Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Especialização em Desenvolvimento de Sistemas com Tecnologia Java

Prov-Se: Um web service semântico para análise de proveniência em experimentos científicos

Lenita Martins Ambrósio

JUIZ DE FORA
OUTUBRO, 2016

Prov-Se: Um web service semântico para análise de proveniência em experimentos científicos

Lenita Martins Ambrósio

Universidade Federal de Juiz de Fora

Instituto de Ciências Exatas

Departamento de Ciência da Computação

Especialização em Desenvolvimento de Sistemas com Tecnologia Java

Orientadora: Regina Maria Maciel Braga Villela

JUIZ DE FORA

OUTUBRO, 2016

Prov-Se: Um web service semântico para análise de proveniência em experimentos científicos

Lenita Martins Ambrósio

MONOGRAFIA SUBMETIDA AO CORPO DOCENTE DO INSTITUTO DE CIÊNCIAS EXATAS DA UNIVERSIDADE FEDERAL DE JUIZ DE FORA, COMO PARTE INTEGRANTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE ESPECIALISTA EM DESENVOLVIMENTO DE SISTEMAS COM TECNOLOGIA JAVA.

Aprovada por:

Regina Maria Maciel Braga Villela
D. Sc. (UFRJ)

Tarcísio de Souza Lima
M. Sc. (PUC - RJ)

Igor Knop
M. Sc. (UFJF)

JUIZ DE FORA

07 DE OUTUBRO, 2016

Resumo

O gerenciamento de dados de proveniência desempenha um papel fundamental no contexto dos experimentos científicos, uma vez que os cientistas estão interessados em examinar e auditar os resultados dos experimentos. Além disso, os dados de proveniência são essenciais para garantir a reprodutibilidade dos resultados obtidos e o reúso do experimento. Em alguns SGWfCs as informações de proveniência são automaticamente capturadas. Entretanto, eles muitas vezes utilizam de formatos proprietários dificultando o compartilhamento de informações de procedência. Além disso, a proveniência do próprio workflow não é contemplada. Desta forma, o objetivo deste trabalho é auxiliar os pesquisadores na verificação, reprodução e reutilização de experimentos científicos através da análise e inferência de informações acerca da proveniência destes experimentos, incluindo os demais artefatos que os compõe. A fim de alcançar este objetivo, o presente trabalho propõe uma arquitetura denominada Prov-SE (Provenance of Scientific Experiments).

Palavras-chave: experimentos científicos, workflows científicos, e-science, proveniência.

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de Abreviações

DCC	Departamento de Ciência da Computação
UFJF	Universidade Federal de Juiz de Fora
SGWfC	Sistema Gerenciador de Workflows Científicos
OWL	Web Ontology Language
RDF	Resource Description Framework
XML	eXtensible Markup Language
OPM	Open Provenance Model
REST	Representational State Transfer

1 Introdução

A ciência atualmente explora novas possibilidades de experimentação científica. Ao longo das últimas décadas a computação estabeleceu-se como o terceiro pilar da ciência (Deelman et al, 2009). Experimentos científicos complexos são agora simulados por supercomputadores através de ferramentas computacionais. Estas ferramentas envolvem etapas de análises de dados e computação em larga escala, e muitas vezes distribuídos, além disso, requerem a colaboração entre os cientistas geograficamente distribuídos.

Para apoiar estes experimentos, surgiram os Sistemas Gerenciadores de Workflows Científicos (SGWfC). Um SGWfC modela explicitamente a dependência entre processos dentro de um experimento, e coordena o comportamento dos recursos em tempo de execução (Belloum et al, 2011).

Um experimento científico envolve a execução de diversas tarefas, através da utilização de serviços externos organizados em um ou vários workflows científicos. O gerenciamento destes experimentos é uma tarefa complexa pois envolve o controle de todo o ciclo de vida do experimento. Neste contexto, a proveniência - informações sobre a origem, o contexto, a derivação, a propriedade, ou a história de algum artefato - desempenha um papel fundamental, uma vez que os cientistas estão interessados em examinar e auditar os resultados dos experimentos científicos. Além disso, os dados de proveniência são essenciais para garantir a reprodutibilidade dos resultados obtidos bem como o reúso do próprio experimento.

Desta forma, o gerenciamento de proveniência tem sido amplamente discutido na comunidade científica (Simmhan et al, 2005; Davidson and Freire, 2008; Lim et al, 2010). Em alguns SGWfCs, as informações de proveniência são automaticamente capturadas sob a forma de traços de execução. No entanto, eles muitas vezes utilizam de formatos proprietários dificultando o compartilhamento das informações. Além disso, a proveniência é voltada para os dados, então dados referentes ao próprio workflow não são contemplados

Outras abordagens de proveniência propostas (Cuevas-Vicenttín et al, 2014; Costa et al, 2014; Siqueira et al, 2016) abrangem apenas os workflows científicos de forma isolada

ao experimento. Estas abordagens, portanto não são capazes de auxiliar a manutenção e a evolução dos experimentos científicos como um todo.

Este trabalho está organizado em quatro capítulos além desta introdução. O segundo capitulo traz uma fundamentação teórica com os principais conceitos envolvidos no trabalho. O capítulo três apresenta os trabalhos relacionados a esta abordagem. O capítulo quatro descreve a arquitetura proposta, bem como um breve cenário de uso da mesma. Por fim, o capítulo cinco apresenta as considerações finais, bem como as sugestões para trabalhos futuros.

2 Pressupostos Teóricos

Neste capítulo são descritos os principais conceitos relacionados à proposta deste trabalho. Algumas considerações sobre experimentos científicos, proveniência e ontologias são apresentadas a fim de embasar a abordagem proposta, bem como facilitar o entendimento do leitor.

2.1 Experimentos Científicos

Um experimento científico é uma série de operações de análise ligadas entre si, as quais são modeladas e executadas através de workflows científicos (Goble et al, 2010). Um workflow científico é um modelo ou template composto por serviços, scripts ou outros workflows, que representa uma sequencia de atividades científicas implementadas por ferramentas, afim de alcançar um determinado objetivo. Para apoiar estes experimentos, permitindo que os pesquisadores pudessem focar em suas pesquisas e não no gerenciamento computacional, surgiram os Sistemas Gerenciadores de Workflows Científicos (SGWfC). Os SGWfC auxiliam a orquestração de vários algoritmos e processamentos computacionais, fazendo-se valer de processamento paralelo e distribuído, bancos de dados, inteligência artificial, dentre outros (Belloum et al, 2011).

Acreditar que um workflow não sofrerá evolução e mudanças no contexto de um experimento pode ser considerado utópico, pois a medida que novos resultados vão surgindo, a pesquisa vai tomando novos rumos, sendo necessário replanejamento, modificação ou adaptação na forma de execução, ou até mesmo dos recursos externos utilizados, como um serviço web ou sub-workflows de pré ou pós processamento (Siqueira et al, 2016).

Para que seja possível o gerenciamento de experimentos científicos, é importante o controle de todo o ciclo de vida do experimento. Neste sentido, diversos modelos já foram propostos para o ciclo de vida dos experimentos científicos. Em geral, eles envolvem quatro etapas principais, conforme o modelo clássico proposto por Deelman and Gil (2006) ilustrado na Figura 2.1.

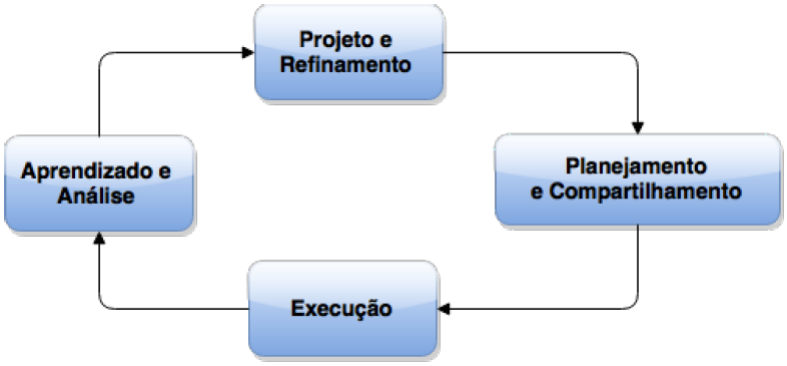

Figura 2.1: Ciclo de vida clássico de um experimento científico (Deelman and Gil, 2006)

A etapa de aprendizado e análise visa estabelecer a concepção do problema e o tratamento devido. No projeto e refinamento são definidos os recursos necessários e os agentes envolvidos, bem como o fluxo entre eles. A etapa de planejamento e compartilhamento aborda a construção do modelo e sua disponibilidade para uso. E por fim, a ultima etapa é a de realização da execução do modelo.

