



UNIVERSIDADE FEDERAL DE PERNAMBUCO

GRADUAÇÃO EM SISTEMAS DE INFORMAÇÃO

CENTRO DE INFORMÁTICA

2016.2

DoWMaker: Uma Ferramenta para Publicação Automática de Dados na Web

Aluno: Wilker Cavalcante do Rego Santos (wcrs@cin.ufpe.br)

Orientadora: Bernadette Farias Lóscio (bfl@cin.ufpe.br)

Recife, 17 de Novembro de 2016

Universidade Federal de Pernambuco

Centro de Informática

Graduação em Sistemas de Informação

DoWMaker: Uma ferramenta para Publicação Automática de Dados na Web

Trabalho apresentado ao Programa de Graduação em Sistemas de Informação do Centro de Informática da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Bacharel em Sistemas de Informação.

Recife, 17 de Novembro de 2016

“Todos os que choram a queda de Jerusalém têm o privilégio de ver sua alegria”
-Talmud

Resumo

Com seu grande crescimento, dados armazenados na Web vem ganhando cada vez mais importância para a humanidade. Informações relevantes podem ser encontradas a partir da análise de grandes massas de dados disponíveis na Web. A publicação dos dados na Web abre portas para que qualquer pessoa, independentemente de nível de experiência, possa consumi-los para a reutilização ou redistribuição. Apesar dos inúmeros benefícios, a publicação de dados na Web ainda encontra diversos desafios, como a necessidade de um conhecimento em linguagens de marcação de forma que todos os consumidores possam compreender e a manutenção dos dados para sempre mantê-los atualizados. Por isso, é necessário a automação de alguns processos, a fim de que os custos para o publicador possam ser diminuídos e que a publicação dos dados na Web se torne cada vez mais fácil.

Palavras-chave: Publicação de Dados, Dados na Web, Dados Abertos, Consumo de dados, Publicação automatizada.

Abstract

Web storage data has grown up and became very important to humanity. Relevant information can be found from the analysis of available great masses of data. The web data publication makes it possible for anyone, no matter how experient, to consume the data to reuse or redistribute them. Despite its benefits, data publication still faces a lot of problems, such as the need of a knowledge in data structure language in a way that all the consumers can understand and the data maintainability in order to keep the data up to date. So, it is necessary the automation of some processes, to reduce costs for the publisher and to make it even more easier to publish data on the Web.

Keywords: *Data Publishing, Data on the Web, Open Data, Data consumption, Automated publishing*

Lista de Figuras

Figura 1: Exemplo de XML

Figura 2. Estrutura do RDF

Figura 3. Exemplo de escrita em RDF

Figura 4. Grafo RDF

Figura 5. Exemplo de JSON

Figura 6. Exemplo de CSV

Figura 7. Classificação em 5 estrelas dos Dados Abertos

Figura 8. Casos de uso do DoWMaker

Figura 9. Exemplo de Versionamento de Conjuntos de Dados

Figura 10. Exemplo de consulta em JSON

Figura 11. Arquitetura do DoWMaker

Figura 12. Arquitetura do módulo Servidor

Figura 13. Transformação dos dados

Figura 14. Esquema do Banco de Publicação sem inserções

Figura 15. Esquema do Banco de Publicação com inserções

Figura 16: Conjunto de Dados Publicado em XML

Figura 17: Histórico de versionamento em Json

Figura 18: Metadados em Json

Figura 19: Tela de metadados do OpenAcademy

Figura 20: Tela de indicadores do OpenAcademy

Sumário

Capítulo 1. Introdução	8
1.1 Motivação	8
1.2 Objetivos e Contribuições	9
1.3 Estrutura do documento	9
Capítulo 2. Contextualização	10
2.1 Ciclo de Vida dos Dados na Web	10
2.2 Boas práticas para Dados na Web	11
2.3 Versionamento dos Conjuntos de Dados na Web	12
2.4 Formatos para publicação de dados na Web	13
2.5 Dados Abertos	16
2.6 Considerações finais	17
Capítulo 3. Publicação automática de Dados na Web	18
3.1 Visão Geral da ferramenta DoWMaker	18
3.2 Casos de uso	18
3.3 Atualização dos Conjuntos de Dados	22
3.4 Versionamento dos Conjuntos de Dados	23
3.5 Estrutura das URIs para consumo dos dados	24
3.6 Criação dos Conjuntos de Dados	25
3.7 Considerações finais	26
Capítulo 4. Implementação e Avaliação	28
4.1 Arquitetura do Sistema	28
4.2 Servidor	29
4.3 Banco de Dados Publicados	31
4.4 Instalação	33
4.5 Avaliação	34
Capítulo 5. Conclusão e Trabalhos Futuros	38
Referências	39

Capítulo 1. Introdução

Neste capítulo será apresentada uma introdução a respeito deste trabalho de graduação, bem como suas motivações e os objetivos propostos, além da estrutura do documento.

1.1 Motivação

No decorrer dos anos, a Web tornou-se um recurso muito importante para a sociedade como um todo. Pessoas se conectam à Web de formas nunca antes imaginadas e tornam este ato cada vez mais presente em suas atividades diárias. Com isto, uma quantidade gigantesca de dados vem sendo criada e armazenada. Entretanto, estes dados são armazenados, em sua grande maioria, de forma desestruturada, tornando processos de análise e compreensão destes dados tarefas muito complexas.

Com uma proposta de estrutura mais organizada, a Web Semântica [4] permite que dados, antes incompreensíveis, sejam compreendidos não apenas por humanos, mas também por máquinas. Além da Web Semântica, a fim de contribuir para a criação de uma Web de Dados, foi proposto um conjunto de princípios, denominado *Linked Data* (Dados conectados) que possibilita que os dados, agora estruturados, sejam interligados, criando assim uma enorme rede de dados conectados [8].

Desta forma, diversas soluções têm sido desenvolvidas ao longo dos anos para a publicação e o compartilhamento de conjuntos de dados na Web. Porém, estas soluções não oferecem mecanismos adequados para o gerenciamento dos conjuntos de dados. Em geral, as soluções atualmente disponíveis não tratam dos seguintes aspectos: gerenciamento de versões dos conjuntos de dados, atualização automática dos conjuntos de dados, criação automática de conexões entre os conjuntos de dados e geração automática de dados em múltiplos formatos.

Considerando as dificuldades mencionadas anteriormente no contexto de publicação de dados na Web, este trabalho propõe uma ferramenta, denominada DoWMaker, para publicação automática de dados na Web. Com o uso da ferramenta proposta, é possível realizar a extração de dados de uma fonte de dados de origem e facilmente publicá-los na Web por meio de uma API¹ que permite o acesso aos dados em múltiplos formatos, incluindo RDF² (Resource Description Framework), CSV³, Json⁴ e XML⁵. Outro benefício com o uso da ferramenta

¹"Api-Wikipédia."

https://pt.wikipedia.org/wiki/Interface_de_programa%C3%A7%C3%A3o_de_aplica%C3%A7%C3%B5es. Acessado em 11 dez. 2016.

² "RDF - Semantic Web Standards - W3C." <https://www.w3.org/RDF/>. Acessado em 11 dez. 2016.

³ "CSV (Comma Separated Values) - Wikipedia."

https://en.wikipedia.org/wiki/Comma-separated_values. Acessado em 11 dez. 2016.

⁴ "JSON." <http://www.json.org/>. Acessado em 11 dez. 2016.

proposta diz respeito à atualização dos conjuntos de dados automaticamente em intervalos de tempos pré-definidos, além da flexibilidade para a modificação na estrutura do conjunto de dados sem a necessidade de alteração de código fonte.

A avaliação da ferramenta foi feita no contexto de um projeto, atualmente em desenvolvimento, que busca realizar a abertura dos dados da UFPE.

1.2 Objetivos e Contribuições

Este trabalho tem como principal objetivo a especificação e desenvolvimento de uma ferramenta para a publicação automática de dados na Web. Como principais características da ferramenta proposta, destacam-se: é multiplataforma, permite o armazenamento dos conjuntos de dados criados pelos usuários em uma base de dados a parte da origem, disponibiliza os dados em diversos formatos por meio de uma API REST, mantém os dados atualizados e garante o versionamento dos dados de forma automatizada.

As principais contribuições deste trabalho são:

- A especificação e implementação de uma ferramenta para facilitar a publicação de dados na Web.
- Uso da ferramenta para a publicação dos dados da UFPE em formato aberto.

