

# PL-SCIENCE: CONECTANDO ONTOLOGIAS E MODELOS DE *FEATURES* EM UMA LINHA DE PRODUTO PARA APLICAÇÕES CIENTÍFICAS

**Gabriella Castro Barbosa Costa**

gabriellacbc@gmail.com

## **Orientadores**

**Regina Maria Maciel Braga Villela**

regina@acessa.com

**José Maria Nazar David**

jmndavid@gmail.com

## **Área de Pesquisa**

Engenharia de Software

Ano de ingresso no programa: 2011

**Resumo.** *A utilização de modelos de features permite a representação das variabilidades existentes em uma Linha de Produto de Software (LPS). No entanto, a representação das restrições existentes entre as variações das features de um produto é limitada. Para tanto, ontologias podem ser utilizadas para aprimorar os modelos de features, fornecendo informações adicionais para o domínio em questão. Este trabalho apresenta a abordagem PL-Science, que propõe o uso conjunto de modelos de features e ontologias, com o objetivo de aprimorar a semântica em uma Linha de Produto de Software Científico (LPSC) para apoiar a criação de workflows científicos.*

**Palavras-chave:** *Linha de Produto do Software, Modelo de features, Ontologias, Workflows Científicos, Alinhamento de Sequências.*

## 1. Introdução

Usuários de *workflows* científicos [Nardi 2009], na maioria das vezes, trabalham em um campo específico de investigação e, nem sempre, têm um treinamento adequado para desenvolvimento de software. Muitas vezes, eles começam uma aplicação copiando um *workflow* existente e, em seguida, ajustam o mesmo às suas necessidades. Portanto, em algumas situações, como é o caso de aplicações científicas, é vantajoso desenvolver uma coleção de produtos de software relacionados, utilizando uma abordagem de Linha de Produto de Software (LPS). Se os produtos de software são semelhantes o suficiente, então deve ser possível prever o que eles têm em comum, o que difere entre eles e, em seguida, reutilizar os aspectos comuns de forma a apoiar o desenvolvimento de aplicações no meio científico.

Segundo Clements e Northrop (2002), modelos de *features* podem ser utilizados para a representação das variabilidades existentes em uma LPS. Neste trabalho, porém, a utilização somente de modelos de *features* para a representação das restrições existentes entre as possibilidades de variação das *features* de um produto apresentou-se limitada. O uso de modelos de *features* apenas não foi suficiente para expressarmos todas as restrições existentes em um determinado domínio (como, por exemplo, o domínio de alinhamento / sequenciamento genético). Tais modelos não oferecem a possibilidade de expressar toda a semântica necessária em aplicações científicas. Restrições por eles representadas estão relacionadas, sobretudo à semântica envolvida nos relacionamentos entre *features*. Portanto, optou-se por unir modelos de *features* com ontologias, de forma a obtermos uma melhor capacidade de especificação dos produtos (*workflows*) a serem gerados através da LPS.

Este artigo está organizado da seguinte forma: na seção 2 são apresentados os principais conceitos que foram usados como base para a criação da abordagem PL-Science, descrita neste trabalho. A seção 3 apresenta os trabalhos relacionados e uma comparação destes com a abordagem proposta. Na seção 4 são apresentadas as principais contribuições deste trabalho e a seção 5 detalha o estado atual do mesmo. Finalmente, na seção 6 é descrita a forma de avaliação dos resultados obtidos com a abordagem PL-Science.

## 2. Fundamentação Teórica

Linha de Produto de Software (LPS) é uma abordagem que busca melhorar a reusabilidade, pois o reuso é planejado, ativado, e executado [Clements e Northrop 2002]. Além disso, LPS também tem como objetivo melhorar a capacidade de manutenção, porque, ao contrário de copiar e modificar o que for necessário, os artefatos que são usados para a criação dos membros da LPS são geridos em conjunto.

*Workflows* científicos diferem-se de *workflows* aplicados a sistemas de negócios por possuírem características que lhes acrescentam maior complexidade, tais como: (i) fluxos com um grande número de etapas; (ii) volatilidade dos fluxos, que devem poder ser alterados frequentemente, durante o processo de avaliação de hipóteses científicas; (iii) necessidade de parametrização para grande número de tarefas [Nardi 2009].