Siqueira et al (2016) propôs um ciclo de vida baseado no de Belloum et al (2011) o qual descreve o ciclo de experimentação como iniciando na fase de investigação do problema, onde ocorre a definição do escopo de pesquisa, avançando para a fase de prototipação do experimento, onde são desenvolvidos os componentes e os workflows necessários para o experimento. Logo após, o experimento é executado de forma controlada com os dados coletados e finaliza com a publicação dos resultados obtidos.

No modelo de Belloum et al (2011) todas as etapas utilizam repositórios compartilhados, o que foi substituído no modelo de Siqueira et al (2016) por atividades de Gerência de Configuração e Gerência de Proveniência, conforme a Figura 2.2.

Esta abordagem de Siqueira et al (2016), enfatiza a necessidade de controle e gerência das diversas versões dos workflows bem como a captura dos processos e dados para posterior análise. Por conta disso, uma das principais etapas deste modelo é a gerência da proveniência de dados. A seção seguinte aborda os principais conceitos sobre a proveniência de dados no contexto de experimentos científicos.

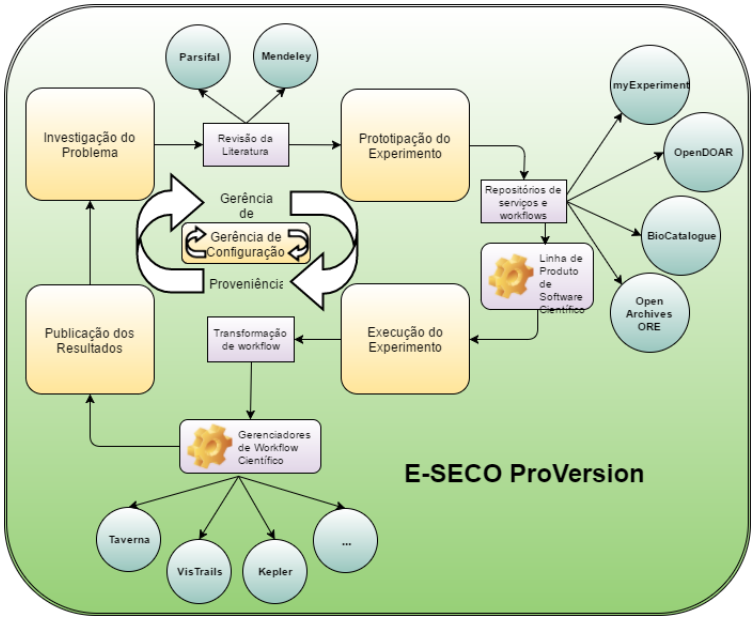

Figura 2.2: Ciclo de vida de um experimento científico no E-SECO ProVersion (Siqueira et al, 2016)

2.2 Proveniência

Com o crescente uso de aplicações em larga escala, o gerenciamento de dados está se tornando cada vez mais complexo. Metadados que descrevem os produtos de dados utilizados e gerados por essas aplicações são essenciais para desambiguar os dados e permitir sua reutilização. Simmhan et al (2005) define a proveniência de dados como um tipo de metadados que traz o histórico de derivação de um artefato de dados a partir de suas fontes, e destaca os principais usos destas informações de proveniência:

- Qualidade dos dados: A proveniência pode ser usada para estimar a qualidade e confiabilidade dos dados baseado na fonte dos mesmos, e nas transformações sofridas por eles;

- Rastreabilidade: A proveniência pode ser usada para traçar uma trilha de auditoria dos dados, determinando os recursos utilizados e os erros ocorridos durante sua derivação;

- Reprodutibilidade: Informações detalhadas de proveniência permitem a reprodução da derivação dos dados;

- Atribuição: A proveniência pode ser usada para estabelecer direitos autorais e pro-

priedade dos dados, permitindo a sua citação e determinando a responsabilidade em caso de dados incorretos.

- Informação: Uma utilização genérica da proveniência é a consulta com base em metadados de linhagem para a descoberta de dados. A proveniência também pode ser consultada para fornecer um contexto para a interpretação dos dados.

Na experimentação científica os metadados sobre a história de derivação dos dados é essencial para garantir a reprodutibilidade dos resultados obtidos através da execução de workflows científicos. Além disso, a proveniência proporciona a verificação da precisão e atualidade dos dados. Desta forma, o gerenciamento de proveniência tem sido considerado um ponto chave na arquitetura de SGWfCs, e amplamente reconhecido na comunidade científica (Lim et al, 2010).

Neste contexto Lim et al (2010) divide a proveniência em dois tipos:

1. Prospectiva: Uma especificação do workflow abstrato como uma receita para derivações futuras dos dados;

2. Retrospectiva: captura informações sobre a execução dos workflows e sobre a derivação passada dos dados.

Existem dois principais modelos para captura de dados de proveniência. O modelo OPM (Open Provenance Model) é o resultado do esforço da comunidade para alcançar a interoperabilidade dos dados de proveniência (Moreau et al, 2011). Seus principais objetivos são:

1. Permitir que informações de proveniência sejam trocadas entre sistemas, por meio de uma camada de compatibilidade com base em um modelo de proveniência compartilhado;

2. Permitir aos desenvolvedores criar e compartilhar ferramentas que operam em tal modelo proveniência;

3. Definir proveniência de forma precisa;

4. Apoiar uma representação digital de proveniência para qualquer 'coisa', quer produzida por sistemas de computador ou não;

5. Permitir a coexistência em vários níveis de descrição;

6. Definir um conjunto de regras que identifiquem as inferências válidas que podem ser feitas na representação de proveniência;

O segundo modelo é o PROV, o qual foi fortemente influenciado pelo OPM e atualmente é o modelo padrão recomendado pela W3C (Moreau and Groth, 2013). Este modelo permite 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. Desta forma, apesar de assim como o OPM não suportar a proveniência prospectiva, o PROV possibilita novas formas de representação do conhecimento inclusive a captura de proveniência centrada em processo, em entidade ou em agente.

O PROV é composto por uma família de 12 documentos, mas para se utilizar este modelo não é preciso estar familiarizado com todos eles. O PROV foi projetado especificamente para que os usuários e desenvolvedores possam começar rapidamente com o uso básico e, em seguida, gradualmente evoluir para cenários de uso mais avançados.

A Figura 2.3 apresenta os documentos do PROV classificados de acordo com o público alvo do documento: Os documentos em verde destinam-se aos usuários - esse público quer entender o PROV e utilizar as aplicações que suportam o PROV. Os em azul são destinados aos desenvolvedores - esse público quer desenvolver ou construir aplicativos que criam e consomem proveniência usando o PROV. E os documentos em rosa são destinados ao uso avançado - este público visa criar validadores, novas serializações do Prov, ou outros sistemas avançados baseados em proveniência.

Entre os principais documentos podem ser citados o PROV-DM, que especifica o modelo de captura de dados, o PROV-CONSTRAINTS, que é um conjunto de restrições aplicáveis ao modelo de dados (PROV-DM) e o PROV-O, uma ontologia para mapeamento do modelo de dados.

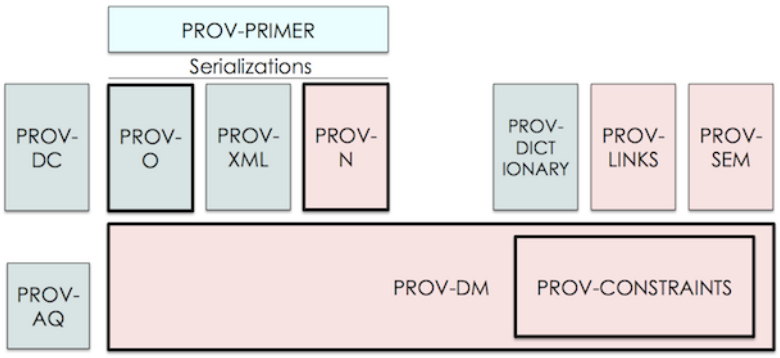

Figura 2.3: Organização do Modelo PROV (Moreau and Groth, 2013)

Simmhan et al (2005) divide as informações de proveniência em dois grupos: Informações sintáticas, utilizadas em sistemas baseados em anotações, os quais frequentemente adotam a linguagem XML para a representação das informações. Informações semânticas, utilizadas em sistemas que possuem ontologias de domínio em linguagens como RDF e OWL. As ontologias expressam precisamente os conceitos e relacionamentos usados na proveniência e proveem boas informações contextuais.

