# A Framework for Creation of Linked Data Mashups: A Case Study on Healthcare

Gabriel Lopes
Instituto Federal do Ceará - IFCE
Fortaleza, CE, Brasil
gabriellopes9102@gmail.com

Vânia Vidal
Universidade Federal do Ceará - UFC
Fortaleza, CE, Brasil
vvidal@lia.ufc.br

Mauro Oliveira
Instituto Federal do Ceará - IFCE
Aracati, CE, Brasil
amauroboliveira@gmail.com

## ABSTRACT

Linked Data promotes the publication of structured data on the Web, easing the development of an homogeneized-view over heterogeneous sources, called *Linked Data Mashup view* (LDM view). But the development of this homogeneized-view still is a challenging task. This article proposes a *framework* Ontology-based that aims to ease the process of creation of LDM views. This *framework* allow users without specifics knowledge to create their own applications, based on their needs. We also present a case study in which we use our approach to develop an integrated view over two heterogeneous sources from Brazilian Public Health System.

## Keywords

Linked Data Mashup; Data Integration ; Semantic Integration; Semantic Mediator

## 1. INTRODUÇÃO

A iniciativa *Linked Data* [2] promove a publicação de bases de dados anteriormente isoladas como fontes RDF interligadas. O *Linked Data* trouxe novas oportunidades para o desenvolvimento de aplicações semânticas. Essas aplicações consomem os dados de um *Linked Data Mashup* (LDM), uma aplicação *web* que promove a integração de bases de dados através da combinação, agregação e transformação de fontes possivelmente heterogêneas [5]. Existem exemplos de aplicações *Linked Data Mashup* em diversos domínios, como na Saúde [1, 11] e na Música [6]. O DrugBank [11], por exemplo, é um *mashup* que integra diversas fontes de dados abertos, reunindo informações sobre mais de 5000 medicamentos, e pode ser utilizado pela comunidade para o desenvolvimento de diversas aplicações.

Entretanto, para o desenvolvimento de aplicações LDM, é necessário a obtenção de uma visão homogênea dos dados provenientes de diferentes fontes, muitas vezes com formatos e/ou *schemas* heterogêneos. De acordo com [9], o desenvolvimento de uma visão integrada, chamada de *visão Linked Data Mashup*, não é uma tarefa simples, pois requer a *integração semântica* das fontes de dados, i.e, as diferenças semânticas entre as fontes precisam ser conciliadas. Os principais desafios para a integração semântica no contexto de *Linked Data* estão relacionados aos seguintes fatores: (i) heterogeneidade das fontes de dados e dos vocabulários usados; (ii) qualidade dos dados que podem estar fragmentados, incompletos, incorretos ou inconsistentes; (iii) resolução de conflito de URIs, visto que diferentes URIs podem se referir ao mesmo objeto. Nesse processo, são necessários conhecimentos específicos que dificultam para um usuário geral, um gestor da área da Saúde por exemplo, criar seu próprio *mashup*.

Esse artigo propõe um *Framework* para facilitar a criação de *visões Linked Data Mashup*. Um dos objetivos da proposta, é permitir que usuários finais, com poucos conhecimentos em Computação, desenvolvam suas próprias visões integradas sobre informações distribuídas. Na nossa abordagem, uma visão LDM é especificada baseada na metodologia proposta em [9]. Essa especificação indica que as fontes de dados utilizadas para o *mashup* foram integradas semanticamente, total ou parcialmente. A partir dessa especificação, usuários finais podem construir seus próprios *mashups*, chamados de *visões de aplicação mashup*, por meio de uma *interface web*, de maneira fácil, intuitiva e sem a necessidade de conhecimentos específicos. Um dos diferenciais da nossa abordagem, é que a integração semântica é realizada uma única vez, e a partir de então, a especificação gerada será reutilizada para a criação de aplicações posteriores. Para isso, fazemos uma reescrita da especificação original, utilizando os parâmetros passados pelo usuário por meio da *interface web* gráfica. Além disso, apresentamos um estudo de caso em que o *framework* proposto é utilizado para especificar uma *visão de mashup* chamada de *Datasus_hub*, que integra semanticamente duas bases heterogêneas do Sistema Público de Saúde brasileiro. Nesse estudo de caso, também demonstramos como um usuário final pode criar seu próprio *mashup*.

