Uma Comparação entre os Modelos de Proveniência OPM e PROV

Bárbara Bivar[1], Lucas Santos[1], Troy C. Kohwalter[1], Anderson Marinho[2], Marta Mattoso[2], Vanessa Braganholo[1]

[1]Instituto de Computação – Universidade Federal Fluminense (UFF)
[2]PESC/COPPE – Universidade Federal do Rio de Janeiro (UFRJ)

{barbarabivar, lsantos}@id.uff.br, {tkohwalter, vanessa}@ic.uff.br,
{andymarinho, marta}@cos.ufrj.br

Abstract. *The large scale production of digital scientific objects involves a variety of processes. The analysis of these processes and its results demands a rigorous management of the changes that occurs throughout its execution. The OPM model has emerged as a reference to represent the provenance of various systems such as databases, web and scientific workflows and was widely adopted. However, an initiative from W3C to define a provenance model named PROV has recently emerged and so many systems based on OPM will probably migrate to PROV. The contribution of this work is a comparative study that aims to ease the migration of the current systems based on OPM to the PROV model.*

Resumo. *A produção em grande escala de objetos digitais científicos envolve uma diversidade de processos. A análise desses processos e seus resultados exige um rigoroso acompanhamento das alterações ocorridas ao longo de sua execução. O modelo OPM surgiu como referência para representar a proveniência de diversos sistemas como bancos de dados, web e workflows científicos e foi largamente adotado. Porém, recentemente surgiu a iniciativa do modelo de proveniência PROV dentro do W3C e com isso muitos sistemas baseados em OPM provavelmente migrarão para o PROV. Este trabalho contribui com um estudo comparativo que visa facilitar a migração dos sistemas atuais do OPM para o modelo PROV.*

1. Introdução

Resultados de experimentos científicos não podem ser compreendidos sem o conhecimento do significado dos dados e das circunstâncias ocorridas em sua criação. Dentre essas informações importantes está a proveniência dos dados [Davidson and Freire, 2008; Freire et al., 2008]. O termo proveniência é bem compreendido no contexto das artes ou bibliotecas digitais onde se refere à documentação histórica de um objeto, ou sua trajetória de vida. Por outro lado, proveniência de dados em experimentação científica tornou-se um tema tão importante que foram criados workshops e conferências específicos sobre o assunto, incluindo o *International Provenance and Annotation Workshop* (IPAW)[1] e o *Provenance Challenge*[2]. Em cada edição do *Provenance Challenge*, a comunidade científica elenca os desafios de proveniência de dados a serem resolvidos e recebe trabalhos de diversos cientistas com

[1] http://www.ipaw.info/
[2] http://twiki.ipaw.info/bin/view/Challenge/

propostas de solução para esses desafios. Durante o primeiro Provenance Challenge, em 2006, os participantes estavam interessados em questões de proveniência para serem utilizadas em dados digitais, envolvendo tópicos relacionados à documentação, derivação e anotação. Como resultado, o primeiro modelo de proveniência digital, *Open Provenance Model* (OPM) [Moreau et al., 2011], foi concebido tentando abordar as questões levantadas durante o *Provenance Challenge*.

Posteriormente, outro modelo de proveniência foi desenvolvido pelo grupo incubador de proveniência do W3C [Gil et al., 2009], do qual participaram diversos pesquisadores responsáveis pelo OPM: o PROV [Gil and Miles, 2010]. De acordo com o grupo, a proveniência de recursos é um registro que descreve as entidades e processos envolvidos na produção de um recurso ou que influenciaram o mesmo. O uso de proveniência, independente do modelo utilizado, fornece um fundamento essencial para avaliar a autenticidade de dados, permitindo confiabilidade e reprodutibilidade [Groth and Moreau, 2013].

Quando do surgimento do PROV, o OPM já vinha sendo utilizado em diversas abordagens. No entanto, existe a possibilidade de o PROV se tornar o modelo padrão de proveniência uma vez que este é apoiado por um órgão de peso como o W3C. De fato, o documento que descreve o modelo de dados do PROV, chamado PROV-DM [Moreau and Missier, 2010a], já se encontra publicado como "Proposta de Recomendação". Isto culminaria em um processo de migração em massa dos sistemas existentes que utilizam o OPM para o PROV. Visando a apoiar abordagens que já trabalhavam com o OPM, o objetivo desse trabalho é realizar um estudo comparativo entre estes modelos de proveniência, apresentando um mapeamento que aponta suas semelhanças e diferenças.

