definição de Clements e Northrop [2], uma Linha de Produto de Software (LPS) é planejada para produzir um conjunto de produtos de software com alto grau de similaridade entre si, que atendam às necessidades específicas de uma missão ou segmento de mercado, e que são desenvolvidas de forma prescritiva a partir de um conjunto de artefatos básicos, chamados ativos centrais ou do núcleo. Os ativos centrais tem que lidar, de maneira sistemática, com as diferenças e semelhanças dos produtos, respectivamente, de maneira informal, variabilidades[6] e comunalidades [7], e customizadas de acordo com as necessidades dos produtos individuais ao serem instanciados [4] [5] [6] [7].

Serviços Web é considerada a tecnologia mais apropriada para a implementação da Arquitetura Orientada a Serviços [8]. Serviços Web são elementos de software, de baixo acoplamento, que, ao implementar Arquitetura Orientada a Serviços, podem ser descobertos, invocados e utilizados através de protocolos padrões da Internet e facilmente integrados para comporem processos de negócios [8] executáveis em sistemas distribuídos e e-business [8]. Em Serviços Web complexos, que resultam da composição de outros Serviços Web, várias operações são especificadas por meio de processos de negócios executáveis, que, por sua vez, agregam outras operações de serviços Web [9]. Ainda, Canfora [10] discute a composição e modificação dinâmicas de serviços como essenciais a evolução de software.

Alguns autores propõem a aplicação da engenharia de Linha de Produto para Serviços Web [4] [11] [12] [5], o que permite benefícios na evolução e desenvolvimento de sistemas orientados a serviços; além de um melhor desempenho da LPS, visto que o baixo acoplamento dos serviços Web favorece a reutilização e conseqüente construção de processos de negócios relacionados. Também, Chang et al. [13] alegam que a LPS tem um grande potencial para modelar variabilidades e comunalidades em

[6]Do inglês, variability

[7]Do inglês, comunalities

[8]Do inglês, Business Process

serviços Web. Contudo, os trabalhos existentes, referentes às Linhas de Produtos orientadas a serviços [4] [12] [14] [15], são muito difíceis de serem aplicados em ambientes industriais, principalmente pelo não detalhamento relativo à execução das fases de projeto e implementação.

Além disso, computação moderna e ambientes de rede, baseados em serviços Web, demandam de um alto grau de adaptabilidade dos sistemas de software. O ambiente do sistema, as necessidades dos usuários e os mecanismos de interfaces entre o software, dispositivos de hardware, como sensores, podem mudar dinamicamente, em tempo de execução. Contudo, nestes ambientes dinâmicos, nos quais o sistema deve se auto-adaptar para acomodar a evolução e mudanças de requisitos em tempo de execução, a estrutura da LPS deve ser alterada de uma perspectiva estática para uma perspectiva dinâmica, em que sistemas sejam capazes de modificar seu próprio comportamento, inclusive, a partir de tomadas de decisões e customização de produtos para transformar um membro de uma linha em outro membro da linha, ambos em tempo de execução. Esta é a idéia da Linha de Produto de Software Dinâmica (LPSD), uma área de pesquisa emergente, bastante promissora e pouco explorada por pesquisadores [4] [15]. Poucos trabalhos consideram, de forma sistemática, uma perspectiva dinâmica da Linha de Produto, inclusive, no tocante ao gerenciamento de variabilidades de software da linha. Há uma carência de abordagens que, efetivamente, tratem das variabilidades em dois níveis: um estático, em que produtos possam ser diferenciados e outro dinâmico, capaz de diferenciar estado de um mesmo produto ou, eventualmente, diferenciar produtos.

Este projeto de pesquisa visa contribuir com o estado da arte de LPS e composição dinâmica de serviços Web, ao propor uma abordagem sistemática para gerenciar e implementar uma Linha de Produto de Software Dinâmica e composta por serviços Web. Ressalta-se que a linha de produto sob uma perspectiva dinâmica, permitirá, também, maior dinamismo na composição de serviços, fator essencial em ambientes de processos de negócios, uma vez que adaptações em tempo de execução fazem-se necessárias, por exemplo, em caso de situações inesperadas ou falhas, como indisponibilidade de um Serviço Web, variações nas propriedades de qualidades do serviço, mudanças nas regras ou condições de composição de serviços. Para a abordagem, destacam-se os seguintes objetivos:

- Propor um modelo para identificar, representar e classificar variabilidades e comunalidades em serviços Web. A abordagem deverá contemplar, também, a identificação, representação e implementação de variabilidades de forma que tomadas de decisão e composição de serviços sejam adaptadas e realizadas em tempo de execução. A estas variabilidades, cujas tomadas de decisão serão realizadas dinamicamente, dá-se o nome de variabilidades dinâmicas [3].

- Extensão de métodos de construção e mapeamento dos artefatos ao longo do processo de desenvolvimento, com a finalidade de mapear pontos de variações entre a fase de projeto até a implementação e execução de produtos individuais.

- Uma vez que pontos de variação e variantes tenham sido devidamente identificados e representados, serão consideradas visões de diferentes perspectivas da Linha de Produtos, através de XML e modelos gráficos, para facilitar a instanciação e customização de processos de negócios individuais, inclusive, em tempo de execução. Este modelo, provavelmente, será uma extensão de linguagens de composição de serviços Web já conhecidas [9] como Business Process Execution Language (BPEL).

- Pretende-se utilizar, também, Programação Orientada a Aspectos para prover maior adaptabilidade, reusabilidade, melhor desenvolvimento incremental e maior capacidade de evolução ao arcabouço proposto, visto que variabilidades podem ser encapsuladas em unidades modulares, chamadas aspectos, e inseridas no restante do sistema em tempo de compilação, carregamento ou execução. Uma alternativa seria considerar a extensão orientada a aspectos de BPEL de forma a separar explicitamente as variabilidades, implementadas como aspectos, das partes semelhantes.

Visto que esta proposta de projeto encontra-se nas fases iniciais de estudos, está a ser feito um levantamento bibliográfico a fim de definir os métodos de avaliação que serão utilizados e métricas que podem ser aplicadas para avaliar o arcabouço proposto. A princípio, sabe-se que será considerada uma avaliação qualitativa da solução através de um estudo de casos representativo de sistemas reais. Será definida uma LPS baseda em serviços Web e produtos serão instanciados, novos produtos e funcionalidades serão inseridos, funcionalidades serão alteradas e mudanças no ambiente serão simuladas a fim de avaliar como a solução proposta comparta-se em tempo de execução. Serão analisados os seguintes atributos: disponabilidade, reusabilidade e adaptabilidade.

Referências

[1] Lehman, M. M., Ramil, J. F., and Perry, D. E. On Evidence Supporting the FEAST Hypothesis and the Laws of Software Evolution. In *PProceedings of the 5th international Symposium on Software Metrics (March 20 - 21, 1998).* METRICS. IEEE Computer Society, Washington, DC, 84, 1998.

[2] Clements P., Northrop L.. Software product lines: Practices and patterns. *Addison-Wesley.* 2002.

[3] Software Engineering Institute (SEI), Carnegie Mellon University, Software Product Lines. In *http:Hsei.cmu.edu/productlines/index.html* 2008.

[4] Capilla, R. and Topaloglu, N. Y. Product Lines for Supporting the Composition and Evolution of Service Oriented Applications. In *Proceedings of the Eighth international Workshop on Principles of Software Evolution. IWPSE. IEEE Computer Society, Washington, DC, 53-56* 2005.

[5] H. Gomaa. Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures. Addison-Wesley. 2004.

[6] Sinnema, M.;Deelstra, S; Nijhuis, J; Bosch, J. COVAMOF: A Framework for Modeling Variability in Software Product Families. In *Software Product Lines, Third International Conference, SPLC 2004, 3154 pp 197.*

[7] Software Engineering Institute. The Product Line Practice (PLP) Initiative , Carnegie Mellon University. *www.sei.cmu.edu/activities/plp/plp_init.html.*

[8] Chung, J.-Y., Lin, K.-J., Mathieu, R. Web services computing: Advancing software interoperability. In *IEEE Computer Special Issue on Web Services Computing, October, pp. 35-37.* 2003.

[9] Charfi, A.; Schmeling, B.; Heizenreder, A.; and Mezini M.. Reliable Secure and Transacted Web Service Compositions with AO4BPEL. In *Proceedings of The 2nd International Conference on Service Oriented Computing (ICSOC'2004), New- York, USA.* 2004.

[10] Canfora, G. Software Evolution in the Era of Software Services. *In Proceedings of the Principles of Software Evolution, 7th international Workshop (September 06 - 07, 2004). IWPSE. IEEE Computer Society, Washington.* 2004.

[11] D. Muthig, I. John, M. Anastasopoulos, T. Forster, J. Doerr, and K. Schmid. Gophone - a software product line in the mobile phone domain. *Technical report, Fraunhofer IESE.* 2004.

[12] Eunsuk Ye, Mikyeong Moon, Youngbong Kim, Keunhyuk Yeom. An Approach to Designing Service-Oriented Product-Line Architecture for Business Process Families. *Advanced Communication Technology, The 9th International Conference on , vol.2, no., pp.999-1002, 12-14 Feb..* 2007.

[13] Chang, S. H. and Kim, S. D.. A Variability Modeling Method for Adaptable Services in Service-Oriented Computing. *Proceedings of the 11th international Software Product Line Conference (September 10 - 14, 2007). International Conference on Software Product Line. IEEE Computer Society, Washington.* 2007.

[14] Gomaa, H. and Saleh, M.. Software product line engineering for Web services and UML. *Proceedings of the ACS/IEEE 2005 international Conference on Computer Systems and Applications. AICCSA. IEEE Computer Society, Washington.* 2005

[15] Herzwurm G. Jesse S. Mikusz M. Helferich. A.. Software product lines, service-oriented architecture and frameworks: Worlds apart or ideal partners? *D. Draheim and G. Weber (Eds.): TEAA 2006, LNCS 4473, pp. 187-201, pages 187-201.* 2007

5. Desenvolvimento de Técnica de Teste Passivo com Base em Algoritmos de Bioinformática

Autores: Gizelle Sandrini Lemos e Eliane Martins (Orientadora)

Introdução

Podem-se utilizar técnicas de testes ativos ou passivos para realizar a validação de um software. No teste ativo, o testador envia uma entrada durante a execução do software e espera por uma saída. Se a saída fornecida pelo software pertencer ao conjunto de saídas esperadas, então o processo continua; senão é detectada a falha no software. Esse tipo de teste é dito ativo pois o testador tem total controle sobre as entradas fornecidas.

A realização de testes ativos é, muitas vezes, prejudicada pela impossibilidade do testador interferir no software a ser testado ou por limitações no espaço de memória e armazenamento do dispositivo no qual se encontra o software. Tais situações podem resultar em softwares não suficientemente testados.

Uma técnica mais recente, denominada de teste passivo, pode ser utilizada nesses casos, pois não altera o funcionamento normal do software e não utiliza memória adicional. Nessa técnica, o comportamento do software, durante a execução, é observado pelo analisador sem que haja interrupções. O analisador capta traços de execução do software e os compara com o comportamento esperado - obtido com base no modelo formal de especificação [1] [2] [6].

Este artigo trata do desenvolvimento de uma técnica de teste passivo a partir de conceitos presentes em abordagens existentes e da adaptação de algoritmos de bioinformática comumente utilizados para o alinhamento de seqüências biológicas.

Contexto

Características dos Testes Passivos

Testes passivos são úteis tanto para analisar se há conformidade do software com relação aos requisitos funcionais quanto aos não-funcionais, como requisitos de segurança, por exemplo. Além disso, têm em geral, um custo de execução menor do que testes ativos.

Testes passivos podem ser realizados quando é impossível aplicar testes ativos. Situações em que o ambiente de produção no qual o software precisa ser testado apresenta restrições de memória e capacidade de armazenamento - como alguns sistemas embarcados - são exemplos. Outra situação favorável à aplicação de testes passivos ocorre quando não é possível que o analisador interfira no software em execução como, por exemplo, no gerenciamento de redes, em que o processo de teste precisa ser realizado também durante a operação normal da rede [4].

Abordagens de Testes Passivos

O conceito de testes passivos foi introduzido por Charles Seitz em 1972 [7] apud [4], mas sua aplicação foi iniciada mais de vinte anos depois. A primeira abordagem de teste passivo consiste da modelagem do comportamento esperado pelo software em forma de FSM*(Finite State Machine)* e na captação do traço de execução [4]. As entradas/saídas presentes no traço são comparadas com as entradas/saídas aceitas pela máquina de especificação, processo realizado em duas etapas:

1. Localização - na máquina de especificação - do estado correspondente ao início do traço;

2. Comparação dos pares entrada/saída do traço com os estados subseqüentes.

Durante a comparação, falhas são detectadas caso sejam encontradas diferenças entre o traço de execução e a máquina de especificação. Porém, utilizando-se essa abordagem não é possível localizar a falha que ocasionou o erro.

Na tentaiva de tentar encontrar mais falhas e facilitar sua localização, foram criadas extensões da primeira abordagem utilizando outras formas de modelagem formal mais abrangentes como CFSM*(Comunicating FSM)* [6] e EFSM*(Extended FSM)* [3]. O uso de CFSM para modelagem da especificação do software foi aplicado a testes passivos em protocolos de rede e, pelo fato de que esse tipo de modelagem permite especificar os canais de comunicação, foi possível localizar em que parte da rede ocorreram erros. Já, o uso de EFSM permite o acompanhamento dos valores de variáveis externas do software. Com essa informação adicional foi possível detectar os erros e localizar as falhas de forma mais eficiente.

Posteriormente, foi desenvolvida uma nova abordagem que procura no traço de execução a presença de invariantes [1] [5]. Invariantes são propriedades, previamente definidas e validadas com base na especificação, requeridas para uma correta implementação. As etapas dessa abordagem são as seguintes:

1. Definição e validação dos invariantes, com base na máquina de especificação;

2. Obtenção e filtragem do traço de execução. O traço é filtrado para que permaneçam nele somente os dados relevantes ao teste;

3. Comparação do traço de execução com os invariantes, usando algoritmos de *pattern matching*.

Objetivos

Este trabalho tem como objetivo principal a elaboração de uma técnica de teste passivo baseada na abordagem de invariantes que utilizará algoritmos de bioinformática - usados comumente no alinhamento de seqüências biológicas - para alinhar os invariantes e o traço de execução. Espera-se que o uso desse tipo de algoritmo proporcione um melhor resultado no alinhamento das seqüências envolvidas no teste sem prejuízo

em termos de complexidade. Como resultado subseqüentes será implementada uma ferramenta para automatizar a aplicação da técnica.

Metodologia de Desenvolvimento

A metodologia a ser empregada neste trabalho abrange o estudo das áreas de interesse mencionadas, em geral a engenharia de software e, em específico, os testes passivos. Além disso, estão sendo estudados os algoritmos de alinhamento de seqüências biológicas para o desenvolvimento do algoritmo da nova técnica de teste passivo. Pretende-se analisar os aspectos de complexidade e recursos envolvidos em cada um dos algoritmos de bioinformática a serem estudados para que sejam selecionados os mais apropriados ao alinhamento das seqüências de teste.

Serão utilizadas EFSMs para modelar a especificação do software e derivar os invariantes na forma de expressões regulares. O estudo dos diversos métodos e algoritmos existentes para o alinhamento de seqüências biológicas possibilitará sua adequação e posterior uso com as seqüências utilizadas nos testes passivos. Caso, durante o alinhamento, sejam identificados pontos de discrepância entre o traço e os invariantes será possível identificar e localizar falhas no software. A validação da técnica e a análise dos resultados ocorrerão com a realização de estudos de caso.

Resultados Esperados

Pretende-se, com o trabalho proposto, desenvolver uma técnica de teste passivo, que utilizará algoritmos derivados dos utilizados em bioinformática no alinhamento de seqüências biológicas, com o objetivo de verificar se o software possui as propriedades desejadas, ou seja, se o software funciona conforme sua especificação. Além disso, espera-se implementar uma ferramenta para automatizar a realização da técnica desenvolvida e incorporá-la às ferramentas de código aberto existentes.

Conclusões

O conceito deteste passivo é relativamente recente e apresenta vários aspectos para estudo e desenvolvimento. A técnica pode ser aplicada em cenários desfavoráveis ao uso de testes ativos nos quais é impossível controlar a execução do software ou quando o espaço de memória e/ou armazenamento é insuficiente para abrigar os dados de teste, pode-se utilizá-la também como complemento para os testes ativos.

A técnica de teste passivo que será desenvolvida neste trabalho poderá ser utilizada em testes de redes, *web services* e sistemas embarcados. Pretende-se difundir o uso da técnica a outros tipos de software e ainda, facilitar sua aplicação por meio da ferramenta que será implementada.

Referências

[1] A. Arnedo, A. Cavalli, and M. Nunez. Fast Testing of Critical Properties through Passive Testing. *Lecture Notes in Computer Science*, Volume 2644, Pages 295–310, 2003.

[2] A. Cavalli, and D. Vieira. An Enhanced Passive Testing Approach for Network Protocols. *IEEE International Conference on Networks*, Pages 169–174, Singapore, September 2006.

[3] M. Tabourier, A. Cavalli,and M. Ionescu. Passive Testing and Application to the GSM-MAP Protocol. *Information and Software Technology*, Volume 41, Issue 11–12, Pages 813–821, September 1999.

[4] D. Lee, A. Netravali, K. Sabnani, B. Sugla, and A. John. Passive Testing and Applications to Network Management. *IEEE International Conference on Network Protocols*, Pages 113–122, October 1997.

[5] E. Bayse, A. Cavalli, M. Nuñez, and F. Zaïdi. A Passive Testing Approach based on Invariants: Application to the WAP. *Journal of Computer Networks*, Volume 48, Issue 2, Pages 247–266, June 2005.

[6] R. Miller. Passive Testing of Networks Using a CFSM Specification. *IEEE International Performance Computing and Communications Conference*, Pages: 111–116. February 1998.

[7] C. Seitz. An Approach to Designing Checking Experiments Based on a Dynamic Model. *Theory of Machines and Computations*, Z. Kohavi and A. Paz Ed., Academic Press, 1972.

6. Multimodal Complex Objects

Authors: Nádia Puchalski Kozievitch, Ricardo da Silva Torres

Introduction

In many digital-based applications involving documents, like research, teaching and diagnosis evolving image, video, catalogs, models, temporal series and geo-referenced data there is a need for preservation, archiving, indexing, and integration of multimedia information. The multiple kinds of information contributed to the concept of Digital Libraries (DL): organized collections of digital information, offering services like search, browsing, recommendation among others. These capabilities is being extended to select and annotate multimedia, link contents, organize information, share and revise information, addressing the concept of Superimposed Information (SI : information layer that is created over information [5]). From the computational view, a DL is composed of simple components named Digital Objects (DO). Theses objects can be aggregated in multiple layers of data, metadata and services, resulting in multimodal complex objects.

At this work we present an overview of existing related work, reference models, specification approaches and open research challenges regarding the definition of an unified framework to manage multimodal complex objects in digital libraries.

Related Work

Digital Libraries and Superimposed Information

In [8], it was presented an OAI-based generic digital library architecture for integrated management of image descriptors and textual information, dealing with fish specimen identification. In [3], it was presented SIERRA, an application that combined text-based and content-based image retrieval in digital libraries, allowing users to link together image content with related data like annotations. To achieve this, the concept of Superimposed Information (SI) was used.

SI refers to a new information laid over information [5]. In this scenario, a user may work with several sources, including books, papers, images, descriptions, audio where only selected and relevant information is extracted. Its second layer includes additional data created to reference, highlight and present information.

Complex Objects

The integration of multiple kinds of data results in complex objects (COs). COs are single entities that are composed of multiple parts, each of which is an entity in and of itself. As different metadata format are created, as XML (Extensible Markup Language) and RDF (Resource Description Format), new patterns arise, as NetCDF, HDF, and ELFS in scientific computing. In industry, CO can be found

in persistent database storage and multimedia codecs. From this evolution emerged other technologies, as multimedia framework MPEG-21 (Moving Picture Experts Group - 21), and digital object formats as MPEG-21 DIDL (Moving Picture Experts Group - 21 Digital Item Declaration Language) and METS (Metadata Encoding and Transmission Standard). Even though there are different standarts for CO, there is still incompatibility, motivating solutions for integration and interoperability. As each pattern is specialized in a specific domain, their differences and their formats hampers interoperability. On the other hand, it is still possible to match them, as proposed on [7], in their comparative study of IMS CP (IMS Content Package) and RAS (Reusable Asset Specification).

Data Models and Standarts

Modeling, defining, understanding and describing DLs is still a challenge, as shown in [6], [1] and [2]. [4] proposed and formalized the 5S [Streams, Structures, Spaces, Scenarios, and Societies] framework, a formal base to capture the complexity of DLs. These 5Ss, along with fundamental set theoretic definitions, are used to define other DL constructs such as digital objects, metadata specification, collection, repository and services. [6] extended the 5S framework to provide specifications, moving towards a minimum DL reference model. The minimum DL is presented as a foundation of various extensions, serving as a base for a DL reference model. In the first extension there is CBIR (Content-Based Information Retrieval) service, the second one is a meta-model for SI and finally the third extension deals with the DL generation based on the DL software, such as Dspace.

The Delos approach [2] uses a framework of three tiers to represent three levels of abstraction: the DL (Digital Library), the DLS (Digital Library System) and DLMS (Digital Library Management System). The first layer is where the digital objects are kept; the second one manages the software application components, providing useful services to the interested users with the support of a DLMS. Here the three levels are hierarchically related: so are their models, i.e., the DL model is included in the DLS one, and later is included in the DLMS model.

An Application Example

Suppose a PHD student inserts a theses into the digital library of an institution. The theses has 12 chapters; a collection of metadata like title, author, area, year; a powerpoint presentation regarding chapter one; a recorded video about chapter two and some annotations about chapter twelve, as shown in Figure 1.

These different digital objects will be referenced as one complex object in an institution digital library and it is possible to create references, links and annotations between other complex objects. The institution digital library offers services, like browsing, searching and recommendation. And year after year, more students send their documents with new formats to the system, requiring an evolution of the

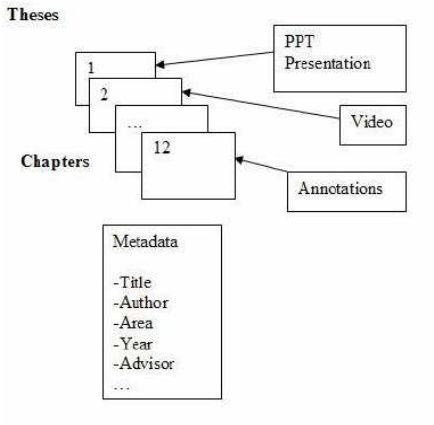


Figura 3: Components of a PHD theses in a digital library

database schema and connection graph.

Further Challenges

All mentioned models do not focus on complex objects, not being suitable for characterizing their multimodal behavior, new formats and standarts, data models, architecture, management and evolution. Our contribution is analyze this scenario through the CO perspective, looking for a unified framework to manage multimodal complex objects in digital libraries and verifying the formal characterization of complex object definition/integration/evolution/architecture and its services.

The main challenges verified through our research are: 1) a formal definition of the connection graph between complex objects and its evolution/services; 2) a formal definition of an unified framework for managing complex objects in digital libraries; 3) evaluation/management of the schema/system/components evolution (new format types can be created, resulting in new collections, catalogs, and services in an existing schema); 4) evaluation/extraction of similar characteristics from different components for further aggregation; comparison of two different COs (different format, different constraints, match queries that evolve image, video and annotation); 5) presentation of a generic data model that includes different COs and finally the integration of this CO reference model with existing DL experiences.

Conclusion

We presented existing related work, reference models, specification approaches and open research challenges regarding the definition of an unified framework to manage multimodal complex objects in digital libraries. Our research contribution will continue analyzing this scenario through complex object perspective verifying the formal characterization of definition, integration and evolution of complex objects and services.

Referências

[1] Agosti, M., Ferro, N., Fox, E.A., Gonçalves, M.A., Lagoeiro, B. *Towards a Reference quality model for digital libraries*. First International Workshop on Digital Library Foundations,2007.

[2] Candela, L., Castelli, D., Ioannidis, Y., Koutrika, G., Pagano, P., Ross, S., Schek, H.J. and Schuldt, H. *A reference model for Digital Library Management Systems Interim Report*. Delos deliverable, n. 1.4.2, 2006.