Modelos de variabilidade são fundamentais para o desenvolvimento e gestão de LPS. Eles podem conter conceitos relacionados a decisões, *features* ou pontos de

variação, dependendo do nível de abstração dos mesmos. Modelos de *features* são uma das notações proeminentes usadas para a modelagem de variabilidades [Andersen *et al.* 2012]. Assim, no desenvolvimento de uma LPS, uma das primeiras atividades a serem realizadas é a análise de *features*, que identifica as características externamente visíveis dos produtos da LPS e as organizam em um modelo de *features*. É através da modelagem de *features* que são especificadas as funcionalidades comuns e variáveis da família de aplicações a serem geradas por meio da linha de produto.

Em Ciência da Computação, uma ontologia define uma especificação formal e explícita de uma conceitualização compartilhada [Guarino 1998]. Ontologias fornecem um modelo formal e manipulável de um domínio. Sua utilização reúne benefícios como reuso, compartilhamento de conhecimento, portabilidade, manutenção e confiabilidade, partindo-se do princípio que elas representam uma conceitualização compartilhada. Na literatura, o alinhamento entre ontologias é comum, porém a contribuição desta pesquisa está no alinhamento entre modelos de *features* e ontologias. Tem-se como proposta extrair as semânticas particulares dos dois tipos de modelos, ou seja, o modelo de *features* será utilizado para a representação da variabilidade da LPS e a ontologia para a especificação de restrições formais e utilização de inferências.

### 3. Trabalhos Relacionados

Johansen *et al.* (2010) exploram modelos de *features* e ontologias, apresentando mecanismos de mapeamento que podem ser estabelecidos entre estes. Embora a abordagem PL-Science faça uso de ontologias em conjunto com modelos de *features*, como em Johansen *et al.* (2010), o objetivo principal desta não é explorar os possíveis mecanismos de mapeamento que podem ser realizados entre estes modelos. Tem-se como propósito utilizar as vantagens oferecidas pelo uso conjunto de modelos de *features* e ontologias para, ao final, obter uma LPS para geração de *workflows* científicos em um domínio previamente estabelecido.

Asikainen *et al.* (2007) propõem o uso de uma ontologia de domínio para a modelagem das variabilidades em LPS chamada Kumbang. Esta ontologia unifica a modelagem de *features* e a modelagem arquitetural de famílias de produtos de software. A ontologia Kumbang é descrita utilizando UML. Por ser fortemente adotada pelos profissionais da área de tecnologia da informação, a UML pode ser utilizada para os mais diversos tipos de modelagem, inclusive para a modelagem de ontologias. Esta representação, porém, não oferece o mecanismo de inferências. Portanto, a ontologia utilizada na abordagem PL-Science utiliza OWL, devido ao fato da possibilidade de utilização de mecanismos de inferência em conjunto com esta linguagem.

Em Czarnecki *et al.* (2006) é explorada a relação entre modelos de *features* e ontologias. É proposta a adição de 'extensões' ao modelo de *features*, de forma a aproximá-lo do formalismo existente nas ontologias. Diferentemente de Czarnecki *et al.* (2006), esta proposta não tem como propósito melhorar a representação de modelos de *features*. Partimos da ideia de que a utilização em conjunto de dois modelos já definidos pode trazer vantagens com relação à adição de semântica no contexto de LPS. Na abordagem PL-Science, o uso em conjunto de modelos de *features* e ontologias tem como objetivo final a geração de uma LPS para *workflows* científicos.

Em Silva (2011), são apresentados modelos semânticos expressos em ontologias para representação de LPS. O autor afirma que esta forma de representação é importante para que possa existir um processo automatizado de construção, configuração e manutenção de uma LPS. Neste trabalho é descrita uma representação semântica de LP, que é utilizada como uma meta-representação de LPS. Assim, através desta representação, é possível descrever LPS usando ontologias. Comparando este trabalho à abordagem PL-Science, tem-se como diferencial que na abordagem PL-Science, modelos de *features* são utilizados em conjunto com ontologias, ou seja, as vantagens de ambos os modelos podem ser utilizadas para a representação da variabilidade e da semântica em LPS.