1.3 Estrutura do documento

O restante deste documento está estruturado da seguinte forma:

- Capítulo 2: Contextualização dos conceitos essenciais para o entendimento do trabalho, como o ciclo de vida dos dados na Web, as boas práticas para a publicação, o versionamento dos conjuntos de dados, formatos para a publicação dos dados e Dados Abertos
- Capítulo 3: Apresentação de uma visão geral da ferramenta DoWMaker.
- Capítulo 4: Apresentação da arquitetura da ferramenta e as tecnologias utilizadas para seu desenvolvimento.
- Capítulo 5: Discussão das conclusões deste trabalho, e indicações para contribuições futuras.

⁵ "Extensible Markup Language (XML) - W3C." <https://www.w3.org/XML/>. Acessado em 11 dez. 2016.

Capítulo 2. Contextualização

Neste capítulo serão apresentados os conceitos fundamentais para a compreensão deste trabalho. Na seção 2.1 será mostrado o ciclo de vida dos dados na Web, na 2.2 aborda um pouco das boas práticas da publicação de dados na Web, na 2.3 o versionamento dos conjuntos de dados e sua importância, na 2.4 os formatos para publicação de dados e na 2.5 o conceito de Dados Abertos.

2.1 Ciclo de Vida dos Dados na Web

Alguns passos existem para a publicação e consumo de dados na Web [7]. Como uma instância do ADLM(*Abstract Data Lifecycle Model*), que é um modelo de representação de ciclo de vida de dados e metadados, foi proposto um ciclo de vida para os dados na Web [2]. O modelo ADLM propõe as seguintes fases [7]: Desenvolvimento de Ontologia, Planejamento, Criação, Arquivamento, Refinamento, Publicação, Acesso, Uso Externo, Feedback e Término. Para o ciclo de vida dos na Web, não foram incluídas o Desenvolvimento de Ontologia e o Término. O Desenvolvimento de Ontologias é considerada atividade independente e, portanto não foi incluída, assim como Término, pois, uma vez publicados na Web, os dados sempre estarão disponíveis para o consumo

Assim, as principais etapas do ciclo de vida dos dados na Web são descritas a seguir [2]:

- **Planejamento:** esta fase se estende desde a intenção na publicação dos dados até a seleção dos mesmos.
- **Criação:** nesta etapa os dados são extraídos da fonte de dados origem e depois transformados para o formato requerido para a publicação.
- **Publicação:** é a fase em que os dados são disponibilizados já para o consumo na Web. Nesta fase, podem ser utilizadas ferramentas de catalogação de dados e APIs.
- **Acesso:** esta etapa se concretiza quando os dados já estão disponibilizados e os consumidores ganham acesso aos dados, ou seja, quando há a divulgação que os dados foram publicados.
- **Consumo:** no momento em que os dados estão sendo consumidos, tanto para a criação de novos dados, quanto para a geração de informação. Está

diretamente ligada aos consumidores, que podem ser desenvolvedores interessados na criação de aplicações para o consumo dos dados.

- **Feedback:** nesta fase são coletadas as informações necessárias em relação à qualidade dos dados e metadados do ponto de vista do consumidor. Com estas informações é possível realizar a evolução dos dados em vários aspectos.
- **Refinamento:** Na fase final, são aplicadas atividades relacionadas às adições e atualizações dos dados publicados, baseados no feedback dos consumidores ou por decisão de melhoria do próprio provedor.

2.2 Boas práticas para Dados na Web

Ainda que a publicação de dados estruturados na Web seja uma realidade que vêm crescendo nos últimos anos, uma falta de consenso para padrões de comunicação e dados entre provedores e consumidores leva à criação de barreiras, especialmente para o consumo.

Nesse sentido, o W3C [14] propôs um conjunto de boas práticas para Dados na Web, o DWBP (*Data On Web Best Practices*) com o intuito de oferecer orientações para os publicadores e consumidores de dados. Esta orientação promove a reutilização dos dados e cultiva a confiabilidade dos dados entre desenvolvedores, independentemente das tecnologias em que utilizaram.

Segundo o W3C, o uso das melhores práticas atacam os principais desafios da publicação e consumo de dados, sendo estes, identificados nos casos e requisitos de uso do DWBP. São eles [14]:

- **Provisão dos metadados:** Este desafio se refere à provisão de metadados que sejam compreensíveis por seres humanos e máquinas.
- **Licença dos dados:** Problema de definições de restrições de acesso aos dados.
- **Qualidade dos dados:** Este, se refere à qualidade dos dados e até que ponto são confiáveis.
- **Versionamento dos dados:** Este desafio é em relação a como identificar as versões dos dados publicados na Web e como fornecer um histórico de versionamento.

- **Identificação dos dados:** Aqui, está o desafio de como identificar os conjuntos de dados e suas distribuições.
- **Formato dos dados:** Se refere à qual formato de dados utilizar para a publicação.
- **Vocabulário dos dados:** Desafio de manter a interoperabilidade dos dados.
- **Acesso aos dados:** Desafio de qual a forma de acesso que os dados terão.
- **Preservação dos dados:** Este desafio se refere a como os dados serão arquivados no decorrer do tempo.
- **Feedback:** Questão de como coletar do consumidor dos dados o seu grau de satisfação.
- **Enriquecimento dos dados:** Aqui, o desafio se refere a como acrescentar valor aos dados.
- **Republicação dos dados:** Se refere à como criar ou respeitar regras de reuso de dados.

Com a utilização das melhores práticas, os ruídos de comunicação entre provedores e consumidores diminuirão e Web estruturada estará mais próxima de seu objetivo final: ser compreensível para seres humanos e máquinas.

2.3 Versionamento dos Conjuntos de Dados na Web

Dados na Web mudam constantemente, sejam eles de forma mais frequente ou não. É importante que os dados que estejam publicados se encontrem sempre atualizados, por isso, segundo o guia das melhores práticas de dados na Web do W3C, é fundamental que os conjuntos de dados estejam versionados [14].

O versionamento do conjunto de dados possibilita ao consumidor entender se um determinado conjunto a ser utilizado está em sua forma mais recente. Um bom indicador de versão permite a reprodução de pesquisa, a comparação e permite ao consumidor entender as mudanças que ocorreram [14] de uma versão para outra.

Segundo o W3C, mesmo que um conjunto de dados venha a sofrer uma alteração, ainda que mínima, deve-se ser gerada uma nova versão. Também é dever do provedor ter a precaução de informar aos seus consumidores quando uma nova versão estará disponível. Em casos de atualização em tempo real, um indicador automático de *timestamp* pode servir como identificador da versão [14].

Muitas das vezes, entender a diferença entre as versões, bem como a dinâmica de atualização dos conjunto de dados pode ser muito custoso. Por isso, o W3C também indica como uma boa prática a disponibilização de um histórico de versionamento, de forma que possa ser lido por humanos e softwares e compreendida a complexidade das atualizações do conjunto de dados em questão [14].

Dependendo das circunstâncias, um consumidor pode (antes mesmo de iniciar uma pesquisa em uma fonte de dados) necessitar saber quando e porque um conjunto de dados foi alterado, para que assim possa refinar sua consulta e escolher qual o momento ideal para que seja realizada. Assim, é de responsabilidade do provedor, oferecer de melhor maneira possível um histórico com qualquer alteração de seus conjuntos de dados.

2.4 Formatos para publicação de dados na Web

Com a necessidade de representação de informações sobre recursos na Web, linguagens de marcação foram desenvolvidas ao longo do tempo. Para este trabalho, será destacado quatro delas: XML, RDF, JSON e CSV.

XML(*eXtensible Markup Language*) é uma linguagem de marcação recomendada pelo W3C para criação de documento de dados organizados hierarquicamente. A Figura 1 mostra um exemplo de XML:

```
<disciplina>
  <nome>Banco de Dados</nome>
  <codigo>IF976</codigo>
  <horario>segunda, 17 - 18:40 / quarta, 18:40 - 20:30</horario>
  <professor>Bernadette Farias Lóscio</professor>
  <sala>E-123</sala>
  <curso>Sistemas de Informação</curso>
</disciplina>
```

Figura 1: Exemplo de XML

Por mais que o XML consiga representar os dados de forma organizada e de fácil leitura, a mesma não permite que recursos venham a referenciar outros. Um recurso entende-se como tudo o que pode ser expressado como idéia e pode ser identificado através de uma URI(*Uniform Resource Identifier*) [1]. Baseado em XML, criou-se o RDF. A partir do RDF, recursos podem ser relacionados com outros através da estrutura sujeito, predicado e objeto conhecida como tripla:

- *Sujeito*: Se refere ao recurso no qual se quer escrever a sentença.
- *Predicado*: Ligação entre o recurso da sentença a um objeto, seja outro recurso ou um dado.

- *Objeto*: Recurso ou dado ligado a o recurso da sentença.