[3] Fox, E.A., Murthy, U., Torres, R. Da S. *Sierra - A Superimposed Application for Enhanced Image Description and Retrieval*. European Conference on Digital Libraries, Alicante, Spain,P. 540-543, 2006.

[4] Gonçalves, M.A., Fox, E.A., Watson, L.T. And Kipp, N.A. *Streams, structures, spaces, scenarios, societies (5S): A formal model for digital libraries*. ACM TOIS, 22 (2). 270-312, 2004.

[5] Murthy, U. *A Superimposed Information-Supported Digital Library*. Doctoral Consortium at the 2007 Joint Conference on Digital Libraries.

[6] Murthy, U., Gorton, D., Torres, R., Gonçalves, M., Fox, E.A. Ad Delcambre, L. *Extending the 5S Digital Library (DL) Framework: from a minimal DL towards a DL Reference Model*. ACM IEEE Joint Conference on Digital Libraries, 2007.

[7] Santanchè, A., Dourado, P. and Ferreira, P. Representação Unificada de Objetos Digitais Complexos: Confrontando o RAS com o IMS CP *III Workshop de Bibliotecas Digitais, 2006*

[8] Torres, R. Da S., Medeiros, C. B., Gonçalves, M.A., Fox, E.A. A Digital Library Framework for Biodiversity Information Systems *International Journal on Digital Libraries, Vol 6, P. 3-17 , 2006*

7. Usando Banco de Dados Espacial em Consultas Geográficas na Web

Autores: Lin Tzy Li e Ricardo da Silva Torres

Considere o seguinte cenário: uma pessoa está interessada em prestar concurso público para cargos existentes em prefeituras municipais próximas à cidade (raio de 50 km) de Campinas-SP. Dado este interesse, esta pessoa gostaria de encontrar as páginas Web destas prefeituras de modo que possa localizar editais em aberto.

A área de recuperação de informação geográfica (GIR) procura tratar de questões deste tipo. Ela lida com os desafios derivados da adição da variável geográfica na tradicional área de recuperação de informação (IR). Os desafios na área incluem interpretar a consulta formulada pelo usuário, buscar a informação armazenada em repositórios, selecioná-la conforme a sua relevância para o usuário, ordená-la (*rank*) e mostrar o conjunto resultado de forma adequada. Como a própria consulta formulada pelo usuário normalmente envolve um grau de imprecisão, o resultado retornado também contém uma margem (pequena) de itens não relevantes. O objetivo principal para IR é maximizar os resultados relevantes e minimizar os irrelevantes [1].

A informação geográfica está presente direta ou indiretamente no dia-a-dia das pessoas, desta forma, não é de se admirar que haja uma grande quantidade de informação na Web sobre entidades geográficas e grande interesse em achá-la ou localizá-la em mapas. Ferramentas como Google Maps e Google Earth vêm popularizando e atendendo necessidades dos usuários Web por informação geoespacial.

Os serviços de buscas convencionais são baseados em casamento de palavras-chaves e em geral não levam em conta que estas palavras podem representar entidades geográficas que podem se relacionar espacialmente com outras entidades geográficas. Mesmo que não tenham sido citadas explicitamente na consulta, elas representam potencialmente uma informação relevante para o usuário [5].

Um exemplo de consulta, relacionada ao cenário apresentado anteriormente, que a maioria dos sistemas GIR existentes não suporta seria: "Quais são as páginas das prefeituras das cidades vizinhas a Campinas?" A dificuldade em se processar este tipo de consulta reside em combinar consultas tradicionais feitas em mecanismos de busca na Web com operadores espaciais, usualmente implementados em bancos de dados espaciais.

Neste trabalho propõe-se uma arquitetura para enriquecer a busca Web tradicional adicionando consultas geográficas com o auxílio de banco de dados espaciais. A idéia é que o usuário expresse diretamente sua consulta geográfica e o sistema expanda esta consulta, envie-a às máquinas de buscas existentes, combine os resultados e retorne ao usuário os resultados ordenados por sua relevância .

A arquitetura proposta neste trabalho é um modelo de 3 camadas conforme ilustrado na Figura 4. Na camada de visualização será tratada a questão de interface humano-computador para entrada da consulta pelo usuário, o retorno do resultado da consulta e o refinamento do resultado. Prevê-se a possibilidade em usar APIs

externas para ajudar na exibição de informação extraída da Web, por exemplo o Google Maps API, que são providas externamente ao sistema para ajudar o desenvolvedor adicionar em suas páginas, funcionalidades providas por outros sites.

Na camada de processamento da entrada, se encontra o desambiguador (*geoparsing*) de termos usados na consulta para indicar lugar (A), o geocodificador – que associará as coordenadas geográficas – da consulta referência (B), o módulo de expansão de consulta (C), o gerenciador de máquinas de buscas (E), o refinador (feedback) de consultas (D) e o módulo de ranking por relevância (F). A gerenciador de busca pode repassar a busca para várias outras máquinas de busca existentes na Web de forma que o resultado do sistema será a combinação dos resultados retornado pelas diversas máquinas de buscas.

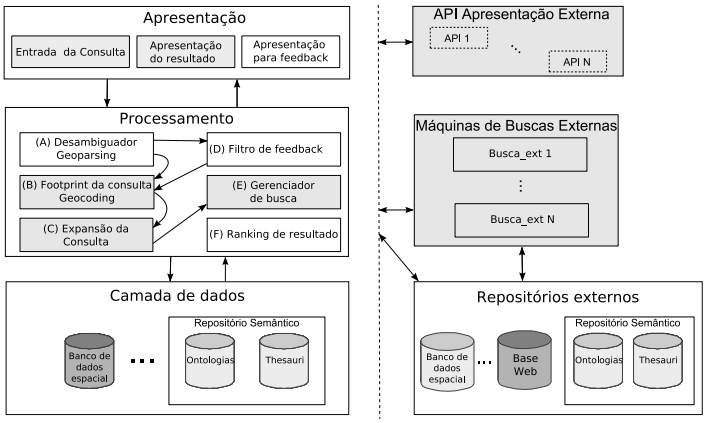


Figura 4: Arquitetura para recuperação informação geográfica na Web.

Por fim, a camada de dados é composta por repositórios locais e os distribuídos pela Web. Estes repositórios consistem de dados, ontologias e thesauri para desambiguar termos ou expandir a consulta do usuário. Os repositórios remotos podem conter também outras ontologias e thesauri, mas também incluem os documentos disponíveis na Web.

Os desafios a serem tratados na camada de apresentação são em relação à forma como o usuário poderia expressar a sua consulta, como o resultado poderia ser apresentado, ou como ele poderia dar um retorno ao sistema indicando quais resultados são realmente relevantes de modo que o sistema possa aprender a refinar melhor o retorno e o ranking de relevância.

Na camada de processamento, um dos desafios é como desambiguar nomes de lugares, pois um nome pode ser comum a vários lugares e objetos ou pode ter um nome alternativo. A outra é como apresentar as alternativas dos nomes para filtragem do usuário e submeter a nova consulta considerando o feedback do usuário após interação anterior. Por outro lado, ao pretender submeter a consulta a vários mecanismos de buscas existentes, o desafio será combinar os resultados de várias fontes, fazer um ranking de relevância deles, retornar para o usuário e tratar o

feedback do usuário com relação à relevância dos resultados apresentados e interagir com os diversos mecanismos de busca.

Se considerar a entrada da consulta espacial em linguagem semi-estruturada ou natural, entre os desafios estão como identificar e manipular referências a lugares nas consultas Web [3, 4, 8] e lidar com imprecisões dessas referências [6].

Considerando a própria Web como um grande repositório de dados, montar um repositório de conhecimento geográfico de forma automática com base em informação disponível na Web se torna um desafio e tanto. Neste caso, lida-se com informação inconsistente [7], desafio de identificar e de geocodificar dados textuais não-estruturados encontrados nas páginas Web [2].

Parte da arquitetura proposta (Figura 4) foi implementada em um protótipo, sendo que os módulos tratados foram destacados em cinza mais escuro. Buscas envolvendo operadores espaciais foram implementadas por meio de consultas enviadas a um banco de dados espacial. Esse banco de dados foi carregado com dados vetoriais obtidos do site do IBGE como cidades, estados, rios, rodovia federal brasileira.

A entrada da consulta é estruturada através de uma interface Web com campos fixos, em que o usuário indica o tipo da informação a ser retornada (por exemplo, páginas de prefeitura), o tipo de objeto espacial ao qual esta informação se relaciona (exemplo, cidade) que aqui será chamado de objeto-alvo e a relação espacial (exemplo, vizinho) que estes objetos-alvos devem ter com um objeto de referência (objeto-referência), sendo que o usuário também especifica o tipo do objeto-referência e o caracteriza (exemplo, cidade Campinas).

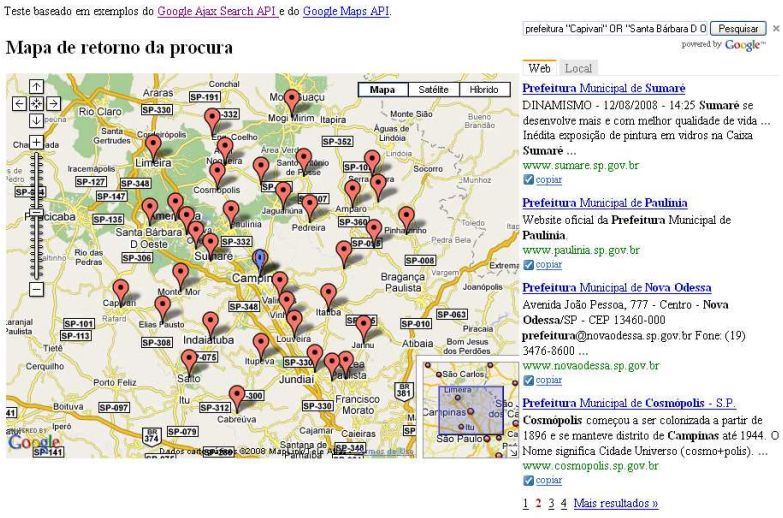


Figura 5: Resultado da consulta espacial na Web: "Quais são as páginas das prefeituras das cidades próximas (até 50 km) da cidade de Campinas?".

No processamento da consulta, tendo o objeto-referência bem caracterizado ele pode ser usado na consulta geográfica equivalente fornecida pelo BDE para busca da lista de objetos-alvos. Com a lista de objetos em mãos, expande-se a consulta

espacial de entrada e envia-se a nova consulta para uma máquina de busca Web (no caso, Google). O resultado da busca é mostrado numa página Web em que se agrega os resultados retornados na busca e a localização espacial dos objetos-alvos no mapa (Figura 5). Este protótipo foi implementado usando a linguagem de programação Javascript e Python sob o framework Django para aplicações Web. A máquina de busca Web foi provida pelo Google AJAX Search API e a exibição da localização no mapa dos objetos-alvos foi fornecida pelo Google Maps API. Como repositório de dados espacial, adotou-se o PostgreSQL com extensão espacial PostGIS.

Referências

[1] R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., New York, NY, USA, 1999.

[2] K. A. V. Borges, A. H. F. Laender, C. B. Medeiros, and J. C. A. Davis. Discovering geographic locations in web pages using urban addresses. In *Proceedings of the 4th ACM workshop on Geographical information retrieval*, pages 31–36, Lisbon, Portugal, 2007.

[3] N. Cardoso and M. J. Silva. Query expansion through geographical feature types. In *Proceedings of the 4th ACM workshop on Geographical information retrieval*, pages 55–60, Lisbon, Portugal, 2007.

[4] G. Fu, C. B. Jones, and A. I. Abdelmoty. Ontology-based spatial query expansion in information retrieval. In *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*, Lecture Notes in Computer Science, pages 1466–1482. Springer Berlin / Heidelberg, 2005.

[5] C. B. Jones, A. I. Abdelmoty, D. Finch, G. Fu, and S. Vaid. The spirit spatial search engine: Architecture, ontologies and spatial indexing. In *Geographic Information Science*, Lecture Notes in Computer Science, pages 125–139. Springer Berlin / Heidelberg, 2004.

[6] R. C. Pasley, P. D. Clough, and M. Sanderson. Geo-tagging for imprecise regions of different sizes. In *Proceedings of the 4th ACM workshop on Geographical information retrieval*, pages 77–82, Lisbon, Portugal, 2007.

[7] A. Popescu, G. Grefenstette, and P. A. M. ellic. Gazetiki: automatic creation of a geographical gazetteer. In *Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*, pages 85–93, Pittsburgh PA, PA, USA, 2008.

[8] M. Sanderson and Y. Han. Search words and geography. In *Proceedings of the 4th ACM workshop on Geographical information retrieval*, pages 13–14, Lisbon, Portugal, 2007.

8. Implementação eficiente de algoritmos criptográficos em arquiteturas modernas

Autores: Diego Aranha, Julio López

A descoberta da criptografia de chave pública [1] revolucionou a forma de se construir sistemas criptográficos e possibilitou a integração de forma definitiva entre teoria e implementação em aplicações reais. Particularmente, trouxe a possibilidade de se estabelecer serviços criptográficos como sigilo e assinatura irretratável em ambientes onde não existe relação de confiança ou canal seguro para distribuição de chaves. O problema da distribuição de chaves converteu-se na dificuldade de obtenção de uma chave pública autêntica. O surgimento de infra-estruturas de chaves públicas solucionou este novo problema e criou problemas adicionais, dados os altos custos de processamento e armazenamento inerentes a este modelo [2].

O advento de sistemas criptográficos baseados no problema do logaritmo discreto em curvas elípticas [3, 4] produziram uma nova revolução na área. Ao apresentarem desempenho superior e exigirem chaves mais curtas para um mesmo nível de segurança que os métodos tradicionais de criptografia assimétrica, alguns dos problemas relacionados às infra-estruturas de chaves públicas foram minimizados. Infelizmente, a dificuldade de gerência e a sobrecarga de desempenho decorrentes dos certificados ainda dificultam a adoção de criptografia assimétrica em ambientes restritos [5].

A busca de alternativas ao paradigma de infra-estruturas de chave pública resultou também na descoberta de sistemas baseados em identidade. Foram concebidos inicialmente por Shamir [6] em 1984 para assinaturas digitais, mas a primeira realização funcional e eficiente para cifração só foi apresentada em 2001 por Boneh e Franklin [7] a partir de emparelhamentos bilineares sobre curvas elípticas. Após esta aplicação de emparelhamentos, novos protocolos com propriedades inovadoras e especiais foram desenvolvidos, o que levou a uma flexibilização considerável das primitivas criptográficas conhecidas. Entre os protocolos baseados em problemas sobre grupos bilineares, destacam-se acordo de chaves para múltiplas entidades [8], assinaturas curtas e agregadas [9], e paradigmas alternativos de certificação implícita [10].

Apesar das propriedades desejáveis, o desempenho de sistemas baseados em emparelhamentos ainda deixa a desejar. O cálculo de um emparelhamento ainda é significativamente mais caro do que a execução de um protocolo convencional [11]. Isto é natural, visto que os métodos bem estabelecidos de criptografia assimétrica puderam receber maior esforço de pesquisa. Esforço similar já é visto em criptografia baseada em emparelhamentos [12], resultando em novos emparelhamentos [13] e novas curvas adequadas à sua instanciação [14].

O desenvolvimento destes novos algoritmos, além de produzir resultados teóricos relevantes, também deve acompanhar as tendências tecnológicas atuais para que sua implementação considere os recursos disponíveis na máquina para obtenção de desempenho. A busca por algoritmos mais eficientes para criptografia consiste tanto em pesquisa algorítmica teórica quanto em pesquisa aplicada de implementação.

Tendências tecnológicas

Tradicionalmente, cada nova geração de processadores de propósito geral obtém ganhos de desempenho significativos de duas formas: aprimoramentos no processo de fabricação e mudanças arquiteturais. Estas últimas são comumente relacionadas à extração de *paralelismo em nível de instruções*, ou seja, à execução concorrente de instruções que não possuem dependências de dados. Entretanto, estas otimizações atualmente se deparam com limitações críticas. A extração de paralelismo em nível de instrução claramente apresenta um limite superior, e muitas aplicações possuem um alto grau de dependência de dados que restringe este paralelismo [15]. O aperfeiçoamento do processo de fabricação também atinge limitações físicas, já que componentes cada vez menores dissipam potência em uma área cada vez menor [16]. Como obstáculo adicional, o poder de processamento vem crescendo muito mais do que a velocidade da memória, e o acesso à memória já é reconhecido como o maior gargalo de execução nas arquiteturas atuais [17]. Estas três limitações provocaram uma mudança radical no projeto de arquiteturas modernas, transportando a ênfase antes colocada em mecanismos para extração automática de paralelismo para mecanismos explícitos, na forma de multiprocessamento e vetorização.

Arquiteturas multiprocessadas de propósito geral são chamadas *multi-core*. Em uma arquitetura *multi-core*, unidades de processamento independentes conectam-se ao sistema de memória. O *paralelismo em nível de thread* marca uma mudança de paradigma de programação [18]. Enquanto nas máquinas uniprocessadas e seqüenciais, o desempenho dos programas crescia automaticamente a cada nova geração de processadores, em arquiteturas *multi-core*, o desempenho dos programas está diretamente relacionado ao grau de paralelismo em nível de *thread* que o programa apresenta. Como esse paralelismo é extraído explicitamente e requer análise profunda do problema e solução sendo tratados, e o modelo de execução em *multithreading* é não-determinístico, paralelizar algoritmos não é uma tarefa trivial [19].

As arquiteturas modernas também contam com instruções especializadas para processamento de múltiplos objetos de dados simultaneamente. Essas instruções são classificadas como SIMD (*Simple Instruction - Multiple Data*) [20], e são extremamente úteis na otimização de programas com alta densidade aritmética. Instruções vetoriais são outro recurso para exploração de paralelismo de dados, mas com uma granularidade muito mais fina do que o multiprocessamento. Há um incentivo ainda maior para a utilização de instruções dessa natureza em arquiteturas modernas, visto que a latência de execução dessas instruções têm diminuído a cada nova geração de processadores.

Objetivos e resultados

Este projeto tem como finalidade desenvolver algoritmos seqüenciais e paralelos eficientes e implementações em *software* otimizadas para criptografia de curvas elípticas e criptografia baseada em emparelhamentos, abrangendo cálculo de emparelhamen-

tos, aritmética em curvas elípticas, corpos finitos e corpos de extensão. O objetivo principal consiste em tornar estes métodos de criptografia mais eficientes nas arquiteturas modernas, considerando métricas de desempenho e utilização de recursos. A implementação irá exigir o projeto de técnicas de otimização de algoritmos em arquiteturas modernas (embutidas, multiprocessadas) e concretizará os algoritmos em código funcional eficiente, fazendo o melhor uso possível dos recursos disponibilizados pelo *hardware*. Para isso, paralelismo em nível de tarefas e em nível de dados serão extensamente utilizados, incluindo a aplicação de multiprocessamento e conjuntos de instruções vetoriais.

Como resultados já obtidos, podemos citar os trabalhos [21] e [22]. O primeiro propõe um protocolo eficiente de cifrassinatura sob um modelo alternativo de certificação de chaves públicas. O segundo aprimora o estado-da-arte de implementações de criptografia de curvas elípticas em redes de sensores sem fio, ampliando sua viabilidade. Nós sensores representam um extremo no espectro de arquiteturas modernas, por terem recursos particularmente limitados e natureza descartável. Aproveitando as características peculiares da plataforma alvo, particularmente a configuração da hierarquia de memória, foi possível desenvolver otimizações para aritmética no corpo finito $\mathbb{F}_{2^{163}}$ e curva elípticas associadas, produzindo as implementações mais eficientes de quadrado, multiplicação, inversão e redução modular já publicadas para esta plataforma. A aritmética eficiente no corpo permitiu o cálculo de uma mutiplicação de ponto 39% mais rápida que a melhor implementação de curvas elípticas sobre corpos binários e 7% mais rápida que a melhor implementação para o caso primo, considerando o mesmo nível de segurança.

Referências

[1] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Trans. Inform. Theory*, IT-22:644–654, 1976.

[2] P. Gutman. PKI: it's not dead, just resting. *Computer*, 35(8):41–49, 2002.

[3] V. Miller. Uses of elliptic curves in cryptography, Advances in Cryptology. In *Crypto'85, LNCS*, volume 218, pages 417–426. Springer, 1986.

[4] N. Koblitz. Elliptic curve cryptosystems. *Math. of computation*, 48:203–9, 1987.

[5] L. B. Oliveira, M. Scott, J. López, and R. Dahab. TinyPBC: pairings for authenticated identity-based non-interactive key distribution in sensor networks. Cryptology ePrint Archive, Report 2007/482, 2007.

[6] A. Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology (CRYPTO '84)*, pages 47–53. Springer-Verlag, 1984.

[7] D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing. In *Proc. of CRYPTO '01*, pages 213–229. Springer-Verlag, 2001.

[8] A. Joux. A one round protocol for Tripartite Diffie-Hellman. *J. Crypto.*, 17(4):263–76, 2004.

[9] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the Weil pairing. *Lecture Notes in Computer Science*, 2248, 2001.

[10] S. S. Al-Riyami and K. G. Paterson. Certificateless Public Key Cryptography. Cryptology ePrint Archive, Report 2003/126, 2003.

[11] M. Scott. Computing the Tate pairing. In *Topics in Cryptology/CT-RSA '05*, volume 3376 of *LNCS*, pages 293–304. Springer, 2005.

[12] P. S. L. M. Barreto, B. Lynn, and M. Scott. Efficient Implementation of Pairing-Based Cryptosystems. *J. Cryptology*, 17(4):321–334, 2004.

[13] P. S. L. M. Barreto, S. Galbraith, C. Ó hÉigeartaigh, and M. Scott. Pairing computation on supersingular abelian varieties. Cryptology ePrint Archive, Report 2004/375, 2004.

[14] D. Freeman, M. Scott, and E. Teske. A taxonomy of pairing-friendly elliptic curves. Cryptology ePrint Archive, Report 2006/372, 2006.

[15] D. W. Wall. Limits of instruction-level parallelism. In *Proc. of the 4th ASPLOS*, volume 26, pages 176–189. ACM Press, 1991.

[16] V. Venkatachalam and M. Franz. Power reduction techniques for microprocessor systems. *ACM Comput. Surv.*, 37(3):195–237, 2005.

[17] Wm. A. Wulf and S. A. McKee. Hitting the memory wall: implications of the obvious. *SIGARCH Comput. Archit. News*, 23(1):20–24, 1995.

[18] H. Sutter and J. Larus. Software and the concurrency revolution. *ACM Queue*, 3(7):54–62, 2005.

[19] E. A. Lee. The Problem with Threads. *Computer*, 39(5):33–42, 2006.

[20] M. J. Flynn. Some computer organizations and their effectiveness. *IEEE Trans. Computers*, C-21(9):948–960, 1972.

[21] D. Aranha, R. Castro, J. López, and R. Dahab. Efficient certificateless signcryption. In *8o. Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, pages 257–258, Gramado, RS, Brasil, 2008.

[22] D. Aranha, D. Câmara, J. López, L. Oliveira, and R. Dahab. Implementação eficiente de criptografia de curvas elípticas em sensores sem fio. In *8o. Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais*, pages 173–186, Gramado, RS, Brasil, 2008.

9. K_r-packing of P_4-sparse graphs

Authors: Vagner Pedrotti and Célia Picinin de Mello

Introduction

The K_r-packing problem is defined as to find the maximum number of pairwise disjoint cliques of size r in a graph. Note that, for $r = 2$ the problem is exactly the maximum matching problem, which has a well-known polynomial time algorithm, but, for $r \geq 3$, this problem is NP-hard for general graphs. Even for restricted graph classes, such as line and total graphs ($r \geq 3$), and split graphs ($r \geq 4$), the problem remains NP-hard. However, Guruswami et al. proposed a polynomial time algorithm for cographs (when r is fixed) [3]. In this work we extend this algorithm to the P_4-sparse graphs.

Throughout this abstract, $G = (V(G), E(G))$ denotes a simple, finite, and undirected graph, and we use standard graph terminology [1].

A subset M of $V(G)$ is called a module of G if do not exist $\{a, b\} \subseteq M$ and $c \in (V(G) \setminus M)$ such that $\{a, c\} \in E(G)$ and $\{b, c\} \notin E(G)$. A module M of G is said to be strong if does not exist another module N of G such that $N \setminus M \neq \emptyset$, $M \setminus N \neq \emptyset$, and $N \cap M \neq \emptyset$.