Em Filho *et al.* (2012) são propostos métodos para adicionar informações de domínio, além de descrições de variabilidades, utilizando uma ontologia ‘topo’ que especifica conceitos genéricos e relações em uma LPS. Tal abordagem reusa o modelo de *features* da LPS, adicionando descrições semânticas à LPS, porém, não modifica a notação já existente do modelo de *features* utilizado. Neste trabalho, de forma a aumentar a semântica de uma LPS, em um momento inicial, o modelo de *features* é automaticamente mapeado em uma ontologia. Tal abordagem é definida como uma “Linha de Produto de Software Semântica” - SSPL (*Semantic Software Product Line*), na qual os artefatos reutilizáveis do núcleo de ativos da LPS e outros elementos organizacionais encontram-se relacionados ao modelo de domínio da LPS, sendo que estes relacionamentos são expressos formalmente através de uma ou mais ontologias. Comparando a abordagem SSPL com a abordagem PL-Science, tem-se que a abordagem SSPL não foi desenvolvida para um domínio específico, ou seja, é genérica. Já a abordagem PL-Science, tem como foco o desenvolvimento de LPS para software científico. Portanto, vale ressaltar que neste domínio de software científico é exigido um grau maior de formalismo para especificação de restrições, principalmente se for considerado que determinados experimentos científicos podem conter processos que envolvem riscos, como vidas humanas, por exemplo. Portanto, na abordagem PL-Science, todo o processo de mapeamento entre modelos de *features* e ontologia deve ser completamente realizado com a interferência do engenheiro de domínio, de forma que não se perca nenhuma informação e que as restrições a serem estabelecidas entre as *features* sejam criadas com base no conhecimento deste engenheiro a respeito do domínio científico em questão. Na abordagem PL-Science, portanto, não se tem um passo de tradução de um modelo de *features* para uma ontologia; é criada uma relação entre estes modelos através de um arquivo de mapeamento, que interliga as *features* do modelo com classes de uma ontologia já existente. Ou seja, é realizado um mapeamento entre o modelo de *features* e uma ontologia já existente, sendo esta consagrada no domínio científico em questão de forma que este mapeamento tira proveito justamente do que cada modelo pode oferecer de melhor. Portanto, o modelo de *features* fornece a representação da variabilidade da LPS e a ontologia é utilizada para a representação da semântica do domínio que não é coberta pela forma de representação da variabilidade de modelos de *features*. Assim, são utilizados dois modelos formais, sem a necessidade de modificar ‘os formatos’ de nenhum deles.

#### 4. Contribuições

Tendo por base os benefícios da utilização de LPS no contexto de aplicações com um elevado grau de semelhança, este trabalho tem como objetivo principal propor uma abordagem cujo foco é o uso conjunto de ontologias e modelos de *features* para o desenvolvimento de uma LPS, voltada para apoiar o desenvolvimento e/ou a execução de aplicações científicas. Através da conexão entre esses modelos, tem-se como objetivo final da LPS a geração de um *workflow* científico com as atividades a serem realizadas de acordo com o domínio escolhido e com as necessidades do usuário [Costa *et al.* 2012a] Esta abordagem, chamada de PL-Science, visa auxiliar o cientista durante o processo de especificação das tarefas necessárias para uma experimentação científica. O objetivo geral desta abordagem pode ser decomposto nos seguintes objetivos específicos: (i) propor uma arquitetura para a implementação de uma LPS voltada para aplicações científicas; (ii) adicionar mais semântica na especificação das variabilidades de uma LPS, através do uso de ontologias em conjunto com modelos de *features*; (iii) implementar uma LPS, apresentando três cenários de uso, no domínio da bioinformática, para sequenciamento / alinhamento genético.

#### 5. Estado Atual do Trabalho

A arquitetura da LPS proposta neste trabalho foi projetada usando gerentes, conforme apresentado na Figura 1 [Costa *et al.* 2012b] e detalhado a seguir: (i) Gerente de Variabilidades (*Variability Manager*): é responsável pela gestão e conexão dos modelos (modelo de *features* e ontologia) que representam as variabilidades na LPS; (ii) Gerente da Linha de Produto (*Product Line Manager*): é responsável pela interação com o usuário da LPS e pela coordenação do comportamento dos demais gerentes presentes na arquitetura; (iii) Gerente Arquitetural (*Architectural Manager*): responsável pela arquitetura da LPS. Tem como objetivo possibilitar a utilização de arquiteturas configuráveis na LPS.