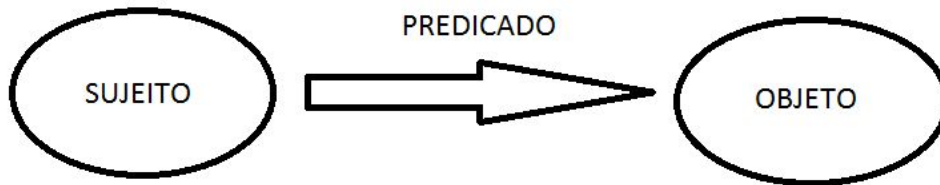


Figura 2: Estrutura do RDF

Na Figura 2 abaixo pode-se ver um exemplo de escrita em RDF, em que a maioria dos objetos são dados e que “idCurso”, faz referência a um outro recurso:

```
<rdf:Description rdf:about="http://www.cin.ufpe.br/opendata/disciplina/if976">
  <rdf:type rdf:resource="http://www.cin.ufpe.br/opendata/disciplina"/>
  <cin:nome>Banco de Dados</cin:nome>
  <cin:codigo>IF976</cin:codigo>
  <cin:horario>segunda, 17 - 18:40 / quarta, 18:40 - 20:30</cin:horario>
  <cin:professor>Bernadette Farias Lóscio</cin:professor>
  <cin:sala>E-123</cin:sala>
  <cin:idCurso rdf:resource="http://www.cin.ufpe.br/opendata/curso/123"/>
</rdf:Description>
```

Figura 2: Exemplo de escrita em RDF

O RDF também pode ser representado com grafos, onde cada nó é um recurso e cada aresta é uma propriedade como pode ser visto na Figura 4:

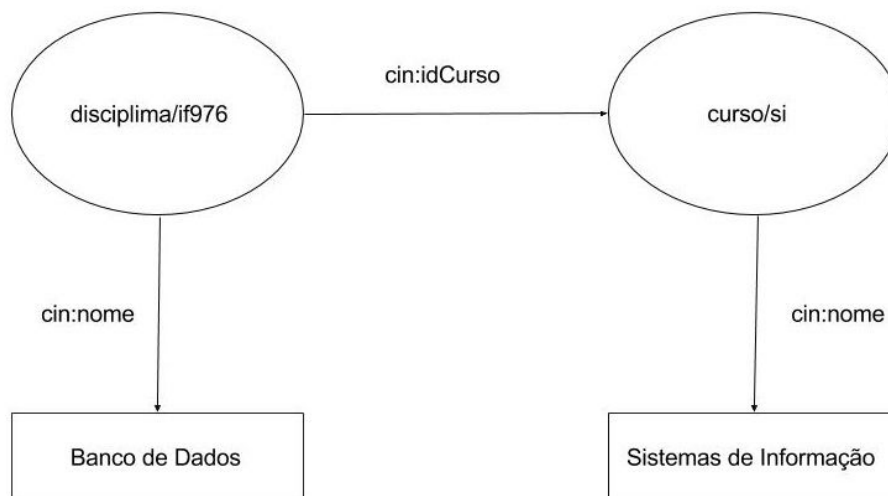


Figura 4: Grafo RDF

Ainda existem muitas outras formas de estruturação de dados disponíveis. Ainda que não consigam expressar uma sintaxe tão complexa como o RDF, são muito populares. Dentre os formatos existentes, para este trabalho, é importante dar destaque a dois: *JSON* e *CSV*.

Originalmente criado por Douglas Crockford *JSON (Javascript Object Notation)* é um formato de dados leve para intercâmbio de dados de origem na formação de objetos *javascript*, ainda que não necessite da linguagem para o seu uso. Sua popularidade é fruto de sua facilidade para leitura em tecnologias utilizadas para o desenvolvimento de software, e isso lhe dá vantagens sobre outros formatos como o XML [15]. A Figura 5 mostra um exemplo de JSON.

```
{ "disciplina": {  
  "nome": "Banco de Dados",  
  "codigo": "IF976",  
  "horario": "segunda, 17 - 18:40 / quarta, 18:40 - 20:30",  
  "professor": "Bernadette Farias Lóscio",  
  "sala": "E-123",  
  "idCurso": 123  
}
```

Figura 5: Exemplo de JSON

Comma-separated values, conhecido popularmente como *CSV*, é um formato que armazena dados em formato tabular. Sua principal característica é que se trata de uma implementação simples de um formato de texto com dados limitados por um

separador (mais comumente ponto e vírgula ou apenas vírgula) e a quebra de linha para separar valores. A Figura 6 mostra um exemplo de csv:

nome	codigo	horario	professor	sala	idCurso
Banco de Dados	IF976	segunda, 17 - 18:40 / quarta, 18:40 – 20:30	Bernadette Farias Lóscio	E-123	123

Figura 6: Exemplo de CSV

2.5 Dados Abertos

Em uma idéia em que dados na Web pudessem estar disponíveis para a livre publicação e consumo, sem qualquer restrição de patentes ou direitos autorais, surgiram os Dados Abertos (*Open Data*). De acordo com a Open Knowledge Foundation [10], Dados Abertos são livremente utilizados reutilizados e redistribuídos por qualquer usuário, sujeito apenas a realizar a atribuição de sua autoria e o compartilhamento pela mesma licença que a original.

David Eaves propôs três leis para os Dados Abertos[11]:

1. Se o dado não pode ser encontrado e indexado na Web, então ele não existe.
2. Se não estiver em formato a ser reconhecido por máquina e não estiver aberto, então não pode ser reutilizado.
3. Se algum dispositivo legal não permitir sua replicação, então ele não é útil.

Além dessas três regras, também foram definidos oito princípios para Dados Abertos Governamentais, que como consequência, também pode ser aplicado para dados abertos em geral, são eles : os dados devem estar completos, com menor granularidade possível, atuais, acessíveis, processáveis por máquina, sem acesso discriminatório e disponíveis em formatos em que todos os usuários compreendam e tenham acesso.

Berners-lee [13], propôs cinco níveis de classificação para a publicação de dados abertos, que é levado em consideração quando o assunto quando o assunto é a distribuição de dados e sua conexão com diversas fontes. A classificação, que ficou conhecida como “Sistema de Cinco Estrelas”, como o nome já propõe, avaliam o grau de abertura dos dados por meio de estrelas.

São os níveis[13]:

1. Dados que estão disponíveis e com licença livre.

2. Dados que estão disponíveis com licença livre e de forma estruturada.
3. Dados que estão disponíveis com licença livre, de forma estruturada e em um formato não proprietário e dentro dos padrões estabelecidos pela W3C.
4. Dados que estão disponíveis com licença livre, de forma estruturada, em um formato não proprietário, dentro dos padrões estabelecidos pela W3C e conectados com outros dados, oferecendo um contexto

A Figura 7 ilustra a classificação:

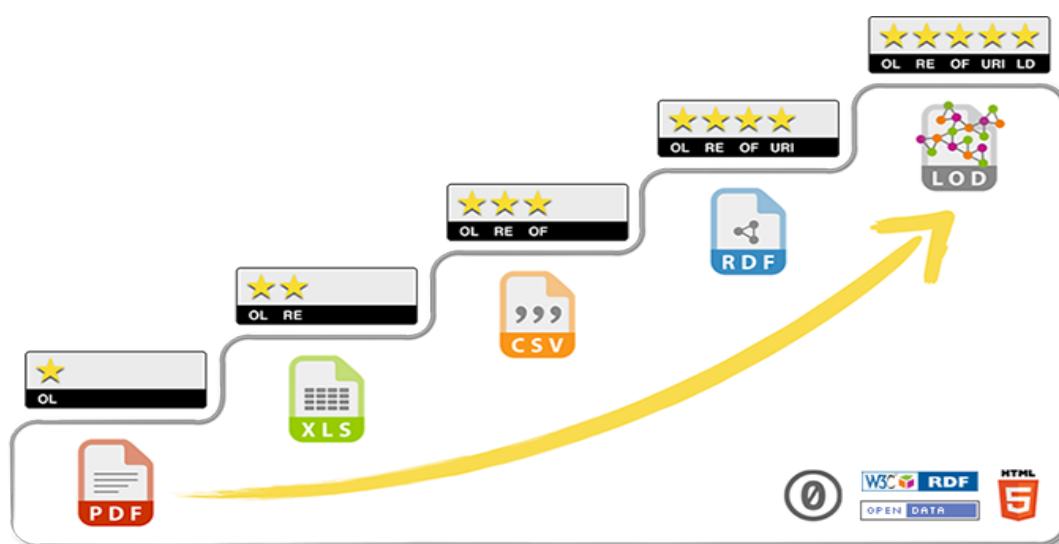


Figura 7: Classificação em 5 estrelas dos Dados Abertos, 2004 [13]

2.6 Considerações finais

Neste capítulo, foram apresentados os principais conceitos relacionados a este trabalho: Ciclo de vida de dados na Web. Algumas das principais tecnologias utilizadas para dados na Web foram abordadas: RDF, XML, JSON e CSV. Também foi apresentado sobre as Melhores Práticas de Dados na Web e sua importância para manter a qualidade dos dados e uma breve introdução à Dados Abertos.