The modular decomposition tree (MDT) of a graph G is defined having one node for each strong module of G. The parent of a node, related to strong module M, is the node associated with the smaller strong module that properly contains M. Hence, the MDT represents inclusions of strong modules, from isolated vertices (leafs) to the module $V(G)$ (the root). If N is a non-leaf node of the MDT and M is its corresponding module in G, then N is called serial (parallel), if $\overline{G[M]}$ ($G[M]$) is not connected. Otherwise, the node is called neighborhood.

A graph is P_4-sparse [4] if the subgraph induced by each module corresponding to a neighborhood node in its MDT is isomorphic to a spider [2], which is a graph whose vertex set may be partitioned into three sets K, S, and H, such that: (1) $|K| = |S| \geq 2$, but H can be empty; (2) K is a clique and S is a stable set; (3) $\{i, j\} \in E(G), \forall i \in K, \forall j \in H$; (4) $\{i, j\} \notin E(G), \forall i \in S, \forall j \in H$; (5) $d(i) = 1$ (thin spider) or $d(i) = |S| - 1$ (thick spider), $\forall i \in S$; and (6) $d(j) = |H| + 1$ (thin spider) or $d(j) = |H| + |S| - 1$ (thick spider), $\forall j \in K$.

K_r-packing of cographs

In this section we recall the following results given by Guruswami et al. [3], that proposed an algorithm to compute in polynomial time a K_r-packing of a cograph, using a dynamic programming technique.

To describe these results, some definitions are necessary. A graph G is $(n_1, n_2, n_3, \ldots, n_r)$-packed if there exists a partition P of $V(G)$ such that there are n_i parts in P which are cliques of size i in G, for all $1 \leq i \leq r$. This obviously implies that

$|V(G)| = \sum_{i=1}^{r} in_i$. The partition P is called a $(n_1, n_2, n_3, \ldots, n_r)$-packing of G. The K_r-packing problem asks for the maximum value of n_r such that there is a $(0, 0, 0, \ldots, n_r)$-packing of G.

Consider the function $f(G, n_3, n_4, \ldots, n_r)$ defined as $\max\{n_2 : G$ has a $(0, n_2, n_3, n_4, \ldots, n_r)$-packing$\}$, and it is undefined if G has no such packing for any value of n_2. Note that, if we compute $f(G, 0, 0, \ldots, n_r)$ for $n_r \geq 0$, we solve the K_r-packing problem for G.

A cograph is a graph whose MDT has only serial and parallel nodes. So, using the following algorithms we can compute $f(G, n_3, n_4, \ldots, n_r)$ recursively for any cograph G.

If N is a parallel node of the MDT of G, M is the associated module, and $M_1, M_2 \ldots, M_k$ are the modules associated to the children of N, then $G[M] = G[M_1] \cup G[M_2] \cup \ldots \cup G[M_k]$. To compute $f(G[M], n_3, n_4, \ldots, n_r)$, we apply repeatedly an algorithm that computes f on a graph $G' \cup G''$. The algorithm returns the maximum of $f(G', n'_3, n'_4, \ldots, n'_r) + f(G'', n''_3, n''_4, \ldots, n''_r)$, for every integers $n'_i \geq 0$ and $n''_i \geq 0$ such that $n_i = n'_i + n''_i$, for each $3 \leq i \leq r$.

Now, if N is a serial node, then $G[M] = G[M_1] + G[M_2] + \ldots + G[M_k]$. By a similar argument, it suffices to apply the algorithm that computes f on a graph $G' + G''$. The algorithm returns the maximum of $n_{2,0} + n_{2,1} + n_{2,2}$, for all integers $n_{i,j}$ such that: (1) for $1 \leq i \leq r$, $n_i = \sum_{j=0}^{i} n_{i,j}$ where $n_{i,j} \geq 0$ for $0 \leq j \leq i$; (2) $n'_j = \sum_{i=j}^{r} n_{i,j}$ and $n''_j = \sum_{i=j}^{r} n_{i,i-j}$ for $1 \leq j \leq r$; (3) $f(G', n'_3, \ldots, n'_r) \geq n'_2$ and $f(G'', n''_3, \ldots, n''_r) \geq n''_2$; (4) $n' = \sum_{j=1}^{r} jn'_j$ and $n'' = \sum_{j=1}^{r} jn''_j$; and (5) $\sum_{i=2}^{r} n_{i,j} = 0$ or $\sum_{i=2}^{r} n_{i,0} = 0$.

K_r-packing of P_4-sparse graphs

To decide the K_r-packing problem for a P_4-sparse graph, we need to solve the function f on spiders using an algorithm similar to the one for joint graphs.

Let G be a spider and K, S, and H be the partition of $V(G)$ as defined in Section . If P is a $(n_1, n_2, n_3, \ldots, n_r)$-packing of G, then each $K_i \in P$ either has a vertex in S or is a subset of $K \cup H$. Hence, we can define, for $1 \leq i \leq r$ and $0 \leq j \leq i$, the integer $n_{i,j}$ as the number of $K_i \in P$ such that $|K_i \cap K| = j$ and $|K_i \cap H| = i - j$, and the integer n_i^S as the number of $K_i \in P$ such that $K_i \cap S \neq \emptyset$. Moreover, we define $n'_j = \sum_{i=j}^{r} n_{i,j}$ as the number of $K_i \in P$, such that $K_i \cap S = \emptyset$ and $|K_i \cap K| = j$; and $n''_j = \sum_{i=j}^{r} n_{i,i-j}$ as the number of $K_i \in P$, such that $K_i \cap S = \emptyset$ and $|K_i \cap H| = j$.

From the following lemmas, we can construct an algorithm for the K_r-packing problem for a spider graph.

Lemma 9.1 *A thin spider G has a $(n_1, n_2, \ldots, n_r)$-packing P if, and only if, there exist non-negative integers n_1^S, n_2^S, and $n_{i,j}$, for $1 \leq i \leq r$ and $0 \leq j \leq i$, such that: (1) $n''_2 \leq f(G[H], n''_3, \ldots, n''_r)$; (2) $|S| = n_1^S + n_2^S$, $|K| = n_2^S + \sum_{i=1}^{r} in'_i$, and $|H| = \sum_{i=1}^{r} in''_i$; and (3) $n_i = \sum_{j=0}^{i} n_{i,j}$ for $3 \leq i \leq r$ and $n_i = n_i^S + \sum_{j=0}^{i} n_{i,j}$ for $i \in \{1, 2\}$.*

Lemma 9.2 *If G is a thin spider and P is a $(n_1, n_2, \ldots, n_r)$-packing of G, then there is another $(n_1, n_2, \ldots, n_r)$-packing, P', of G, such that every K_2 in P' either is contained in H or has one vertex in S and the other in K.*

Lemma 9.3 *A thick spider G has a $(n_1, n_2, \ldots, n_r)$-packing if, and only if, there exist non-negative integers $n_{i,j}$, for $1 \leq i \leq r$, $0 \leq j \leq i$, and n_i^S, for $1 \leq i \leq r$, such that: (1) $n_2'' \leq f(G[H], n_3'', \ldots, n_r'')$; (2) $|S| = \sum_{i=1}^{r} n_i^S$, $|K| = \sum_{i=2}^{r}(i-1)n_i^S + \sum_{i=1}^{r} in_i'$, and $|H| = \sum_{i=1}^{r} in_i''$; (3) $n_i^S = 0$, for $i > |K|$; and (4) $n_i = n_i^S + \sum_{j=0}^{i} n_{i,j}$, for $1 \leq i \leq r$.*

Lemma 9.4 *If a thick spider G has a $(n_1, n_2, \ldots, n_r)$-packing, then there is another partition P', which is also a $(n_1, n_2, \ldots, n_r)$-packing of G, but every K_2 in P' either is contained in H or has a vertex in S.*

Now we are ready to describe the algorithm that solves a K_r-packing for a P_4-sparse graph G.

The function f is computed in each node of the MDT of G, processing serial and parallel nodes as in Section . For neighborhood nodes, f is computed as the maximum of $n_{2,0} + n_2^S$ over all integers $n_{i,j}$ and n_i^S satisfying the conditions given in lemmas 9.1 and 9.3 and the condition $\sum_{i=2}^{r} n_{i,i} = 0$ or $\sum_{i=2}^{r} n_{i,0} = 0$.

The expression maximized is due to lemmas 9.2 and 9.4. The additional condition comes from the fact that $G[K \cup H] = G[K] + G[H]$ and from the following lemma:

Lemma 9.5 (Lemma 4.2 of [3]) *If P is a $(n_1, n_2, \ldots, n_r)$-packing of $G = G' + G''$, then there exists a $(n_1, n_2, \ldots, n_r)$-packing P' covering precisely the same vertices as P does and P' does not contain C' and C'' such that $C' \subseteq V(G')$ and $C'' \subseteq V(G'')$.*

The MDT of any graph is obtained in linear time [5]. We also can identify if a graph is spider, as well as identify the partition of the spider in the three sets K, S, and H, in linear time [2]. Since the number of possibilities evaluated for spiders is a subset of the possibilities evaluated for joint graphs, the time complexity of the proposed algorithm is also polynomial.

We conclude this abstract observing that MDT can be applied to solve the K_r-packing problem for other graphs that have well characterized neighborhood nodes in its MDT, such as P_4-tidy graphs [2].

Referências

[1] J. A. Bondy and U. S. R. Murty. *Graph Theory and its Applications*. MacMillan Press, 1976.

[2] V. Giakoumakis, F. Roussel, and H. Thuillier. On P_4-tidy graphs. *Discrete Mathematics & Theoretical Computer Science*, 1(1):17–41, 1997.

[3] V. Guruswami, C. Pandu Rangan, M. S. Chang, G. J. Chang, and C. K. Wong. The K_r-Packing Problem. *Computing*, 66:79–89, 2001.

[4] C.T. Hoàng. A class of perfect graphs. Master's thesis, School of Computer Science, Montreal, 1983.

[5] R.M. McConnell and J.P. Spinrad. Modular decomposition and transitive orientation. *Discrete Mathematics*, 201(1-3):189–241, 1999.

10. Um Middleware para Controle e Monitoramento de Instrumentação em Tempo Real em Grades Computacionais

Autores: Carlos Roberto Senna e Edmundo Roberto Mauro Madeira (Orientador)

Resumo

O uso de instrumentação de forma remota pode ser feito de várias formas. Instrumentos conectados diretamente a computadores, acessíveis ou não via rede local, ou até mesmo pela Internet. Nesta proposta pretendemos criar uma infra-estrutura (*middleware*) de alto desempenho que permita o controle e o monitoramento de instrumentação de forma remota, trabalhando em tempo real sobre uma arquitetura de grade computacional.

O modelo proposto adiciona às grades computacionais características que são fundamentais em aplicações que requerem controle total à distância com sensação de presença, agregando às grades um novo e importante nível de qualidade de serviço (*QoS*). A Grade incorporada dessa infra-estrutura agrega as características colaborativas da Web 2.0, tornando-se uma plataforma adequada para e-Ciência, permitindo que centros de excelência em pesquisa possam compartilhar seus equipamentos, experimentos, bases operacionais, dados e informações com centros com menor capacidade, atingindo um novo patamar em pesquisa colaborativa (*WikiScience*).

Introdução

Instrumentos são recursos caros, escassos e concentrados, permitindo acesso a poucas pessoas. A sofisticação dos instrumentos requer pessoas especializadas em seu uso. Atividades como pesquisa científica, controle de qualidade e desenvolvimento colaborativo, unem essas características, fazendo com que poucos possam fazer uso efetivo da instrumentação. Um ambiente que permita o acesso a esses recursos, físicos (instrumentos) ou humanos (técnicos e pesquisadores) traz benefícios importantes. Laboratórios sofisticados podem ser acessados por pessoas de vários níveis de especialização, supervisionadas por profissionais qualificados gerando o ambiente ideal para colaboração [1].

No entanto a integração de instrumentos em aplicações científicas, comerciais ou industriais, ainda não conta com suporte apropriado quando usada em conjunto com grades computacionais. Para fazer essa integração, propomos um novo modelo que permita à grade computacional a composição de serviços criando novas capacidades (*"mashups"*) acessíveis como novos serviços. Essa nova infra-estrutura trata serviços como plataforma, permitindo aos usuários selecionar e compor serviços e também publicar resultados como novos serviços da Grade.

A plataforma incorpora ainda características que são fundamentais em aplicações que requerem controle total à distância com sensação de presença, agregando às grades um novo e importante nível de qualidade de serviço (*QoS*). O modelo permite a virtualização de instrumentos e experimentos, organizando-os em classes conforme características como aplicação, requisitos operacionais, tipos de interfaces, etc. O modelo implementa acessibilidade aos instrumentos com facilidades para integração com a Grade e suporte a vários tipos de interfaces e controles.

Projeto

O objetivo é modelar e desenvolver uma infra-estrutura para e-Ciência [3], que incorpore os conceitos colaborativos da Web 2.0 (comunidades, *blogs*, *mashups*, etc) [7], com facilidades e recursos para o controle e monitoramento de instrumentação de forma remota, trabalhando em tempo real sobre uma arquitetura de grade computacional [2]. Essa nova infra-estrutura incorpora às grades características para promover a troca de dados, produtos e código, visando a criação de "Metadados e Metaserviços da Comunidade". Isso torna o ambiente mais interativo facilitando a colaboração, transformando indivíduos em comunidades, ciência em computação (*WikiScience*).

A Arquitetura Proposta

A arquitetura é composta por uma infra-estrutura de Grade baseada em OGSA [6], na qual são agregados conjuntos de serviços para virtualização de instrumentos e experimentos (VI). Para fazer a orquestração dos serviços da Grade é usado o GPO, que será expandido recebendo recursos para gerência de metaworkflows e virtualização de recursos (MW).

Toda essa infra-estrutura será acessada pelos usuários através de um portal, incrementado de facilidades que o coloquem dentro do conceito de Web 2.0. A Figura 6 mostra os módulos da arquitetura proposta cujos detalhes são apresentados a seguir.

Modelos Desenvolvidos

Dois modelos desenvolvidos em trabalhos anteriores são relevantes na presente proposta. O primeiro modelo é o GPO (*Grid Process Orchestration*) [8] um middleware para interoperabilidade de aplicações distribuídas que requerem composição de serviços em uma grade computacional. O GPO permite a criação e gerência de fluxos de aplicações, tarefas e principalmente serviços das grades computacionais.

O segundo modelo desenvolvido é um sistema para controle de instrumentos de fotônica de forma remota através da Internet [5]. O sistema permite o controle de diferentes instrumentos (lasers sintonizáveis, analisadores de espectro óptico, medidores de potência, analisadores PMD) para gerar, operar e analisar sinais ópticos utilizando a rede óptica do Projeto Kyatera [4]. O usuário configura/reconfigura

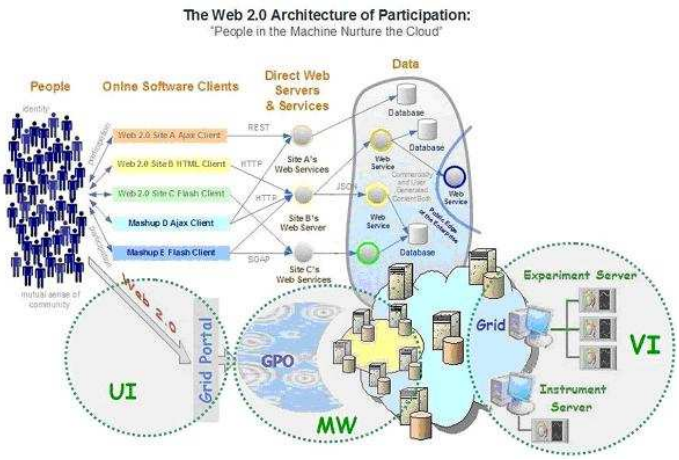


Figura 6: Arquitetura Proposta

seu experimento conforme a sua necessidade usando chaves ópticas remotamente controladas.

Módulo para Virtualização de Instrumentos (VI)

Para suportar a virtualizaçao dos instrumentos serão criadas regras para definição de serviços em três níveis: físico, abstração de instrumentos e interfaces.

No nível físico estão os serviços que controlam diretamente os instrumentos (*drivers*). Nesses serviços serão usadas as particularidades de cada instrumento como tipo de interface de comunicação, linguagem de controle, tipos de dados, etc.

No nível de abstração os serviços usam um repositório onde estão informações sobre as classes de instrumentos e as definições específicas, como por exemplo, os serviços do nível físico disponíveis para o instrumento alvo.

No nível de interfaces estão as definições sobre modelos de dados a serem usados como argumentos e como resultado no acesso aos instrumentos.

Módulo de Gerência de MetaWorkflow (MW)

O módulo de Gerência de Metaworkflow (MW) recebe as submissões feitas através do portal ficando responsável pela sua execução. As submissões podem ser metaworkflows escritos pelo usuário ou requisições geradas pelos *mashups*. Em ambos os casos tais submissões são interpretadas pelo MetaGPO que gera um "workflow abstrato" intermediário a ser trabalhado pelo GPOWC (*GPO Workflow Creator*). O GPOWC analisa os recursos da Grade, consulta o "Repositório de Workflows e Metadados", gera um workflow concreto e submete-o ao GPOWE (GPO Workflow Engine).

Módulo de Interface com o Usuário (UI)

O módulo de interface com o usuário é constituído por um portal que agrega à Grade recursos com característicos da Web 2.0. Esses recursos agregados permitem que novas facilidades, como por exemplo, a criação de um workflow resultante da combinação de outros workflows, através de *mashups*, possam ser incorporados ao ambiente operacional.

O portal permite aos usuários classificar e organizar seus resultados e compará-los com os da coletividade. Esse automatismo pode qualificar os resultados obtidos imediatamente tornando-se mais uma forma de validação dos experimentos.

Referências

[1] Chiculita, C., Frangu, L. *A Web Based Remote Control Laboratory.* The 6th World Multiconference on Systemics, Cybernetics and Informatics, Orlando, Florida, EUA, 14-18 Julho, 2002.

[2] Foster, I., Kesselman, C., Tuecke, S. *Anatomy of the Grid: Enabling Scalable Virtual Organizations.* International Jounal of High Performance Computing Applications, 15(3):200-222, Thousand Oaks, CA, EUA, 2001.

[3] Foster, I. *Service-Oriented Science.* Science Magazine Vol. 308, pp 814-17, 6 Maio, 2005.

[4] FAPESP Fundação de Amparo à Pesquisa do Estado de São Paulo. *Projeto TIDIA KyaTera.* http://kyatera.incubadora.fapesp.br/portal.

[5] Fragnito, H. L. *Laser - Noise Interactions: Modulation Instability.* 8th J. A. Swieca Summer School on Quantum Optics and Nonlinear Optics, IFGW, UNICAMP, Campinas, Brazil, January, 2002.

[6] OGF Open Grid Forum. *The Open Grid Services Architecture, Version 1.5.* www.ogf.org/documents/GFD.80.pdf, 24 Julho, 2006.

[7] O'Reilly, T. *What is Web 2.0. Design Patterns and Business Models for the Next Generation of Software.* http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html, 30 Setembro, 2005.

[8] Senna, C. R. e Madeira, E. R. M. *A middleware for instrument and service orchestration in computational grids.* 7th IEEE International Symposium on Cluster Computing and the Grid - CCGrid 2007, Rio de Janeiro - Brasil, 14-17 Maio, 2007.

11. Specification of a Framework for Semantic Annotation of Geospatial Data on the Web

Authors: Carla Geovana do N. Macário, Claudia Bauzer Medeiros

Agriculture is very important for the world economies and the availability of systems for agro-environmental planning and management is a key issue to improve this activity. The term "Geospatial data" refers to all kinds of data on objects and phenomena in the world that are associated with spatial characteristics and that reference some location on the Earth´s surface. Examples include historical information of climate, soil and temperature data, and others like maps and satellite images. Such date are a basis for decision mapping in agriculture. Their combined use is useful to answer questions like '*when will be the best time to start the harvesting coffee in this area?* or '*what was the productivity of sugar-cane in this region that had the same sampled pattern?*. Such questions are important for production planning and the definition of public policies concerning agricultural practices, also allowing the environmental control of protected areas. Although agricultural research generates a huge amount of information, the decision on what to produce (and when, where and how) requires a reliable access to updated data. So, it becomes necessary solutions for data acquisition, processing and exchange in a small time.

WebMAPS is multidisciplinary project under development at UNICAMP (Brazil). Its goal is to develop a platform based on Web Services for agricultural planning. It requires state of the out research in specification and implementation of software that combines satellite maps with data from surface sensors and from agricultural production. It aims to provide a decision support platform for public policies in agricultural planning [Medeiros et al., 2006].

WebMAPS intends to use information distributed all over the world and available in specific repositories on the Web, where sometimes only their owners know their meaning, to answer the user questions. Although very useful, this distribution and heterogeneity poses many problems. First, because only the people or organizations that produce the data know what it means. Second, because these data are inherently heterogeneous, being produced and maintained by distinct entities, according to a variety of temporal and spatial criteria. Third, in order to take advantage of available information, new discovery tools must be devised, since at present search engines are mostly centered on syntactic properties of data. According to [Lutz and Klien, 2006] this conduct us to an absent of semantic interoperability, which implies in three levels of problems at three levels:

- metadata level, blocking the discovery of geographic information;
- schema level, hampering the retrieval of geographic information;
- data level, complicating the interpretation and integration of geographic information

One challenge of this project is thus to enable the use of these data by the several users from different cultures, needs and organizations. The use of annotation as a way to provide semantics to resources is not new and has been used for a long time. The Semantic Web, different from traditional approaches, proposes the use of a formal vocabulary to annotate Web, providing to all users the same understanding of objects described. In this way, anyone can find a document that meets the desired meaning. The Semantic Web, and more specifically the Geospatial Semantic Web proposed by [Egenhofer, 2002], is an effort to achieve this, through the use of semantic annotation to describe the meaning of each available data. There are several proposals to provide semantics to data on the Web. However, even the research concerned with geospatial data is centered on annotating textual documents, disregarding other kinds of data that are very important for agriculture, such as satellite maps. When these data are considered, they are annotated using a keywords schema.

As an answer to this, we are developing a mechanism for semantic annotation of the different geospatial data available on the web. The intended mechanism, to be implement as a web service in WebMAPS, should automate the annotation process and allow the integration of heterogeneous data to generates the annotations. As main challenges on this work we consider the following issues: how to combine the available data? how to deal with heterogeneity questions? how to obtain the data? which annotation schema is better to be used?

Our research is based in on the use of ontologies, GIR (Geographic Information Retrieval), workflows and web services. The first step is focused on the analysis of the data to be considered, expecting to obtain a core of metadata that can be automatically extracted, such as date and locality information (latitude and longitude). According to the data being considered, other metadata (e.g. temperature, locality name, agricultural zoning) should be generated/obtained based on scientific workflows that describe how to do this. As a way to do the query, we decided to initially consider a structure proposed by the SPIRIT project (<*theme, spatial relationship, locality*>), where each term is associated to an ontology. The management of these ontologies will be done by the service provided by [Daltio and Medeiros, 2007]. Finally we are considering to use the available metadata standards as a way to promote reuse and improve the mechanism. Although we expect that will be necessary to extend then to accomodate the established requirements.

In this work we intend to present the proposed mechanism to support management of semantic annotations on digital content on the Web, considering not only textual but also non-textual data, for agricultural planning.

Referências

[Daltio and Medeiros, 2007] Daltio, J. and Medeiros, C. B. (2007). An ontology service for biodiversity information system (in portuguese). In *XXXIV SEMISH:*

Brazilian National CS Conference, pages 2143–2157, Rio de Janeiro, Brazil.

[Egenhofer, 2002] Egenhofer, M. J. (2002). Toward the semantic geospatial web. In *GIS '02: Proceedings of the 10th ACM international symposium on Advances in geographic information systems*, pages 1–4, New York, NY, USA. ACM Press.

[Fileto et al., 2003] Fileto, R., Liu, L., Pu, C., Assad, E. D., and Medeiros, C. B. (2003). Poesia: An ontological workflow approach for composing web services in agriculture. *The VLDB Journal*, 12(4):352–367.