O restante do artigo está organizado da seguinte forma. A Seção 2 apresenta o *framework* proposto. Na Seção 3, nossa abordagem é utilizada para construir uma visão integrada sobre duas bases da Saúde. Finalmente, a Seção 4 apresenta nossas expectativas e trabalhos futuros.

## 2. ESPECIFICAÇÃO DO FRAMEWORK

### 2.1 Visão geral

Nessa Seção, apresentamos o *framework* com 4 camadas baseado em Ontologias (Figura 1), que tem como objetivo



*WebMedia '16, November 08-11, 2016, Teresina, PI, Brazil*

 ISBN 978-1-4503-4512-5/16/11...$15.00
DOI: http://dx.doi.org/10.1145/2976796.2988185

permitir que usuários finais obtenham uma visão integrada de informações distribuídas. Na nossa abordagem, uma *integração semântica* é realizada baseada nos conceitos de [9], brevemente discutido na Seção 1, que apresenta cinco passos para a especificação e materialização de uma visão LDM. O resultado dessa integração semântica é a especificação de uma *Visão Integrada* $M$, que será utilizada para a criação de *visões de aplicação mashups*, ou simplesmente *visões de mashup*. Cada visão de *mashup* é criada por um usuário final por intermédio de uma *interface web* gráfica. As quatro camadas do *framework* são descritas a seguir.

### 2.1.1 Framework 4 camadas

Na Camada de Integração Semântica, a *Ontologia de Domínio* $O_M$ representa uma visão homogênea sobre as fontes de dados $S_1, ..., S_n$, que deseja-se unificar. Na Camada de Dados, cada fonte $S_i$ é descrita por uma ontologia $O_{Si}$ e exporta uma ou mais visões, chamadas de *Visões Exportadas*. Cada visão exportada $E_i$ é composta por uma *Ontologia Exportada* $O_{Ei}$, cujo vocabulário é um subconjunto de $O_M$, e um conjunto de mapeamentos $M_{Ei}$, que mapeia os conceitos de $O_{Si}$ em $O_{Ei}$. Também é na Camada de Visões Exportadas que são definidas as regras para descobertas de *Links Semânticos*, $EL_1, ..., EL_m$, que apontam a similaridade entre dois objetos do mundo real em duas bases distintas.

Usuários finais utilizam os parâmetros $(O_{Vi}, F_i)$ para construir visões de *mashup* baseadas na especificação da *Visão Integrada* previamente criada. A *Ontologia de Aplicação* $O_{Vi}$, cujo vocabulário deve ser um subconjunto do vocabulário de $O_M$, representa os conceitos de interesse do usuário; enquanto $F_i$ representa um conjunto de filtros (e.g. cidade, ano etc.) que serão aplicados sobre os dados. Os usuários finais interagem com o *framework* por meio de uma *interface web* gráfica, que permite, de forma intuitiva, a construção de uma visão de *mashup*. Como será discutido na subseção 2.3, para a materialização da visão $V_i$, é necessário combinar os parâmetros $(O_{Vi}, F_i)$ com a especificação da visão integrada $M$, gerando uma especificação de $V$. Essa especificação $V$ denota uma integração semântica parametrizada sobre as bases $S_1, ..., S_n$. A especificação de $V$ será então utilizada na materialização dos dados, realizada automaticamente com o auxílio de *frameworks* específicos.

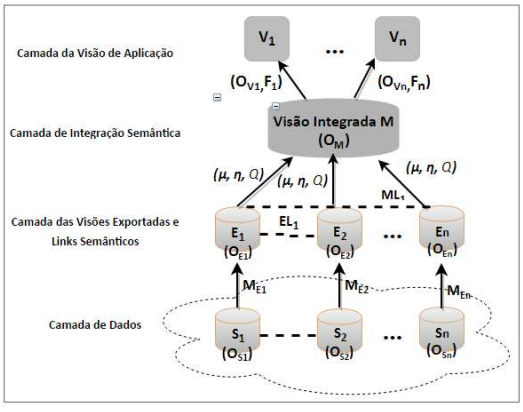


**Figura 1: *Framework* 4 Camadas**

## 2.2 Especificação da Visão Integrada

Seja $M$ uma visão em um formato homogêneo sobre as fontes de dados que deseja-se unificar. A especificação de M é uma 6-tupla:

- $\lambda_m = \{M, O_M, E_M, EL_M, \mu_m, Q\}$.