Capítulo 3. Publicação automática de Dados na Web

Neste capítulo será apresentada uma visão geral da ferramenta DoWMaker, assim como todos os seus casos de uso.

3.1 Visão Geral da ferramenta DoWMaker

A ferramenta desenvolvida neste trabalho, denominada DoWMaker, tem como principal objetivo a automação de algumas etapas do processo de publicação de dados na Web, bem como a abstração de alguns conhecimentos técnicos. Por ser multiplataforma, a ferramenta também poderá ser instalada nos ambientes computacionais mais populares do mercado.

A DoWMaker permitirá diminuir os esforços do publicador de dados em manter seus dados em um padrão esperado pela maioria dos consumidores de dados na Web. A ferramenta proposta permite a extração dos dados a partir de um Banco de Dados Relacional, bem como a atualização e versionamento dos conjuntos de dados de forma automática.

Outra característica importante é a disponibilidade dos conjuntos de dados em quatro formatos: RDF, JSON, XML e CSV. Os metadados também serão disponibilizados pela ferramenta, sendo estas informações obrigatórias a serem inseridas pelo provedor no momento da publicação de um novo conjunto de dados.

O consumidor, por sua vez, será capaz de realizar o acesso aos dados e metadados por meio de APIs, podendo escolher o formato de dados que preferir extrair. Também poderá escolher qual versão do conjunto de dados será a adequada para a sua pesquisa, havendo a possibilidade de se basear no histórico de versionamento com os detalhes de cada versão, disponibilizados pela ferramenta.

3.2 Casos de uso

Nesta seção, são apresentados todos os casos de uso relacionados à ferramenta. A Figura 8 ilustra o diagrama de casos de uso.

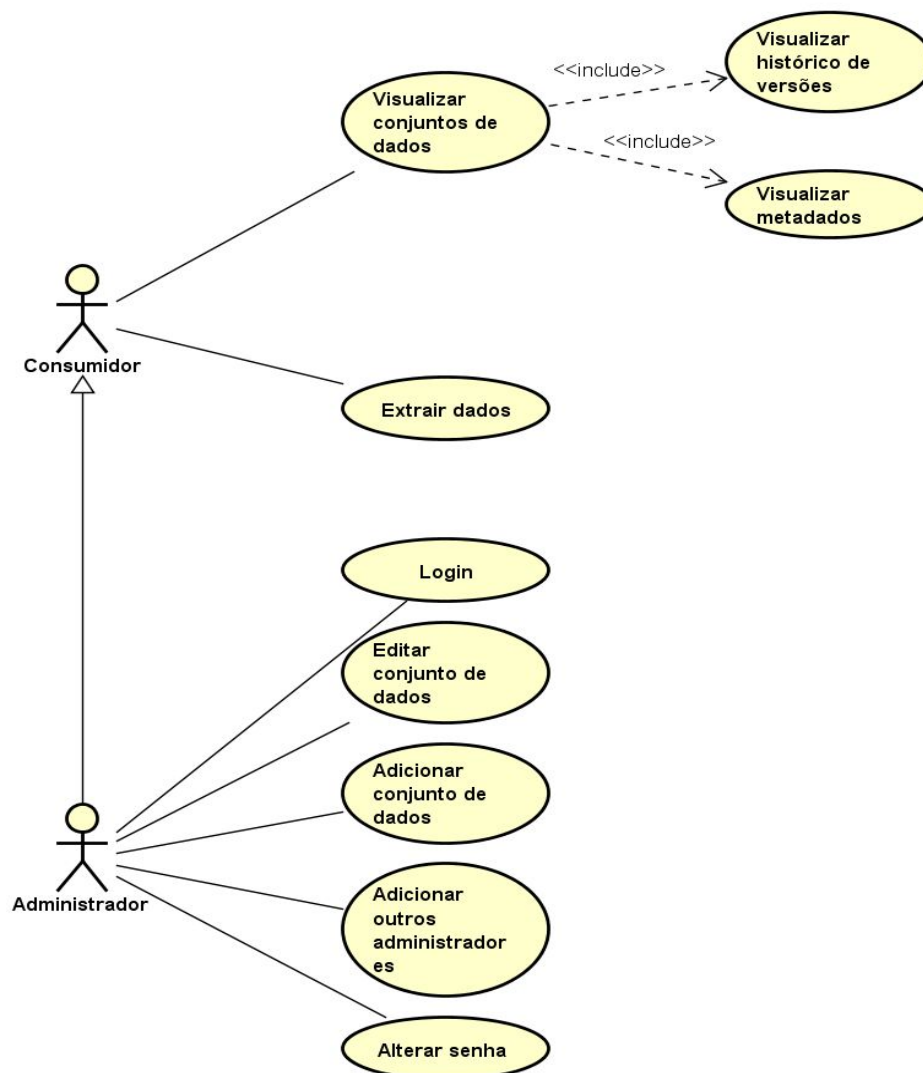


Figura 8: Casos de uso do DoWMaker

- **Visualizar Conjuntos de Dados**

Descrição	O consumidor poderá visualizar a lista de todos conjuntos de dados disponíveis, bem como seus metadados.
Ator	Consumidor de dados na Web
Pré-Condições	Criar um serviço que realiza requisições http GET ou acessar a URI via navegador.
Fluxo de Eventos	Realizar uma requisição http GET à URI de descrição dos conjuntos de dados.
Fluxo Alternativo	Caso não haja conjuntos de dados, o sistema não irá retornar nenhum dado

- **Extrair Dados**

Descrição	O consumidor poderá extrair os dados do conjunto de dados requerido, escolhendo um dos formatos de dados disponíveis, bem como a versão dos dados.
Ator	Consumidor de dados
Pré-Condições	Criação de um serviço que realiza requisições http GET ou acessar a URI via navegador.
Fluxo de Eventos	Realizar uma requisição http GET, passando como parâmetros o nome do conjunto de dados, a versão e o formato de dados.
Fluxo Alternativo	1. Caso o consumidor não venha a fornecer a versão do conjunto de dados, o sistema retornará como padrão a versão mais recente. 2. Caso o consumidor não venha a fornecer o formato de dados da extração, o sistema retornará como padrão o formato RDF. 3. Caso o conjunto de dados requisitado não exista, o sistema não irá retornar dados.

- **Login**

Descrição	O usuário irá se identificar ao sistema para que tenha autorização de acesso às funcionalidades de administrador.
Ator	Administrador do sistema.
Pré-Condições	Criação de um serviço que pode realizar requisições http POST, ou uso de ferramenta que faça o mesmo.
Fluxo de Eventos	Realização de uma requisição http POST, passando como parâmetros o login e a senha e o sistema retornará o código token.
Fluxo Alternativo	Caso o usuário informe os dados incorretamente ou em branco, o sistema retornará a requisição como estado http 401 (Não autorizado).

- **Adicionar Conjunto de Dados**

Descrição	O administrador irá informar ao sistema os dados necessários para que um novo conjunto de dados seja adicionado e os dados extraídos de sua fonte original.
Ator	Administrador do sistema
Pré-Condições	1. Criação de um serviço que pode realizar requisições http POST, ou uso de ferramenta que faça o mesmo. 2. O administrador deve ter realizado o login e possuir o código token.
Fluxo de Eventos	Realização de uma requisição http POST, passando todos os dados requeridos para a criação de um novo conjunto de dados.
Fluxo Alternativo	Caso o usuário informe um dos parâmetros incorretamente ou em branco, o sistema responderá com o estado http 400 (<i>Bad Request</i> , ou requisição incorreta).

- **Editar Conjunto de Dados**

Descrição	O administrador poderá realizar alterações no conjunto de dados que desejar.
Ator	Administrador do sistema
Pré-Condições	1. Criação de um serviço que pode realizar requisições http POST, ou uso de ferramenta que faça o mesmo. 2. O administrador deve ter realizado o login e possuir o código token. 3. O administrador deve informar o código de identificação única do conjunto de dados.
Fluxo de Eventos	Realização de uma requisição http POST, passando todos os dados requeridos para a alteração de um conjunto de dados.
Fluxo Alternativo	Caso o usuário informe um dos parâmetros incorretamente ou em branco, o sistema responderá com o estado http 400 (<i>Bad Request</i> , ou requisição incorreta).