O restante desse trabalho está organizado da seguinte forma. Os modelos OPM e PROV são apresentados, respectivamente, na Seção 2 e Seção 3. A Seção 4 apresenta o mapeamento entre os modelos. Por fim, a Seção 5 apresenta as conclusões deste trabalho e discute trabalhos futuros.

2. Open Provenance Model

O *Open Provenance Model* (OPM) [Moreau et al., 2011] surgiu como resultado dos Desafios de Proveniência (*Provenance Challenges*) propostos no contexto do IPAW [Freire et al., 2008]. Os Provenance Challenges aconteceram em quatros edições, uma em cada ano, de 2006 à 2010 ["Provenance Challenge WIKI," 2010]. O OPM resultou dos dois primeiros desafios, descritos a seguir.

1º Desafio: Visou a promoção de um fórum para a comunidade entender as capacidades dos diferentes sistemas de proveniência e expressar suas representações de proveniência.

2º Desafio: Visou o estabelecimento de interoperabilidade entre os sistemas, por meio de troca de informações de proveniência.

2.1. Conceitos Básicos

O *OPM* representa os dados de proveniência usando um grafo causal que registra a trajetória de vida de um processo ou artefato. O *OPM* é baseado em três entidades principais, descritas a seguir.

Artefato (Círculo - A): Entidades imutáveis que podem representar um objeto físico ou uma representação digital de um sistema de computador.

Processo (Retângulo - P): Ação ou conjunto de ações realizadas ou causadas por artefatos que irão resultar em novos artefatos.

Agentes (Octógono - Ag): Entidade contextual que age como catalisador de um processo, possibilitando, facilitando, controlando ou afetando sua execução.

Vale lembrar que o OPM visa sempre representar os acontecimentos passados, podendo até registrar um evento que ainda não tenha sido finalizado, mas nunca representando um evento do futuro. Dessa forma, seu foco está na proveniência retrospectiva em detrimento da prospectiva [Davidson and Freire, 2008].

2.2. Relações de Causalidade

As entidades do OPM são relacionadas no grafo de proveniência por meio de relações de dependência/causalidade. A seguir são apresentadas as cinco definições das relações de causalidades do OPM.

used: Relação entre *processo* e *artefato*. Indica a dependência do *artefato* para a execução do *processo*. Um *artefato* pode ser solicitado por diversos *Processos*.

wasGeneratedBy: Relação entre *artefato* e *processo*, indicando que o *Artefato* foi gerado pelo *Processo*.

wasControlledBy: Relação entre *processo* e *agente*, indicando que o *processo* foi controlado pelo *agente*. Essa relação especifica que todas as etapas do *processo*, desde a sua inicialização até sua finalização, foram controladas pelo *agente*. Um mesmo *agente* pode controlar mais de um *processo*, mas também pode ocorrer de um *processo* ser controlado por mais de um *agente*.

wasTriggeredBy: Relação entre *processos*, indicando que um *processo* foi desencadeado por outro *processo*.

wasDerivedFrom: Relação entre *artefatos*, indicando que um *artefato* foi originado por outro *artefato*.

2.3. Restrições Temporais

O OPM é compatível com a relação de causalidade temporal em sistemas distribuídos, ou seja, podemos considerar a relação *happens before* de Lamports [Lamport, 1978]. Mas para isso, é necessário que o tempo seja associado diretamente às ocorrências instantâneas.

Para *artefatos*, considera-se a criação e o uso. Em *processos*, atenta-se para seu início e fim de execução. Entretanto, essas informações temporais não são obrigatórias no modelo OPM, mas sim opcionais como uma forma de detalhar as informações de proveniência das entidades.