Os gerentes interagem com o **Repositório da LP** e também com repositórios distribuídos. Estes repositórios incluem os artefatos necessários para a geração dos *workflows* da LPS. Como exemplos destes artefatos pode-se citar os modelos de *features*, as ontologias, os arquivos de mapeamento entre ontologias e modelos de *features*, *web services*, algoritmos, entre outros.

Considerando a abordagem de uso de modelo de *features* e ontologias, foi proposta uma metodologia para utilização da LPSC, que é composta de nove passos principais, divididos em duas fases: (1) Desenvolvimento do Núcleo de Artefatos ou Engenharia de Domínio, que descreve os artefatos produzidos na fase de engenharia de domínio e (2) Desenvolvimento do Produto usando o Núcleo de Artefatos ou Engenharia de Aplicação, que descreve a configuração dos produtos, utilizando o núcleo de artefatos. A Fase 1 é composta pelos seguintes passos: (1) Definição do escopo da linha de produto; (2) Definição do modelo de *features* de acordo com as possibilidades de variação existentes no domínio escolhido; (3) Definição da ontologia de domínio, que contém as ‘restrições’ a respeito do domínio escolhido que não podem ser expressas apenas através do modelo de *features*; (4): Realização do mapeamento entre os termos do modelo de *features* e a ontologia do domínio; (5) Definição de um modelo de *workflow* base, de acordo com o domínio escolhido; (6) Definição e conexão dos outros

artefatos (algoritmos, Web Services e outras aplicações) da linha de produto que serão armazenados no repositório. Já a Fase 2 possui os seguintes passos: (1) Seleção das *features* disponíveis no domínio escolhido, de acordo com o produto a ser desenvolvido (*workflow* científico). Esta seleção é baseada tanto no modelo de *features* e seu mapeamento para o modelo ontológico, quanto no(s) modelo(s) de *workflow*(s) disponível para o domínio. Neste passo deve-se levar em consideração o escopo da LPS. Este passo é controlado pelo Gerente de Variabilidade; (2) Para cada tarefa do *workflow*, são analisadas as possibilidades de variação (definidas no Passo 1) e o usuário define qual o algoritmo, Web Service ou Aplicação que será ‘instanciado’ no *workflow* base; (9) Geração de um arquivo especificado em XML com o detalhamento das tarefas que deverão compor o *workflow* científico, considerando os artefatos selecionados.