Algumas definições importantes sobre ontologias são descritas a seguir, bem como algumas das principais ontologias utilizadas para proveniência em experimentos científicos.

2.3 Ontologias

Para apoiar o compartilhamento e reutilização de conhecimento entre os diferentes sistemas é preciso definir um vocabulário comum de representação deste conhecimento. Neste sentido, Gruber (1995) tomou o termo ontologia emprestado da filosofia, e o definiu para a computação como uma especificação formal e explicita de uma conceituação compartilhada. Onde uma conceituação é uma visão simplificada e abstrata do mundo que se deseja representar para algum propósito.

De forma geral, uma ontologia é uma especificação de um vocabulário de representação de um domínio compartilhado, composta por: Definições de classes; Relações e Funções. As principais funções de uma ontologia são:

- Compartilhar o entendimento comum sobre a estrutura da informação entre pessoas ou agentes de software;

- Permitir a reutilização do conhecimento de domínio;
- Explicitar suposições sobre o domínio;
- Separar o conhecimento do domínio do conhecimento operacional;
- Analisar o conhecimento de domínio.

As ontologias permitem descrever a semântica das classes e propriedades usadas em documentos na web. Com isso, se tornou o terceiro componente básico da Web Semântica. O tipo mais comum de ontologias para a web tem uma taxonomia e um conjunto de regras de inferência. A taxonomia define classes de objetos e as relações entre eles. Classes, subclasses e relações entre entidades são uma ferramenta muito poderosa para o uso da Web. Podemos expressar um grande número de relações entre entidades através da atribuição de propriedades às classes e subclasses permitindo a herdar tais propriedades. As regras de inferência fornecem mais energia às ontologias permitindo que o conhecimento seja interpretado e inferido logicamente por máquinas Lee et al (2001).

No domínio da proveniência de dados, conforme dito anteriormente, as ontologias expressam precisamente os conceitos e relacionamentos usados e proveem boas informações contextuais. Visto isso, o modelo PROV definiu uma ontologia para a modelagem de dados de proveniência camada PROV-O Moreau and Groth (2013).

2.3.1 PROV-O

A ontologia PROV-O expressa o modelo de dados do PROV (PROV-DM) usando a OWL 2. Ela fornece um conjunto de classes, propriedades e restrições que podem ser usados para representar e trocar informações de proveniência gerada em sistemas diferentes e em diferentes contextos. Ela também pode ser especializada para criar novas classes e propriedades para modelar informações de proveniência para diferentes aplicações e domínios.

A Figura 2.4 representa o modelo inicial de classes e relacionamentos da ontologia PROV-O expandida. Onde as entidades são representadas por formas ovais amarelas, as atividades por retângulos azuis, e os agentes são pentágonos em laranja.

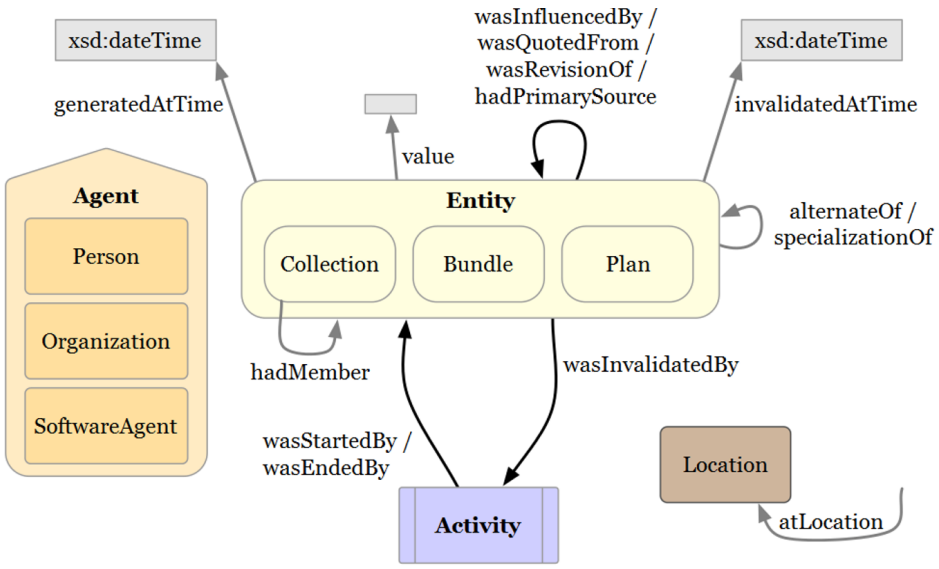

Figura 2.4: Modelo de classes e relacionamentos do PROV expandido (Belhajjame et al, 2013)

Apesar de fornecer diversas construções importantes para derivação de conhecimento, a PROV-O não expressa todo o conhecimento necessário para a gerência tanto de experimentos quanto dos workflows associados. Uma maneira de trazer esse suporte é através da proposição de regras ontológicas específicas relacionadas a este contexto. Com esta finalidade, Cuevas-Vicenttín et al (2016) apresenta a ontologia ProvONE descrita na subseção seguinte.

2.3.2 ProvONE

Em alguns SGWfCs as informações de proveniência são automaticamente capturadas sob a forma de traços de execução. No entanto, eles muitas vezes utilizam de formatos proprietários dificultando o compartilhamento de informações de procedência. Além disso, a proveniência do próprio workflow não é contemplada. A ontologia ProvONE é um modelo para a representação de proveniência de workflows científicos criado a fim de preencher estas lacunas (Cuevas-Vicenttín et al, 2016).

Esta ontologia é uma extensão do padrão PROV, inicialmente denominada D-PROV, com o objetivo de capturar as informações mais relevantes sobre os processos computacionais dos workflows científicos, e fornecer pontos de extensão para acomodar as especificidades de determinados sistemas de workflows científicos (Missier et al, 2013).

Criada no contexto da rede DataONE (Data Observation Network for Earth), uma rede de dados federados de observações da Terra (Michener et al, 2016), a ontologia ProvONE, adiciona elementos de proveniência (entidades e tipos de relacionamentos) para descrever a estrutura do processo juntamente com as dependências de dados que se originam da execução de um processo.

A ProvONE cobre tanto a proveniência prospectiva quanto a retrospectiva. Além disso, conforme pode-se observar na Figura 2.5 ela provê informações sobre os aspectos dos workflows (classes em azul), dos processos de execução (classes em laranja) e dos artefatos de dados (classes em lilás).

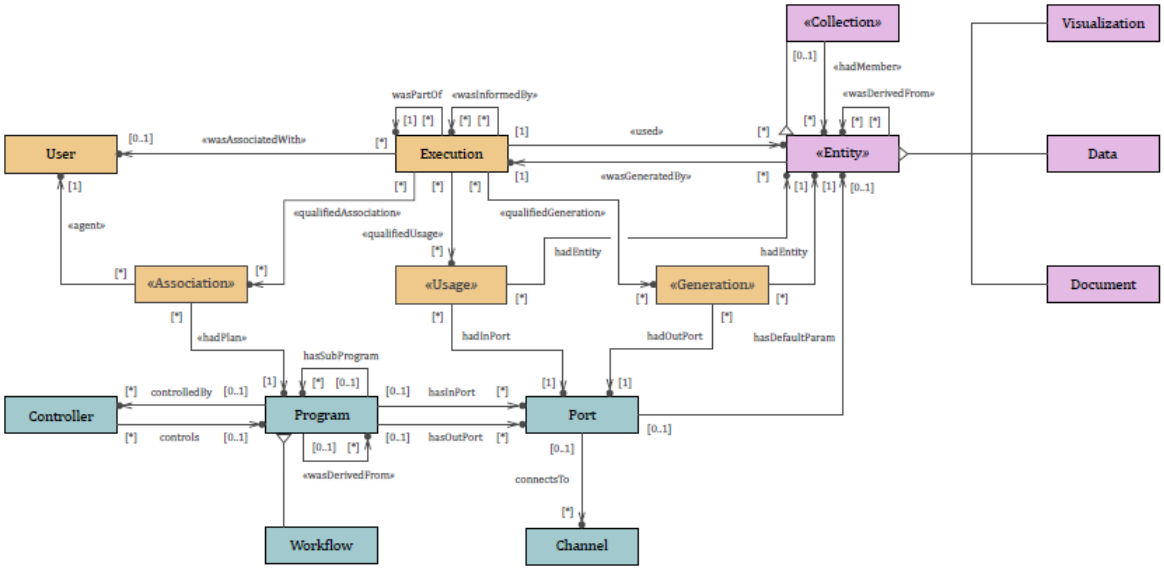

Figura 2.5: Modelo conceitual da ProvONE (Cuevas-Vicenttín et al, 2016)

3 Trabalhos Relacionados

Atualmente, com o reconhecimento, por parte da comunidade científica, da importância do armazenamento dos dados de proveniência dos workflows científicos, quase todos os SGWfCs existentes já apoiam a gestão de proveniência como uma funcionalidade chave. Além disso, surgiram outras abordagens independentes de workflow para tratar este tipo de informação.