O OPM assume que existe um observador que toma conta de um relógio e sabe exatamente quando uma ocorrência aconteceu. No entanto, nem sempre é fácil obter o tempo instantâneo de criação/uso de um artefato ou definir quando um processo começou/terminou sua execução. Para isso, o modelo OPM assume que um tempo *t*

pertence ao intervalo [t_m, t_M], e que um evento ocorrido no tempo t nunca ocorre antes de t_m e nem depois de t_M. Consequentemente, um artefato nunca é criado antes do t_m estabelecido, e nem depois de t_M, assim como um processo não pode ter começado antes de t_m e nem terminando após t_M [Freire et al., 2008].

É possível incluir marcadores de tempo no grafo a fim de representar as restrições temporais. Para as relações *wasGeneratedBy*, *user*, *wasDeliveredFrom* e *wasTriggeredBy*, basta um marcador de tempo. Já para a relação *wasControlledBy* precisa-se de dois marcadores, um para indicar o início e o outro o fim, pois um processo pode ser controlado por mais de um agente.

3. PROV

PROV é uma família de especificações proposta pela W3C para expressar a proveniência de registros, contendo descrições de entidades e atividades envolvidas em produzir, entregar ou enunciar um determinado objeto. O grupo de discussão que deu origem ao PROV foi lançado oficialmente concomitantemente ao quarto *Provenance Challenge* ["Provenance Challenge WIKI," 2010]. As especificações que compõem o modelo de proveniência PROV foram divididas em diversos documentos, com detalhes sobre aspectos diferentes do modelo: PROV *Overview (PROV-OVERVIEW)* [Groth and Moreau, 2013], PROV *Primer (PROV-PRIMER)* [Gil and Miles, 2010], Prov *Ontology (PROV-O)* [Lebo et al., 2010], PROV *Data Model (PROV-DM)* [Moreau and Missier, 2010a], PROV *Constraints (PROV-CONSTRAINTS)* [Nies et al., 2010], PROV *Notation (PROV-N)* [Moreau and Missier, 2010b], PROV XML (PROV-XML) [Hua et al., 2013], PROV *Dublin Core Mapping (PROV-DC)* [Garijo et al., 2013], PROV *Links* [Moreau and Lebo, 2013], PROV *Access and Query (PROV-AQ)* [Weitzner et al., 2008], PROV *Dictionary (PROV-DICTIONARY)* [Missier et al., 2013], PROV *Semantics (PROV-SEM)* [Cheney, 2013] e PROV *Implementations* [Groth et al., 2012]. O estudo apresentado nesse artigo é baseado no *PROV-PRIMER*, que apresenta os conceitos e regras que compõem o modelo, além do *PROV-DM* e do *PROV-CONSTRAINTS*, que oferecem especificações completas e detalhadas de todos os conceitos apresentados no *PROV-PRIMER*.

Diante da compreensão de que os dados de proveniência podem ter diferentes perspectivas e finalidades, o modelo PROV procura acomodar todos os tipos de uso da proveniência. Ele classifica a proveniência em três tipos: *Centralizado em Agentes*, *Centralizada em Objetos* e *Centralizada em Processos*.

***Centralizado em Agentes**:* Os dados de proveniência visam especificar/registrar *quem* executou certas ações ou gerou certos objetos.

***Centralizada em Objetos**:* Os dados de proveniência visam especificar *quais* são as partes que compõem um objeto.

***Centralizada em Processos**:* Os dados de proveniência visam especificar *como* foram geradas as informações em questão. Isto é, quais as atividades ou processos que foram executados para que ela existisse.

3.1. Conceitos Básicos

O PROV também utiliza um grafo para representar informações de proveniência. Esse grafo é caracterizado por três tipos de vértices, descritos a seguir.

Entidade (Circulo - E): Para o PROV, uma entidade é algo físico, digital, conceitual ou outro tipo de coisa com alguns aspectos fixos, podendo as entidades serem reais ou imaginárias.

Atividade (Retângulo - A): Uma atividade é algo que ocorre durante um período de tempo, atuando sobre/com *entidades*. Assim, *atividades* podem consumir, processar, modificar, realocar, ou gerar *entidades*. Atividades de processamento de informações podem, por exemplo, incluir, mover, copiar ou duplicar uma *entidade* digital.