[Jones et al., 2004] Jones, C., Abdelmoty, A., Finch, D., Fu, G., and Vaid, S. (2004). The SPIRIT spatial search engine: Architecture, ontologies and spatial indexing. In *Geographic Information Science: Third International Conference, Gi Science 2004*, pages 125 – 139, Adelphi, Md, USA.

[Lutz and Klien, 2006] Lutz, M. and Klien, E. (2006). Ontology-based retrieval of geographic information. *International Journal of Geographical Information Science*, 20(3):233–260.

[Medeiros et al., 2006] Medeiros et al., C. M. B. (2006). *WebMAPS II - System based on Web for agricultural monitoring and planning*. CNPq Universal Project. Started in 2003 - renewed in 2006.

[Tsalgatidou et al., 2006] Tsalgatidou, A., Athanasopoulos, G., Pantazoglou, M., Pautasso, C., Heinis, T., Gronmo, R., Hoff, H., Berre, A., Glittum, M., and Topouzidou, S. (2006). Developing scientific workflows from heterogeneous services. *SIGMOD Record*, 35(2):22–28.

12. On-Line Dynamic Traffic Grooming Algorithms for WDM Mesh Networks

Authors: André Costa Drummond, Nelson Luis Saldanha da Fonseca

In WDM networks, there is a huge discrepancy between high capacity channels (OC-192 or OC-768) and the low speed of the flows, especially IP flows transported by these channels. In order to use resources efficiently, it is, thus, necessary to multiplex or "groom" low speed flows onto high capacity WDM channels.

In the routing and wavelength assignment problem (RWA), the employment of shortest path algorithms for choosing routes can lead to an unbalanced utilization of resources, thus increasing the probability of blocking. The drawbacks of these algorithms are especially crucial when the approach is used for large networks with high sustainable request arrival rates, since nodes can receive large numbers of connection requests.

One possibility for ameliorating the growth in computational complexity, as well as reducing the probability of blocking, is the search for a lightpath in a reduced network region, called a zone. This zone will be proportional in size to the number of nodes in the shortest physical route from the source to destination node of each connection request. The central idea here is to increase scalability and reduce computational complexity by avoiding lightpaths that may exhaust network resources. Moreover, a zone can be dynamically enlarged as needed to ensure flexibility in lightpath selection. Although this approach reduces the probability of blocking, as will be shown here, this reduction in blocking is generally not fairly distributed among all source-destination pairs.

The use of auxiliary graphs to represent network resources and topology is another development for the solution of the traffic grooming problem. The Single-Layered Route-Computation (SLRC) algorithm [1] builds, upon the arrival of a request for connection establishment, an auxiliary graph with vertices representing all OXCs in the network, and edges between nodes representing groomable lightpaths, or those potentially allocable. The SLRC algorithm tries to establish a route from source to destination using a traditional shortest-path algorithm in the auxiliary graph. If no path can be found the call is considered to be blocked. The major drawback of SLRC is not being scalable given the need for a full representation of the network.

Auxiliary graphs can represent single regions (zones) rather than the whole network, thus leading to solutions with lower computational complexity. The Zone Based With Neighbor Expansion (ZWNE) algorithm proposed in [2] employs such an auxiliary graph with a size proportional to the number of hops on the shortest path. The auxiliary graph in this solution includes the vertices of the shortest path between source and destination of a connection request, with this path sought upon the arrival of a request. Edges between these nodes represent existing lightpaths or those allocable (given by the solution of a routing and wavelength assignment

problem). If such a lightpath cannot be found in this restricted topology, an expansion auxiliary topology is introduced. This expansion adds vertices neighbors to the vertices already present in the auxiliary graph as well as new edges representing the relevant lightpaths of the physical topology. A similar expansion is utilized until an adequate route is determined. Although scalable, ZWNE leads to unbalanced distribution of resources among the source destination pairs.

The novel algorithm proposed here, the Alternative Routing With Virtual Topology Expansion (ARVTE) algorithm, is based on the main principals of the ZWNE algorithm. Although it also constructs a reduced auxiliary graph, the construction of this graph differs fundamentally. Two variants of ARVTE which provide different ways to compute the disjoint paths for building the auxiliary graph are proposed.

The initial step is an off-line determination of a path for each source-destination pair. This determination involves the execution of the traditional shortest path algorithm on a graph with edges which will receive high costs if used by other source destination pairs. The goal is to have a set of unique paths even if these do not represent the shortest possible route from a specific source node to a specific destination. The idea is that these disjointed paths will orient the establishment of lightpaths without the creation of network bottlenecks, a procedure entitled alternative routing. The two variants assign weights to the edges of the paths already chosen by other source destination pairs so that these edges can be avoided by new path selections. One of the variants assigns the highest possible value to edge weights and the other assigns weight values which accounts for the total number of paths that include the edge.

Whenever a request arrives for the establishment of a connection for a source-destination pair, an auxiliary graph is created to choose a lightpath for that connection. Vertices of this auxiliary graph are initially those of the path associated with the source destination path in the original offline procedure. Further edges will be added, however, if lightpaths already exist between these nodes or if the solution to a routing and wavelength assignment algorithm suggests that a lightpath between the two nodes can be established. The shortest path between the specific source-destination pair is then sought. If it exists, it will carry the connection to be established. Otherwise the auxiliary graph will be expanded by including vertices neighboring the vertices of the auxiliary graph in the virtual topology. If there are no neighbors to these virtual vertices, the auxiliary graph incorporate neighboring vertices in the physical topology. A maximum of p vertices can be added in this expansion. The shortest path algorithm is then executed for this expanded auxiliary graph. This procedure is repeated a certain number of iterations; the connection is considered to be blocked if no solution is found.

This paper introduces a novel algorithm and two of its variants for dynamic traffic grooming for WDM mesh networks. The efficiency and fairness promoted by this algorithm and its variants are compared to those of two others previously proposed algorithms. Simulation results using the NSF and the USA topologies evince that the algorithm introduced here is scalable and yet promotes a fair distribution of resources

among source destination pairs; these two characteristics are not simultaneously achieved by any other existing algorithm.

Referências

[1] K. Zhu and B. Mukherjee, "On-line approaches for provisioning connections of different bandwidth granuralities in wdm mesh networks," in *Proceedings of OFC*, 2002, pp. 549–551.

[2] Q.-D. Ho and M.-S. Lee, "A zone-based approach for scalable dynamic traffic grooming in large wdm mesh networks," *IEEE Journal of Lightwave Technology*, vol. 25, no. 1, pp. 261–270, 2007.

13. Grafos Pfaffianos e Problemas Relacionados

Autores: Alberto Alexandre Assis Miranda e Cláudio Leonardo Lucchesi

Seja D um grafo orientado. Seja Q um circuito com número par de vértices de D. Diz-se que a orientação de Q em D é *ímpar* se, ao fixarmos um sentido de percurso de Q, o número de arestas orientadas no mesmo sentido do percurso é ímpar. Um subgrafo H de um grafo G é *conforme* se $G - V(H)$ tem emparelhamento perfeito. As seguintes afirmações sobre D são equivalentes:

- D é uma orientação *Pfaffiana*;
- todo circuito conforme de D com um número par de vértices tem orientação ímpar;
- seja M um emparelhamento perfeito qualquer de D, então todo circuito M-alternado tem orientação ímpar.

Um grafo não orientado é Pfaffiano se e somente se existe uma orientação de suas arestas que é uma orientação Pfaffiana.

A definição de grafo Pfaffiano deriva da idéia de Tutte de usar a estrutura matemática Pfaffiano (não definida aqui) na teoria de emparelhamentos. Em seu livro "Graph Theory As I Have Known It" [14], ele descreve como chegou à idéia de usar os Pfaffianos para determinar uma fórmula para o número de emparelhamentos perfeitos de um grafo. Apesar de não ter sido bem sucedido em encontrar essa fórmula, Tutte conseguiu utilizar identidades envolvendo Pfaffianos para demonstrar o seu teorema famoso que caracteriza grafos que têm emparelhamentos perfeitos [13].

Surpreendentemente, o problema de se decidir se um dado grafo é Pfaffiano está relacionado a outros problemas fundamentais de teoria dos grafos, e aparentemente não relacionados. Por exemplo, o problema de decidir se um dado grafo orientado tem ou não um circuito orientado par é equivalente ao problema de decidir se um grafo bipartido dado é ou não Pfaffiano [15].

O problema está também relacionado a outras questões em física, química e economia (veja o livro de Lovász e Plummer [6, Capítulo 8] e o artigo de McCuaig [7].) Motivado por problemas externos à teoria dos grafos, Kasteleyn [3] demonstrou que todo grafo planar tem uma orientação Pfaffiana. O grafo $K_{3,3}$ é o menor grafo não Pfaffiano.

Little [4] demonstrou que um grafo bipartido é Pfaffiano se e somente se não contém subgrafo conforme que é uma bissubdivisão de $K_{3,3}$. Uma *bissubdivisão* de um grafo é obtida substituindo-se arestas do grafo por caminhos com um número par de vértices internos. Um subgrafo H de um grafo G é *conforme* se $G-V(H)$ tem um emparelhamento perfeito. Assim, o problema de decidir se um dado grafo bipartido é Pfaffiano está em co-NP. Este resultado imediatamente sugere uma pergunta natural: decidir se o problema geral está ou não em NP. Vazirani e Yanakakis [15] mostraram que decidir se uma dada orientação de um grafo é ou não Pfaffiana é tão difícil quanto decidir se o grafo tem ou não uma orientação Pfaffiana.

Outras duas classes interessantes de grafos são os grafos "quase-bipartidos" e os grafos "sólidos". Recentemente, Little e Fischer caracterizaram grafos quase-bipartidos Pfaffianos [5], de uma forma similar àquela dos bipartidos, através de subgrafos conformes proibidos. Recentemente, Carvalho, Lucchesi e Murty caracterizaram grafos sólidos Pfaffianos [1], também de uma forma similar àquela dos bipartidos.

As caracterizações citadas acima estão intimamente relacionadas com uma classe de operações sobre grafos, que chamaremos de *reduções Pfaffianas*. Dizemos que uma operação f sobre grafos é uma *redução Pfaffiana* se $f(G) < G$ e $f(G)$ não é Pfaffiano somente se G não é Pfaffiano. Dizemos que um grafo não Pfaffiano G é *irredutível* com relação a um conjunto de operações, se para qualquer aplicação de uma destas operações o grafo obtido é Pfaffiano. As caracterizações citadas anteriormente podem ser refraseadas como: (i) o único bipartido irredutível é o $K_{3,3}$, (ii) os únicos quase-bipartidos irredutíveis são o $K_{3,3}$, Γ_1 e $\Gamma 2$, e (iii) o único sólido irredutível é o $K_{3,3}$. Recentemente, Norine e Thomas [10] mostraram uma classe infinita de grafos irredutíveis para as operações comumente utilizadas com grafos Pfaffianos. Neste mesmo artigo, sugeriram novas operações e considerando estas operações fizeram a seguinte conjectura:

Conjectura 1 *Os únicos grafos irredutíveis são: $K_{3,3}$, Γ_2 e o grafo de Petersen.*

Em 1998, McCuaig [7], e, independentemente, Robertson, Seymour e Thomas [11], descobriram algoritmo polinomial para decidir se um grafo bipartido tem ou não uma orientação Pfaffiana. Em 2007, descobrimos um algoritmo de tempo polinomial que decide se um grafo "quase-bipartido" é Pfaffiano [8]. Ainda não se conhece algoritmo de tempo polinomial para decidir se um grafo sólido é Pfaffiano.

Dizemos que um grafo G é *k-Pfaffiano*, para algum inteiro k, se existe uma k-tupla de orientações $(D_1, D_2, \ldots, D_k)$ de G e uma k-tupla de reais $(\alpha_1, \alpha_2, \ldots, \alpha_k)$ tais que para todo emparelhamento perfeito M de G

$$\sum_{i=1}^{k} \alpha_i D_i(M) = 1.$$

Na década de 60, Kasteleyn conjecturou que todo grafo imersível em uma superfície orientável de genus g é 4^g-Pfaffiano. Galluccio e Loebl [2] e independentemente Tesler [12] provaram esta conjectura. Recentemente, Norine [9] generalizou este resultado provando que a existência de um desenho em uma superfície orientável de genus g onde todo emparelhamento perfeito tem um número par de cruzamentos entre suas arestas é suficiente para que o grafo seja 4^g-Pfaffiano. Além disso, Norine provou que a recíproca vale para $g \leq 1$. Norine provou também que um grafo 3-Pfaffiano é Pfaffiano e que um grafo 5-Pfaffiano é 4-Pfaffiano. A partir destes resultados fica a seguinte conjectura, feita por Norine:

Conjectura 2 *Um grafo é k-Pfaffiano mas não $(k-1)$-Pfaffiano somente se $k = 4^g$ para g inteiro.*

O foco do nosso projeto a partir deste ponto serão as Conjecturas 1 e 2.

Referências

[1] M. H. de Carvalho, C. L. Lucchesi, and U. S. R. Murty. A characterization of solid pfaffian matching covered graphs. Comunicação Particular.

[2] A. Galluccio and M. Loebl. On the theory of Pfaffian orientations. I. Perfect matchings and permanents. *Eletron. J. Combin.* , 6, 1999.

[3] P. W. Kasteleyn. Dimer statistics and phase transitions. *J. Math. Phys.*, 4:287–293, 1963.

[4] C. H. C. Little. A characterization of convertible $(0, 1)$-matrices. *J. Combin. Theory Ser. B*, 18:187–208, 1975.

[5] C. H. C. Little and I. Fischer. A characterisation of Pfaffian near bipartite graphs. *J. Combin. Theory Ser. B*, 82:175–222, 2001.

[6] L. Lovász and M. D. Plummer. *Matching Theory*. Number 29 in Annals of Discrete Mathematics. Elsevier Science, 1986.

[7] W. McCuaig. Pólya's permanent problem. *The Electronic J. of Combin.*, 11, 2004.

[8] A. A. A. Miranda and C. L. Lucchesi. A polynomial time algorithm for recognizing near-bipartite Pfaffian graphs. Technical Report IC-07-15, May 2007.

[9] S. Norine. Drawing 4-Pfaffian graphs on the torus. *Combinatorica*, Aceito para publicação http://www.math.princeton.edu/~snorin/papers.html.

[10] S. Norine and R. Thomas. Minimally non-Pfaffian graphs. *J. Combin. Theory Ser. B*, Aceito para publicação.

[11] N. Robertson, P. D. Seymour, and R. Thomas. Permanents, Pfaffian orientations and even directed circuits. *Ann. of Math. (2)*, 150:929–975, 1999.

[12] G. Tesler. Matchings in graphs on non-orientable surfaces. *J. Combin. Theory Ser. B*, 78:198–231, 2000.

[13] W. T. Tutte. The factorization of linear graphs. *J. London Math. Soc.*, 22:107–111, 1947.

[14] W. T. Tutte. *Graph Theory as I Have Known It*. Number 11 in Oxford Lecture Series in Mathematics and its Applications. Clarendon Press, Oxford, 1998.

[15] V. V. Vazirani and M. Yannakakis. Pfaffian orientation of graphs, 0,1 permanents, and even cycles in digraphs. *Discrete Applied, Math.*, 25:179–180, 1989.

14. Um Protocolo de Roteamento Geográfico de Tempo Real para Redes de Sensores Sem Fio Multimídia

Autores: Cláudio S. de Carvalho e Edmundo R. M. Madeira

Rotear fluxos de vídeo de tempo real é uma tarefa complicada em redes [1]. Além de requerer uma grande largura de banda, um fluxo de vídeo de tempo real possui requisitos de qualidade de servico (QoS) bastante restritivos. Um fluxo requer que a entrega de pacotes seja feita com um baixo atraso e uma confiabilidade moderada, e a variacão de atraso (*jitter*) entre pacotes também seja baixa. Quando se trata de redes com recursos computacionais abundantes (por exemplo, as redes ópticas conhecidas pela sua alta largura de banda), esta tarefa se torna um pouco menos complicada, no entanto, em redes com recursos computacionais limitados (por exemplo, as redes de sensores sem fio), rotear fluxos de vídeo de tempo real se torna uma tarefa ainda mais desafiadora.

O objetivo geral deste trabalho de doutorado é investigar duas novas direções para se rotear geograficamente fluxos de vídeo de tempo real em redes de sensores sem fio (RSSF). Uma das direções é desenvolver uma arquitetura DiffServ simplificada para RSSF, com "reserva" de recursos. A outra direção é desenvolver um protocolo para se rotear pacotes com base no conceito de velocidade já conhecido, no entanto, investigando a diferenciação de pacotes, congestionamento, e o consumo de energia.

Qualidade de Serviço

Várias técnicas e arquiteturas foram projetadas para prover qualidade de serviço em redes. Em particular, as arquiteturas para serviços integrados (IntServ) [2] e para serviços diferenciados (DiffServ) [3] foram projetadas especificamente para o tratamento de fluxos multimídia em redes cabeadas. No entanto, nenhuma destas arquiteturas podem ser aplicadas diretamente em RSSF.

Ambas as arquiteturas IntServ e DiffServ são caracterizadas pela entrega de pacotes utilizando-se rotas com recursos reservados. A principal diferença entre elas está na forma em que os pacotes da rede são encaminhados ao longo das rotas. Na arquitetura IntServ, os roteadores realizam o encaminhamento com base no fluxo em que os pacotes pertencem, isto é, pacotes pertencentes a um mesmo fluxo são encaminhados pela mesma rota. Na arquitetura DiffServ, o encaminhamento é realizado com base em classes de serviço que expressam os requisitos a serem providos para um determinado conjunto de fluxos. Estes requisitos são definidos por um acordo (SLA) entre a aplicação ou domínio cliente e a rede provedora do serviço. Assim, na arquitetura DiffServ, cada roteador da rede encaminha os pacotes pertencentes a uma mesma classe de serviço pela mesma rota. Isto faz da arquitetura DiffServ mais escalável quanto ao número de fluxos presentes na rede do que a

arquitetura IntServ, embora a IntServ seja mais simples.

Redes de Sensores Sem Fio

Uma RSSF consiste de um grande número de nós pequenos, instalados em uma região de interesse e que colaboram para realizar tarefas comuns como monitoramento de ambiente, supervisão de áreas militares, e controle de processos industriais [4]. Estes nós são equipados com unidades limitadas para sensoriamento, processamento de dados, e comunicação; além de uma fonte de energia também limitada, tipicamente uma bateria.

As principais funcionalidades de um nó são detectar, processar e transmitir dados de eventos para um ponto de acesso da rede, o *sink*. A detecção é feita através de um ou mais sensores instaladados em nós espalhados na região de interesse. Dentre estes sensores estão inclusos sensores escalares como sísmico, magnético, térmico, infravermelho, acústico e radar; e sensores multimídia como câmera de vídeo e microfone. Embora o maior número de aplicações para RSSF utilizem sensores escalares, com o recente avanço da capacidade de transmissão de 256 kb/s [5] para até 10 Mb/s utilizando a camada física *Ultra Wide Band* (UWB) [6], as aplicações multimídia para RSSF se tornam cada vez mais promissoras.

Uma vez que a rede possui recursos limitados e os nós da rede são capazes de rotear e gerar dados (assim como nas redes Ad Hoc), a entrega de dados para o *sink* é realizada em múltiplos *hops*. Na forma mais simples, cada nó encaminha os pacotes recebidos para todos os seus nós vizinhos, que por sua vez realizam o mesmo processo até que os pacotes alcancem o *sink*. Uma outra opção é o roteamento geográfico [7]. Nesta alternativa, cada nó conhece a sua localização geográfica, a localização geográfica dos seus vizinhos, e a localização geográfica do *sink*. Assim, cada nó envia os pacotes recebidos para o nó vizinho que estiver mais próximo do *sink*, obtendo rotas mais curtas.

Proposta

Várias aplicações para RSSF possuem requisitos de tempo real. As direções de pesquisa propostas neste trabalho foram concebidas para aplicações de supervisão voltadas para RSSF com somente um *sink* e nós sensores equipados com uma câmera de vídeo e um sensor de presença. Um nó pode ter os dois tipos de sensores. Para economizar energia, somente nós com sensores de presença podem detectar eventos. Uma outra funcionalidade destes nós é acordar o nó sensor mais próximo com uma câmera para que a câmera grave um vídeo do objeto enquanto o objeto estiver se movimentando na área de cobertura da câmera. dirigida a eventos

Os dados da câmera e do nó sensor de presença são transmitidos para o *sink* somente quando um evento for detectado. No *sink*, estes dados devem ser analisados o mais rápido possível para que uma potencial tentativa de intrusão seja contida. Cada um dos tipos de dados possuem requisitos de QoS diferentes. Os fluxos de vídeo

obtidos pelas câmeras devem ser entregues ao sink com baixo atraso, confiabilidade moderada e baixo *jitter*. Já os dados obtidos pelos sensores de presença devem ser entregues com alta confiabilidade na tentativa de garantir que todo evento será reportado para o *sink*, e baixo atraso. O *jitter* não é importante para este tipo de dado.

A existência de mais de um tipo de dado na rede deixa clara a necessidade de se fazer diferenciação de pacotes. Quanto aos requisitos requeridos pela aplicação, o baixo *jitter* é um dos mais difíceis de ser provido e um dos mais importantes para os fluxos de vídeo. Se a aplicação receber um fluxo de vídeo com uma alta variação de atraso entre os pacotes, a qualidade de exibição do vídeo pode ficar comprometida. No contexto das redes cabeadas, a arquitetura DiffServ é uma alternativa bastante utilizada para se reduzir o *jitter*. Isto se deve ao uso de rotas com recursos reservados e ao encaminhamento diferenciado baseado em classes de serviço. No entanto, a arquitetura DiffServ é bastante complexa para ser diretamente aplicada em RSSF pois os recursos dos nós são limitados. Uma das direções de pesquisa propostas neste trabalho é investigar o desenvolvimento de uma versão simplificada da arquitetura DiffServ para RSSF. É importante que esta arquitetura atenda os requisitos mínimos de QoS da aplicação descrita e considere a limitação de recursos da rede. Inicialmente, a abordagem adotada é desenvolver um algoritmo de roteamento geográfico para "reservar" recursos em uma rota por um tempo determinado, evitando que a energia desta rota seja exaurida rapidamente. O tempo de vida da rota será calculado com base na taxa de amostragem do vídeo e na estimativa do tempo em que o evento permanecerá na cobertura do nó com sensor de presença que o detectou ou do nó sensor de vídeo que o está gravando.

Um outra maneira de se oferecer baixo *jitter* e provendo garantias de baixo atraso para todos os pacotes do fluxo de vídeo. O protocolo de roteamento geográfico de tempo real SPEED [8] é um exemplo. O SPEED não aplica os conceitos da arquitetura DiffServ, cada nó encaminha pacotes para o nó vizinho que estiver mais próximo do destino e que apresentar o menor atraso para o processamento de um pacote. Esta relação é introduzida como sendo velocidade pelo SPEED. A outra direção de pesquisa deste trabalho é obter baixo *jitter* utilizando o conceito de velocidade, no entanto, preocupando-se com a diferenciação de pacotes, a energia dos nós e o congestionamento.

A validação deste trabalho será feita no *Network Simulator 2* (NS2) utilizando uma implementação da camada de enlace definida pela especificação IEEE 802.15.4 e uma implementação recente da camada física UWB [9] definida pela especificação IEEE 802.15.4a. Ambas as especificações são voltadas para RSSF. O objetivo desta validação é verificar se os requisitos mínimos de QoS da aplicação foram satisfeitos e verificar o tempo que leva para o primeiro nó da rede ter sua energia exaurida.

Referências

[1] Andrew S. Tanenbaum *Redes de Computadores* Quarta Edição Traduzida, 2003.

[2] J. Wroclawski *The Use of RSVP with IETF Integrated Services* http://www.ietf.org/rfc/rfc2210.txt Setembro, 1997.

[3] S. Blake et. al. *An Architecture for Differentiated Services* http://www.ietf.org/rfc/rfc2475.txt Dezembro, 1998.

[4] Ian F. Akyildiz, W. Su, Yogesh Sankarasubramaniam, e Erdal Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, 2002

[5] IEEE 802.15.4 standard *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)* http://standards.ieee.org/getieee802/802.15.html

[6] IEEE 802.15.4a standard *Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: Add Alternate PHYs* http://standards.ieee.org/getieee802/802.15.html

[7] Brad Karp e H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. Em *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking*, páginas 243–354, Agosto 2000

[8] Tian He, John A. Stankovic, Chenyang Lu, e Tarek F. Abdelzaher. A spatiotemporal communication protocol for wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 16(10):995–1006, 2005.