- **Adicionar outros administradores**

Descrição	O administrador poderá adicionar outros administradores no sistema.
Ator	Administrador do sistema
Pré-Condições	1. Criação de um serviço que pode realizar requisições http POST, ou uso de ferramenta que faça o mesmo. 2. O administrador deve ter realizado o login e possuir o código token.
Fluxo de Eventos	Realização de uma requisição http POST, passando como parâmetro o nome do login do novo administrador e sua senha.
Fluxo Alternativo	Caso o usuário informe um dos parâmetros incorretamente ou em branco, o sistema responderá com o estado http 400 (<i>Bad Request</i> , ou requisição incorreta)

- **Alterar Senha**

Descrição	O administrador poderá alterar sua senha de acesso ao sistema
Ator	Administrador do sistema
Pré-Condições	1. Criação de um serviço que pode realizar requisições http POST, ou uso de ferramenta que faça o mesmo. 2. O administrador deve ter realizado o login e possuir o código token.
Fluxo de Eventos	Realização de uma requisição http POST, passando todos os dados requeridos para a alteração da senha do administrador.
Fluxo Alternativo	Caso o usuário informe a antiga senha incorretamente ou a nova senha em branco, o sistema responderá com o estado http 400 (<i>Bad Request</i> , ou requisição incorreta).

3.3 Atualização dos Conjuntos de Dados

Pensando em uma solução em que os conjuntos de dados estejam sempre atualizados, de acordo com a fonte de dados de origem ou com uma frequência de atualização pré-estabelecida, a ferramenta DoWMaker foi desenvolvida com um mecanismo de atualização automática dos dados. Por meio de uma configuração de frequência pré-estabelecida pelo publicador de dados, a ferramenta irá verificar quando cada conjunto de dados terá que passar por uma atualização, e quando o momento chegar, ela realizará a consulta automaticamente no Banco de Dados de origem, retornará e irá inserir os dados atualizados no Banco de Dados em que os dados publicados estão. As frequências de atualização consideradas são: Por hora, Diário, Semanal, Mensal, Trimestral, Semestral e Anual.

Também será permitido ao publicador (ou Administrador do Sistema) realizar atualizações manuais, quando necessário, apenas informando ao sistema qual conjunto de dados que deseja atualizar. Se uma alteração na estrutura do conjunto de dados for requisitada, assim como como a atualização manual, o sistema realizará todo o processo normalmente, só que desta vez, com as alterações realizadas.

3.4 Versionamento dos Conjuntos de Dados

Com o intuito de fornecer mais liberdade, transparência e compreensão dos conjuntos de dados por parte do consumidor, a ferramenta DoWMaker conta com um mecanismo de versionamento que mantém não só a versão mais atual dos conjuntos de dados, mas também todas as outras versões anteriores.

Quando um conjunto de dados é referenciado, e não há especificação de versão, o sistema automaticamente entende que os dados a serem utilizados são os da última versão disponível.

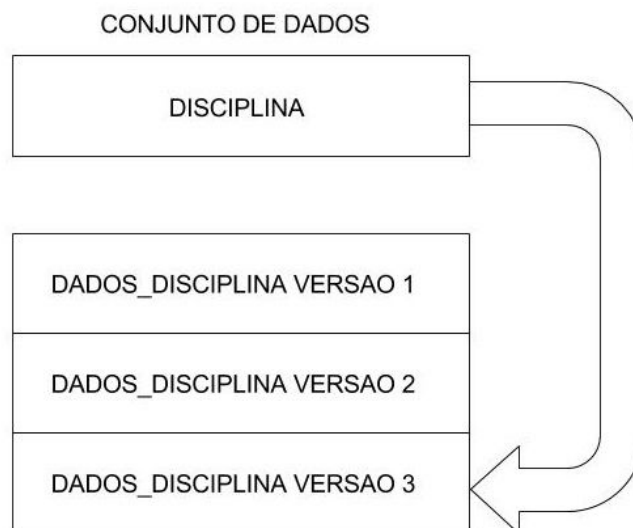


Figura 9: Exemplo de Versionamento de Conjuntos de Dados

A Figura 9 apresenta o exemplo de um conjunto de dados “disciplina”, que possui três versões. Nota-se que, sempre quando o conjunto de dados for referenciado, a versão mais recente, neste caso a versão 3, será indicada, então a ferramenta retornará todos os dados pertencentes à esta versão. Percebe-se também que as versões antigas também estão disponíveis da mesma forma, logo, quando indicado na consulta a uma versão específica, a mesma também retornará seus dados. Neste caso, se o usuário preferir utilizar os dados da versão 1, então, ele especificará isto para o sistema e o mesmo trará os dados da versão requerida.

Devido à sua estratégia automatizada de atualização de dados, a ferramenta utiliza como código de indicador de versão uma *tag* baseada em *timestamp*, construída a partir da data e hora do momento em que a atualização (ou inserção) é realizada.

3.5 Estrutura das URIs para consumo dos dados

A ferramenta DoWMaker foi desenvolvida com a possibilidade dos dados serem consumidos a partir de requisições *http* GET, podendo o usuário acessar os dados e metadados de cada conjunto de dados a partir de URIs. a seguir, são apresentados alguns exemplos que ilustram as URIs de acesso aos dados. Em todos os exemplos a seguir, será suposto que o usuário queira acessar os dados de um hipotético conjunto de dados “disciplina”.

Exemplo 1: Acesso a um conjunto de dados específico

<http://www.cin.ufpe.br/dadosabertos/find/disciplina>

Para todas as consultas de dados, o parâmetro “find” deve ser inserido na URI, seguido pelo nome do conjunto de dados requerido. Por padrão, o sistema retornará todos os dados no formato RDF em sua última versão. Quando for necessário realizar a busca por um recurso específico dentro conjunto de dados, um identificador único deve ser especificado. Como pode ser visto no Exemplo 2.

Exemplo 2: Acesso a um recurso específico dentro de um conjunto de dados

<http://www.cin.ufpe.br/dadosabertos/find/disciplina/if971>

Exemplo 3: Acesso a uma versão específica de um conjunto de dados

Para a especificação de uma versão do conjunto de dados, o parâmetro do identificador deve ser seguido com um *underline* e o número da versão:

http://www.cin.ufpe.br/dadosabertos/find/disciplina/if971_201612121430

Exemplo 4: Solicitação de dados em um formato específico

Ainda é possível a especificação do formato de dados. O exemplo abaixo mostra a mesma consulta, desta vez, com retorno em XML.

<http://www.cin.ufpe.br/dadosabertos/find/disciplina/format/xml>

Exemplo 5: Solicitação do histórico de versionamento do conjunto de dados

É possível também, o acesso ao histórico de versionamento do conjunto de dados, como pode ser visto no exemplo.

<http://www.cin.ufpe.br/dadosabertos/find/history/disciplina>

Seguido do parâmetro “format”, o usuário deverá especificar o formato que desejar, podendo ser ele RDF, JSON, CSV e XML. Para consultas mais personalizadas, diferentemente das outras, uma requisição JSON http POST deve ser realizada. Em seu corpo, devem ser especificados o nome do conjunto de dados, o formato, a versão e os parâmetros, como demonstrado na Figura 10, que retorna como resultado uma disciplina cujo nome seja “Banco de Dados” e sua sala a “E-123”.

```

{
  "datasetName": "disciplina",
  "format": "rdf",
  "version": "201612121430",
  "where": {
    "nome": "Banco de Dados",
    "sala": "E-123"
  }
}

```

Figura 10: Exemplo de consulta em JSON

3.6 Criação dos Conjuntos de Dados

Para que um novo conjunto de dados possa ser publicado pela ferramenta DoWMaker, o usuário deverá realizar uma solicitação à API do sistema através de uma aplicação que realize requisições HTTP, e informar alguns dados em formato JSON. Todos os atributos foram definidos de acordo com as recomendações fornecidas pela DWBP [14]. São eles:

Nome	Descrição
datasetTitle	Nome escolhido para o conjunto de dados
sqlQuery	Query SQL que será executada para a extração dos dados no Banco de Dados de origem
collectionIdColumnName	Se refere ao nome do atributo que será utilizado como identificador único de cada tupla no conjunto de dados
updateFrequency	Campo de configuração da frequência de atualização do conjunto de dados
creator	Nome do criador do conjunto de dados
keywords	Palavras-chave
publisher	Nome do provedor de dados
contactPoint	Aqui é informada alguma forma de contato com o provedor de dados
period	Período temporal que o conjunto de dados abrange.
spacialCoverage	Espaço geográfico que os dados se referem.

theme	Tema que o conjunto de dados se refere
language	Língua em que os dados do conjunto de dados estão representados
dateTimeFormats	Formato dos dados do tipo data
description	Descrição do conjunto de dados

3.7 Considerações finais

Neste capítulo, foram apresentados as principais características da ferramenta DoWMaker bem como a descrição de seus casos de uso. Também foram apresentados mais detalhes sobre as atividades de Atualização dos Conjuntos de Dados, Versionamento dos conjuntos de dados, Acesso aos conjuntos de dados e criação de um novo conjunto de dados.