Agentes (Losango- Ag): Um *agente* é algo que possui certa responsabilidade por uma *atividade*, pela existência de uma *entidade* ou pelas *atividades* de outro *agente*. Um *agente* pode ser, por exemplo, uma pessoa, uma organização ou até um *software* que gerencia certa execução.

Note que os tipos apresentados no modelo PROV, assim como no modelo OPM, são nomeados para serem utilizados em afirmações sobre o passado. Desta forma, descrevem como os objetos foram criados ou apresentados (*i.e.*, proveniência retrospectiva).

3.2. Relações de Causalidade

Assim como no OPM, as arestas do grafo de proveniência do PROV representam relações de causalidade entre os vértices. Estas relações são direcionadas do efeito para a causa. As definições das dez relações presentes no PROV são apresentadas a seguir.

used: É uma relação que sempre ocorre entre *atividades* e *entidades*. Implica que uma determinada *atividade* usou uma *entidade*.

wasGeneratedBy: É uma relação que ocorre entre *entidades* e *atividades*. Indica que uma *entidade* foi gerada por uma *atividade*.

wasAssociatedWith: É uma relação que ocorre entre *atividades* e *agentes*. Implica que uma determinada *atividade* foi associada a um *agente* específico.

wasAttributedTo: É uma relação que ocorre entre *entidades* e *agentes*. Implica que uma *entidade* foi atribuída a um *agente* específico.

actedOnBehalfOf: Esta relação ocorre entre *agentes*. Indica que um *agente* tem responsabilidade ou autoridade sobre um outro *agente*.

wasRevisionOf: Esta relação ocorre entre *entidades*. Indica que uma *entidade* foi derivada de uma outra *entidade*. Uma *entidade* pode ser gerada a partir de outra para correção de um erro, por exemplo. A relação *wasRevisionOf* registra esse fato.

wasDerivedFrom: Esta relação também ocorre entre *entidades*, de forma semelhante à anterior. Neste caso, a relação representa que uma *entidade* foi originada de outra. Aqui, a derivação é evolutiva ao invés de corretiva, como no caso anterior.

wasInformedBy: Esta relação ocorre entre *atividades* e indica que uma *atividade* informada usou uma *entidade* que foi gerada pela *atividade* que a informou, mas esta *entidade* é desconhecida ou não é de interesse.

wasStartedBy: É uma relação que ocorre entre *atividades* e *entidades*, de forma a registrar que uma *atividade* iniciou uma *entidade*. Esta relação é semelhante à *wasGeneratedBy*. O que as diferencia é que *wasGeneratedBy* cria a *entidade*, portanto ela não existia antes de essa relação ocorrer, enquanto que *wasStartedBy* é uma *atividade* que inicia uma *entidade* previamente existente.

wasEndedBy: Também é uma relação que ocorre entre *atividades* e *entidades*, de forma a registrar que uma atividade finalizou uma *entidade*. Por exemplo, essa relação pode registrar uma atividade que terminou após a aprovação de um documento específico.

3.3. Restrições Temporais

O modelo PROV oferece a possibilidade de guardar dados relativos ao tempo das informações de proveniência, devido à importância das informações temporais em alguns cenários. É permitido guardar data e hora relativas a *entidades* ou *atividades*. Para *entidades,* é permitido armazenar informações de geração ou uso. Já para *atividades*, é permitido guardar informações de início e término de execução. Essas informações são armazenadas em *tickets* que ficam nas relações entre as *atividades* e *entidades* que correspondem à informação temporal. Esses *tickets* também podem ser usados para guardar informações de detalhes sobre qualquer um dos tipos de vértices apresentados ou das relações causais.

4. Mapeamento entre os modelos OPM e PROV

Em termos dos elementos principais dos dois modelos de proveniência estudados, é possível fazer um mapeamento direto entre os principais conceitos, associando o *artefato* em OPM à *entidade* em PROV, o *processo* em OPM à *atividade* em PROV e o *agente* em OPM ao *agente* em PROV. Algumas relações dos dois modelos são compatíveis porque, além de possuírem o mesmo nome, realizam de fato a mesma relação de causalidade entre os objetos. As relações mapeadas como iguais entre os dois modelos são: *used*, *wasGeneratedBy* e *wasDerivedFrom*.