[9] Ruben Merz, Jean-Yves Le Boudec, e Jörg Widmer *An Architecture for Wireless Simulation in NS-2 Applied to Impulse-Radio Ultra-Wide Band Networks 10th Communications and Networking Simulation Symposium*, Norfolk, VA, Março 25-29, 2007

15. Verification and Testing of Fault-Tolerant Software Architectures

Authors: Patrick H.S. Brito and Cecília M.F. Rubira (supervisor)

Introduction

The adoption of software components, which used to be restricted to the construction of enterprise systems, has expanded to other application areas where the cost of failure might be unacceptable. Software systems that can cause risks for human lives or great financial losses can be made fault-tolerant, so that they are capable of providing their intended operations, even if only partially, despite the presence of faults. Amongst the several existing techniques for building fault-tolerant systems, exception handling is a well-known mechanism for structuring error recovery in software systems [8]. The use of exception handling to develop large-scale software systems [9], together with the fact that it is implemented by several modern object-oriented languages, such as, Java, Ada, C#, and C++, and component models, such as, CCM, EJB, Ice, and .NET, confirms its importance to the current practice of software development. On the other hand, it is also accepted that the exception handling mechanism might have its disadvantages, if we consider the fact that a large part of a system's code is devoted to error detection and handling [8, 9].

To cope with the inherent complexity of the exception handling mechanism, it has been claimed that the abnormal behaviour should be systematically incorporated as early as possible in the software development process, specially during the requirements engineering and the architectural design [10]. Fault tolerance at the architectural level has received considerable attention, mostly in the context of fault handling. In particular, issues related to architectural reconfiguration that includes replacing, adding, removing architectural elements, or changing the topology of the configuration [2]. The same cannot be said about error propagation and error handling at the architectural level. In order to identify and remove faults related to the system's abnormal behaviour, verification and testing techniques should be used during the architectural design and implementation. Few contributions have exploited the verification of the abnormal behaviour at the architectural level. For instance, the work by Castor et al. [7] proposes a solution for specifying and verifying exception control flows at the software architecture using the Alloy specification language. However, their approach does not specify the behaviour of exception handlers as part of the verification process. Also, this solution does not scale very well when the verification process has to deal with many different types of exceptions [7].

Before verifying the software architecture, it is necessary to specify it using a formal notation. Architecture Description Languages (ADLs) are formal notations with the specific purpose of representing software architectures. Although these languages are normally considered intuitive, they lack on support for representing specific aspects of the system. Examples of such limitations concerns the representation of exception types, exception control flow involving architectural elements, scenarios of exception handling, and operations's signature for generating model-based test cases. To overcome these limitations, it is necessary to use a formal notation that allows the representation of distinct exception types. Moreover, for representing the chaining of exception control flows, conversion and masking, the formal notation should also support the specification of scenarios involving architectural elements.

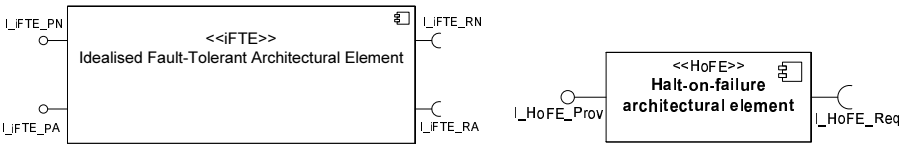
In this paper, we present a rigorous architectural approach for developing fault-tolerant software systems. This approach is based on architectural abstractions for the specification of fault-tolerant software architectures, and provides support for their verification and testing, thus aiming to improve the system dependability. Based on architectural abstractions, fault-tolerant software architectures can be described using stereotyped UML2.0, which can then be used as a basis for automatically generating the formal specification of the software architecture. This formal model

allows the formal verification of error handling properties, as well as the automatic generation of test cases for assessing the correctness of the final system. For the formal specification, we use a combination of B-Method [1] and CSP [6] for representing the structure and behaviour of the software architectures. Model checking is used for verifying the architectural properties of software fault tolerance, in particular, the signalling, propagation and handling of exceptions. Finally, unit and integration test cases are generated for assessing the implementation of the software architecture.

The rest of this paper is organised as follows. Section presents two architectural abstractions used by our approach. Section describes the rigorous development approach proposed for developing dependable component-based systems. Finally, Section presents a preliminary evaluation of the overall approach, as well as some concluding remarks and future directions of research.

Fault-Tolerant Architectural Abstractions

The *idealised fault-tolerant architectural element* (iFTE) is an architectural abstraction for structuring fault-tolerant systems. This abstraction enforces the principles associated with the concept of the idealised fault-tolerant component [1], and incorporates mechanisms for detecting errors, as well as propagating and handling them in a structured way. The iFTE abstraction provides an explicit separation of concerns between two types of behaviour: (i) the normal behaviour, which realises the services of the application, and (ii) the abnormal (exceptional) behaviour, which realises the detection, propagation and handling of errors. In order to provide this separation, the iFTE abstraction defines four types of interfaces, which are presented in Figure 7(a). While the I_iFTE_PN and I_iFTE_RN are responsible for the normal behaviour, I_iFTE_PA and I_iFTE_RA are responsible for the abnormal behaviour.



(a) iFTE Abstraction (b) HoFE Abstraction

Figura 7: Two Architectural Abstractions

The *halt-on-failure architectural element* (HoFE) is an architectural abstraction for the provision of error confinement and fault tolerance, and which enforces the principles associated with the *crash failures* fault model [5]. When an HoFE fails, it fails silently without producing any error signal. The HoFE abstraction defines two types of interfaces, which are presented in Figure 7(b). It is assumed that an HoFE is able to detect failures on other architectural elements from which requests operations, e.g., by associating *time-outs* with the I_HoFE_Req interfaces.

A Rigorous Development and Testing Approach

In our approach, the software architecture is considered a first-level unit, which guides the development from the specification to the implementation of the application. Figure 8 presents an overview of the proposed approach for developing fault-tolerant software architectures. Activity 1 specifies the software architecture, which can be done graphically using a CASE tool. From the use case abnormal scenarios, two artefacts should be specified: a UML component diagram representing the structure of the software architecture, and a set of UML sequence diagrams representing the abnormal architectural scenarios of exception control flows and handlers. For generating the architectural scenarios, the use case scenarios are refined according to the architectural configuration of the system. Activity 2 formally specifies the software architecture (architectural configuration and scenarios). This activity consists of an automatic model transformation from UML (XMI files) to B-Method and CSP. This transformation consists of instantiating formal templates with

the structural and behavioural specifications extracted from the UML models. These templates are provided as part of the solution and are available elsewhere [3, 5]. Activity 3 is the formal verification of the software architecture, in order to identify design faults related to the exception control flows and handlers. Activity 4, which consists of the generation of test cases and the system source code, is not detailed in this paper. The overall process presented in Figure 8 is considered recursive, since it can be executed either for the entire system, or for the internal structure of an architectural element.

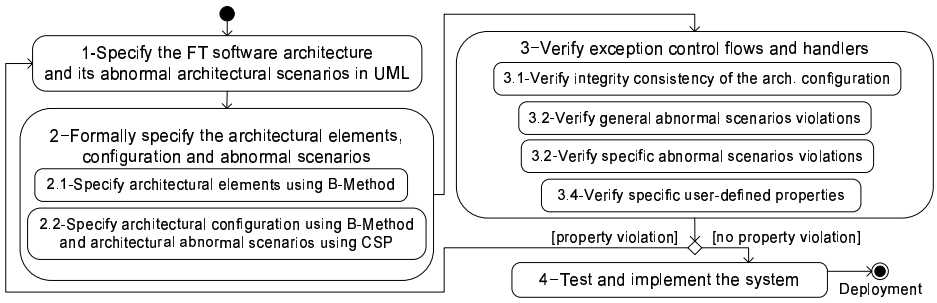


Figura 8: A process for developing abnormal behaviour

Conclusions

We have presented an architecture-centred solution for developing fault-tolerant software systems. Architectural abstractions are used to abstract away from system details while providing the means for analysing how errors are propagated, detected and handled. Fault-tolerant software architectures can be described using stereotyped UML2.0, which can then be used as a basis for automatically generating the formal specification of the software architecture. These formal models, which were not presented in this paper, allow the formal verification of error handling properties and the automatic generation of test cases for assessing the correctness of fault-tolerant software architectures. The preliminary results of our approach were evaluated by two case studies of critical applications: a mining control system [3, 4] and a real banking application [5].

A limitation of the proposed solution is the assumption that the communication between architectural elements follows a call-return protocol. Although this assumption simplifies the specification of software architectures, it lacks concurrency, which is essential in the context of a wide range of applications. To overcome this limitation, current work is looking on how to adapt our solution to support the specification of event-based software architectures. For this, the new formal framework has to resolve concurrent exceptions and coordinate scenarios involving the execution of concurrent components.

Referências

[1] J.-R. Abrial, M. K. O. Lee, D. Neilson, P. N. Scharbach, and I. Sorensen. The b-method. In *Proc. of the 4th International Symposium of VDM Europe on Formal Software Development (VDM '91) - Volume 2*, pages 398–405, London, UK, 1991. Springer-Verlag.

[2] J. S. Bradbury. Organizing definitions and formalisms for dynamic software architectures. Technical Report 2004-477, School of Computing, Queen's University, March 2004.

[3] P. H. S. Brito, R. de Lemos, E. Martins, and C. M. F. Rubira. Architecture-centric fault tolerance with exception handling. In *Proc. of the 3rd Latin American Symposium on Dependable Computing, LNCS 4746*, pages 75–94, 2007.

[4] P. H. S. Brito, R. de Lemos, and C. M. F. Rubira. Development of fault-tolerant software systems based on architectural abstractions, lncs. In *Proc. of Second European Conference on Software Architecture*, page (to appear). Springer-Verlag, Paphops, Cypros, 2008.

[5] P. H. S. Brito, R. de Lemos, and C. M. F. Rubira. Verification of exception control flows and handlers based on architectural scenarios. In *Proc. of 11th IEEE High Assurance Systems Engineering Symposium*, page (to appear). IEEE Press, Nanjing, China, 2008.

[6] S. D. Brookes, C. A. R. Hoare, and A. W. Roscoe. A theory of communicating sequential processes. *J. ACM*, 31(3):560–599, 1984.

[7] F. Castor Filho, P. H. da Silva Brito, and C. M. F. Rubira. Specification of exception flow in software architectures. *Journal of Systems and Software*, October 2006.

[8] F. Cristian. Exception handling. In *Dependability of Resilient Computers*, pages 68–97. Blackwell, 1989.

[9] D. Reimer and H. Srinivasan. Analyzing exception usage in large java applications. In *Proc. of ECOOP'2003 Workshop on Exception Handling in Object-Oriented Systems*, July 2003.

[10] C. M. F. Rubira, R. de Lemos, G. Ferreira, and F. Castor Filho. Exception handling in the development of dependable component-based systems. *Software – Practice and Experience*, 35(5):195–236, March 2005.

16. Técnicas de Análise e Otimização de Consumo de Energia para Sistemas Embarcados

Autores: Felipe Klein e Rodolfo Azevedo

Com a constante redução no tamanho dos transistores e o conseqüente aumento no número de transistores por *chip*, a potência dissipada pelos circuitos digitais está crescendo exponencialmente. Aliado a isto, a demanda por dispositivos portáteis, tais como PDAs, MP3 *players*, telefones celulares, entre outros, não pára de crescer e, a cada nova geração destes produtos, mais funcionalidades são agregadas que ocasionam o aumento de sua complexidade e consequentemente do consumo de energia.

Este aumento de complexidade foi previsto em meados da década de 70, na famosa "Lei de Moore", que estabeleceu que o número de transistores dobraria a aproximadamente cada 18 meses [1]. Uma das implicações do aumento da dissipação de potência é a redução do tempo de vida útil do circuito, pois o aumento da densidade de potência ($\frac{W}{cm^2}$) pode levar o circuito a falência.

Outro resultado do crescente aumento da dissipação de potência é o inevitável aumento do custo final do produto, devido a elaboração de soluções sofisticadas para o resfriamento do *chip*. Este é um fator importante, que pode tornar o produto inviável do ponto de vista do mercado. Uma outra implicação importante, principalmente para os dispositivos portáteis, geralmente alimentados por bateria, é que o aumento do consumo de energia pode reduzir drasticamente a autonomia do dispositivo, podendo assim inviabilizar o seu uso na prática.

Por isso, fica claro que o projeto de sistemas digitais visando a redução do consumo de energia se torna um elemento-chave no fluxo de projeto – o chamado *low power design*.

Porém, ao contrário de ferramentas para análise de desempenho e área, que têm um grande leque de opções disponíveis, tanto na indústria quanto no meio acadêmico, relativamente poucas opções estão disponíveis para a análise/otimização de potência em níveis mais altos de abstração.

Para os níveis mais baixos, como de transistor (*transistor-level*) e de portas lógicas (*gate-level*), há diversas opções de ferramentas de CAD[9], como o SPICE [2]. No entanto, estimar potência de circuitos médios e grandes nestes níveis pode ser uma tarefa excessivamente custosa em termos de tempo. Uma outra dificuldade é que estas ferramentas se localizam nos últimos estágios do ciclo de projeto, retardando a detecção de uma escolha de projeto mal-feita. Isso geraria novas rodadas completas no ciclo de desenvolvimento, até que as restrições de projeto fossem alcançadas.

Por isso, conforme a complexidade do sistema aumenta, é necessário prover ao projetista um meio de subir nos níveis de abstração, onde estão as maiores oportunidades de otimização para potência [3]. Sabe-se que a precisão absoluta das técnicas de estimativa de potência de alto nível são menos apuradas que as de baixo nível.

[9]CAD – Computer-Aided Design

Porém, mesmo uma estimativa grosseira em alto nível pode economizar muito tempo de projeto, dado que **gargalos de consumo** podem ser detectados mais cedo, acelerando o processo de otimização.

Aliado a isso, a pressão do mercado para um *time-to-market* cada vez menor tem evidenciado a tecnologia dos chamados SoCs[10], que permitem a integração de processadores, memórias e uma grande variedade de módulos de *hardware* e *software*, de forma a aumentar a produtividade (via reusabilidade) durante o projeto de arquiteturas voltadas para uma aplicação específica. Como conseqüência, cada vez mais técnicas de estimativa/otimização de potência são necessárias para uma análise apropriada de tais sistemas.

O projeto de doutorado que vem sendo desenvolvido, é uma ampliação e continuação do trabalho de mestrado [4] do mesmo autor, realizado no Instituto de Computação da UNICAMP, e concluído em abril de 2005.

Em [4] foi desenvolvida a biblioteca PowerSC, que estende a linguagem de descrição de sistemas SystemC [5]. A PowerSC permite que os módulos (IPs[11]) descritos em SystemC RTL[12] sejam analisados em função de potência, de uma maneira muito transparente para o projetista.

Uma outra vertente deste trabalho está focada no componente de software de sistemas embarcados, com o intuito de otimizá-los de forma a reduzir o consumo de energia. Este componente é complementar ao componente de hardware, que é devidamente tratado pela PowerSC.

A hierarquia de memória contribui significativamente para o consumo de energia [13] e, portanto, deve ser explorada de forma a se otimizar o consumo de energia. Em sistemas embarcados, é comum haver memórias muito pequenas na hierarquia de memória. Conhecidas como SPMs (*scratch-pad memory*) ou TCMs (*tightly-coupled memory*), estas memórias são geralmente utilizadas de forma complementar às *caches*, ou mesmo as substituindo completamente.

Além de um baixo custo em termos de espaço, estas memórias têm um baixo custo de energia para leitura/escrita se comparados à memória principal [14].

Um problema com relação à estas memórias é que, diferentemente das *caches*, deve existir um suporte explícito do compilador para seu uso efetivo. Ao contrário disto, as *caches* são transparentes para o programador. Todavia, dado seu baixo custo de energia, além da melhor previsibilidade de temporização em aplicações *time-constrained* [15] (típicas em sistemas embarcados), o esforço adicional de se criar o suporte às *scratch-pad memories* no compilador acaba compensando.

Neste trabalho são investigados métodos para explorar as SPMs em arquiteturas multi-processadas de forma dinâmica. Isto é, objetos de memória são selecionados para serem alocados na SPM de forma que possam compartilhá-la. Desta forma, durante a execução da aplicação, objetos de memória são copiados da memória principal para a *scratch-pad memory* (e vice-versa) de acordo com o seu uso. Mais

[10]SoC – System-on-Chip
[11]IP – Intellectual Property
[12]RTL – Register-Transfer Level

especificamente, a técnica desenvolvida explora as chamadas *virtually-shared SPMs* [16], existentes em algumas arquiteturas multi-processadas, as quais permitem que as SPMs sejam compartilhadas entre os processadores via um barramento especial de alta velocidade. Com isso, o espaço disponível para alocação para cada processador é virtualmente aumentado o que, por sua vez, resulta em maiores oportunidades de otimização.

SoCs são sistemas complexos e heterogêneos e, como não é difícil de notar, a análise de potência precisa de tais sistemas também é complexa. A interação entre os diversos componentes dos SoCs altera a forma como a energia é consumida no sistema. Como exemplo ilustrativo, considere um sistema com 2 processadores, 1 barramento e 1 memória. A concorrência pela memória aumenta a quantidade de *stalls* nos processadores, aumentando o consumo de energia do sistema em relação ao sistema monoprocessado. Outro fator agravante no consumo de energia é o aumento no número de transações no barramento, causado pela intensificação de tráfego de dados entre os dois processadores.

Em vista disso, uma metodologia eficiente que permita ao projetista modelar os diversos componentes (através de linguagens de descrição de arquiteturas [11], por exemplo) e, durante este tempo, efetuar *trade-offs* arquiteturais com relação à potência é essencial, mas ainda inexistente.

Referências

[1] Gordon E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8):114–117, 19 April 1965.

[2] L. W. Nagel. Spice2: A computer program to simulate semiconductor circuits. Erl-m520, Univ. California, Berkeley, 1975.

[3] Jan Rabaey. Low-power system design. In *EUROCHIP Course on Methods and Tools for Digital System Design*, 1995. Leuven, Netherlands.

[4] Felipe V. Klein. Powersc: Uma extensão de systemc para a captura de atividade de transição. Master's thesis, Instituto de Computação, UNICAMP, April 2005.

[5] Open SystemC Initiative. *SystemC Language Reference Manual*, revision 1.0 edition, 2003. See http://www.systemc.org.

[6] Paul E. Landman and Jan M. Rabaey. Activity-sensitive architectural power analysis. In *IEEE Transactions on Computer-Aided Design of Integrated Circuits*, pages 571–587. IEEE Computer Society Press, June 1996.

[7] Vivek Tiwari, Sharad Malik, and Andrew Wolfe. Power analysis of embedded software: a first step towards software power minimization. In *ICCAD '94: Proceedings of the 1994 IEEE/ACM international conference on Computer-aided design*, pages 384–390. IEEE Computer Society Press, 1994.

[8] Praveen Kalla, Jörg Henkel, and Xiaobo Sharon Hu. Sea: Fast power estimation for micro-architectures. In *ASP-DAC '03: Proceedings of the 2003 conference on Asia South Pacific design automation*, pages 600–605, 2003.

[9] Reinaldo A. Bergamaschi and Yunjian W. Jiang. State-based power analysis for systems-on-chip. In *DAC '03: Proceedings of the 40th conference on Design automation*, pages 638–641. ACM Press, 2003.

[10] F. Klein, R. Azevedo, and G. Araujo. High-level switching activity prediction through sampled monitored simulation. In *Proc. of International Symposium on System-on-Chip (SOC)*, Tampere, Finland, November 2005. Accepted for publication.

[11] Marcus Bartholomeu Sandro Rigo, Guido Araujo and Rodolfo Azevedo. Archc: A systemc-based architecture description language. In *to appear in the 16th Symposium on Computer Architecture and High Performance Computing - Foz do Iguacu, Brazil*, October 2004.

[12] Nikil Dutt, Alex Nicolau, Hiroyuki Tomiyama, and Ashok Halambi. New directions in compiler technology for embedded systems (embedded tutorial). In *ASP-DAC '01: Proceedings of the 2001 conference on Asia South Pacific design automation*, pages 409–414, New York, NY, USA, 2001. ACM Press.

[13] Manish Verma and Peter Marwedel. Advanced Memory Optimization Techniques for Low-power Embedded Processors. 2007 Springer.

[14] Rajeshwari Banakar, Stefan Steinke, Bo-Sik Lee, M. Balakrishnan and Peter Marwedel. Scratchpad memory: design alternative for cache on-chip memory in embedded systems. In *CODES '02: Proceedings of the tenth international symposium on Hardware/software codesign.*

[15] Lars Wehmeyer and Peter Marwedel Influence of Memory Hierarchies on Predictability for Time Constrained Embedded Software In *DATE '05: Proceeding of the Design Automation and Test in Europe.*

[16] Mahmut Kandemir, Ismail Kadayif, Alok Choudhary, J. Ramanujam and Ibrahim Kolcu. Compiler-directed Scratch Pad Memory Optimization for Embedded Multiprocessors. In *IEEE Transactions on Very Large Scale Integration Systems, March 2004.*

17. Branch-Cut-and-Price para o problema do m-anéis-estrelados capacitado

Autores: Edna A. Hoshino e Cid C. de Souza

No problema do m-anéis-estrelados capacitado (CmRSP), um conjunto de clientes deve ser visitado por m veículos inicialmente localizados em um depósito. Cada veículo realiza uma *rota* ou *anel* que começa e termina no depósito e é caracterizada por um conjunto ordenado de clientes e pontos de *Steiner*. Há, também, uma estrela associada a cada veículo. A *estrela* de um veículo t é um conjunto de *conexões*, definidas por pares da forma (u, v), onde u é um cliente e v é um cliente ou um ponto de *Steiner* pertencente ao anel de t. Cliente no anel-estrelado (isto é, no anel ou na estrela) de t é dito *coberto* por t e sua quantidade é limitada pela capacidade do veículo que é assumida ser a mesma para toda a frota de veículos. Uma solução para o CmRSP é definida por um conjunto de m anéis-estrelados cobrindo todos os clientes. Custos de roteamento incidem em todo par de elementos consecutivos no anel, enquanto custos de conexão ocorrem para toda conexão na estrela. O custo de uma solução é definido pela soma de todos os custos de roteamento mais os custos de conexão dos seus m anéis-estrelados. O CmRSP consiste em encontrar uma solução de custo mínimo e é um problema $\mathcal{NP}$-difícil uma vez que generaliza o problema do caixeiro viajante.

O CmRSP foi introduzido por Baldacci et al. [1], que descreve uma aplicação no projeto de redes de fibras óticas e um algoritmo de *branch-and-cut* (BC) para o problema. Nos experimentos reportados, instâncias de tamanhos moderados foram resolvidos em tempo razoável. Por outro lado, Mauttone et al. [2] propuseram uma heurística combinando GRASP e busca Tabu e obtiveram boas soluções para as mesmas instâncias testadas em [1]. Problemas correlatos incluem o problema do anel-estrelado (RSP) e suas variações [3, 4]. O RSP pode ser visto como um caso especial do CmRSP onde um único veículo não-capacitado está disponível. O problema de alocação e roteamento, *Vehicle Routing-Allocation Problem*, (VRAP) apresentado por Beasley e Nascimento em [5] é uma generalização do CmRSP, onde clientes podem não ser cobertos, mas, sujeitos a uma penalização na função objetivo. Atençao especial tem sido dada para uma variação do VRAP em que apenas um veículo está disponível e é conhecida por *Single Vehicle Routing-Allocation Problem* [6].

Em um trabalho anterior [7], uma formulação linear inteira para o CmRSP, baseada no modelo de cobertura, foi apresentada e um algoritmo *branch-and-price* para o problema foi proposto. O uso do método da geração de colunas para o CmRSP foi motivado pelo sucesso desta técnica na resolução do problema do roteamento de veículos capacitados (CVRP) [8], que pode ser interpretado como um caso especial do CmRSP. De fato, os resultados computacionais obtidos com esta técnica mostraram que o algoritmo BP é competitivo com o BC, que era o único algoritmo exato conhecido na literatura para o problema. Na média, o BP apresentou um

desempenho similar ao BC. Em algumas instâncias, o BP superou o desempenho do BC e vice-versa. Uma das principais contribuições daquele trabalho é o uso de uma estrutura chamada k-stream para fortalecer a relaxação do problema de geração de colunas, que no CmRSP também é um problema $\mathcal{NP}$-difícil. Essa idéia é similar àquela usada para o problema do caminho elementar com restrições de recursos [9, 10]. Também importante foi o uso de autômatos finitos determinísticos dentro da rotina para resolver o algoritmo de geração de colunas.