Onde M é o nome da visão integrada; $O_M$ é a ontologia de domínio; $E_M$ representa o conjunto das visões exportadas; $EL_M$ define o conjunto de regras para descoberta dos *links* semânticos; $\mu_m$ determina as regras de fusão e, finalmente, $Q$ determina os critérios para avaliação de qualidade das fontes de dados. O processo de especificação da visão $M$ consiste em 5 etapas:

1. **Seleção das fontes de dados relevantes para a aplicação**. As fontes de dados $S_1, ..., S_n$ são escolhidas de acordo com a relevância para a aplicação. Por exemplo, se a aplicação requer dados geográficos, GeoNames[1]pode ser uma escolha. Além disso, deve haver uma ontologia $O_{Si}$ descrevendo cada fonte $S_i$.

2. **Modelagem da Ontologia de Domínio**. Denotada por $O_M$, essa ontologia representa todos os conceitos do *framework* que podem ser utilizados para a criação das visões de aplicação *mashup*.

3. **Especificação das Visões Exportadas**. A especificação de uma $E_i \in E_M$ consiste em: (i) modelar a ontologia exportada $O_{Ei}$, cujo vocabulário deve ser um subconjunto do vocabulário de $O_M$ e (ii) especificar o conjunto de mapeamentos $M_{Ei}$ que mapeia os conceitos de $O_{Si}$ para $O_{Ei}$.

4. **Especificação dos conjuntos de Links Semânticos**. Cada conjunto $EL_i \in EL_M$ denota regras para definir que $x$ objetos do mundo real em visões exportadas distintas, $E_u$ e $E_v$, representam uma mesma entidade.

5. **Especificação as regras de fusão e de avaliação de qualidade**. As regras de fusão $\mu_m$ explicitam como duas representações distintas de um mesmo objeto do mundo real serão combinadas em uma única representação. As regras de avaliação $Q$ são utilizadas para quantificar a qualidade das fontes de dados.

## 2.3 Construção de uma Visão de Aplicação

Para a construção de uma Visão de Aplicação *Mashup*, denotada por $V$ na Figura 1, são necessárias 3 etapas: (i) geração da especificação $V$ sobre $M$; (ii) geração da especificação $V$ sobre as fontes de dados e (iii) materialização de $V$. Essas etapas são descritas a seguir.

### 2.3.1 Geração da Especificação de V sobre M

A especificação de V sobre M é uma tupla $V = (O_V, F_V)$, onde $O_V$ é a ontologia de aplicação, que deve ser um subconjunto da ontologia de domínio $O_M$ e representa os conceitos de interesse do usuário; e $F_V$ é um conjunto de filtros definidos sobre os conceitos de $O_V$. A especificação de $V$ pode ser realizada através de uma *interface web* gráfica, onde, por meio de uma seleção intuitiva, o usuário define a ontologia de aplicação e os filtros.

[1]http://www.geonames.org/

### 2.3.2 Geração da Especificação de $V$ sobre as Fontes de Dados

Nessa etapa, a especificação de $V$ sobre as fontes de dados $S_1, ..., S_n$ é gerada através da combinação dos parâmetros do usuário com a especificação da Visão Integrada $M$. Seja os parâmetros do usuário $\delta = (O_V, F_V)$, a combinação de $\delta$ com $\lambda_m$ (subseção 2.2) resulta em: $\lambda_v = \{V, O_V, F_V, E_V, EL_V, \mu_v, Q\}$. Onde $V$ é o nome da visão de aplicação. As visões exportadas e as regras de fusão são definidas como se segue:

- $\forall E_{Vi} \in E_V$, $O_{EVi} = O_V \cap O_{EMi}$, onde $O_{EMi} \in E_{Mi}$. Os conjuntos de mapeamentos são definidos de forma que: $\tau \in M_{EVi}$ é uma regra de mapeamento de $C_S$ para $C_T$, sse $C_S \in O_{Si}$ e $C_T \in O_{EVi}$
- $\mu_{vi} \in \mu_v$ é uma regra de fusão para duas entidades, $C_U$ e $C_V$, sse $\mu_{vi} \in \mu_m$ em $\lambda_m$ e ($C_U \in O_{EVi}$ e $C_V \in O_{EVj}$).