Por outro lado, a relação *wasControlledBy* do OPM não existe com o mesmo nome no PROV, mas a relação *wasAssociatedWith* do PROV apresenta a mesma função, ligando *atividades* (*processos* no OPM) a *agentes* (em ambos os modelos). A Figura 1 ilustra o mapeamento entre os principais conceitos e relações do OPM e do PROV.

A relação *wasTriggedBy* é uma relação entre dois *processos* do OPM. Apesar de o PROV também possuir uma relação entre duas *atividades* (equivalentes aos *processos* no OPM) dentre do seu conjunto de relações, a relação *wasInformedBy* tem uma finalidade diferente. Esta relação visa a mostrar que uma determinada *atividade* informou algo para outra, enquanto que a relação *wasTriggedBy* do OPM indica que um *processo* foi desencadeado por outro. Por fim, existe uma relação no PROV (*wasStartedBy*) que equivale à relação *wasTriggedBy* do OPM. Entretanto, esta relação

é mais abrangente, pois pode ocorrer não apenas entre duas atividades, mas também entre uma *atividade* e uma *entidade*, indicando que a *entidade* iniciou a *atividade*.

No modelo PROV existem ainda quatro relações que não foram mapeadas no OPM: a previamente citada *wasInformedBy*, e as relações *wasEndedBy*, *actedOnBehalfOf* e *wasAttributedTo*. Estas duas últimas (*actedOnBehalfOf* e *WasAttributedTo)* são relações de delegação e associação dos *agentes* às *entidades* e *atividades*. Estas são extremamente importantes no PROV porque visam a fornecer informações de proveniência também centralizadas em *agentes,* algo que não ocorre no OPM. Por último, a relação *wasEndedBy* visa a representar a finalização de *processos*.

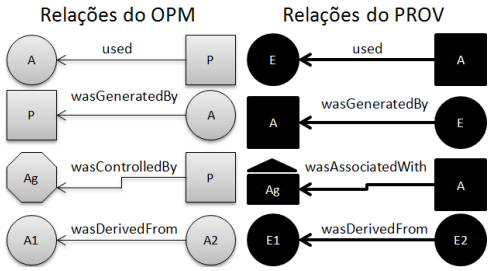

Figura 1. OPM x PROV

Tabela 1. OPM x PROV

OPM	PROV
used	*used*
wasGeneratedBy	*wasGeneratedBy*
wasControlledBy	*wasAssociatedWith*
wasDerivedFrom	*wasDerivedFrom*
wasTriggeredBy	*wasStartedBy*
	wasEndedBy
	wasRevisionOf
	wasAttributtedTo
	wasInformedBy
	actedOnBehalfOf

A partir destas relações sem equivalência direta, é possível observar diferenças entre os modelos. O OPM é mais simples, aparentemente voltado para o controle de fluxos de execução, indicando que um determinado processo foi iniciado por outro. Enquanto isso, o PROV aparenta ser mais voltado para questões de responsabilidades e histórico de dados, possuindo diversas relações entre *agentes* e os demais tipos (*atividades* e *entidades*). PROV também é bastante abrangente, possuindo relações equivalentes a todas as do OPM. A Tabela 1 ilustra a comparação das relações entre os dois modelos.

5. Conclusão

Registrar os dados de proveniência é uma necessidade em vários cenários, principalmente aqueles que possuem fluxos de execução complexos onde é necessário ter o histórico de todos os passos. Este trabalho apresentou um estudo comparativo entre os dois principais modelos de proveniência: OPM e PROV. De fato, os dois modelos apresentados permitem representar diversos aspectos das informações de proveniência, porém apresentam algumas diferenças.

O modelo PROV é mais voltado para armazenar dados de proveniência de maneira mais detalhada, focando nas responsabilidades dos *agentes* nos itens de proveniência. Este fato pôde ser constatado, pois o modelo PROV possui relações específicas para *agentes*, sem equivalências no OPM, mostrando-se um modelo mais abrangente. Já o OPM é mais simples e, aparentemente, mais usado devido ao fato de ter sido definido quatro anos antes que o PROV.