Neste trabalho, um novo algoritmo exato para o CmRSP é proposto. Trata-se de um *branch-and-cut-price* (BCP) combinando os algoritmos BC e BP. Algoritmos BCP misturam planos de cortes e geração de colunas em um algoritmo branch-and-bound para resolver problemas de programação linear inteira. Os experimentos mostram que o BCP é altamente competivo em relação ao BC. A diferença de performance é bastante expressiva especialmente quando aumenta-se o número de veículos e, consequentemente, a capacidade em cada veículo é diminuída. Como esperado, o limitante dual na raiz da árvore de branch-and-bound é mais apertado. Em alguns casos, esse limitante foi apertado o suficiente para fechar o gap de otimalidade na raiz.

Referências

[1] Baldacci, R., Dell'Amico, M., Salazar, J.: The capacitated m-ring star problem. Operations Research **55** (2007) 1147–1162

[2] Mauttone, A., Nesmachnow, S., Olivera, A., Robledo, F.: A hybrid metaheuristic algorithm to solve the capacitated m-ring star problem. In: International Network Optimization Conference. (2007)

[3] Labbé, M., Laporte, G., Martín, I.R., González, J.S.: The ring-star problem: Polyhedral analysis and exact algorithm. Networks **43** (2004) 117–189

[4] Dias, T., de Souza Filho, G., Macambira, E., Cabral, L., Fampa, M.: An efficient heuristic for the ring star problem. In: Experimental Algorithms WEA 2006. Volume 4007 of Lecture Notes in Computer Science., Springer (2006) 24–35

[5] Beasley, J., Nascimento, E.: The vehicle routing-allocation problem: A unifying framework. Trabajos de OPerativa **4** (1996) 65–86

[6] Vogt, L., Poojari, C., Beasley, J.E.: A tabu search algorithm for the single vehicle routing allocation problem. Journal of the Operational Research Society (2007) 467–480

[7] Hoshino, E., de Souza, C.: Column generation algorithms for the capacitated m-ring-star problem. In: Computing and Combinatorics: 14th Annual Inter-

national Conference, COCOON. Volume 5092 of Lectures Notes in Computer Science., Springer Berlin (2008) 631–641

[8] Fukasawa, R., Longo, H., Lysgaard, J., de Aragão, M.P., Reis, M., Uchoa, E., Werneck, R.: Robust branch-and-cut-and-price for the capacitated vehicle routing problem. Mathematical Programming **106**(3) (July 2006) 491–511

[9] Irnich, S., Desaulniers, G.: Shortest path problems with resource constraints. In: Column Generation. Springer (2005) 33–65

[10] Irnich, S., Villeneuve, D.: The shortest path problem with resource constraints and k-cycle elimination for $k \geq 3$. Informs Journal on Computing **18**(3) (2006) 391–406

18. Comportamento Autônomo de Multidões

Autores: Fernanda A. Andaló, Siome K. Goldenstein

A movimentação agregada de indivíduos (humanos ou animais, por exemplo) em grupos ou multidões é bastante complexa. Tal movimentação é dependente de muitos parâmetros que provêem características como sincronização, homogeneidade e unidade [1]. É de longa data a preocupação investida no entendimento e controle desta movimentação, sendo motivada, principalmente, pela necessidade de se modelar tais multidões em computador, por meio de simulações.

Entende-se multidão como um grande grupo de indivíduos em um mesmo ambiente físico, compartilhando um objetivo comum e podendo agir diferentemente do que quando estão sozinhos [2].

A importância de se modelar multidões no computador é crescente, pois existem várias áreas do conhecimento que necessitam de aplicações que envolvam simulações em tempo real ou não-real. Na indústria de entretenimento, simulações de multidões podem ser utilizadas na produção de animações e jogos de computador [3]. No treinamento policial e militar, simulações podem ser usadas para demostração e controle de rebeliões [4]. Em arquitetura, pode-se planejar e visualizar construções e cidades [5]. Na área de engenharia de segurança, simulações podem ser utilizadas para estudo de desocupação emergencial de construções, navios e aviões [6]. Na sociologia e psicologia, pode-se estudar o comportamento de multidões, analisar a relação entre diferentes pessoas, a hierarquia existente em grupos, a perda de individualidade, entre outros fatores [1]. Finalmente, na física pode-se estudar a dinâmica de multidões [7].

Em todas as áreas citadas, observa-se que a necessidade por simulações de multidões advém de duas situações que podem ocorrer no mundo real. Primeiramente, pode ser perigoso, para os indivíduos, realizar as ações objetivadas (como cair de um prédio em um filme, por exemplo). Segundo, é muito complexo e oneroso lidar com um grande número de indivíduos no mundo real [8]. Ambas as situações podem ser resolvidas pela simulação, no computador, da situação real.

A multidão simulada em computador é formada por agentes virtuais animados que operam de forma autônoma em ambientes. Por agente autônomo entende-se um sistema situado em um ambiente, e parte do mesmo, que percebe este ambiente e age sobre ele ao longo do tempo, seguindo seu próprio roteiro (plano) de modo a afetar o que será percebido no futuro [9].

A modelagem de um único agente autônomo, em sua forma elementar, pode ser trabalhosa. Considerando-se vários agentes, bem como as interações entre eles, a complexidade da modelagem cresce substancialmente. A complexidade é afetada por diversos fatores relacionados à necessidade de variedade na modelagem: variedade de visualização de agentes, de trajetórias individuais, de comportamentos, de animações e de reações às mesmas situações. Tais variações são obtidas modelando-se diferentes níveis de comportamento e de inteligência para os agentes e suas interações entre si e com o ambiente [10].

O comportamento de agentes ainda deve ser adaptativo em termos de tempo e espaço (mudanças contínuas no ambiente). Por último, o método de modelagem deve possuir boa escalabilidade com o crescimento da complexidade da geometria do ambiente, do número e inteligência dos agentes, e das várias interações dos agentes com o ambiente [11].

Atualmente, os desafios técnicos de tais simulações residem na crescente demanda por recursos computacionais que suportem simultaneamente: visualização de grande número de agentes, sistema para evitar colisão entre eles, interações agente-agente e agente-ambiente, além de, em alguns tipos de simulações, interação com os usuários [7].

Além de estudar modelos para melhor aproveitamento dos recursos computacionais, na tentativa de achar a solução para os desafios técnicos atuais (e muitas vezes com este objetivo), subáreas de simulação de multidões trabalham com [7]:

- geração do comportamento (como uma multidão virtual deve se comportar?);
- controle da movimentação (como entidades virtuais se movimentam pelo ambiente e evitam colisões?);
- integração de multidões em ambientes virtuais (quais aspectos do ambiente precisam ser modelados?);
- renderização (como visualizar, de maneira rápida, muitos agentes?);
- interação com multidões virtuais (como e quais informações devem ser trocadas entre agentes virtuais e reais?);
- geração de agentes virtuais (como gerar multidões heterogêneas?);
- criação de cenários (como criar cenas complexas de multidões eficientemente?)

Este trabalho está relacionado à uma combinação das subáreas apresentadas, abordando desde geração de agentes virtuais e comportamento a controle da movimentação. Todos os aspectos a serem estudados devem ser englobados em uma metodologia. Em particular, deve-se enfatizar a geração e aprendizado de comportamento, pois os agentes virtuais não possuem somente visão. Eles também devem possuir comportamento, percepção, memória e algum raciocínio, tornando-os autônomos e inteligentes.

Comportamento é definido como o jeito de agir de humanos e animais. Geralmente é descrito em uma linguagem natural que contém significância social, psicológica ou fisiológica, e que não é facilmente reduzida a movimentos de músculos e juntas. Sendo assim, modelar um comportamento não é apenas fazer o agente reagir ao ambiente, mas também incorporar o fluxo de informação pelo qual o ambiente age neste agente e como o agente codifica e utiliza esta informação [12]. Considerando estes aspectos, a metodologia que será proposta neste projeto deverá incorporar mecanismos de geração deste comportamento.

O objetivo deste trabalho é projetar esta metodologia com vários modos de operações para caracterizar propriedades estruturais distintas (como diferentes comportamentos de interação entre agentes). Além disso, pretende-se implementar sistema de simulação que utiliza a metodologia desenvolvida para fins de análise de resultado.

Pretende-se projetar um modelo de simulação que não possua as limitações existentes nos modelos atuais. Este modelo deve ser dinâmico e híbrido, ou seja, deve combinar dinâmica contínua e discreta. Este modelo híbrido discreto/contínuo deve conseguir combinar a teoria de sistemas híbridos e as abstrações naturais do comportamento de agentes autônomos.

A teoria formal de sistemas híbridos tem sido empregada em diversos domínios de aplicação, mas para animação não é tipicamente considerada. Geralmente as dinâmicas contínua e híbrida são combinadas, porém é pouco claro como tais sistemas se relacionam com a base teórica de sistemas híbridos.

Além disso, deve-se investigar a utilização de método físico para simulação (como as equações de Navier-Stokes) e representações não-paramétricas de distribuição de probabilidade para representações alternativas das multidões.

Referências

[1] S. R. Musse, B. Ulicny, A. Aubel, and D. Thalmann. Groups and crowd simulation. In *International Conference on Computer Graphics and Interactive Techniques - ACM SIGGRAPH 2005 Courses*, 2005.

[2] M. E. Roloff. *Interpersonal Communication - The Social Exchange Approach*, volume 6. SAGE Publications, London, 1981.

[3] C. Reynolds. Crowd simulation on PS3, 2006.

[4] M. D. Petty, F. D. McKenzie, R. C. Gaskins, and E. W. Weisel. Developing a crowd federate for military simulation. In *Proceedings of the Spring 2004 Simulation Interoperability Workshop*, pages 483–493, 2004.

[5] W. Shao and D. Terzopoulos. Populating reconstructed archaeological sites with autonomous virtual humans. In *Proceedings of the 6th International Conference on Intelligent Virtual Agents*, 2006.

[6] N. Courty and S. R. Musse. Simulation of large crowds in emergency situations including gaseous phenomena. In *Proceedings of the Computer Graphics International 2005*, pages 206–212, 2005.

[7] D. Thalmann, C. Hery, S. Lippman, H. Ono, S. Regelous, and D. Sutton. Crowd and group animation. In *International Conference on Computer Graphics and Interactive Techniques - ACM SIGGRAPH 2004 Course Notes*, 2004.

[8] C. O'Sullivan, J. Cassell, H. Vilhjálmsson, J. Dingliana, S. Dobbyn, B. McNamee, C. Peters, and T. Giang. Levels of detail for crowds and groups. *Computer Graphics Forum*, 21(4), 2002.

[9] S. Franklin and A. Graesser. Is it an agent, or just a program?: a taxonomy for autonomous agents. In *Proceedings of the Workshop on Intelligent Agents III, Agent Theories, Architectures, and Languages*, pages 21–35, 1996.

[10] S. Goldenstein, E. Large, and D. Metaxas. Non-linear dynamical system approach to behavior modeling. *The Visual Computer*, 15:349–364, 1999.

[11] S. Goldenstein, M. Karavelas, D. Metaxas, L. Guibas, E. Aaron, and A. Goswami. Scalable nonlinear dynamical systems for agent steering and crowd simulation. *Computer and Graphics*, 25(6):983–998, 2001.

[12] H. Noser, O. Renault, D. Thalmann, and N. M. Thalmann. Navigation for digital actors based on synthetic vision, memory, and learning. *Computers and Graphics*, 19(1):7–19, 1995.

19. Códigos Corretores de Erros e Reticulados Aplicados à Criptografia de Chave Pública

Autores: Rosemberg André da Silva (doutorando), Ricardo Dahab (orientador)

A obtenção de soluções polinomiais em computadores quânticos para problemas considerados intratáveis em computadores clássicos tem sido objeto de intensa pesquisa recentemente, com aplicações claras em cripto-análise. Numa outra vertente, várias áreas da matemática têm sido estudadas visando a formulação de problemas-candidatos que sejam intratáveis tanto do ponto de vista computacional clássico quanto do quântico, objetivando a utilização como base para criptossitemas. Neste contexto, a avaliação dos quesitos de segurança, de usabilidade e de complexidade dos sistemas obtidos faz-se particularmente relevante. Dentre as áreas da Matemática que se mostram mais promissoras para implementações de criptossistemas robustos a ataques quânticos destacam-se: Grupos não-Abelianos, Reticulados, Códigos Corretores de Erros, Sistemas Quadráticos Multivariáveis, e Núcleos e Perceptrons Permutados.

Alguns algoritmos criptográficos de chaves públicas baseados na Teoria de Códigos Corretores de Erros e Reticulados têm ganho popularidade por sua robustez a algoritmos quânticos, os quais prometem tornar inefetivos métodos já consagrados, tais como RSA e ECC. Desde 1994, é conhecido um algoritmo quântico capaz de resolver polinomialmente os problemas da fatoração de números inteiros e do logaritmo discreto, que são a base matemática de tais métodos clássicos [7]. No presente projeto de pesquisa de doutorado, pretende-se explorar os aspectos de análise de segurança e eficiência de implementações (com eventuais otimizações) que se utilizam de códigos e reticulados, buscarem-se provas de NP-dificuldade para problemas em aberto nesta área e derivarem-se algoritmos criptográficos seguros. Entre os problemas a serem analisados temos a decodificação de códigos lineares (como os de Goppa), o cálculo de vetores mais curtos num reticulado e a obtenção de vetores mais próximos a um alvo. Eles são comprovadamente NP-difíceis, mesmo para soluções aproximadas. Já servem, inclusive, como suporte a várias aplicações em criptogrfia, posto que não se conhecem ainda algoritmos que os resolvam num computador quântico com número polinomial de portas. Entretanto, há questões em aberto (listadas na seçao) que serão objeto de análise aprofundada.

Códigos Corretores de Erros

A *Teoria de Códigos* teve sua origem na década de 1940 com os trabalhos de Golay, Hamming e Shannon como ferramenta na solução de um problema de engenharia: correção erros em canais de comunicação ruidosos. Rapidamente, passou a fazer uso de técnicas matemáticas mais sofisticadas. Assim, de problemas mais simples, como os abordados na codificação e decodificação das famílias de códigos de Hamming, BCH, cíclicos e Reed-Muller, chegou-se a problemas mais avançados, como os

vistos em códigos de resíduos quadráticos, Golay, Goppa, alternantes, Kerdoc, Preparata, auto-duais, etc. Recentemente, o advento de algoritmos quânticos aos quais alguns códigos corretores de erros são ainda resistentes tem despertado o interesse da comunidade de criptografia.

Criptossistemas Baseados em Códigos Corretores de Erros

Criptossistema McEliece Utiliza códigos de Goppa em sua formulação. Foi proposto em 1978 por Robert McEliece. Tal criptossitema disfarça um código de Goppa (de fácil decodificação) como um código linear geral, de forma que a inversão de tal disfarce seja computacionalmente difícil. Em tal dificuldade reside a segurança do sistema. Um dos fatores que desencorajaram seu uso é o tamanho relativamente grande de suas chaves. Reduzi-las de forma a manter o sistema ainda seguro é um objeto de pesquisa. Mesmo com tal restrição, por não se conhecerem algoritmos quânticos que o quebrem, este criptossistema é de muito interesse no contexto de segurança da informação.

Criptossistema de Niederreiter Este criptossistema, a exemplo do anterior, também usa códigos de Goppa em sua formulação. Originalmente, utilizava códigos generalizados Reed-Solomon (GRS), contra os quais foram descobertas uma série de fragilidades. As seguranças dos criptossistemas McEliece e Niederreiter são equivalentes, apesar do primeiro incluir aleatorizações e o segundo ser determinístico.

Como visto acima, os códigos corretores de erros podem ser usados satisfatoriamente na construção de criptossistemas. Entretanto, nem todas as escolhas apresentam boas características de eficiência e segurança. Algumas instâncias têm parâmetros de segurança frágeis e devem ser evitadas. Um exemplo destes ó o criptossistema McEliece com máxima distância de rank (MRD), também conhecidos como códigos inseguros de Gabidulin [1]. Tem-se ainda como inseguro o Niederreiter com códigos GRS [4].

Reticulados

Um reticulado $\mathcal{L}$ é um sub-grupo aditivo discreto de $\mathbb{R}^n$, onde $n \in \mathbb{N}$. Caso $\mathcal{L} \subseteq \mathbb{Z}^n$, denomina-se reticulado inteiro. A base B de um reticulado inteiro $\mathcal{L} \subseteq \mathbb{Z}^n$ corresponde ao conjunto de vetores linearmente independentes $B = [b_1, b_2, ..., b_m]$, com $m \in \mathbb{N}$, cujas combinações lineares geram todos os elementos de $\mathcal{L}$. A dimensão de $\mathcal{L}$ é dada pelo número de elementos que formam uma de suas bases. A base B de um reticulado $\mathcal{L}$ pode ser representada na forma de matriz $\in \mathbb{R}^{n \times m}$. Dado um reticulado $\mathcal{L}$ de dimensão n, define-se como i-ésimo mínimo sucessivo $\lambda_i(\mathcal{L})$, para $i = 1, ..., n$ o menor número r tal que a bola com centro na origem e com raio r contenha i vetores linearmente independentes.

Dado um reticulado $\mathcal{L}$, define-se como problema de encontrar o vetor de menor comprimento (SVP) a tarefa de encontrar em $\mathcal{L}$ o vetor v mais próximo da origem, com $\|v\| \leq \gamma\lambda_1(\mathcal{L})$, onde γ é um fator de aproximação. Não são conhecidos algoritmos polinomiais que resolvam os problemas SVP (mesmo que aproximadamente) para $\gamma = n^{O(1)}$. Os melhores algoritmos polinomiais ([8]) apresentam aproximação $\gamma = 2^{O(n \log\log n/\log n)}$. Além disso, já foi demonstrado que aproximações por um fator constante são NP-difíceis [3]. Ainda não se sabe, no entanto, se a aproximação por fatores polinomiais também esteja em NP-difícil, permanecendo como simples conjectura. Da dificuldade de aproximar SVP resultaram algumas aplicações em criptografia, tais como cifragem e funções de hash. Dado um reticulado $\mathcal{L}$ e um vetor $v \in \mathcal{L}$, define-se como problema de encontrar o vetor mais próximo a v (CVP) a tarefa de encontrar em $\mathcal{L}$ o vetor u tal que $\|v - u\|$ seja mínimo. Este problema tem resultados similares aos de SVP em relação ao caráter NP-difícil das soluções exata e aproximada.

Criptossistemas Baseados em Reticulados

Criptossistema Ajtai-Dwork Este criptossistema foi o primeiro proposto com caso médio tão seguro quanto o pior caso [6]. Sua é segurança baseada no SVP único. Erros na decriptação foram posteriormente corrigidos por Goldreich, Goldwasser e Halevi [5]. De forma semelhante ao que foi visto para o caso de códigos corretores de erros, as chaves utilizadas ocupam espaço consideravelmente grande, da ordem de $O(n^4)$. Alguns ataques descritos por Nguyen e Stern [9] mostram como utilizar heurísticas baseadas no algoritmo LLL para explorar este criptossistema. Desta forma, para termos instâncias seguras de Ajtai-Dwork devemos usar chaves maiores que as descritas neste ataque.

Criptossistema Goldreich-Goldwasser-Halevi Diferentemente do criptossistema proposto acima, este baseia-se no CVP ao invés do SVP único [2]. Além disso, guarda algumas semelhanças com o criptossistema McEliece. Os tamanho das chaves empregadas são tipicamente $O(n^2)$ bits, podendo ir até $O(n^3 \log n)$. Um sério problema deste criptossistema reside no fato de que o texto cifrado vaza informações. Assim, a decriptação reduz-se a um caso especial CVP bem mais fácil de ser resolvido que o CVP geral [9]. Micciancio propôs uma extensão deste criptossistema empregando matrizes na forma hermitiana normal, o qual é seguro para entradas de tamanho $n \geq 780$.

Problemas em Aberto

Alguns candidatos a avaliação no escopo deste projeto quanto a aplicabilidade em criptografia estão listados a seguir: estudo de complexidade para os problemas SVP e CVP em diferentes tipos de reticulados (cíclicos, ideais); reduções de bases de reticulados; conexões entre teoria algébrica de números e reticulados; emprego de

códigos quase-cíclicos para redução do tamanho das chaves em criptosistemas baseados em códigos corretores de erros; construção de códigos quase-cíclicos com raios de cobertura de ordem semelhante aos dos códigos de Goppa.

Metodologia

A metodolgia utilizada na condução deste trabalho envolverá derivação de teoremas e algoritmos, com respectivas demonstrações de segurança e complexidade. Além disso, serão realizadas implementações visando estabelecer comparações com soluções existentes, permitindo avaliar o estado da arte através de aplicações concretas, complementando desta forma os valores assintóticos determinados teoricamente.

Referências

[1] J. K. Gibson. Severely denting the Gabidulin versionof the McEliece public key cryptosystem. In *Designs, Codes and Cryptography*, 1995.

[2] O. Goldreich, S. Goldwasser, S. Halevi. Public-key cryptosystems from lattice reduction problems. In *Crypto'1997*, LNCS 1294, 1997.

[3] D. Micciancio. Improving lattice based cryptosystems using the Hermitenormal form. In *CaLC'2001*, LNCS 2146, 2001.

[4] V. Sidelnikov, S. Shestakov. On cryptosystems based on generalized Reed-Solomon codes. In *DiskretnayaMat* 4(3), 1992.

[5] O. Goldreich, S. Goldwasser, S. Halevi. Eliminating Decryption Errors in the Ajtai-Dwork Cryptosystem. In *CRYPTO '97: Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology*, Springer-Verlag, 1997, pp. 105–111.

[6] M. Ajtai, C. Dwork. A public-key cryptosystem with worst-case average-case equivalence. In *roceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing (El Paso, Texas, United States, May 04 - 06, 1997).* ACM, New York, NY, 284-293.

[7] C. P. Schnorr, M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. In *Math. Programming* 66 (1994), 181-199.

[8] H. Alt, M. Habib, C. P. Schnorr. Lattice Reduction by Random Sampling and Birthday Methods. 2003.

[9] P. Q. Nguyen. A Note on the Security of NTRUSign. In *Ecole Nornale Supérieure & CNRS*, France, 2006.

20. Distribuição de Chaves Criptográficas em Redes de Sensores Sem Fio

Autores: Leonardo B. Oliveira e Ricardo Dahab

Redes de Sensores Sem Fio (RSSFs) [1] são um tipo particular de Redes Móveis Ad hoc (*Mobile Ad hoc Networks* – MANETs). Elas são compostas em sua maioria por pequenos nós (*nodes*) sensores cujos recursos (energia, largura de banda, processamento etc.) são extremamente limitados. Estes sensores, por sua vez, se conectam com o mundo externo por meio de dispositivos poderosos chamados de sorvedouros (*sink*) ou Estações Rádio Base (ERBs). Elas são utilizadas com o intuito de monitorar regiões, oferecendo dados sobre a área monitorada, também chamada de *área de interesse (interest area)* para o resto do sistema. Dentre sua vasta gama de aplicações estão operações de resgate em áreas de conflito e/ou desastre, espionagem industrial e detecção de exploração ilegal de recursos naturais. Ainda vale mencionar que, em 2003, ocorreu um *workshop* [16] patrocinado pelo *National Science Foundation* para identificar tópicos de pesquisa fundamentais em redes e a área de RSSFs foi um dos seis selecionados.

Embutir segurança em RSSFs é uma tarefa complexa e muito desafiadora. Ela é comumente justificada em RSSFs por causa das aplicações militares. Isto é, aplicações executadas durante batalhas e, portanto, na presença de adversários dos quais informações *sensoriadas* precisam ser protegidas. Acreditamos, contudo, que uma vez que as RSSFs comecem a ser empregadas em larga escala o emprego de mecanismos de segurança tornar-se-á mais e mais comum. Isso porque o sigilo deverá ser uma propriedade imperativa também em RSSFs rurais e urbanas. Por exemplo, fazendeiros e indústrias que lançarem mão das redes para monitorar sua cadeia de plantações e suprimento, respectivamente, desejarão manter os dados monitorados secretos, impedindo que os mesmos cheguem ao conhecimento de competidores. Ademais, autenticação – outra propriedade de segurança – poderá ser útil até mesmo em RSSFs domésticas, evitando que sensores de redes vizinhas interajam entre si acidentalmente.