Capítulo 4. Implementação e Avaliação

Neste capítulo, alguns detalhes da implementação da ferramenta DoWMaker serão apresentados, como linguagem de programação utilizada, arquitetura da ferramenta sistema, gerência de configuração, banco de dados, *frameworks* e bibliotecas auxiliares. Também serão discutidos detalhes de instalação da ferramenta e a implantação para a publicação dos dados do SIGA - UFPE.

4.1 Arquitetura do Sistema

A ferramenta DoWMaker foi desenvolvida com o objetivo de ser acessada via Web, por isso, foi subdividida em duas partes: servidor e API (*Application Programming Interface*), que é uma interface que pode ser acessada via Web, como apresentado na Figura 11. O servidor se mantém em comunicação com dois bancos de dados, sendo um a base de origem, em que estão os dados que serão extraídos e o outro que armazena apenas os dados que serão publicados. Em todo o desenvolvimento da ferramenta, foi utilizado o *Git* como framework de gerência de configuração, e todo o código foi armazenado em um repositório público do *BitBucket*⁶.

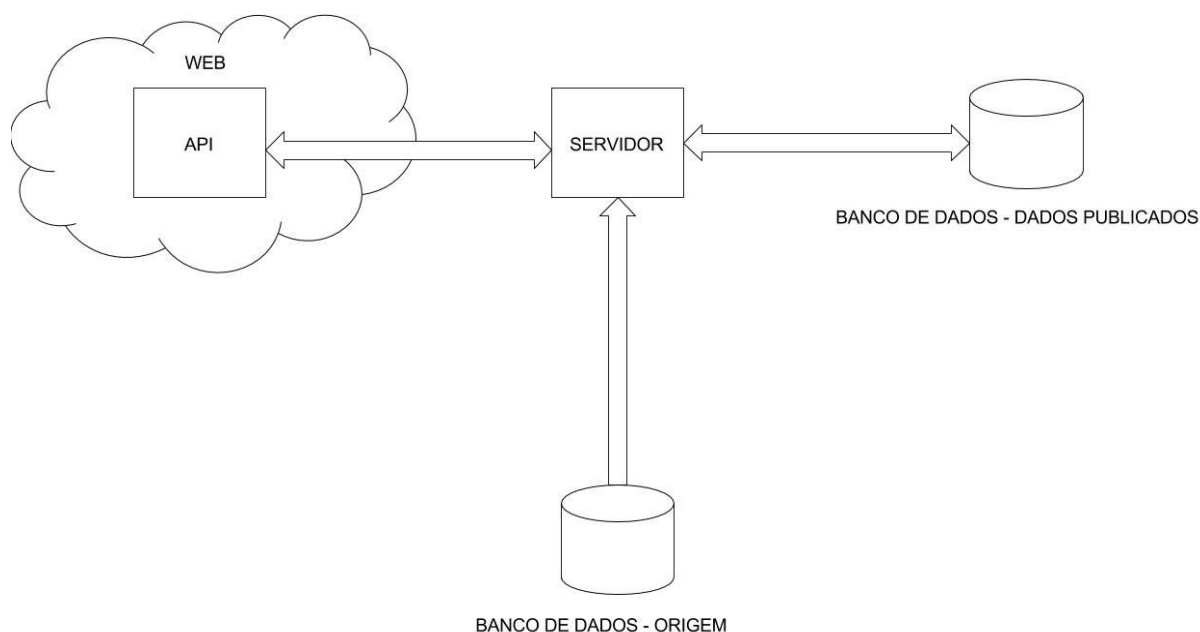


Figura 11: Arquitetura do DoWMaker

⁶ "Bitbucket | The Git solution for professional teams." <https://bitbucket.org/>. Acessado em 14 dez. 2016.

4.2 Servidor

DoWMaker é uma ferramenta que concentra todos os esforços no servidor de aplicação. É no servidor que todas as funcionalidades estão, desde a execução de consultas no banco de dados de origem e a captura dos dados, até a criação das APIs e a conversão dos dados para os quatro formatos disponíveis. Concebida para ser uma ferramenta multiplataforma e de fácil instalação, todo o desenvolvimento foi realizado com a linguagem de programação *Java*⁷, já que sua tecnologia permite que a aplicação seja executada em qualquer hardware ou software que contenha uma versão da JVM (*Java Virtual Machine*).

Para melhor implementação do padrão *JEE 8*⁸, foi utilizado o *Spring Framework*⁹, para que houvesse a injeção de dependências nas camadas do código. Além disso, por meio do *Spring*, são criadas as APIs, e os dados recebidos por elas são mapeados em classes *Java*. Também é possível a execução de rotinas temporais que são responsáveis por garantir que os conjuntos de dados estejam sempre atualizados.

Em questão da persistência dos dados da Base de Origem, o *Hibernate*¹⁰ é utilizado para a execução das consultas SQL¹¹ e o mapeamento em classes *Java*. Com múltiplas bibliotecas instaladas em suas pastas internas, a ferramenta DoWMaker não se limita apenas ao reconhecimento de um único SGBD relacional, podendo estabelecer uma comunicação com quatro dos SGBDs mais populares: *PostgreSQL*¹², *MySQL*¹³, *SQL Server*¹⁴ e *Oracle*¹⁵.

⁷ "java.com: JAVA + Você." https://www.java.com/pt_BR/. Acessado em 11 dez. 2016.

⁸ "Java Platform, Enterprise Edition (Java EE) | Oracle Technology Network." <http://www.oracle.com/technetwork/java/javaee/overview/>. Acessado em 11 dez. 2016.

⁹ "Spring." <https://spring.io/>. Acessado em 11 dez. 2016.

¹⁰ "Hibernate. Everything data. - Hibernate." <http://hibernate.org/>. Acessado em 11 dez. 2016.

¹¹ "SQL – Wikipédia, a enciclopédia livre." <https://pt.wikipedia.org/wiki/SQL>. Acessado em 11 dez. 2016.

¹² "PostgreSQL." <https://www.postgresql.org/>. Acessado em 11 dez. 2016.

¹³ "MySQL." <https://www.mysql.com/>. Acessado em 11 dez. 2016.

¹⁴ "SQL Server 2016 | Microsoft."

<http://www.microsoft.com/pt-br/server-cloud/products/sql-server/default.aspx>. Acessado em 11 dez. 2016.

¹⁵ "Database 12c | Oracle." <https://www.oracle.com/database/>. Acessado em 11 dez. 2016.

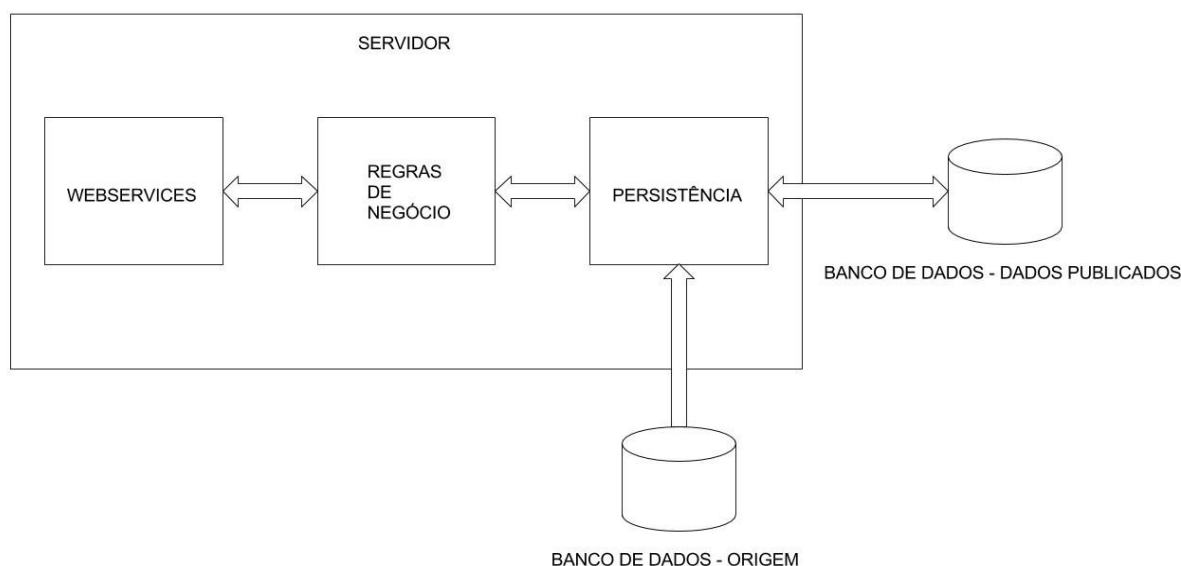


Figura 12: Arquitetura do módulo Servidor

A Figura 12 mostra como está organizado o módulo Servidor da DoWMaker. A camada de *Webservices* é responsável pela comunicação do servidor com a API. A segunda camada, realiza a implementação das regras de negócio relacionadas à ferramenta. A camada de Persistência faz a conexão e persistência dos dados nos servidores de banco de dados.