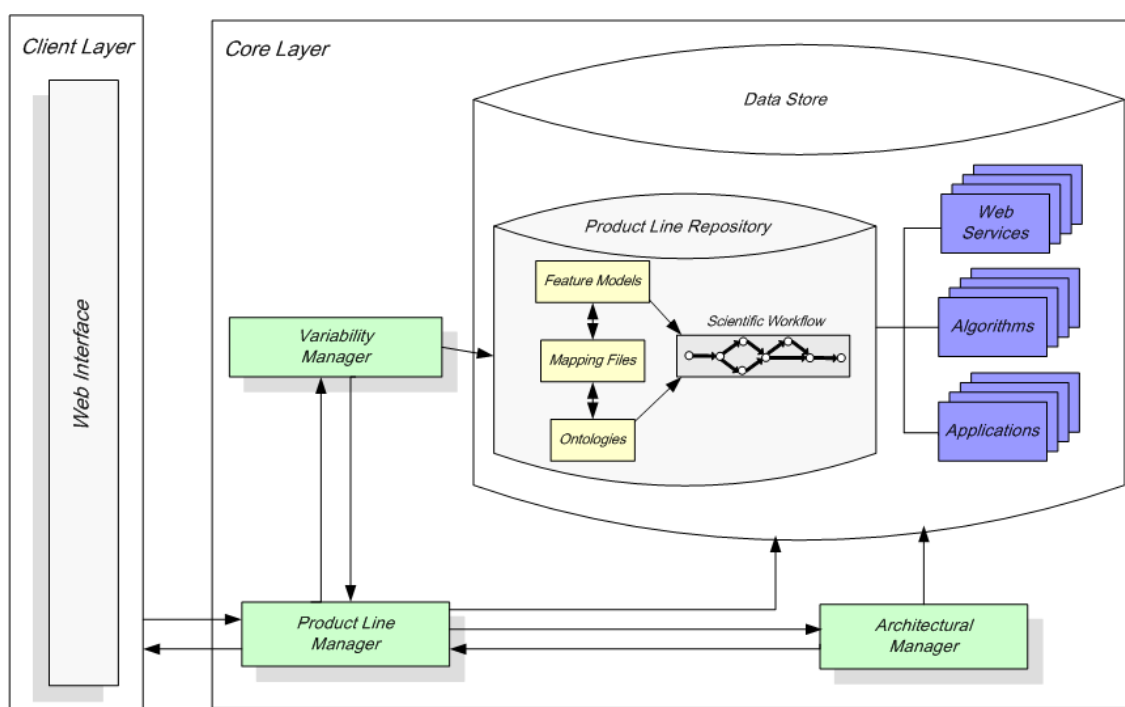
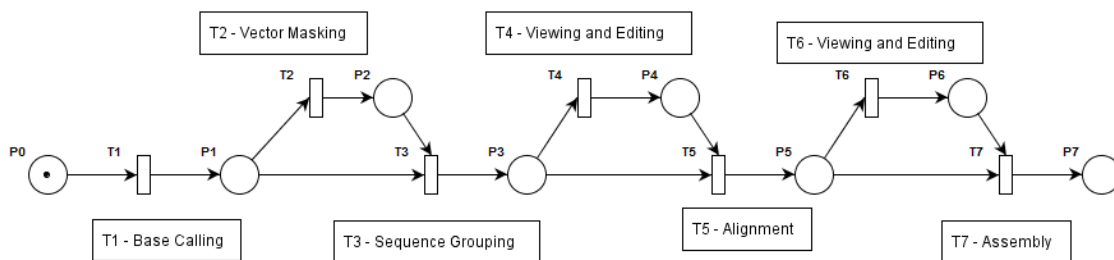


Figura 1. Visão Geral da Arquitetura

## 6. Avaliação dos Resultados

De forma a avaliar a proposta apresentada, a PL-Science está sendo aplicada em três estudos de caso distintos, no domínio da Bioinformática. Esta avaliação tem como objetivo a geração de um *workflow* científico, especificado no formato XML, que englobe atividades envolvidas no processo de sequenciamento / alinhamento genético.

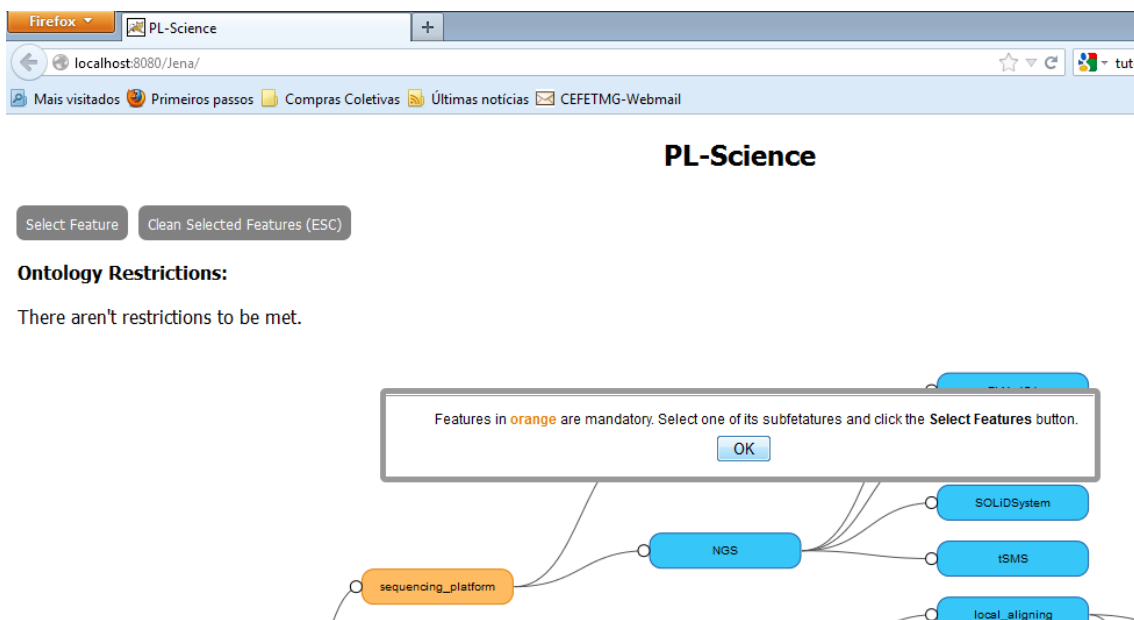
Um dos estudos de caso que estão sendo conduzidos, baseado em Arbex (2009), engloba a definição de um *workflow* que seja capaz de realizar as tarefas T1 (*base calling*), T2 (*vector masking*) e T3 (*sequence grouping*), exibidas na Figura 2. Esta figura apresenta o modelo de *workflow* base do domínio escolhido, que é definido no passo 5, conforme mencionado na metodologia da abordagem PL-Science.