3.1 Kepler (Altintas et al, 2004)

É um SGWfC usado no domínio de ecologia, geologia e bioinformática. Implementa o framework para proveniência COMAD (Collection Oriented Modeling and Design), que apoia a captura de dados aninhados e captura dependências de dados explícitos. Desta forma, trata a proveniência dos dados gerados na execução do workflow.

3.2 Taverna (Oinn et al, 2007)

É um SGWfC voltado principalmente ao apoio à comunidade de ciências da vida (biologia, química e médica). Implementa um plug-in para coleta de dados de proveniência que captura tanto proveniência interna gerada localmente por execuções dos workflows, como a proveniência externa gerada a partir de provedores de dados de terceiros. Mais recentemente, apresentou um modelo de proveniência de dados para aumentar os dados de proveniência com anotações.

3.3 VisTrails (Freire et al, 2006)

É um SGWfC que fornece suporte para a exploração e visualização de dados. Usa um mecanismo de proveniência baseado em mudanças para capturar informações de proveniência dos produtos de dados e da evolução dos workflows usados para gerar esses produtos. O modelo de proveniência é composto por três camadas: a camada de evolução do workflow,

que capta a relação evolutiva entre as especificações do workflow; a camada do workflow, que consiste em especificações individuais de workflows; e a camada de execução, que armazena informações de tempo de execução da execução do workflow.

3.4 Pegasus (Deelman et al, 2004)

É também um SGWfC que engloba um conjunto de tecnologias baseadas em workflow, permite a modelagem, execução e acompanhamento dos workflows em execução. Possui uma abordagem para criação e refinamento de workflows que usa representações semânticas. Desta forma, permite a geração de dados de proveniência a nível de aplicação durante a criação do workflow, e a nível de execução, durante o refinamento e execução do workflow. Estas informações relacionam o workflow final executado, com sua especificação de inicial (Kim et al, 2008).

3.5 Karma (Simmhan, 2006)

É um framework para captura de proveniência de experimentos científicos focados em workflows, utiliza um serviço web para captura dos dados e os armazena em um repositório no formato XML. Captura metadados de proveniência uniformes e usáveis, de maneira independente do workflow ou framework de serviço. O modelo Karma captura duas formas de proveniência: proveniência de processo, que são metadados que descrevem a execução do workflow e invocações associadas; e proveniência de dados, que fornece metadados semelhantes sobre a história da derivação de um produto de dados.

A proveniência neste modelo é dada em dois níveis: o nível de registo, que regista os metadados de serviços e dados que podem ser utilizados numa sequência de execução; e o nível de execução, que modela as instâncias do nível de registro e grava as informações relacionadas à invocações de métodos e aos produtos de dados utilizados ou gerados por cada invocação.

3.6 PReServ (Groth et al, 2015)

É um mecanismo de captura de proveniência para experimentos científicos independente do SGWfC. Ele foi desenvolvido para o contexto de bioinformática e todos os dados coletados são armazenados em metadados no formato XML. É uma implementação baseada em web services Java subjacente à arquitetura Pasoa (Groth et al, 2006). O PReServ captura as interações entre componentes internos e agrupamento de interações por meio do protocolo PReP (Provenance Recording Protocol), que especifica as mensagens que os atores podem trocar com o banco de proveniência.

3.7 ProvSearch (Costa et al, 2014)

Propõe uma arquitetura de gerenciamento de dados de proveniência independente de SGWfC e voltada para experimentos em ambientes distribuídos. Combina técnicas de gerenciamento de workflows distribuídos com gerenciamento de dados de proveniência distribuído. Permite que os dados de proveniência sejam capturados, armazenados e consultados em tempo de execução, sem interferir na execução do workflow.

Nesta arquitetura, os dados são fragmentados em múltiplos repositórios de proveniência na nuvem e podem ser acessados por diferentes SGWfCs. Os dados de proveniência são tratados em um modelo chamado PROV-Wf, uma extensão do modelo PROV para o domínio dos workflows científicos (Costa et al, 2013). A Figura 3.1 ilustra esta arquitetura, composta por quatro componentes:

1. Nós de banco de dados (Provenance DW): Formam uma rede descentralizada de servidores de bancos de proveniência. Cada nó contém um sistema de gerenciamento de banco de dados distribuído instalado com duas bases de dados diferentes. Uma para armazenar todos os dados de proveniência tradicionais (como a hora inicial e final, etc.) e os resultados do experimento; E a outra apenas com as estatísticas (por exemplo, o tempo médio de execução de um programa específico, porcentagem de erros para uma máquina específica, etc.);

2. Nó de controle (Control Node): É responsável por identificar qual o nó de banco de

dados irá armazenar os dados de proveniência para uma execução específica;

3. Depósito integrado e global de proveniência (Integrated DW): Armazena um resumo de todas a bases locais de dados estatísticas, agindo como um repositório de proveniência, ou seja, as estatísticas de todas as execuções de todas as experiências são armazenados neste depósito de proveniência integrado e pode ser consultado por qualquer usuário sem acessar resultados de experimentos de terceiros;

4. API: É a interface entre o ProvSearch e os SGWfC existentes.

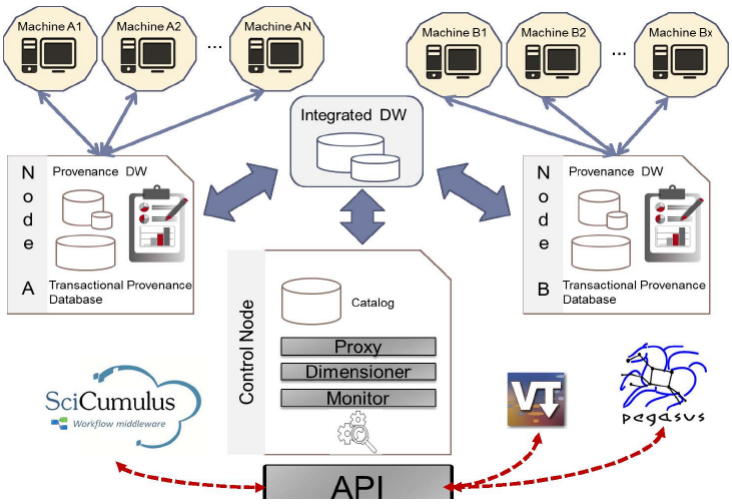

Figura 3.1: Arquitetura integrada do ProvSearch (Costa et al, 2014)

3.8 PBase (Cuevas-Vicenttín et al, 2014)

É um repositório de proveniência de workflows científicos que implementa a ontologia ProvONE, permitindo armazenamento, analise e replicação de experimentos científicos. Este repositório, assim como a ontologia ProvONE, é parte do projeto DataONE uma rede de dados federados de observações da Terra (Michener et al, 2016).

A Figura 3.2 apresenta a arquitetura do PBase o qual utiliza a arquitetura da plataforma JAVA em tês níveis:

1. Nível de visualização (Web Client): Um cliente web o qual possui uma interface adaptada para a visualização dos dados de proveniência de workflows científicos,

tornando a especificação de consultas e a interpretação dos seus resultados mais fácil e eficaz;

2. Nível de Aplicação: Implementação dos serviços web para atender às consultas feitas pelo usuário.

3. Nível de dados (PBase GraphDB): Conta com um banco de dados gráfico Neo4j oferecendo assim consultas declarativas e eficientes.