Como as fontes de dados são as mesmas, $Q$ não é alterado. A especificação dos *links semânticos* também permanecem inalterados de $\lambda_m$ para $\lambda_v$. Como o vocabulário de $O_V$ é um subconjunto do vocabulário de $O_M$, quaisquer regras de *links* $EL_i \in EL_M$ em $\lambda_m$ também serão válidas para $\lambda_v$.

### 2.3.3 Materialização da Visão de Aplicação Mashup ($V$)

No nosso *framework*, a materialização de $V$ é realizada automaticamente baseada na especificação $\lambda_v$ e consiste em três passos. 1. **Materialização das Visões Exportadas**: nessa etapa as fontes de dados $S_{Vi}$ são traduzidas para o vocabulário de $O_{EVi}$, utilizando os mapeamentos $M_{EVi}$. 2. **Materialização dos Links Semânticos**: dada uma especificação de *links* $EL_{Vi}$ sobre um conjunto $E_V$ de visões exportadas, essa etapa computa os *links owl:sameAs* entre entidades similares, mas em visões exportadas distintas. 3. **Materialização da Visão Aplicação Mashup**: essa etapa materializa a visão de *mashup* $V$ aplicando as regras de fusão sobre a materialização das visões exportadas e dos links semânticos. Nessa etapa serão avaliadas as fontes de dados, segundo critérios de qualidade e também serão resolvidas possíveis inconsistências.

# 3. ESTUDO DE CASO

Nessa Seção, utilizamos o *framework* apresentado na Seção 2 para desenvolver uma *visão integrada* sobre duas bases do Sistema Público de Saúde brasileiro (SUS): Sistema de Informações sobre Nascidos Vivos (SINASC) e SUS eletrônico (e-SUS). Além disso, demonstramos como construir *visões de aplicação mashup* a partir dessa visão integrada, que chamamos de *Datasus_hub*.

## 3.1 Datasus_hub

*Datasus_hub* foi desenvolvido seguindo os passos descritos na subseção 2.2, apresentados a seguir.

### 3.1.1 Fontes de Dados

As fontes de dados selecionadas foram SINASC e e-SUS, respectivamente representadas por $S_{sinasc}$ e $S_{esus}$ na especificação. Ambas as fontes estão em um formato de dados relacional e possuem *schemas* distintos. A base SINASC contém informações sobre uma gestação e sobre o recém-nascido, como: a quantidade de consultas pré-natal da gestante; a quantidade de abortos em gestações anteriores; o peso do recém-nascido e possíveis anomalias-congênitas (malformação). A base e-SUS contém informações sobre um indivíduo, como hábitos e doenças. Nessa base é informado, por exemplo, se determinado indivíduo é fumante, usa drogas, é diabético etc.

### 3.1.2 Ontologia de Domínio

A ontologia desenvolvida, representada na Figura 2, reutiliza termos de vocabulários bem definidos, como *foaf (Friend of a Friend)*[1], *dbo (DBPedia Ontology)*[2]e *bio (Biographical Information)*[3]. Além disso, foi criado o vocabulário *gissa* para representar os novos termos.

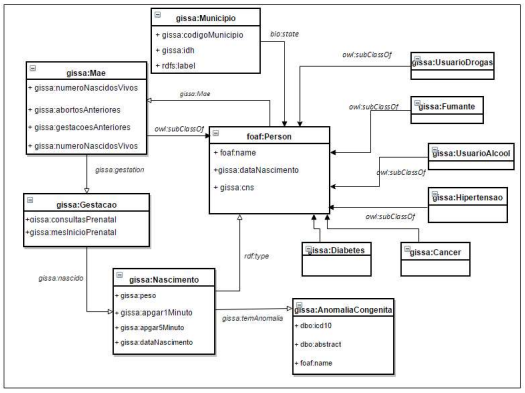


**Figura 2: Datasus_OWL**

### 3.1.3 Visões Exportadas

As visões exportadas $E_{sinasc}$ e $E_{esus}$ foram definidas com o mapeamento entre as fontes $S_{sinasc}$ e $S_{esus}$ e a ontologia *Datasus_OWL*. Note que o resultado de cada mapeamento é uma ontologia exportada cujo vocabulário é um subconjunto de *Datasus_OWL*. Devido à limitação de espaço, as ontologias exportadas $O_{Esinasc}$ e $O_{Esus}$ e seus respectivos mapeamentos são omitidos neste trabalho. Para o mapeamento das bases relacionais em RDF, foi utilizado a linguagem padrão R2RML [10].