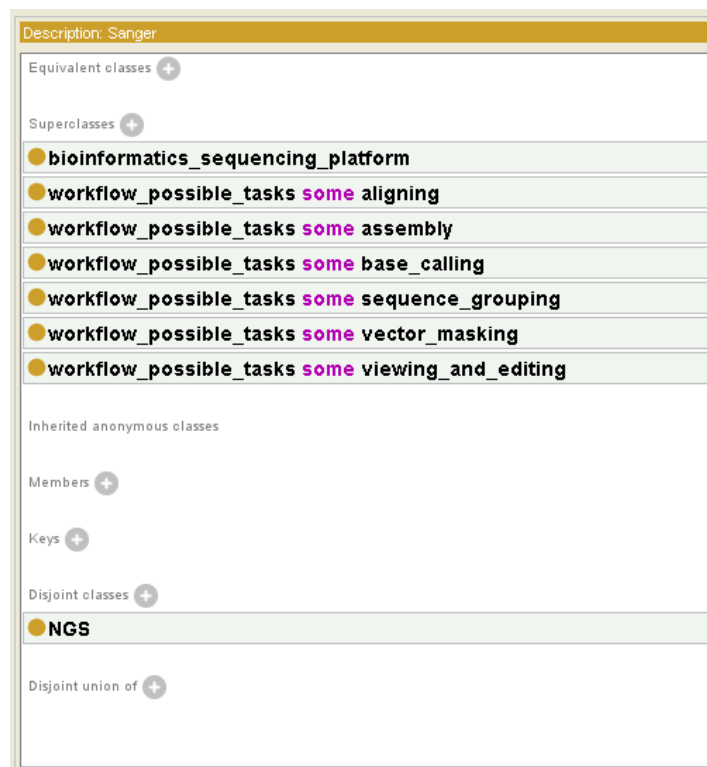
**Figura 2. Modelo do *Workflow* base para alinhamento/sequenciamento genético**

Ao executar a aplicação desenvolvida para dar suporte à PL-Science, é exibida a tela da Figura 3. De acordo com as restrições dos modelos de domínio (modelo de *features* e ontologia), a aplicação exibe caixas de diálogo ao usuário, de forma a auxiliá-lo na definição / seleção das *features*, de acordo com a necessidade e o objetivo do *workflow* a ser gerado.

Como primeiro passo para execução da aplicação, é necessária a seleção da *feature* que estabelece a plataforma de sequenciamento a ser utilizada. No caso deste estudo de caso, seria o sequenciamento pelo método de Sanger [Sanger *et al.*, 1977]. Após a seleção desta *feature*, a aplicação, através do Gerente de variabilidades, analisa-se quais as restrições relacionadas à mesma. Neste estudo de caso, as restrições deste método de sequenciamento abrangem as tarefas (descritas no modelo de *workflow*, Figura 2) que poderão compor o *workflow* final da aplicação. A Figura 4 exibe as restrições da classe Sanger.