Como trabalhos futuros, estuda-se a possibilidade da criação de um tradutor automático de modelos OPM para PROV representados em XML. Este tradutor seria utilizado para auxiliar sistemas que queiram migrar para a especificação PROV, uma

vez que esta tende a acabar se tornando a especificação padrão. Além disso, o tradutor facilitaria a comunicação entre sistemas. A conversão pode ser complexa, pois uma informação do OPM pode ser representada de mais de uma maneira no PROV. Para isso, a tradução poderia ser um processo interativo que permite ao usuário escolher possíveis alternativas de tradução.

Agradecimentos. Os autores gostariam de agradecer ao CNPq, Capes e FAPERJ pelo financiamento parcial deste trabalho.

Referências

Cheney, J., 2013. Semantics of the PROV Data Model [www Document]. URL http://www.w3.org/TR/2013/WD-prov-sem-20130312/

Davidson, S.B., Freire, J., 2008. Provenance and scientific workflows: challenges and opportunities, in: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD '08. ACM, New York, NY, USA, pp. 1345–1350.

Freire, J., Koop, D., Santos, E., Silva, C.T., 2008. Provenance for Computational Tasks: A Survey. Computing in Science Engineering 10, 11–21.

Garijo, D., Eckert, K., Miles, S., Trim, C.M., Panzer, M., 2013. Dublin Core to PROV Mapping [www Document]. URL http://www.w3.org/TR/2013/WD-prov-dc-20130312/

Gil, Y., Cheney, J., Groth, P., Hartig, O., Miles, S., Moreau, L., Silva, P., 2009. W3C Provenance Incubator Group [www Document]. URL http://www.w3.org/2005/Incubator/prov/wiki/Main_Page

Gil, Y., Miles, S., 2010. PROV Model Primer [www Document]. URL http://www.w3.org/TR/prov-primer/

Groth, P., Lebo, T., Moreau, L., Soiland-Reyes, S., Missier, P., Sahoo, S., 2012. ProvImplementations [www Document]. URL http://www.w3.org/2011/prov/wiki/ProvImplementations

Groth, P., Moreau, L., 2013. PROV-Overview [www Document]. URL http://www.w3.org/TR/prov-overview/

Hua, H., Tilmes, C., Zednik, S., Moreau, L., 2013. PROV-XML: The PROV XML Schema [www Document]. URL http://www.w3.org/TR/prov-xml/

Lamport, L., 1978. Time, clocks, and the ordering of events in a distributed system. Commun. ACM 21, 558–565.

Lebo, T., Sahoo, S., McGuiness, D., 2010. PROV-O: The PROV Ontology [www Document]. URL http://www.w3.org/TR/prov-o/

Missier, P., Moreau, L., Cheney, J., Lebo, T., Soiland-Reyes, S., Nies, T.D., Coppens, S., 2013. PROV Dictionary [www Document]. URL http://www.w3.org/TR/2013/WD-prov-dictionary-20130312/

Moreau, L., Clifford, B., Freire, J., Futrelle, J., Gil, Y., Groth, P., Kwasnikowska, N., Miles, S., Missier, P., Myers, J., Plale, B., Simmhan, Y., Stephan, E., Den Bussche, J.V., 2011. The Open Provenance Model core specification (v1.1). In: Future Generation Computer Systems 27, 743–756.

Moreau, L., Lebo, T., 2013. Linking Across Provenance Bundles [www Document]. URL http://www.w3.org/TR/2013/WD-prov-links-20130312/

Moreau, L., Missier, P., 2010a. PROV-DM: The PROV Data Model [www Document]. URL http://www.w3.org/TR/prov-dm/

Moreau, L., Missier, P., 2010b. PROV-N: The Provenance Notation [www Document]. URL http://www.w3.org/TR/prov-n/

Nies, T.D., Cheney, J., Missier, P., Moreau, L., 2010. Constraints of the PROV Data Model [www Document]. URL http://www.w3.org/TR/prov-constraints/

Provenance Challenge WIKI [www Document], 2010. . URL http://twiki.ipaw.info/bin/view/Challenge/

Weitzner, D.J., Abelson, H., Berners-Lee, T., Feigenbaum, J., Hendler, J., Sussman, G.J., 2008. Information accountability. Communications of the ACM 51, 82–87.