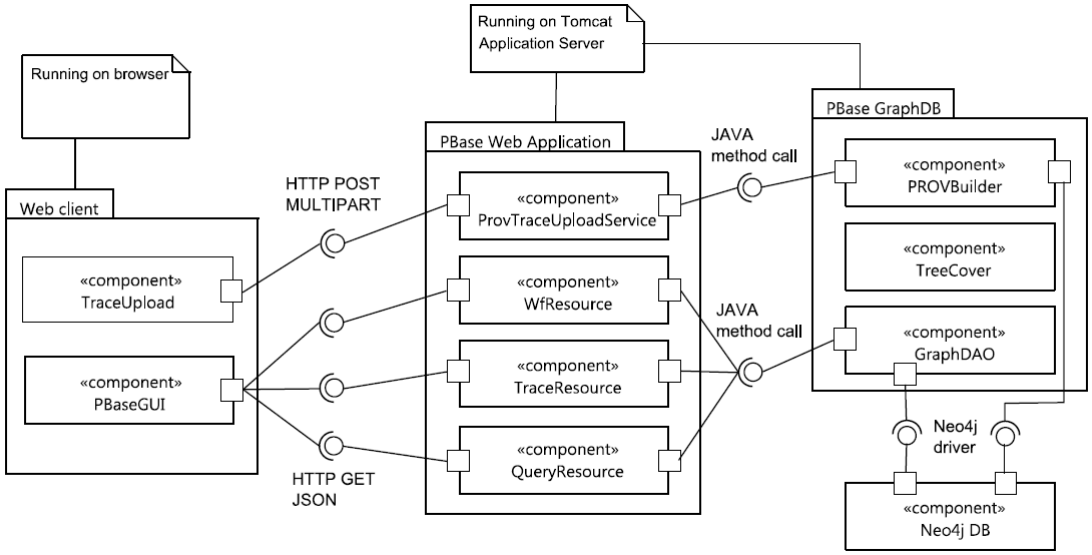

Figura 3.2: Arquitetura do sistema PBase (Cuevas-Vicenttín et al, 2014)

3.9 E-SECO ProVersion (Siqueira et al, 2016)

Apresenta uma abordagem para viabilizar o suporte a gerência de configuração na arquitetura E-SECO (E-Science Software ECOsystem) (Freitas et al, 2014). O E-SECO propõe o gerenciamento de todo o ciclo de vida dos experimentos científicos, utilizando como base o modelo de ciclo de vida proposto por Belloum et al (2011). Este ciclo de experimentação foi expandido englobando a abordagem E-SECO ProVersion, conforme já demonstrado anteriormente na Figura 2.2.

Esta abordagem utiliza uma extensão do modelo PROV, que abrange tanto a ontologia quanto o modelo de dados, aplicado ao domínio de workflows científicos. O módulo de proveniência permite ao pesquisador capturar os dados do workflow em diferentes

SGWfCs por meio de um serviço web. Estes dados alimentam a ontologia PROV-OEXT, que por meio de regras específicas do domínio detectam informações sobre a evolução e manutenção em workflows. As informações são armazenadas no módulo de histórico dos workflows, e disponibilizadas ao pesquisador por meio da interface web do E-SECO. Esta arquitetura é ilustrada pela Figura 3.3.

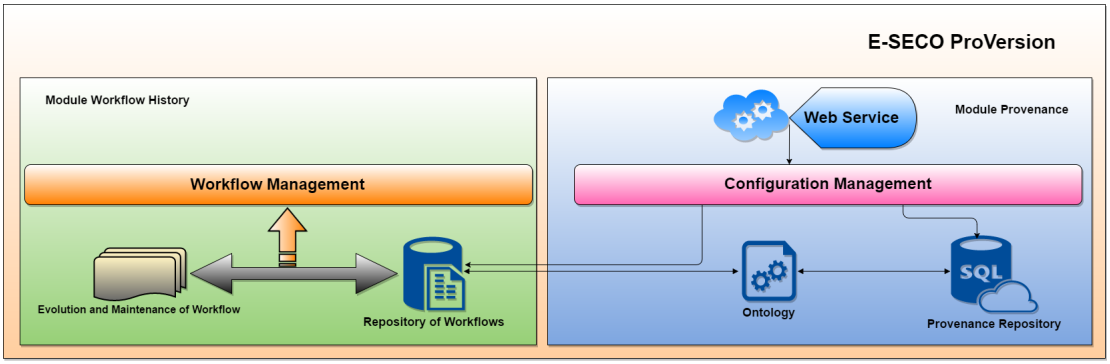

Figura 3.3: Arquitetura do E-SECO ProVersion (Siqueira et al, 2016)

3.10 Considerações finais do capítulo

A Tabela 3.1 apresenta um quadro comparativo entre as abordagens que tratam proveniência em experimentos científicos, considerando critérios importantes para a arquitetura proposta neste trabalho.

Tabela 3.1: Comparativo entre trabalhos relacionados

Abordagem	**Independente de SGWfC**	**Livre**	**Padrão**	**Inferência**	**Experimento Todo**
Kepler	NÃO	NÃO	NÃO	NÃO	NÃO
Taverna	NÃO	NÃO	NÃO	NÃO	NÃO
VisTrails	NÃO	NÃO	NÃO	NÃO	NÃO
Pegasus	NÃO	NÃO	NÃO	NÃO	NÃO
Karma	SIM	SIM	NÃO	NÃO	NÃO
PReServ	SIM	SIM	NÃO	NÃO	NÃO
PBase	SIM	SIM	SIM	SIM	NÃO
ProvSearch	SIM	SIM	SIM	NÃO	NÃO
E-SECO ProVersion	SIM	SIM	SIM	SIM	NÃO
Prov-SE	SIM	SIM	SIM	SIM	SIM

Conforme apresentado, muitos SGWfCs (Kepler, Taverna, VisTrails, Pegasus) já

contam com funcionalidades para o gerenciamento de dados de proveniência. Entretanto, estes sistemas utilizam modelos proprietários, o que dificulta o acesso a estes dados. Outras abordagens independentes dos SGWfCs foram surgindo para permitir acesso livre ao dados de proveniência capturados. Contudo, abordagens como o Karma e PReServ, que não utilizam um modelo padrão de proveniência, dificultam a interoperabilidade dos dados, bem como sua consulta e a análise pelo pesquisador (Davidson and Freire, 2008).

Arquiteturas como a do ProvSerarch e do E-SECO ProVersion que utilizam o padrão PROV, e a PBase que utiliza o modelo ProvONE, facilitam a interoperabilidade dos dados, permitindo integrar dados de proveniência de vários SGWfCs, o que é necessário quando os resultados científicos foram obtidos através da execução de uma sequência de workflows científicos em diversos SGWfCs (Lim et al, 2010). Apesar disso, estas abordagens não são abrangentes o suficiente, pois tratam a proveniência apenas a nível de workflow, não considerando os experimentos científicos como um todo. Desta forma, são ignoradas informações a cerca da derivação e semelhança entre os experimentos, e sobre os demais artefatos que o compõe.

4 Proposta

Este capítulo tem como objetivo apresentar a arquitetura Prov-SE. O Prov-SE pode ser visto como uma aplicação independente, mas projetada inicialmente para a plataforma de ecossistema E-SECO (Freitas et al, 2014), proporcionando funcionalidades específicas para a análise de dados de proveniência dos experimentos. Para um melhor entendimento do Prov-SE, a seção seguinte apresenta algumas considerações importantes sobre a plataforma E-SECO. Em seguida são apresentados detalhes sobre a arquitetura e sobre o desenvolvimento do Prov-SE. E por fim, para exemplificar o funcionamento da arquitetura proposta, um cenário de uso é apresentado.

4.1 E-SECO

O E-SECO é uma plataforma baseada em ecossistema de software, orientada a serviços e apoiada por uma rede ponto a ponto, desenvolvida com o objetivo de proporcionar uma arquitetura flexível, extensível e escalável para tratar as etapas do ciclo de vida de um experimento científico (Freitas et al, 2014).

Conforme mencionado anteriormente o ciclo de vida considerado pelo E-SECO, ilustrado pela figura 2.2, foi baseado no modelo proposto por Belloum et al (2011). Foram adicionadas atividades como a realização de revisões sistemáticas da literatura durante a etapa de investigação do problema e a utilização do conceito de uma LPSC (Linha de Produto de Software Científico) na etapa de prototipação do experimento.