Idealmente, um esquema de segurança para RSSFs tem que prover perfeita conectividade e resiliência. Em outras palavras, sensores devem ser capazes de (i) comunicar-se com quaisquer outros sensores de forma segura e (ii) os danos do comprometimento de um sensor devem ficar restritos ao mesmo – note-se que essas propriedades têm que ser satisfeitas mesmo para sensores que foram dispostos [13] em diferentes momentos. Ademais, o esquema deve ser de baixo custo tanto em termos de processamento, como de comunicação e armazenamento.

Segurança, por sua vez, é comumente alavancada (*bootstrapped*) através de esquemas de distribuição de chaves. O baixo poder computacional dos sensores, contudo, inviabiliza o emprego de criptossistemas assimétricos, também chamados de sistemas baseados em Criptossistemas de Chave Pública (*Public Key Cryptosystem* – PKC,

[13] Neste documento, empregamos o verbete dispor como tradução do inglês *deploy*.

convencionais (RSA, por exemplo) e, até recentemente, propriedades de segurança eram alcançadas por meio de criptossistemas simétricos (RC5, SkipJack etc. [3]) em RSSFs.

Apesar de mais eficientes, criptossistemas simétricos possuem algumas inconveniências. Antes de tudo, as partes que desejam comunicar-se de forma segura precisam, *a priori*, decidir por uma chave em comum e então compartilhar essa chave através de um canal seguro. Logo, o primeiro (e maior) desafio é encontrar tal canal, já que a necessidade do compartilhamento de chaves advém justamente da necessidade de se viabilizar um canal seguro. Este problema é ainda pior em RSSFs, pois ao contrário das redes convencionais, em que existem canais alternativos (telefone, correio, múltiplas rotas etc.) que podem ser usados para a troca de chaves, o único canal existente em RSSFs é o enlace sem fio. A segunda dificuldade é que para se comunicar de forma segura, um indivíduo – idealmente – deveria compartilhar uma chave distinta com cada um dos outros participantes da rede. Isso causa um sério problema de escalabilidade, uma vez que em geral RSSFs são compostas por centenas ou milhares de dispositivos sensores. Uma alternativa, é verdade, seria vários sensores compartilharem uma mesma chave, mas, neste caso, a violação de um único sensor comprometeria todo o grupo. Por fim, o compartilhamento de uma mesma chave por mais de um indivíduo em criptossistemas simétricos possibilita ataques de refutabilidade (*repudiation*) e personificação (*spoofing*).

É bem verdade que existem esquemas de pré-distribuição de chaves baseados em criptossistemas simétricos (e.g., [3,11,14,15]) especialmente projetados para RSSFs. Mesmo eles, contudo, possuem pontos fracos. Isto é, embora sejam apropriados para as aplicações e arquiteturas para os quais foram concebidos, não são adequados a outras. Tais esquemas oferecem um compromisso entre conectividade e robustez, mas não fornecem um nível ideal de ambos. Além disso, a maioria dos esquemas depende de alguma interação entre os sensores para efetuar o acordo de chaves (*key agreement*).

Mais recentemente, descobriu-se que PKCs alternativos são viáveis em RSSFs [6, 12]. Já que nesses sistemas as partes comunicantes possuem apenas um par de chaves, PKCs são escaláveis e simples de ser utilizados. Tal conveniência, entretanto, tem custo: um esquema de autenticação das chaves públicas precisa ser fornecido. E autenticação de chaves, por sua vez, seja tradicional (PKI e/ou certificados) ou especialmente voltada para RSSFs (como em [13]), resulta em sobrecarga (*overhead*) – o que vai particularmente de encontro aos quesitos de RSSFs.

Resumindo, dotar RSSFs de segurança é uma tarefa especialmente desafiadora e fundamental para a ampla adoção da tecnologia de RSSFs. Em nossa tese propusemos diferentes soluções de segurança, cujos focos principais são as distribuições de chaves. Como será mostrado em nossa apresentação no *workshop*, iniciamos nosso trabalho com soluções personalizadas para certas arquiteturas de RSSFs e evoluímos para soluções flexíveis em que a segurança é alavancada de forma não interativa, o que é ideal para este tipo de rede. Até onde sabemos, nosso trabalho é pioneiro em soluções de segurança para RSSFs hierárquicas e o primeiro a realizar distribuição

de chaves não interativa usando Criptografia Baseada em Emparelhamentos.

Referências

[1] Deborah Estrin, Ramesh Govindan, John S. Heidemann, and Satish Kumar. Next century challenges: Scalable coordination in sensor networks. In *Mobile Computing and Networking (MobiCom'99)*, p. 263–270, Seattle,1999.

[2] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2–3):293–315, 2003. Also apeared in 1st IEEE International Workshop on Sensor Network Protocols and Applications.

[3] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and J. D. Tygar. SPINS: Security protocols for sensor networks. *Wireless Networks*, 8(5):521–534, 2002.

[4] V. Miller. Uses of elliptic curves in cryptography, advances in cryptology. In *Crypto'85, Lecture Notes in Computer Science*, volume 218, pages 417–426. Springer-Verlag, 1986.

[5] N. Koblitz. Elliptic curve cryptosystems. *Mathematics of computation*, 48:203–209, 1987.

[6] N. Gura, A. Patel, A. Wander, H. Eberle, and S. C. Shantz. Comparing elliptic curve cryptography and rsa on 8-bit cpus. In *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, p. 119–132, 2004.

[7] R. Sakai, K. Ohgishi, and M. Kasahara. Cryptosystems based on pairing. In *Symposium on Cryptography and Information Security(SCIS'00)*, p. 26–28,2000

[8] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. *SIAM J. Comput.*, 32(3):586–615, 2003. also appeared in CRYPTO '01.

[9] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO'84: on Advances in cryptology*, p. 47–53. Springer-Verlag, 1984.

[10] L. B. Oliveira and R. Dahab. SecLEACH – a random key distribution solution for securing clustered sensor networks. In *5th IEEE International Symposium on Network Computing and Applications*, Cambridge, MA, 2006. fast abstract.

[11] Leonardo B. Oliveira, Adrian Ferreira, Marco A. Vilaça, Hao Chi Wong, Marshall Bern, Ricardo Dahab, and Antonio A. F. Loureiro. SecLEACH– on the security of clustered sensor networks. *Signal Process.*, 87(12):2882–2895, 2007.

[12] David J. Malan, Matt Welsh, and Michael D. Smith. A public-key infrastructure for key distribution in tinyos based on elliptic curve cryptography. In *1st IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON'04)*, 2004.

[13] Wenliang Du, Ronghua Wang, and Peng Ning. An efficient scheme for authenticating public keys in sensor networks. In *6th ACM MobiHoc '05*, pages 58–67, New York, 2005.

[14] Sencun Zhu, Sanjeev Setia, and Sushil Jajodia. LEAP: efficient security mechanisms for large-scale distributed sensor networks. In *10th ACM conference on Computer and communication security (CCS'03)*, pages 62–72. ACM Press, 2003.

[15] Laurent Eschenauer and Virgil D. Gligor. A key management scheme for distributed sensor networks. In *9th ACM conf. on Computer and communications security (CCS'02)*, pages 41–47, 2002.

[16] Report of the national science foundation workshop on fundamental research in networking, April 2003. http://www.cs.virginia.edu/ jorg/workshop1.

21. Aplicação de algoritmos evolutivos na geração automática de dados de teste de conformidade

Autores: Thaise Yano, Eliane Martins e Fabiano Luís de Sousa

A atividade de teste tem sido apontada como uma das mais onerosas no desenvolvimento de software. Um dos desafios mais relevantes na pesquisa de teste de software é a geração de dados de teste. E nesse contexto, existe a promissora relação entre o teste de software e outras áreas de pesquisa, tal como o uso de abordagens baseada em busca [3, 4] na geração de dados de teste. Outro desafio é o teste baseado em modelos, sendo que a idéia principal é usar modelos no desenvolvimento de software para auxiliar o processo de teste, particularmente, na geração automática de casos de teste.

Dessa maneira, um problema importante na atividade de teste de software é o desenvolvimento de dados de teste, que consiste em identificar dados de entrada que satisfaçam um determinado critério de teste. Automatizar a geração de dados de teste contribuiria para a redução de custo e esforços ao processo de desenvolvimento de software. Pela abordagem dinâmica, a geração de dados de teste é reformulada como um problema de otimização [5]. Quando um requisito de teste não é satisfeito, uma função objetivo é associada a ele e métodos de otimização de funções são utilizados para buscar dados que mais se aproximam de satisfazer o requisito. Com base nessas informações, os dados de teste são incrementalmente modificados até que um deles satisfaça o requisito. Como métodos de otimização utilizados nessa abordagem incluem o recozimento simulado, método do gradiente, *hill climbing* e algoritmos evolutivos.

A área de pesquisa que investiga o uso de algoritmos evolutivos na atividade de teste é denominada como Testes Evolutivos. A idéia geral consiste em otimizar uma determinada entrada de acordo com o critério de teste expresso em uma função objetivo. Os indivíduos sobre os quais o processo de otimização ocorre constituem os casos de teste. Nos últimos anos houve um grande crescimento no interesse por essa área, sendo aplicada para diferentes técnicas e critérios de teste. Muitos trabalhos focam o teste estrutural, mas também abordam com menor intensidade o teste funcional [4]. Entre os algoritmos evolutivos mais utilizados na geração de dados de teste está o algoritmo genético (AG). Porém outros algoritmos têm sido propostos para o teste evolutivo que apresentam a vantagem de possuir menos parâmetros a serem ajustados que o AG, tal como o GEO (*Generalized Extremal Optimization*) [2]. Isso facilita o processo de configuração do algoritmo para obter um melhor desempenho. No trabalho de Abreu [1] foi explorado pela primeira vez o uso do GEO em uma atividade da engenharia de software. Os estudos de casos mostraram que o GEO também é competitivo com o AG na geração de dados de teste estrutural, exigindo menos esforço computacional para tal. Isso motivou a aplicação do GEO como uma opção interessante a ser utilizada na geração de dados de testes funcionais. Então neste trabalho é realizada uma investigação do algoritmo

GEO no teste de conformidade, tendo MEF (Máquina de Estados Finitos) como especificação.

O critério de teste adotado é cobrir um determinado conjunto de transições T_{alvo}, que podem ser não seqüenciais, em um único caso de teste. O motivo pela escolha desse critério, ao invés de cobrir todas as transições, é que pretende-se gerar seqüências que cubram transições que sejam críticas ao sistema em teste. Por exemplo, pode-se testar as transições que correspondem às entradas inválidas para realizar o teste de robustez do sistema. Esse critério de teste é codificado em uma função objetivo para a aplicação dos algoritmos evolutivos na atividade de teste. Vale ressaltar a importância da função objetivo, uma vez que captura as informações cruciais para a otimização, diferenciando uma boa solução de uma ruim [3]. A função objetivo definida neste trabalho verifica as transições disparadas por uma seqüência de entrada *seq* de uma MEF M e calcula o número de transições n comuns a T_{alvo}. O valor de aptidão de *seq* é definido como: $n/|T_{alvo}|$, em que $|T_{alvo}|$ é a cardinalidade do conjunto T_{alvo}. Assim os valores da função objetivo estão no domínio de [0,1].

Na Figura 9 é apresentada uma ilustração do procedimento de geração dos dados para o teste de conformidade. É utilizada a ferramenta SMC[14] (*State Machine Compiler*) capaz de gerar um código que faz a simulação de uma máquina de estados nas linguagens Java, C++ e Perl, por exemplo. Como primeiro passo, uma MEF M é passada para a ferramenta SMC que, por sua vez, gera um código Java P que simula essa máquina. O código gerado é instrumentado para informar em quais transições uma seqüência de entradas passou. Em seguida, dado um conjunto de transições T_{alvo}, o GEO começa a gerar dados de teste tentando cobrí-lo. Cada dado gerado é uma seqüência de entrada para M. O caminho coberto pela seqüência é obtido graças à instrumentação inserida em P e será entrada para a função objetivo que calculará a cobertura desse caminho em relação a T_{alvo}. O critério de parada do GEO é (i) quando se atinge um número máximo de avaliações da função objetivo ou (ii) quando todo conjunto T_{alvo} for coberto. Caso o critério de parada não seja satisfeito, o GEO reinicia o processo de geração de dados.

Figura 9: Ilustração da abordagem utilizada.

Um estudo de caso da abordagem foi feito para o protocolo WTP (*Wireless Transaction Protocol*), uma das camadas do WAP (*Wireless Application Protocol*) cujo objetivo é oferecer acesso à Internet para dispositivos móveis, como telefone celular. WTP é um protocolo de transação confirmada que provê os serviços essenciais para aplicações interativas pedido/resposta. A fim de testar o receptor das transações,

[14]Disponível em http://smc.sourceforge.net.

uma MEF M_1 foi construída considerando apenas suas operações. Embora exista um fluxo de dados no WTP, tais como variáveis e parâmetros, somente o fluxo de controle foi levado em consideração até o momento na MEF. A máquina é determinística, reiniciável e parcialmente especificada, ignorando-se eventos de entrada que não foram especificados para um estado e mantendo-a no mesmo estado. A MEF M_1 possui 6 estados e 45 transições. O conjunto de transições a serem cobertas é aquele correspondente às exceções especificadas na documentação do protocolo e constitui u m total de 25 transições em M_1.

O processo de otimização no GEO é sobre um conjunto de variáveis de projeto. Neste trabalho cada variável de projeto representa um evento de entrada $x \in I$ da máquina. Em M_1, existem 23 eventos de entrada e as variáveis de projeto estão no intervalo de inteiros [0, 22]. Assim o GEO busca encontrar uma seqüência de eventos de entrada que cubra T_{alvo}. O tamanho da seqüência de teste gerada é determinado pelo número de variáveis de projeto. Antes de iniciar o processo de geração de dados de teste, é recomendável fazer um ajuste fino do único parâmetro ajustável do GEO τ, visto que para cada problema existe um que torna a busca mais eficiente. Por exemplo, se $\tau \to \infty$, somente o elemento menos adaptado sofrerá mutação em cada iteração do algoritmo, o que é equivalente a uma busca totalmente determinística. Por outro lado, se $\tau \to 0$, qualquer elemento escolhido como candidato sofrerá mutação.

A fim de avaliar a aplicação da abordagem, o GEO foi comparado com o teste aleatório. A comparação do algoritmo GEO e o teste aleatório foi realizada com base na evolução dos valores da função objetivo em relação ao número de avaliações da mesma. É importante ressaltar que a função objetivo permite avaliar a cobertura de T_{alvo} por um caso de teste. Um experimento de 20 execuções com máximo de 100000 avaliações da função objetivo foi feito para diferentes números de variáveis de projeto (32, 64 e 128). Foram utilizadas na comparação diferentes implementações do GEO: GEOcan, GEOvar e GEOdis. A primeira é o algoritmo canônico na qual a variável de projeto é codificada por um conjunto de valores binários. O GEOvar tem um processo de mutação diferente ao GEOcan. O GEOdis é como o canônico mas cada variável de projeto é um valor discreto. Todas as implementações do GEO foram executadas com τ igual a 4, ao se observar que para este problema o GEO deve ser mais determístic o do que aleatório. O resultado da comparação com 128 variáveis de projeto é mostrado na Figura 10. Pode-se notar que o GEOdis teve o melhor desempenho, seguido pelo GEOcan e ambas implementações obtiveram melhor resultado que o teste aleatório. No experimento pode-se observar também que quanto maior o número de variáveis de projeto, melhor é o desempenho do GEO. Isso deve-se ao fato da MEF ser reiniciável e assim o caminho percorrido pela seqüência de eventos de entrada gerada é como se fosse consistuído por diferentes caminhos na máquina que se iniciam e terminam no estado inicial. Consequentemente, uma seqüência formada por um número maior de variáveis de projeto possibilita percorrer um número maior desses caminhos da máquina.

Este trabalho é o passo inicial para a investigação do algoritmo GEO no teste

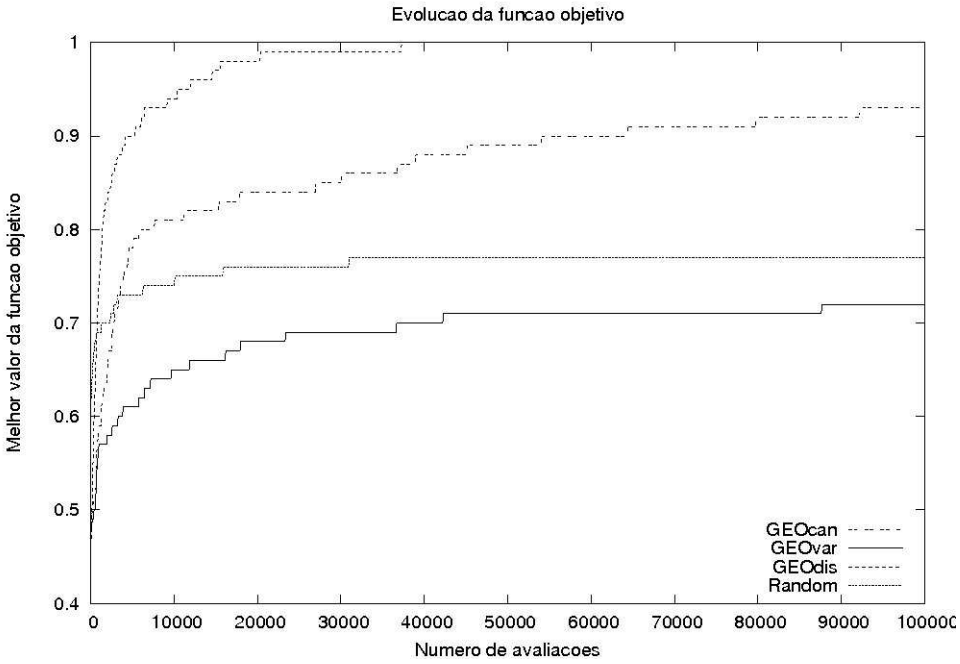


Figura 10: GEO X Teste aleatório.

baseado em modelo. Pretende-se utilizar MEF estendida (MEFE), que permite representar o fluxo de dados de um sistema, ao invés do modelo clássico que apenas representa o fluxo de controle.

Referências

[1] B. T. Abreu. Uma abordagem evolutiva para a geração automática de dados de teste. Master's thesis, IC/Unicamp, Campinas, SP, 2006.

[2] F. L. DeSousa. *Otimização Extrema Generalizada: Um novo algoritmo estocástico para o projeto ótimo*. PhD thesis, INPE, São José dos Campos, SP, Brasil, 2002.

[3] M. Harman. The current state and future of search based software engineering. In *Future of Software Engineering 2007*. IEEE Computer Society, 2007.

[4] P. McMinn. Search-based software test data generation: a survey. *Software Testing, Verification and Reliability*, 14(2):105–156, 2004.

[5] C.C. Michael, G. McGraw, and M. A. Schatz. Generating software test data by evolution. *IEEE Transactions on Software Engineering*, 27(12):1085–1110, December 2001.

22. Requisitos de Interação para o Design de Interfaces para Todos

Autores: Vania Paula de Almeida Neris and Maria Cecília Calani Baranauskas

Introdução

Atualmente, diversos serviços vêm sendo oferecidos à população via computadores e internet como: pagamento de contas, comunicação com amigos e com instituições públicas e privadas, procura de empregos, acesso a informações, entre outros. No entanto, apesar da redução dos preços dos computadores, da implantação de telecentros e *lan houses* e da disseminação de telefones celulares, a grande maioria da população brasileira ainda não se beneficia desses serviços. O que se percebe é que as interfaces de usuário, da maneira como são concebidas hoje, não favorecem a interação da população de maneira geral ao não considerar as diferentes necessidades dos usuários presentes na população.

Nesse contexto, a tese relacionada se propõe a investigar como desenvolver interfaces de usuário que atendam às diferentes possibilidades de interação, seguindo os preceitos do Design para Todos [3]. Um dos caminhos que se apresenta é desenvolver interfaces que sejam ajustáveis e que possam propiciar o acesso aos serviços à maior extensão possível de cidadãos. Assim, o objetivo da tese é formalizar e validar um *framework* que apóie o projeto de interfaces ajustáveis de acordo com requisitos de interação de usuários com competências diversas. Como contexto para a pesquisa estamos considerando a diversidade presente na população brasileira, incluindo idosos e pessoas com baixo letramento. O termo *framework* é utilizado aqui no seu sentido mais amplo como uma estrutura composta por diretrizes, mecanismos, artefatos e sistemas usados no planejamento e na tomada de decisões de design.

A formalização e validação do *framework* terão como base dois estudos de caso, nos quais grupos de usuários estão envolvidos em atividades participativas para o entendimento da diversidade de competências de usuários relacionadas à temática de ajuste de software. Neste resumo, apresentamos resultados preliminares obtidos com a realização de algumas dessas atividades: uma caracterização desses usuários e o levantamento de requisitos de interação que se espera contemplar em uma solução ajustável.

Demandas da diversidade cultural

As demandas de interação que serão apresentadas foram identificadas por meio de atividades desenvolvidas com um conjunto de usuários, denominado Cenário*, que pode ser entendido como um microcosmo da população, uma vez que foi constituído levando-se em conta a diversidade sócio-cultural da população brasileira. Com base nos números do IBGE, foram verificadas as porcentagens da população referentes a

gênero, idade, grau de letramento e renda familiar e esses números foram utilizados para selecionar usuários participantes. Mais detalhes sobre a formação do Cenário* podem ser encontrados em [1].

As atividades desenvolvidas incluíram: **StoryTelling** - participantes organizaram-se na sala em círculo, e cada um contou algumas de suas experiências de sucesso e de fracasso, ligadas ao uso de algum tipo de tecnologia; **Jogo dos painéis** - grupos de usuários deveriam representar uma frase usando figuras e palavras. Os participantes dos outros grupos deveriam identificar qual era a frase original e ganharia pontos tanto o grupo de conseguisse transmitir a mensagem (3 pontos), como o grupo que conseguisse acertar a mensagem transmitida (1 ponto); **Interação com sistemas de login** - quatro protótipos para cadastro de usuários foram criados por alunos de uma disciplina de Interação Humano-Computador (IHC) e levados para teste no Cenário*. Um dos protótipos era todo textual e com campos para preenchimento padrão. Outros faziam maior uso de som e imagens. Alguns utilizaram Língua Brasileira de Sinais e outro adaptou algumas das teclas do *laptop* para melhor reconhecimento pelos digitalmente iletrados.

Usuários e suas habilidades

As atividades realizadas no Cenário* permitiram observar que além da dificuldade de manuseio da tecnologia, pela falta de conhecimento do "alfabeto digital", isto é, a lógica de funcionamento dos computadores, a não identificação com o mapeamento do mundo real expresso na aplicação traz severas complicações à interação. A Figura 1 apresenta uma caracterização em quadrantes que relaciona o conhecimento do domínio com a habilidade com o uso de tecnologia.

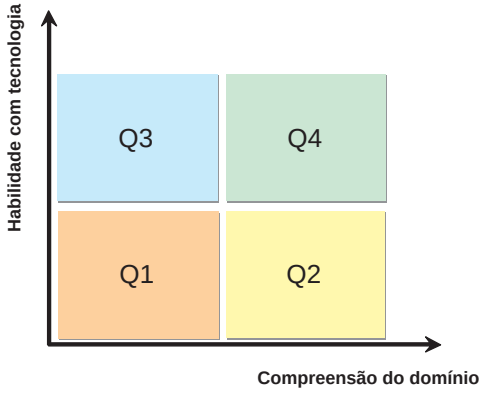


Figura 11: Caracterização de usuários em quadrantes.

Os usuários do Q1 são os menos favorecidos em relação à interação com interfaces computacionais. Além de não terem conhecimento do "alfabeto digital" (uso do teclado, mouse, janelas, botões, ícones, menus etc), também apresentam dificuldades para compreender o mapeamento atual do mundo real para as aplicações computacionais. Normalmente são as pessoas com idade mais avançada, maior incidência

do gênero feminino, com dificuldades de visão, falta de destreza com o mouse, apresentam problemas de leitura, muitas vezes trocando palavras ou não terminando a leitura.