Para que os todos os formatos de dados estejam disponíveis, a DoWMaker utilizou-se de tecnologias mais específicas, incluindo: para o formato RDF foi utilizado o *Apache Jena*¹⁶, para JSON foi utilizado o *GSON*¹⁷, para XML o *JDOM*¹⁸ e para CSV um algoritmo simples que utiliza separadores de ponto e vírgula e quebra de linha. Todo o processo de transformação dos dados ocorre na camada de Regras de Negócio.

¹⁶ "Apache Jena - Home." <https://jena.apache.org/>. Acessado em 11 dez. 2016.

¹⁷ "GitHub - google/gson: A Java serialization/deserialization library that" <https://github.com/google/gson>. Acessado em 11 dez. 2016.

¹⁸ "JDOM." <http://www.jdom.org/>. Acessado em 11 dez. 2016.

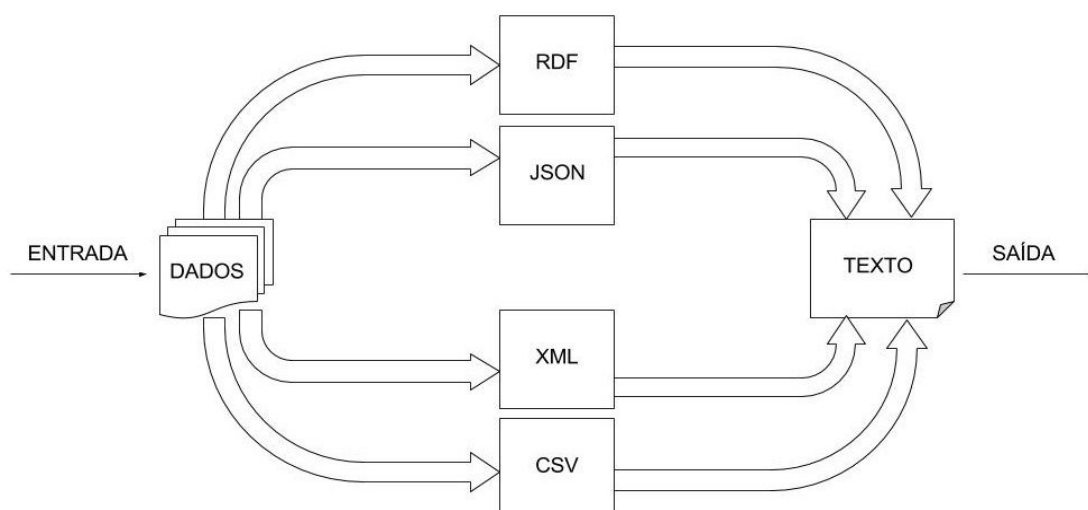


Figura 13: Transformação dos dados

Como pode ser visto na Figura 13, assim que os dados são extraídos do Banco de Dados, dependendo do formato escolhido pelo usuário, o sistema envia para um dos quatro pacotes e o selecionado realiza a conversão dos dados e os retorna em formato de texto, que será utilizado como resposta da API.

4.3 Banco de Dados Publicados

A estrutura do Banco de Dados Publicados foi criada usando um modelo de Banco de Dados Documental não relacional (*NoSQL*¹⁹), com o *MongoDB*²⁰. Esta tecnologia foi escolhida porque, além de permitir que uma grande quantidade de dados sejam consultados em uma velocidade maior do que seria em um banco relacional, oferece maior a flexibilidade nas estruturas de suas coleções de dados. Com isso, as atualizações dos conjuntos de dados realizadas pela DoWMaker permitem a alteração do esquema sem precisar reiniciar o servidor e realizar a transação de uma grande quantidade de dados sem perder desempenho.

¹⁹ "NoSQL Databases." <http://nosql-database.org/>. Acessado em 11 dez. 2016.

²⁰ "MongoDB." <https://www.mongodb.com/>. Acessado em 11 dez. 2016.

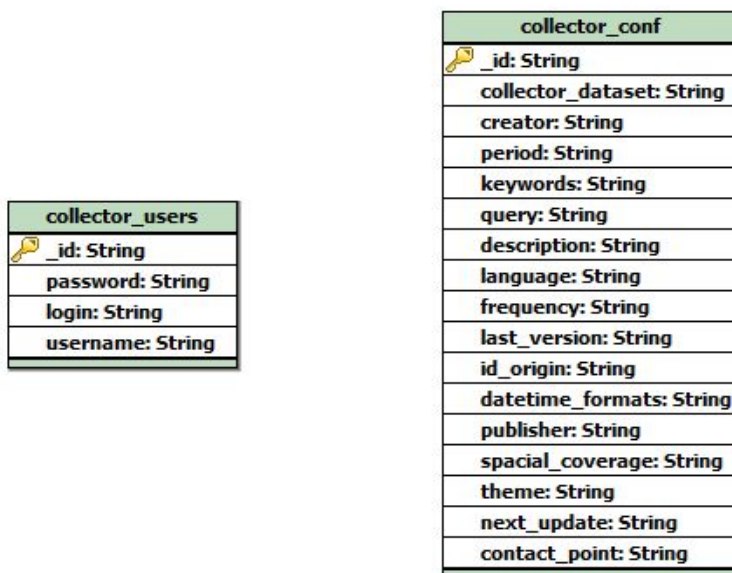


Figura 14: Esquema do Banco de Publicação sem inserções

A Figura 14 demonstra uma representação gráfica das coleções de dados do Banco de Dados Publicados, sem nenhum conjunto de dados inserido. Na coleção “collector_users”, são armazenados os dados dos Administradores do Sistema, esta coleção é acessada para a autenticação dos mesmos. Responsável pelo armazenamento dos metadados dos conjuntos de dados e outros recursos para funcionalidade do sistema, o “collector_conf” é o coração do Banco de Dados Publicados. Além da importância dos dados contidos nesta coleção, ela também pode ser usada como um índice para que o servidor saiba onde estão armazenados os dados de conjuntos de dados específicos.

Quando há a inserção de um novo conjunto de dados, novas coleções serão inseridas.

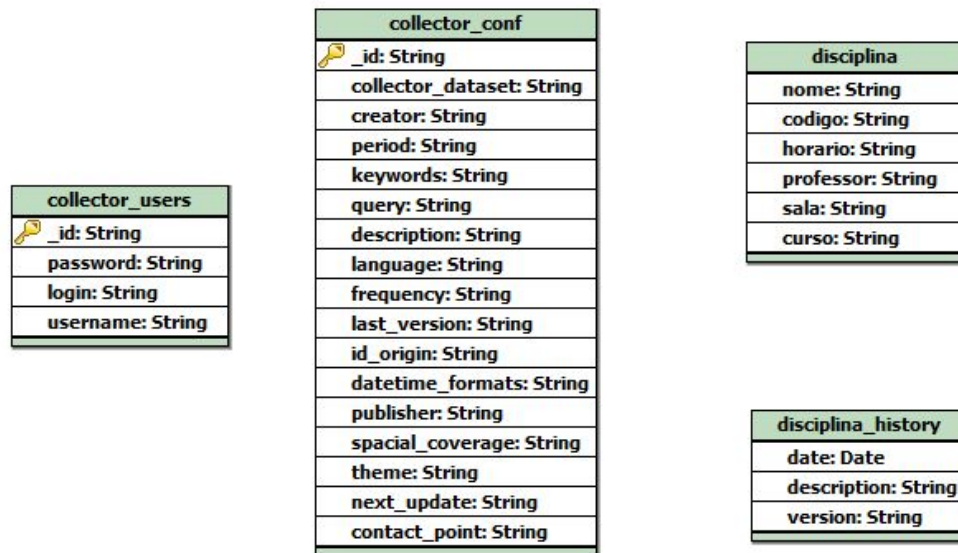


Figura 15: Esquema do Banco de Publicação com inserções

Para cada novo conjunto de dados inserido, duas novas coleções serão criadas, como pode ser visto na Figura 15. A primeira, com o nome do conjunto de dados, neste caso “disciplina” e a outra com o sufixo “_history”. Respectivamente, para armazenar os dados extraídos da origem e para armazenar o histórico de versionamento do conjunto de dados.