Durante a etapa de prototipação são disponibilizados, aos cientistas, recursos para utilizarem workflows e serviços web de plataformas como myExperiment e BioCatalogue. A linha de produto de software científico chamada de Collaborative PL-Science Ecosystem é então incorporada na etapa de prototipação do experimento, aumentando o nível de reuso e a qualidade no desenvolvimento de workflows científicos, além de reduzir o tempo e consequentemente o custo do desenvolvimento (Freitas et al, 2014).

Finalmente, não somente os resultados da execução do experimento são armazena-

dos, mas também todos os dados relativos ao processo de experimentação, possibilitando que outros pesquisadores possam consultar (Freitas et al, 2014). As etapas de gerência de configuração e gerência de proveniência se estendem por todo o ciclo de vida do experimento, e foram adicionadas posteriormente em Siqueira et al (2016) pelo E-SECO ProVersion.

A Figura 4.1 detalha cada componente desta plataforma, na forma como ela se encontra atualmente.

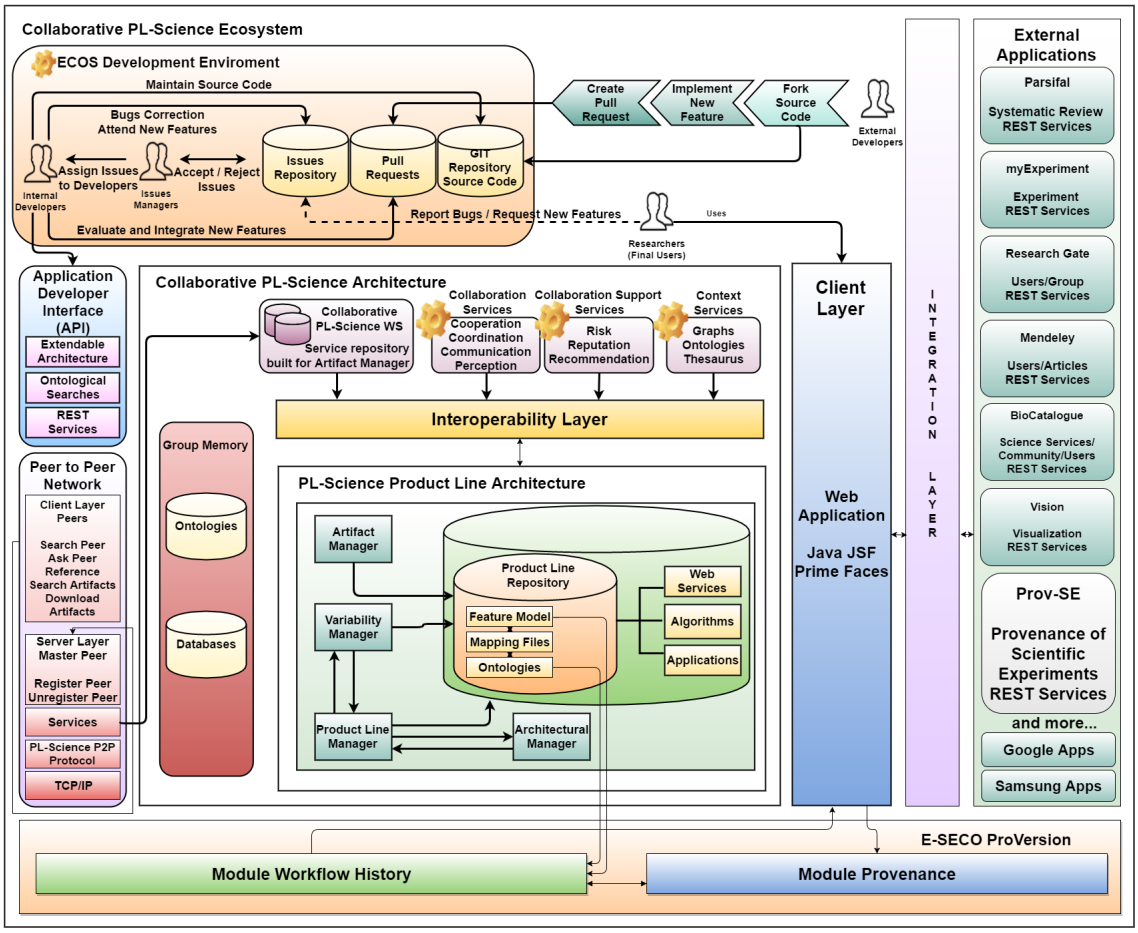

Figura 4.1: Arquitetura do E-SECO

A camada de visualização, ilustrada na figura representa a aplicação web propriamente dita, onde os cientistas interagem com a aplicação em um ambiente multiusuário, executando atividades relativas a condução de um experimento científico. A rede ponto a ponto trabalha diretamente no núcleo da aplicação, onde ocorre o gerenciamento dos artefatos da LPSC. Cada instância do E-SECO exerce o papel de um ponto na rede, compartilhando seus artefatos com outras aplicações, sendo estas conhecidas ou não. O núcleo

da aplicação faz parte da proposta Collaborative PLScience Architecture, onde elementos e serviços de colaboração foram associados à LPSC, onde estão representados os processos envolvidos na etapa de concepção de workflows científicos (Freitas et al, 2014).

A interação dos usuários externos ocorre em dois níveis, primeiro na camada de visualização da aplicação, atuando na condução de experimentos e no desenvolvimento de artefatos. O segundo nível ocorre no ambiente de desenvolvimento do ECOS, gerenciado na plataforma GitHub. Através dela, desenvolvedores externos podem auxiliar na construção da plataforma, propondo melhorias e desenvolvendo novas funcionalidades. Essas funcionalidades serão avaliadas pela equipe de desenvolvimento interno da plataforma, podendo ou não serem integradas no código fonte. Cientistas ganham um canal de comunicação, através do qual podem solicitar novas funcionalidades ou reportarem problemas na plataforma (Freitas et al, 2014).

4.2 Prov-SE

Embora a plataforma E-SECO já conte com um módulo de proveniência integrado, o E-SECO ProVersion conforme discutido anteriormente, não abrange o escopo do experimento científico com um todo, coletando e analisando apenas os dados de proveniência dos workflows.

Desta forma, a arquitetura Prov-SE (Provenance of Scientific Experiments) tem como objetivo auxiliar os pesquisadores na verificação, reprodução e reutilização de experimentos científicos através da análise e inferência de informações acerca da proveniência dos experimentos científicos, incluindo os demais artefatos que os compõe.

Esta arquitetura é composta por um web service semântico RESTFul, o qual é dividido em cinco componentes, conforme ilustrado na Figura 4.2. O primeiro componente (Web Service) é o serviço propriamente dito, que provê acesso aos recursos implementados pela ferramenta. O segundo (Service Resources) é a camada de recursos do serviço, a qual implementa os métodos oferecido pelo serviço. Os principais recursos oferecidos são:

- Buscar por experimentos similares;
- Buscar por experimentos derivados;

- Buscar reproduções do Experimento;
- Consultar os SGWfCs utilizados no experimento;
- Consultar workflows e programas utilizados direta ou indiretamente no experimento;
- Consultar pesquisadores e grupos de pesquisa envolvidos no experimento.

O terceiro (Data Handler) faz o tratamento das informações recebidas via serviço para incluí-las na ontologia. O quarto componente (Inference Layer) é responsável por executar o motor de inferência com a ontologia a fim gerar maior conhecimento dos dados, e permitir que a camada de recursos execute as consultas dos dados da ontologia.

O último componente (Ontology) é uma ontologia, estendida do modelo ProvONE, com novas classes, relacionamentos e regras, para que esta a ontologia permita modelar não só o domínio dos workflows, mas também do experimento científico com um todo.

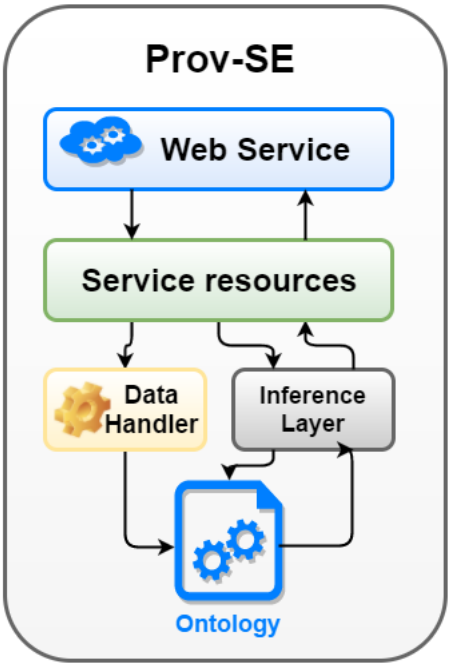

Figura 4.2: Arquitetura do Prov-SE

Para permitir que usuários e agentes de software possam descobrir, invocar e compor o web service implementado com um alto grau de automação. A arquitetura conta também com a descrição semântica do serviço, através da ontologia OWL-S (Filho and Ferreira, 2009). OWL-S é uma ontologia padrão da W3C para a descrição serviços

web, a qual descreve semanticamente o perfil do serviço, como o serviço pode ser utilizado, e como é possível interagir com o serviço (Martin et al, 2004).