Os usuários do Q2 têm maior compreensão do domínio do que os do Q1, mas também não detêm conhecimento sobre o alfabeto digital. Normalmente, são pessoas entre 45 e 60 anos, com menor incidência de problemas de visão, sem habilidades com o mouse e apresentam problemas não tão sérios de leitura quando comparados aos do Q1. Apesar de não conseguirem interagir devido às dificuldades operacionais, esses usuários detêm informações do contexto, tendo alguma orientação do que é possível encontrar nas aplicações computacionais.

O Q3 reúne os usuários que já utilizam computadores em suas residências ou telecentros, para atividades esporádicas, normalmente comunicação com amigos e acesso a informações. Normalmente são pessoas com menos de 45 anos, já conseguem ter domínio sobre o mouse e não apresentam problemas de leitura das opções de interação (no entanto, evitam ler textos longos). Não utilizam serviços bancários ou de compras por terem receios relacionados à segurança das operações. Detêm conhecimento sobre o alfabeto digital, mas não utilizam ou utilizam pouco as teclas de atalho ou navegação via teclado.

O Q4 reúne os usuários que utilizam os computadores com freqüência. Têm domínio do mouse e utilizam teclas de atalho. Realizam compras pela Internet e conhecem algumas políticas de segurança. Entendem o mapeamento do mundo real para o digital, muitas vezes questionando opções feitas pelos designers e apontando problemas de usabilidade e acessibilidade.

Requisitos e diretivas de interação

Considerando a caracterização apresentada, as atividades desenvolvidas no Cenário* e estudos de IHC, foi possível levantar requisitos de interação e apontar diretivas que visam auxiliar designers. A Tabela 1 exemplifica alguns requisitos e aponta algumas sugestões de como endereçá-los para facilitar a interação dos usuários. Um conjunto maior pode ser encontrado em [4].

Vale destacar que os requisitos e diretivas aqui não são condição única e suficiente para garantir a interação dos usuários dos respectivos quadrantes. Os requisitos e diretivas têm o objetivo de auxiliar designers refletindo resultados de práticas experimentadas no Cenário*. Também em pesquisas que envolvem práticas participativas, uma questão importante é determinar o número de integrantes das práticas e a representatividade dos mesmos. Considerando as premissas do Design Participativo, entendemos que um conjunto de 10 a 15 participantes, reunidos segundo estatísticas que representam a população, tem tamanho e representatividade adequados para embasar os resultados preliminares aqui apresentados. Para mais detalhes sobre a formação de grupos de usuários para atividades participativas de design veja [1].

Tabela 1: Exemplos de Requisitos e Diretivas de Interação.

Requisitos e Diretivas de Interação	**Usuários**
Instruções sonoras devem ser disparadas assim que uma nova página é carregada, bem como o correspondente em LIBRAS no caso do usuário surdo.	Q1
O fundo de caixas de texto deve ter cor que contraste com a cor de fundo da interface.	Q1
Opção para apagar e espaço devem estar ressaltadas no teclado.	Q1 e Q2
Esses usuários não têm domínio do uso de scroll. Oferecer a opção de descer ou subir o conteúdo na própria página.	Q1 e Q2
Fornecer feedback para todas as ações. Quando se tratar de serviços que emitem protocolos, oferecer a possibilidade de impressão. O protocolo impresso representa a concretização da ação, aumentando o sentimento de confiança de que a tarefa foi realizada com sucesso.	Q3
Garantir que o usuário tenha controle sobre as ações do sistema. Usuários experientes questionam quando o sistema reage de maneira automática, sem o consentimento explícito do usuário.	Q4

Considerações finais

Considerando o conjunto de requisitos e diretivas formalizado, algumas soluções de design foram geradas. Próximos passos incluem a construção de uma solução ajustável que permita que cada usuário possa interagir com uma proposta de design que mais se aproxime das suas necessidades de interação. Para a construção dessa solução ajustável está se considerando a infra-estrutura proposta por [2] e uma prova de conceito está atualmente em desenvolvimento.

Agradecimentos. FAPESP (proc. nro. 2006/54747-6) e Microsoft Research - FAPESP (proc. nro. 2007/54564-1).

Referências

[1] Baranauskas, M.C.C.; Hornung, H.H.; Martins, M. C. Design Socialmente Responsável: Desafios de Interface de Usuário no Contexto Brasileiro. *Proc of the 35o. SEMISH*. XXVIII CSBC.

[2] Bonacin, R.; Baranauskas, M.C.C. An Organizational Semiotics Approach Towards Tailorable Interfaces. *Proc of the 11th HCII*, v. 3. p. 1-12, 2005.

[3] Connell, B.R., Jones, M., Mace, R. et al. The Principles of Universal Design. *Center for Universal Design*, 1997. http://www.design.ncsu.edu/cud/about_ud/udprinciples.htm . Acesso-set/08.

[4] Neris, V.P.A.; Martins, M.C.; Prado, M.E.B.B, Hayashi, E. C. S.; Baranauskas, M. C. C. Design de Interfaces para Todos - Demandas da Diversidade Cultural e Social. *Proc of the 35o. SEMISH 2008* XXVIII CSBC.

23. Automated Negotiation of Multi-party Electronic Contracts in Agricultural Supply Chains

Authors: Evandro Bacarin, Edmundo R.M. Madeira, Claudia Bauzer Medeiros

SPICA Project

SPICA stands for "SuPply chain Integration, Coordination, contracting and Auditing framework". It aims at providing a comprehensive framework for agricultural supply chains.

A *supply chain* is a network of retailers, distributors, transporters, storage facilities and suppliers concerning a specific product. These elements are by their nature distributed, heterogeneous and autonomous and their relationships are inherently dynamic.

An agricultural supply chain has a few particularities. To start with, the flow of products within a chain is subject to a wide range of controls. Besides the economic and delivery schedule limitations found in B2B negotiations, agricultural supply chains are sensitive to geographic location, season, climate, product perishability, and cultural and religious backgrounds.

Examples of concerns are, for instance, whether the production process is harmful to the environment or whether it uses genetically modified substances. This requires setting up strict monitoring at all stages, as well as enforcing a large set of rules, which may be product, region or season-sensitive. A parallel concern is the quality of the final product, which involves auditing all production and distribution stages.

Another peculiarity is the so-called *return flow* within such chains, in which the refuse of a given stage of the chain may be recycled and re-enter the chain at another stage. Recycling is not a problem restricted to agricultural chains, but the constraints imposed on these cycles are. Finally, the number and kinds of actors encountered allow limitless possibilities of chain configurations, and the same kind of raw material may originate a large set of interrelated chains.

SPICA goals are three-fold. First, presenting a flexible and comprehensive, yet simple, model for agricultural supply chains. Second, providing support for controlling the product flow within a chain. Finally, organizing the activities and interactions within such a chain.

The Model

SPICA proposes a model for agricultural supply chains. Literature on Economics and Agriculture often presents descriptions of specific supply chains, e.g., the rice supply chain. However, such descriptions differ significantly in what the chain elements are, their relationship and the description's level of details.

SPICA's model comprises a few elements, namely: production, transportation, storage elements and actors. The chain's dynamics is modeled by means of coordination plans, contracts, regulations and summaries. A *coordination plan* specifies a sequence of activities to be performed within a supply chain. A *contract* specifies the relationship between chain's element. A contract can be established among several chain's elements (i.e., it is a multi-party contract). A *regulation* imposes restrictions on a product's flow. Finally, *summaries* are used to track the flow and transformation of products within a chain.

The Framework

The second and third goals are to be accomplished by means of a framework that is currently under development. It is composed of a few managers: negotiation manager (NM), coordination manager (CM), sumary manager (SM) and regulation manager (RM) and a number of respositories.

Each element (production, transportation, storage) may implement some of those managers. Managers can be organized hierarchically.

Currently, the framework's development aims at the negotiation process. We have proposed an XML-based multi-party contract model and a negotiation protocol for this kind of contracts. The negotiation protocol allows several negotiation styles to be employed within a single negotiation, namely: bargain (peer-to-peer negotiation), auction (if there is competition), and ballot (if consensus is needed).

The Negotiation Process

The SPICA's negotiation process proposes a *contract format* and a *negotiatiotion protocol*.

The contract's distinguishing feature is that it is a multi-party contract. Most contracts proposed in the literature are bilateral, i.e., there are only two signatories. In agricultural supply chains, colaboration among partners is strongly required. Expressing such a colaboration within a set of bilateral contracts may, at least, harden the management of existing contracts and make more difficult the contract enforcement.

In this protocol, the negotiation process is orchestrated by a leader negotiator and is guided by a contract model. The contract model is a predefined contract template containing a set of so-called *properties* which are filled in with values agreed upon by the negotiators. After a successful negotiation, a new contract is created from the model and the negotiated values.

A negotiation process involves two or more negotiators. One of them is the leader. There is a notary responsible for bureaucratic chores (e.g., constructing the final contract) or acting as a trusted third-party (e.g., to control ballots). These players exchange information within a negotiation process through asynchronous messages. The messages may be peer-to-peer or broadcasted.

There are three generic styles of negotiation: ballots, auctions and bargains. *Ballots* are used when the negotiators have to reach consensus on a property's value. *Auctions* are used when there is competition among different negotiators in order to bind a property to a value. *Bargains* are used when there are two negotiators and they want to interact to reach a value that is convenient for both.

The negotiation styles are built on a few negotiation primitives, i.e., types of messages exchanged among the participants. These primitives rely on two basic confirmed mechanisms: request for proposals (RFP) and offers; Request of Information (RFI) and information (Info).

An RFP is an invitation. A negotiator A sends an RFP to a negotiator B asking for a value for one or more properties. An RFP may prescribe some restrictions on the expected answer and may also bind the value of other properties.

An offer is a promise. A negotiator who wants to assign a value to one or more properties sends an offer to another negotiator. The offer indicates the properties the first negotiator is interested in and the values it proposes for them. If the other negotiator accepts such offer, both negotiators are committed to the proposed values. A negotiator can answer to an RFP by sending back an offer that proposes values for the desired properties and that complies with the restrictions indicated in the RFP.

RFIs and Infos are similar to RFPs and offers, respectively. However, there are two distinguishing differences. First, besides asking a value for a given property, an RFI may ask upper and lower bounds for it. Second, Infos provide the asked information, but the negotiator is not committed to the provided values.

The papers [BvdAMM07,BMM08] describe how these primitives are combined to construct diverse kinds of negotiation styles. It is noteworthy that most negotiation protocols describe in the literature implement only one of them. Typically, they implement a flavour of an auction or a bargain. To our best knowledge, none protocol combines all the three styles we advocate are necessary to supply chains.

Implementation

Currently, an infrastructure for supporting the negotiation protocol has been implemented. It comprises hundreds of Java classes that help the implementation of negotiators and provide a middleware that supports negotiation message exchanges according to the proposed protocol.

The negotiators and other supporting agents are Web services and negotiation messages are formatted as XML files.

The middleware builds on the YAWL's worflow engine. The negotiation protocol is specified by means of YAWL language [vdAtH05] and runs on YAWL's engine augmented with a new tailor made YAWL Custom Service.

Main Contributions

To sum up, the thesis' main contributions are:

- a model and an architecture for agricultural supply chain integration;
- a multi-party contract format;
- a negotiation protocol that uses a few primitives to construct different negotiation styles;
- a prototype implementation for the negotiation protocol over an existing workflow management system.

Project's Publications

Further details about the project's results may be find in the following publications: [BMM04, BMM08, BvdAMM07, BMM07, KMBM07].

Referências

[BMM04] E. Bacarin, C.B. Medeiros, and E.R.M. Madeira. A Collaborative Model for Agricultural Supply Chains. In *CoopIS 2004, LNCS 3290*, pages 319–336, 2004.

[BMM07] Evandro Bacarin, Edmundo R. M. Madeira, and Claudia M. B. Medeiros. Using choreography to support collaboration in agricultural supply chains. Technical Report IC-07-07, Institute of Computing/UNICAMP, March 2007.

[BMM08] E. Bacarin, E.R.M. Madeira, and C.B. Medeiros. Contract e-negotiation in agricultural supply chains. *Intl. Journal of Electronic Commerce*, 12(4):71–97, summer 2008.

[BvdAMM07] E. Bacarin, W.M.P van der Aalst, E. Madeira, and C.B. Medeiros. Towards modeling and simulating a multi-party negotiation protocol with colored petri nets. In *Proc. CPN 07 - Eighth Workshop and Tutorial on Practical Use of Coloured Petri Nets and the CPN Tools*, 2007.

[KMBM07] A.K. Kondo, C.B. Medeiros, E. Bacarin, and E.R.M. Madeira. Traceability in Food for Supply Chains. In *Proc. 3rd International Conference on Web Information Systems and Technologies (WEBIST)*, pages 121–127, March 2007. Barcelona,Spain.

[vdAtH05] W.M P. van der Aalst and A.H.M. ter Hofstede. Yawl: yet another workflow language. *Inf. Syst.*, 30(4):245–275, 2005.

24. Adoção e Evolução de Variabilidades em Linhas de Produto baseada em Componentes

Autores: Leonardo Pondian Tizzei e Cecília Mary Fischer Rubira

O **Desenvolvimento Baseado em Componentes** (DBC) é um paradigma de desenvolvimento que visa reduzir custo, esforço e tempo de chegada ao mercado de produtos de software. No DBC, sistemas de software são desenvolvidos a partir da composição de blocos interoperáveis e reutilizáveis chamados de componentes de software [11]. Os métodos de DBC como UML Components [4] visam reutilizar componentes desenvolvidos previamente. No método UML Components, durante a fase de especificação da arquitetura o arquiteto deve consultar o repositório de componentes e verificar se existe algum componente que pode implementar alguma funcionalidade do sistema. Para simplificar o desenvolvimento, o método adota uma arquitetura pré-definida em camadas.

Visando garantir alguns atributos de qualidade como manutenibilidade ou facilidade de integração, componentes de software podem seguir modelos de padronização como o COSMOS* [7]. Em particular, o **modelo de implementação COSMOS*** faz um mapeamento entre arquitetura de software e código-fonte, mantendo a conformidade entre eles.

Assim como DBC, uma **Linha de Produto de Software** (LPS) também visa reutilizar software, com a diferença que uma linha de produto define um núcleo comum de artefatos que são reutilizados. É possível unir ambas disciplinas criando uma LPS em uma organização já inserida no contexto de DBC, com o objetivo de intensificar ainda mais o reúso. Isso ocorre porque uma LPS é planejada para produzir um conjunto de produtos de software com alto grau de similaridade entre si, que atendem às necessidades específicas de um segmento de mercado ou missão, e que são desenvolvidos de forma prescritiva a partir de um conjunto de artefatos básicos[15] [5]. Uma LPS geralmente é descrita através de características[16], que podem ser mandatórias, opcionais ou alternativas [8]. Por exemplo, um celular poderia ter como característica mandatória o envio de mensagens por SMS, uma característica opcional seria a manipulação de vídeos e uma característica alternativa seria a tecnologia GSM ou CDMA. A criação de diferentes produtos a partir de um núcleo comum é possível devido ao conceito de variabilidade. **Variabilidade** é a capacidade que um sistema de software ou artefato tem de ser modificado, customizado ou configurado para ser usado em contexto específico [12]. Usualmente os métodos para desenvolvimento de LPSs adaptam métodos e notações usadas no desenvolvimento de sistemas tradicionais. Por exemplo, o método PLUS [8] descreve um processo evolucionário para o desenvolvimento de linhas de produto de software, estendendo as notações padrões de UML para dar suporte aos conceitos de variabilidades inerentes a linhas de produto.

Contudo a variabilidade de uma LPS pode tornar sua evolução mais complexa do que a de um sistema tradicional, dificultando a previsão do impacto de introduzir uma nova característica a um produto. Em parte, esse problema se deve à falta de rastreabilidade dos mecanismos de variabilidade entre os artefatos de uma LPS [3]. Também contribui para esse problema o fato de que muitas vezes as características não são mapeadas para

[15] do inglês, *core assets*
[16] do inglês, *features*

um único componente do sistema. Ou seja, tanto um componente pode implementar mais de uma característica como uma característica pode ser implementada por mais de um componente. Esses problemas são conhecidos como **emaranhamento** e **espalhamento**, respectivamente.

O **desenvolvimento de software orientado a aspectos** (AOSD[17]) é uma abordagem que visa modularizar os interesses transversais[18] das aplicações. Esses interesses transversais são características ou propriedades com escopo abrangente que geralmente presentes em diversos módulos em um sistema de software [1]. Alguns trabalhos utilizam AOSD para amenizar os problemas causados pelo emaranhamento e pelo espalhamento de características [1, 6].

A evolução de LPS pode ser mais difícil que a evolução de um sistema tradicional por ter que lidar com a variabilidade. O mapeamento entre os pontos de variação é importante para apoiar a evolução de linhas de produto de software [3]. Diversos trabalhos discutem a gerência de variabilidades na fase de análise(e.g. [9, 10]). Contudo, apesar de alguns trabalhos estabelecerem um relacionamento entre pontos de variação, eles não provêem uma abordagem sistemática para mapear pontos de variação durantes as fases de projeto e implementação [3, 10, 12]. Outros trabalhos utilizam AOSD para amenizar os problemas causados pelo emaranhamento e pelo espalhamento das características [1,6]. Todavia, estes trabalhos não adotam uma abordagem baseada em componentes que poderia intensificar o reúso de software. Além disso, os trabalhos também não utilizam técnicas de AOSD em todas as fases do ciclo de vida de um software, que poderia contribuir para a rastreabilidade e conseqüentemente a evolução do sistema. Desta forma, essa pesquisa almeja responder a seguinte questão:

> *O mapeamento explícito e sistemático entre análise, projeto e implementação de variabilidade, realizado durante a adoção de uma linha de produto baseada em componentes, pode facilitar sua evolução?*

Nossa proposta é desenvolver uma abordagem para gerenciar a evolução de variabilidades em uma linha de produto baseada em componentes. Essa abordagem consiste em um arcabouço envolvendo a definição de artefatos de software com variabilidade, métodos para criar e mapear esses artefatos durante o processo de desenvolvimento e métodos prescritivos que possam ser integrados a um processo de desenvolvimento baseado em componentes. Mais especificamente, será proposta uma combinação dos métodos PLUS e UML Components, que será validada através de estudos de caso. O método deve considerar a utilização de aspectos desde as fases iniciais promovendo a separação de interesses e reduzindo o emaranhamento e espalhamento de características. Outro objetivo é estender o ambiente Bellatrix [14] e adaptar ferramentas [13] para apoiar o o desenvolvimento de LPS através da inclusão de mecanismos de variabilidade. Por fim, como o enfoque é em LPS baseadas em componentes, o modelo de implementação de componentes COSMOS* [7] será estendido para dar suporte a mecanismos de variabilidade.

[17] sigla em inglês para *Aspect-oriented Software Development*

[18] do inglês, *crosscutting concerns*

Resultados preliminares

Os resultados preliminares da pesquisa são três: (i) uma versão inicial do método para apoiar a evolução de linhas de produtos baseadas em componentes; (ii) um meta-modelo que estabelece o relacionamento entre os diversos conceitos relacionados ao método e (iii) uma extensão do modelo de implementação de componentes COSMOS*, para permitir a implementação de variabilidades através de aspectos.

Para desenvolver um método para apoiar a adoção e evolução de linhas de produto baseadas em componentes, combinamos dois métodos: PLUS [8] e UML Components [4]. Do método PLUS, adotamos o modelo de características e a utilização de estereótipos para determinar o que é mandatório, opcional ou alternativo. Do método UML Components, adotamos os modelos e passos usados na especificação do sistema. Assim, pretendemos permitir a rastreabilidade entre características e elementos arquiteturais, inclusive a associação das características a interfaces ou operações. Dessa forma, ao evoluir o modelo de características, é possível seguir os passos do mapeamento proposto e identificar em que pontos a arquitetura pode ser alterada. Contudo, essa abordagem não oferece apoio a especificação de aspectos desde as fases iniciais da especificação. Outro ponto crítico, é que o método UML Components induz a criação de um sistema em camadas, o que pode não ser recomendado em aplicações embarcadas, por exemplo.

O meta-modelo para apoiar a variabilidade é importante para estabelecer o relacionamento entre os conceitos e artefatos envolvidos no método. Por exemplo, definir o relacionamento entre uma característica e um componente de software. O meta-modelo também pode auxiliar na construção de ferramentas para automatizar parte do método, uma vez que ele descreve algumas das meta-informações necessárias para essa automatização. Para especificar o meta-modelo, definimos um conjunto de propriedades necessárias e nos baseamos em trabalhos anteriores para obter essas propriedades [2, 12].

Por fim, estendemos o modelo de implementação de componentes COSMOS* para apoiar a implementação de variabilidades através de mecanismos de aspectos e desta forma facilitar a evolução do sistema. Para estender o modelo COSMOS* sem quebrar o encapsulamento, explicitamos os pontos de acesso ao componente via aspectos. Cada componente também possui um contrato que especifica o nível de acesso que outros componentes têm em relação às informações produzidas por ele.

Trabalhos futuros

Os próximos passos envolvem estudos de caso para avaliar os trabalhos realizados. Pretendemos especificar e refatorar uma linha de produto chamada de MobileMedia [6] para sistemas de telefones móveis com o objetivo de: (i) verificar a viabilidade da utilização meta-modelo; (ii) identificar as vantagens e deficiências do método, principalmente em relação a facilidade para evoluir o sistema, e propor soluções para essas deficiências e (iii) analisar se a utilização do COSMOS* estendido contribui para a evolução do sistema.

Referências

[1] V. Alves, P. M. Jr., L. Cole, A. Vasconcelos, P. Borba, and G. Ramalho. Extracting and evolving code in product lines with aspect-oriented programming. In *Trans. on*

Aspect-Oriented Software Development (TAOSD), volume IV, pages 117–142, 2007.

[2] F. Bachman, M. Goedicke, J. Leite, R. Nord, K. Pohl, B. Ramesh, and A. Vilbig. A meta-model for representing variability in product family development. In *5th Int. Workshop on Sw. Product-Family Eng.*, volume 3014 of *Lecture Notes in Computer Science*, pages 66–80. Springer, 2003.

[3] K. Berg, J. Bishop, and D. Muthig. Tracing software product line variability: from problem to solution space. In *SAICSIT '05: Proc. of the 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*, pages 182–191, 2005.

[4] J. Cheesman and J. Daniels. *UML components: a simple process for specifying component-based software*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2000.

[5] P. Clements and L. Northrop. Software product lines: Practices and patterns. *Addison-Wesley*, 2002.

[6] E. Figueiredo, N. Camacho, C. S. M. Monteiro, U. Kulesza, A. Garcia, S. Soares, F. Ferrari, S. Khan, F. Filho, and F. Dantas. Evolving software product lines with aspects: an empirical study on design stability. In *ICSE*, 2008.

[7] L. A. Gayard, C. M. F. Rubira, and P. A. de Castro Guerra. COSMOS*: a COmponent System MOdel for Software Architectures. Tec.Rep. IC-08-04, Instituto de Computação, 2008.

[8] H. Gomaa. *Designing Software Product Lines with UML: From Use Cases to Pattern-Based Software Architectures*. Addison-Wesley, 2004.

[9] K. Kang, S. Kim, J. Lee, K. Kim, E. Shin, and M. Huh. Form: A feature-oriented reuse method with domain-specific reference architectures. *Ann. Softw. Eng.*, 5:143–168, 1998.

[10] P. Sochos, M. Riebisch, and I. Philippow. The feature-architecture mapping (farm) method for feature-oriented development of softw. product lines. In *ECBS '06: Proc. of the 13th Annual IEEE Int. Symp. and Workshop on Eng. of Computer Based Systems (ECBS'06)*, pages 308–318, 2006. IEEE Computer Society.

[11] C. Szyperski. *Component Software*. Addison-Wesley, 2002.

[12] S. Thiel and A. Hein. Systematic integration of variability into product line architecture design. In *SPLC 2: Proc. of the Second Int. Conf. on Sw. Product Lines*, pages 130–153, 2002. Springer-Verlag.

[13] L. Tizzei, P. Guerra, and C. Rubira. Uma abordagem sistemática para reutilização e versionamento de componentes de software. In *Work. de Manut. de Sw. Moderna(SBQS)*, 2007.

[14] R. Tomita. Bellatrix: Um ambiente para suporte arquitetural ao desenvolvimento baseado em componentes. Master's thesis, Instituto de computação (UNICAMP), 2006.