**Figura 3. Tela inicial da Abordagem PL-Science**



**Figura 4. Restrições da classe Sanger**

Após a verificação de todas as restrições da *feature* selecionada, o Gerente da LP apresenta ao usuário uma nova mensagem, sugerindo os próximos passos na seleção. De acordo com este estudo de caso, são apresentadas todas as possibilidades de tarefas, levando em consideração o método de Sanger. Neste ponto, é exibida uma mensagem ao usuário para que ele selecione, entre as *features* exibidas em destaque, a última tarefa do *workflow* a ser criada. Neste exemplo, foi selecionada a *feature* *sequence\_grouping*, que corresponde à tarefa T3, do modelo de *workflow*. Novamente, após a seleção da *feature*, todas as restrições relacionadas tanto às tarefas que serão realizadas (da tarefa T1 até a tarefa T3) são analisadas e são mostrados ao usuário os *web services*, algoritmos ou aplicações que são capazes de realizar as tarefas previamente definidas. As descrições destes serviços são estabelecidas como propriedades dos mesmos e estas informações serão utilizadas para a geração do arquivo XML que contém a especificação do *workflow* final do usuário.

Tem-se como trabalhos futuros a finalização deste estudo de caso e a realização dos outros dois, como forma de verificar o funcionamento da abordagem PL-Science.

## Referências

- Andersen, N., Czarnecki, K., She, S., Wąsowski, A. (2012) "Efficient synthesis of feature models", In *Proceedings of the 16th International Software Product Line Conference - Volume 1 (SPLC '12)*, Vol. 1. ACM, New York, NY, USA, p. 106-115.
- Arbex, W. (2009) "Modelos computacionais para identificação de informação genômica associada à resistência ao carrapato bovino". Tese de Doutorado. Doutorado em Engenharia de Sistemas e Computação - Universidade Federal do Rio de Janeiro.



- Asikainen, T., Männistö, T., and Soininen, T. (2007) “Kumbang: A domain ontology for modelling variability in software product families”. *Adv. Eng. Inform.*, 21:23–40.
- Clements, P. and Northrop, L. (2002) “Software Product Lines: Practices and Patterns”, Addison-Wesley Publishing Company, Boston.
- Costa, G. C. B., Braga, R., David, J. M. N. (2012a) “Apoiando uma Linha de Produtos de Workflow Científico através da associação de Ontologias e Modelos de Features”, In: II Workshop de Teses e Dissertações do CBSOft (WTDSOft), III Congresso Brasileiro de Software: Teoria e Prática (CBSOft), p. 1-5, Natal - RN.
- Costa, G. C. B., Braga, R., David, J. M. N., and Campos, F. C. A. (2012b) “Connecting feature models and ontologies in software product lines”, In: *International Conference on Enterprise Information Systems, Proceedings of ICEIS*, Wroclaw.
- Czarnecki, K., Peter Kim, C. H., and Kalleberg, K. T. (2006) “Feature models are views on ontologies”, In: *Proceedings of the 10th International on Software Product Line Conference*, pages 41–51, Washington, DC, USA. IEEE Computer Society.
- Filho, J. B. F., Barais, O., Baudry, B., Viana, W., Andrade, R. M. C. (2012) “An approach for semantic enrichment of software product lines”, In: *Proceedings of the 16th International Software Product Line Conference (SPLC '12)*, Vol. 2. ACM, New York, NY, USA, p. 188-195.
- Guarino, N. (1998) “Formal Ontology and Information Systems”, In: *Proceedings of the 1st International Conference*, Trento, Italy. IOS Press, Amsterdam, Netherlands.
- Johansen, M. F., Fleurey, F., Acher, M., Collet, P., and Lahire, P. (2010) “Exploring the Synergies Between Feature Models and Ontologies”, In: *International Workshop on Model-driven Approaches in SPLE*, pages 163–171, Jeju Island, South Korea.
- Nardi, A. R. (2009) “Uma arquitetura de baixo acoplamento para execução de padrões de controle de fluxo em grades”. Tese de Doutorado. Instituto de Matemática e Estatística, Universidade de São Paulo.
- Sanger, F., Nicklen, S., Coulson, A. R. (1977) “DNA sequencing with chain-termination inhibitors”, In: *Proc. Natl. Acad. Sci. USA*, v.74, p.5463-5467.
- Silva, A. P. (2011) “Uma Linha de Produto de Software baseada na Web Semântica para Sistemas Tutores Inteligentes”. Tese de Doutorado. Doutorado em Ciência da Computação. Universidade Federal de Campina Grande.