4.3 Implementação

O Prov-SE foi implementado através da IDE Netbeans 8.1 na linguagem JAVA. O web service utiliza a arquitetura Restful (Representational State Transfer) através da API JAX-RS (Java API for XML-RESTful Services) implementada pelo Framework Jersey já integrado ao Netbeans. Para facilitar a importação de bibliotecas externas, o projeto faz uso da ferramenta de automação de compilação Apache Maven. A publicação do serviço foi feita através do servidor Glassfish.

O programa foi organizado segundo os padrões de implementação de web services. Desta forma, possui um pacote para a modelagem de objetos, outro para classes de implementação do serviço, e um último que contém as classes de controle e acesso à ontologia. A Figura 4.3 apresenta o diagrama de classes da ferramenta.

A ontologia foi desenvolvida com o auxilio da ferramenta Protege, utilizando a linguagem OWL 2 (Web Ontology Language). Esta ontologia foi criada a partir da importação da ontologia ProvONE, e em seguida foram adicionadas novas classes relações e regras em SWRL (Semantic Web Rule Language) para permitir que a ontologia englobasse os experimentos científicos como um todo, e fosse capaz de explicitar conhecimentos implícitos sobre a proveniência dos mesmos.

Para fazer o acesso do serviço à ontologia, foi utilizada a API do Apache Jena, um framework JAVA open source que possui suporte a ontologias na linguagem OWL. Em conjunto com a API Jena, foi utilizado o motor de inferência Pellet, o qual permitiu a execução da ontologia, e a extração de conhecimento a partir das relações e regras implementadas, e das consultas implementadas na linguagem SPARQL.

Com relação à descrição semântica do serviço, utilizou-se o Framework Jersey para a criação automatizada do arquivo WADL (Web Application Description Language) no Netbeans. As ontologias de descrição dos serviços foram feitas manualmente. As ontologias Service, Profile, Process seguem o padrão OWL-S (Martin et al, 2004). A ontologia Grounding, a qual descreve como interagir com o serviço, utiliza uma aborda-

Figura 4.3: Diagrama de classes do Prov-SE

gem alternativa para mapeamento de arquivos WADL à OWL-S proposta por (Filho and Ferreira, 2009), uma vez que a OWL-S suporta apenas serviços em WSDL (Web Services Description Language).

4.4 Cenário de uso

O pesquisador **José** desenvolveu o experimento **Exp Sum** através da plataforma E-SECO. Este experimento realiza a soma de números inteiros, utilizando o workflow **Wf Simple Sum** implementado no SGWfC **Kepler**. Um grupo de pesquisadores denominado **Nenc** criou um outro experimento na plataforma E-SECO chamado **Exp Addition** o qual permite a soma de quais quer valores reais.

Após um tempo, **José** precisou realizar um novo experimento onde ele pudesse testar a soma de números decimais. Através da plataforma E-SECO, ele buscou por experimentos similares ao que ele já havia criado.

A plataforma E-SECO então fez um requisição ao web service Prov-SE, buscando por experimentos similares ao **Exp Sum**. O Prov-SE através de sua camada de inferência consegue identificar que o experimento **Exp Addition** é similar ao **Exp Sum** uma vez que ambos utilizam o workflow **Wf Simple Sum**.

Desta forma, o Prov-SE responde à requisição do E-SECO, informando que o **Exp Addition** é um experimento similar ao **Exp Sum**. Além disso, ele fornece ao E-SECO informações adicionais sobre a proveniência do **Exp Addition**, como os pesquisadores que influenciaram na criação deste experimento, e o SGWfC que ele utilizou.

Analisando os dados oferecidos pelo E-SECO, **José** percebeu que já existe um experimento similar ao seu que permite a soma de números decimais. Sabendo que este experimento foi influenciado pelas pesquisadoras **Lenita** e **Regina**, já conhecidas nesta área, o pesquisador resolveu então reutilizar o **Exp Addition** para realizar seu teste.

As relações de proveniência utilizadas neste cenário são ilustradas na Figura 4.4. Onde as relações fornecidas ao serviço são representadas por setas contínuas em preto, e as relações inferidas pela ferramenta são representadas pelas setas pontilhadas em verde.

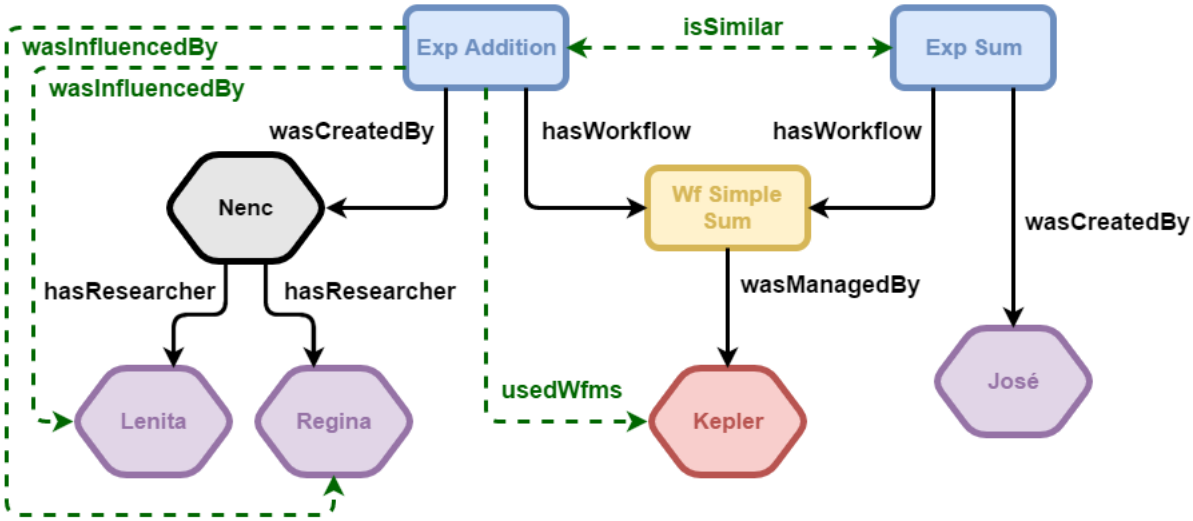

Figura 4.4: Relações de proveniência no cenário de uso do Prov-SE

5 Conclusões

O presente trabalho apresentou a arquitetura Prov-SE, um web service semântico para a análise de proveniência em experimentos científicos. Esta arquitetura foi proposta com o objetivo de auxiliar a verificação, a reprodução e o reúso de experimentos científicos através da análise e inferência de informações acerca da proveniência deste experimentos, incluindo os demais artefatos que os compõe.

A partir dos trabalhos relacionado apresentados no capítulo 3, percebe-se que a proveniência na área de experimentação cientifica já vem sendo explorada de diversas formas. Entretanto, nenhuma das abordagens apresentadas permite a análise e inferência de dados de proveniência dos experimentos científicos como um todo.

Desta forma, a arquitetura apresentada, mostra-se bastante útil à comunidade de pesquisadores, principalmente devido à sua capacidade inferir informações implícitas acerca da proveniência dos experimentos científicos. Através desta informações, o Prov-SE facilita a verificação dos experimentos realizados, bem como a reprodutibilidade e o reúso dos mesmos.

Desenvolvido inicialmente para integrar à plataforma E-SECO, como este web service foi anotado semanticamente utilizando OWL-S, ele interopera facilmente com outros sistemas. Além disso, o modelo de proveniência utilizado baseado no ProvONE, também facilita na interoperabilidade dos dados.