A descoberta dos *links semânticos* entre $E_{sinasc}$ e $E_{esus}$ foi realizada utilizando as propriedades de uma entidade *foaf:Person*: *foaf:name*, *gissa:cns*, que representa um identificador do cidadão nas bases da Saúde; *gissa:Municipio* e *gissa:dataNascimento*.

### 3.1.4 Fusão e Qualidade

Para quantificar a qualidade das bases, utilizamos alguns dos critérios abordados em [8], como: (i) quantidade de registros ausentes nas bases de dados; (ii) quantidade de registros duplicados e (iii) quantidade de registros com erros, como nomes de pessoas com erros de escrita, por exemplo. Atribuímos peso 1 para cada critério, e a fonte de dados com maior pontuação representa a menos confiável. Na fusão, utilizamos todas as propriedades originadas da fonte mais confiável.

[1]http://xmlns.com/foaf/spec/
[2]http://purl.org/vocab/bio/0.1/
[3]http://dbpedia.org/ontology/

## 3.2 Construção de uma visão de mashup com Datasus_hub

Nessa subseção, utilizamos a especificação de *Datasus_hub* e as etapas definidas na Seção 2.3 para criar uma *visão de aplicação mashup*. Em nosso exemplo fictício, um gestor da saúde no Brasil quer alertar a população de seu município sobre os perigos dos maus-hábitos durante a gravidez. Para isso, o gestor quer correlacionar o uso de drogas, do tabaco e de álcool durante a gestação com a malformação em recém-nascidos. No Brasil, essas informações estão distribuídas em fontes de dados com formatos heterogêneos. Assim, utilizamos nosso *framework* para criar uma *visão de aplicação mashup* $V$, que atenda às necessidades do gestor, baseada na visão integrada *Datasus_hub*. As etapas para construção da visão são apresentados a seguir.

### 3.2.1 Seleção dos Filtros e da Ontologia de Aplicação

Nessa etapa, utilizamos a *interface* gráfica para selecionar os conceitos (classes) da ontologia de domínio (Fig. 2) importantes para nossa aplicação. Dessa seleção, extraímos a ontologia de aplicação $O_V$. As classes selecionadas foram: *foaf:Person, gissa:Mae, gissa:Gestacao, gissa:UsuarioDrogas, gissa:Fumante, gissa:Municipio, gissa:AnomaliaCongenita, gissa:Nascimento, gissa:UsuarioAlcool*. No nosso exemplo, o gestor quer construir uma aplicação sobre as informações de seu município. Portanto, nosso $F_V = \{gissa{:}Municipio\}$. As próximas etapas acontecem de forma transparente para o usuário.

### 3.2.2 Geração das Visões Exportadas

Nosso *framework* combina a especificação de *Datasus_hub* com os parâmetros $O_V$ e $F_V$ para gerar a especificação de $V$. O conjunto das ontologias exportadas de $V$, $E_V$, é definido por: $E_V = \{(O_V \cap O_{Esinasc}) \cup (O_V \cap O_{Esus})\}$, denotadas por $E_{Vs}$ e $E_{Ve}$ respectivamente. As regras de mapeamento e de descoberta dos *links semânticos* são importadas da especificação de *Datasus_hub* conforme previamente definido.