O cenário de uso apresentado, exemplifica o uso desta arquitetura, entretanto, a abordagem ainda carece de uma avaliação formal de suas funcionalidades. Além disso, como trabalhos futuros, é necessária a implementação da integração deste web service com a plataforma E-SECO, uma vez que o Prov-SE foi desenvolvido de forma independente. Com relação ao dados de proveniência inferidos pela ontologia, pode-se como trabalho futuro incluir novas relações e regras à ontologia, permitindo que a ferramenta tenha uma capacidade ainda maior de extração de conhecimento.

Bibliografia

Altintas, I.; Berkley, C.; Jaeger, E.; Jones, M.; Ludascher, B. ; Mock, S. **Kepler: an extensible system for design and execution of scientific workflows**. In: Scientific and Statistical Database Management, 2004. Proceedings. 16th International Conference on, p. 423–424, June 2004.

Belloum, A.; Inda, M. A.; Vasunin, D.; Korkhov, V.; Zhao, Z.; Rauwerda, H.; Breit, T. M.; Bubak, M. ; Hertzberger, L. O. Collaborative e-science experiments and scientific workflows. **IEEE Internet Computing**, v.15, n.4, p. 39–47, 2011.

Belhajjame, K.; Cheney, J.; Corsar, D.; Garijo, D.; Soiland-Reyes, S.; Zednik, S. ; Zhao, J. **Prov-o: The prov ontology**. https://www.w3.org/TR/2013/REC-prov-o-20130430/, 2013. Acessado: 14/09/2016.

Costa, F.; Silva, V.; de Oliveira, D.; Ocaña, K.; Ogasawara, E.; Dias, J. ; Mattoso, M. **Capturing and querying workflow runtime provenance with prov: A practical approach**. In: Proceedings of the Joint EDBT/ICDT 2013 Workshops, EDBT '13, p. 282–289, New York, NY, USA, 2013. ACM.

Costa, F.; Oliveira, D. d. ; Mattoso, M. **Towards an adaptive and distributed architecture for managing workflow provenance data**. In: Proceedings of the 2014 IEEE 10th International Conference on e-Science - Volume 02, E-SCIENCE '14, p. 79–82, Washington, DC, USA, 2014. IEEE Computer Society.

Cuevas-Vicenttín, V.; Kianmajd, P.; Ludäscher, B.; Missier, P.; Chirigati, F.; Wei, Y.; Koop, D. ; Dey, S. The pbase scientific workflow provenance repository. **International Journal of Digital Curation**, v.9, n.2, p. 28–38, 2014.

Cuevas-Vicenttín, V.; Ludäscher, B.; Missier, P.; Belhajjame, K.; Chirigati, F.; Wei, Y.; Dey, S.; Kianmajd, P.; Koop, D.; Bowers, S.; Altintas, I.; Jones, C.; Jones, M. B.; Walker, L.; Slaughter, P.; Leinfelder, B. ; Cao, Y. **Provone: A prov extension data model for scientific workflow provenance**. http://jenkins-1.dataone.org/jenkins/view/Documentation%20Projects/job/ProvONE-Documentation-trunk/ws/provenance/ProvONE/v1/provone.html, 2016. Acessado: 14/09/2016.

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

Deelman, E.; Blythe, J.; Gil, Y.; Kesselman, C.; Mehta, G.; Patil, S.; Su, M.-H.; Vahi, K. ; Livny, M. **Pegasus: Mapping Scientific Workflows onto the Grid**, p. 11–20. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

Deelman, E.; Gil, Y. **Managing large-scale scientific workflows in distributed environments: Experiences and challenges**. In: 2006 Second IEEE International Conference on e-Science and Grid Computing (e-Science'06), p. 144–144, Dec 2006.

Deelman, E.; Gannon, D.; Shields, M. ; Taylor, I. Workflows and e-science: An overview of workflow system features and capabilities. **Future Gener. Comput. Syst.**, v.25, n.5, p. 528–540, Mai 2009.

Filho, O. F. F.; Ferreira, M. A. G. V. Semantic web services: a restful approach. **IADIS International Conference WWW/Internet**, p. 169–180, 2009.

Freire, J.; Silva, C. T.; Callahan, S. P.; Santos, E.; Scheidegger, C. E. ; Vo, H. T. **Managing rapidly-evolving scientific workflows**. In: Proceedings of the 2006 International Conference on Provenance and Annotation of Data, IPAW'06, p. 10–18, Berlin, Heidelberg, 2006. Springer-Verlag.

Freitas, V.; David, J. M.; Braga, R. ; Campos, F. **Uma arquitetura para ecossistema de software científico**. In: 9th Workshop on Distributed Software Development, Software Ecosystems and Systems-of-Systems (WDES 2015), p. 41–48, Belo Horizonte, Brazil, 2015.

Goble, C. A.; Bhagat, J.; Aleksejevs, S.; Cruickshank, D.; Michaelides, D.; Newman, D.; Borkum, M.; Bechhofer, S.; Roos, M.; Li, P. ; others. myexperiment: a repository and social network for the sharing of bioinformatics workflows. **Nucleic acids research**, v.38, n.suppl 2, p. W677–W682, 2010.

Groth, P.; Miles, S.; Fang, W.; Wong, S. C.; Zauner, K.-P. ; Moreau, L. **Recording and using provenance in a protein compressibility experiment**. In: Proceedings of the High Performance Distributed Computing, 2005. HPDC-14. Proceedings. 14th IEEE International Symposium, HPDC '05, p. 201–208, Washington, DC, USA, 2005. IEEE Computer Society.

Groth, P.; Jiang, S.; Miles, S.; Munroe, S.; Tan, V.; Tsasakou, S. ; Moreau, L. **An architecture for provenance systems**. University of Southampton, 2006.

Gruber, T. R. Toward principles for the design of ontologies used for knowledge sharing. **Int. J. Hum.-Comput. Stud.**, v.43, n.5-6, p. 907–928, Dez. 1995.

Kim, J.; Deelman, E.; Gil, Y.; Mehta, G. ; Ratnakar, V. Provenance trails in the wings/-pegasus system. **Concurrency and Computation: Practice and Experience**, v.20, n.5, p. 587–597, 2008.

Berners-Lee, T.; Hendler, J.; Lassila, O. ; others. The semantic web. **Scientific american**, v.284, n.5, p. 28–37, 2001.

Lim, C.; Lu, S.; Chebotko, A. ; Fotouhi, F. **Prospective and retrospective provenance collection in scientific workflow environments**. In: Services Computing (SCC), 2010 IEEE International Conference on, p. 449–456, July 2010.

Martin, D.; Burstein, M.; Hobbs, J.; Lassila, O.; McDermott, D.; McIlraith, S.; Narayanan, S.; Paolucci, M.; Parsia, B.; Payne, T.; Sirin, E.; Srinivasan, N. ; Sycara, K. **Owl-s: Semantic markup for web services**. https://www.w3.org/Submission/OWL-S/, 2004. Acessado: 14/09/2016.

Michener, W.; Koskela, R.; Vieglais, D. ; Budden, A. **Data observation network for earth (dataone)**. https://www.dataone.org/, 2016. Acessado: 14/09/2016.

Missier, P.; Dey, S.; Belhajjame, K.; Cuevas, V. ; Ludaescher, B. **D-PROV: extending the PROV provenance model with workflow structure**. In: Procs. TAPP'13, Lombard, IL, 2013.

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. The open provenance model core specification (v1.1). **Future Generation Computer Systems**, v.27, n.6, p. 743 – 756, 2011.

Moreau, L.; Groth, P. **An overview of the prov family of documents**. https://www.w3.org/TR/prov-overview/, 2013. Acessado: 14/09/2016.

Oinn, T.; Li, P.; Kell, D. B.; Goble, C.; Goderis, A.; Greenwood, M.; Hull, D.; Stevens, R.; Turi, D. ; Zhao, J. **Taverna/myGrid: Aligning a Workflow System with the Life Sciences Community**, p. 300–319. Springer London, London, 2007.

Simmhan, Y. L.; Plale, B. ; Gannon, D. A survey of data provenance in e-science. **SIGMOD Rec.**, v.34, n.3, p. 31–36, Set. 2005.

Simmhan, Y. L.; Plale, B.; Gannon, D. ; Marru, S. **Performance Evaluation of the Karma Provenance Framework for Scientific Workflows**, p. 222–236. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

Sirqueira, T. F.; Dalpra, H. L.; Braga, R.; Araújo, M. A.; David, J. M. N. ; Campos, F. E-seco proversion: Manutenção e evolução de experimentos científicos. **BreSci - 10º Brazilian e-Science Workshop**, 2016.