### 3.2.3 Materialização da Aplicação Mashup

Nessa etapa, o *framework* retorna uma visão homogênea dos dados anteriormente isolados. Como definido em 2.3.3, a materialização ocorre em 3 etapas. Na primeira etapa, utilizamos o SILK [3] para descoberta dos *links semânticos* representados por *owl:sameAs*, resultando em 326 *links* entre $E_{Vs}$ e $E_{Ve}$. Para a materialização das visões exportadas, utilizamos uma versão modificada D2R-Server [4] para processar os mapeamentos R2RML [10]. Na materialização da visão de aplicação, utilizamos o SIEVE [7] para definir as regras de qualidade e de fusão dos dados. Na avaliação de qualidade dos dados, a base $S_{sinasc}$ obteve uma pontuação de 643, enquanto $S_{esus}$, 248. Desta forma, em caso de fusão, utilizamos as informações do $S_{esus}$.

# 4. CONCLUSÃO E TRABALHOS FUTUROS

Neste artigo, apresentamos um *framework* baseado em Ontologias para auxiliar que usuários gerais, com poucos conhecimentos específicos em Computação, possam desenvolver visões integradas sobre dados em fontes heterogêneas. Dentre os benefícios que esperamos com a nossa abordagem, podemos citar os seguintes. (1) Fácil integração de novas fontes de dados. (2) Auxiliar gestores na tomada de decisão. (3) A partir da integração semântica realizada pelo *framework*, quaisquer bases que utilizem os mesmos *schemas*, ou conceitos parecidos, poderão ser integradas com o mínimo de esforço. (4) Modificação mínima em casos de mudanças nos *schemas* das fontes de dados. O *framework* apresentado está na fase de conceptualização, portanto, um dos nossos trabalhos futuros é o desenvolvimento de uma ferramenta *web* que vai implementar nossa metodologia. Além disso, futuramente esperamos avaliar nossa abordagem utilizando-a para auxiliar tomadores de decisão na Saúde Pública do Brasil.

# 5. REFERENCES


[1] F. Belleau, M.-A. Nolin, N. Tourigny, P. Rigault, and J. Morissette. Bio2rdf: Towards a mashup to build bioinformatics knowledge systems. *J. of Biomedical Informatics*, 41(5):706–716, Oct. 2008.

[2] C. Bizer, T. Heath, and T. Berners-Lee. Linked data - the story so far. *Int. J. Semantic Web Inf. Syst.*, 5(3):1–22, 2009.

[3] C. Bizer, J. Volz, G. Kobilarov, and M. Gaedke. Silk - a link discovery framework for the web of data. In *18th International World Wide Web Conference*, April 2009.

[4] D2RQ. *D2RQ - Accessing Relational databases as Virtual RDF Graphs*, mar 2012. available at http://d2rq.org/.

[5] H. H. Hoang, T. N. Cung, D. K. Truong, D. Hwang, and J. J. Jung. Semantic information integration with linked data mashups approaches. *IJDSN*, 2014, 2014.

[6] G. Kobilarov, T. Scott, Y. Raimond, S. Oliver, C. Sizemore, M. Smethurst, C. Bizer, and R. Lee. *Media Meets Semantic Web – How the BBC Uses DBpedia and Linked Data to Make Connections*, pages 723–737. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[7] P. N. Mendes, H. Mühleisen, and C. Bizer. Sieve: Linked Data Quality Assessment and Fusion. In *2nd International Workshop on Linked Web Data Management (LWDM 2012) at the 15th International Conference on Extending Database Technology, EDBT 2012*, page to appear, March 2012.

[8] L. L. Pipino, Y. W. Lee, and R. Y. Wang. Data quality assessment. *Commun. ACM*, 45(4):211–218, Apr. 2002.

[9] V. M. P. Vidal, M. A. Casanova, N. Arruda, M. Roberval, L. P. Leme, G. R. Lopes, and C. Renso. *Advanced Information Systems Engineering: 27th International Conference, CAiSE 2015, Stockholm, Sweden, June 8-12, 2015, Proceedings*, chapter Specification and Incremental Maintenance of Linked Data Mashup Views, pages 214–229. Springer International Publishing, Cham, 2015.

[10] W3C. *R2RML RDB to RDF Mapping Language*, jun 2016. available at https://www.w3.org/TR/r2rml/.

[11] D. S. Wishart, C. Knox, A. Guo, D. Cheng, S. Shrivastava, D. Tzur, B. Gautam, and M. Hassanali. Drugbank: a knowledgebase for drugs, drug actions and drug targets. *Nucleic Acids Research*, 36(Database-Issue):901–906, 2008.