4.4 Instalação

Para que a ferramenta DoWMaker possa ser instalada, é necessário o servidor de aplicação *Tomcat 8*. Quando a aplicação for iniciada, três arquivos de configuração em uma pasta no sistema operacional (podendo variar de acordo com o sistema operacional, a ferramenta avisará o caminho para o diretório) serão criados. A princípio, a ferramenta não irá iniciar até que as configurações estejam prontas. Os arquivos e seus campos de configuração estão descritos a seguir:

- databaseconfig.properties

Propriedade	Descrição
sgbd	Seleção do SGBD do Banco de Dados de Origem, podendo ser: 1- PostgreSQL 2- MySQL 3- Oracle 4 - SQL Server

Propriedade	Descrição
url	URL de caminho para a conexão com o Banco de Dados de Origem
usuario	Usuário de autenticação do Banco de Dados de Origem
senha	Senha de autenticação do usuário no Banco de Dados de origem

- mongoconfig.properties

Propriedade	Descrição
url	URL de caminho para a conexão com o Banco de Dados Publicados

- uri.properties

Propriedade	Descrição
uriBase	Base URL da qual o sistema está hospedado. Informação necessária para
domain	Nome do domínio dos dados

Com os campos preenchidos de forma correta, o servidor de aplicação deverá ser reiniciado e a ferramenta já poderá ser utilizada normalmente.

4.5 Avaliação

Como prova de conceito e avaliação da ferramenta, a implantação da DoWMaker foi realizada para a publicação dos dados do SIGA - UFPE em formato aberto. Sistema este que contém dados sobre os discentes, docentes e cursos da UFPE. O trabalho foi desenvolvido em conjunto com outros dois trabalhos de graduação, sendo um com o objetivo da criação dos conjuntos de dados a partir do Banco de Dados do SIGA [16] e o outro na construção de um portal para o consumo dos dados [15].

A princípio, uma cópia de um subconjunto das tabelas do Banco de Dados do SIGA foi realizada. Então, a partir desta cópia, foi criada uma nova instância no

Oracle 11g. Dessa forma, os conjuntos de dados foram inseridos, podendo ser acessados de maneira instantânea pelas suas devidas URIs.

Após implantação da DowMaker, foi possível a publicação de uma total de 6 Conjuntos de Dados [16]. A Figura 16 mostra uma tupla do Conjunto de dados “orgão”, quando se é requisitada a visualização de seus dados em XML:

```
▼<XML>
  ▼<orgao>
    <_id>324_20170110202414</_id>
    <abrevc_orgao_comp>null</abrevc_orgao_comp>
    <codg_orgao_siape>null</codg_orgao_siape>
    <data_excs_orgao>null</data_excs_orgao>
    <CNPJ_instc_ensn>24134488000108</CNPJ_instc_ensn>
    <nivel_hierqc>1</nivel_hierqc>
    <codg_campus>1</codg_campus>
    <hpag_orgao>null</hpag_orgao>
    <version>20170110202414</version>
    <desc_naturz_orgao>null</desc_naturz_orgao>
    <codg_orgao_sup>54</codg_orgao_sup>
    <desc_detld>null</desc_detld>
    <last_version>true</last_version>
    <nome_orgao>COORDENAÇÃO DO CURSO DE PSICOLOGIA</nome_orgao>
    <sigla_orgao>null</sigla_orgao>
    <codg_predio>5</codg_predio>
    <data_criac_orgao>null</data_criac_orgao>
    <email_orgao>null</email_orgao>
    <seqnc_comp_orgao>null</seqnc_comp_orgao>
    <codg_tipo_orgao>14</codg_tipo_orgao>
  </orgao>
▼<orgao>
```

Figura 16: Conjunto de Dados Publicado em XML

Ainda no mesmo Conjunto de Dados, as Figuras 17 e 18 mostram a visualização dos metadados e do histórico de versionamento, respectivamente:

```
{
  "_id" : ObjectId("58756d1ee032b31bccaf798b"),
  "date" : "Tue Jan 10 20:24:14 GMT-03:00 2017",
  "description" : "Primeira inserção",
  "version" : "20170110202414"
}
{
  "_id" : ObjectId("587573c8e032b31bccaf79a8"),
  "date" : "Tue Jan 10 20:52:39 GMT-03:00 2017",
  "description" : "Alteração de estrutura",
  "version" : "20170110205239"
}
>
```

Figura 17: Histórico de versionamento em Json

```

{
  "datasetTitle": "curso",
  "creator": "Priscila Thyfani",
  "period": "Possui dados desde 1970.",
  "keywords": "Programa Formação, Curso",
  "description": "Conjunto de dados que contém informações sobre os Programas de Formação, ou seja, cursos da UFPE.",
  "language": "Portuguese",
  "last_version": "20170110212239",
  "id_origin": "codg_progm_formc",
  "dateTimeFormats": "CSV, XML, JSON",
  "publisher": "Cin - UFPE",
  "spacialCoverage": "Recife, Brasil",
  "theme": "Educação",
  "contactPoint": "ptrg@cin.ufpe.br",
  "frequency": "Yearly",
}

```

Figura 18: Metadados em Json

A API da DoWMaker foi utilizada pelo portal OpenAcademy [15], como pode ser visto na Figura 19, para o uso dos metadados e dos conjuntos de dados para geração de indicadores, visto na Figura 20:

OpenAcademy

Home

Consultas

Indicadores

Metadados

OpenAcademy UFPE

Metadados

Descrição do Conjunto de Dados

Título	Pessoa
URI	http://localhost:8080/find/pessoa
Palavras-chave	segurança, automação
Publication date	2017
Criadores	Wilker Santos
Frequência de Atualização	Every Hour
Tema	Segurança
Linguagem	Portuguese
Versão Atual	20170107111216

Formatos dos Conjuntos de Dados

RDF

Título	RDF do Conjunto de Dados - 18-07-2016
Descrição	Arquivo RDF do Conjunto de Dados
Tipo de Arquivo	RDF/XML
Licença	

Figura 19: Tela de metadados do OpenAcademy [15]



Figura 20: Tela de indicadores do OpenAcademy [15]

Capítulo 5. Conclusão e Trabalhos Futuros

A ferramenta DoWMaker possibilita que os provedores de dados na Web possam realizar a publicação de forma automatizada. Ela pode ser instalada na maioria dos ambientes que possuam como forma de armazenamento de seus dados um Banco de Dados Relacional. Por meio dela, os provedores de dados diminuirão seus esforços no ato da publicação e o conhecimento sobre o formato dos dados publicados poderá ser abstraído.

Dentre as principais contribuições deste trabalho, destacam-se:

- Especificação e implementação da ferramenta DoWMaker.
- Realização de uma prova de conceito com dados do SIGA da UFPE.
- Disponibilidade da ferramenta para ser utilizada por qualquer entidade que queira realizar a publicação de seus dados na Web.

Este trabalho contribui como parte inicial de um estudo ainda maior sobre futuros Sistemas Gerenciadores de Dados da Web. Algumas limitações são conhecidas, dando necessidade a futuras ações. Dentre elas, destacam-se

- Implementar soluções para que os dados possam ser extraídos em mais de uma fonte ao mesmo tempo, sendo elas não só Banco de Dados Relacionais, mas também de Banco de Dados Não Relacionais, APIs e dados armazenados em documentos.
- Desenvolver uma interface gráfica para o sistema, de forma que facilite ainda mais o seu uso.
- Desenvolver estratégias para que consultas mais complexas possam ser realizadas na ferramenta, podendo ocorrer combinações entre os múltiplos conjuntos de dados e dar mais liberdade ao usuário no momento do consumo dos dados.

Com o forte crescimento da publicação de dados na Web, e os benefícios que trazem, a necessidade de soluções que automatizam todo o processo de publicação torna-se cada vez maior. É conhecido que muitos dos problemas dos dados na Web são frutos de uma publicação realizada sem muitas preocupações com quem consumirá, e isso tem como consequência dificuldades para o consumo de dados. Por isso, com a DoWMaker, é esperado que os publicadores encontrem

menos barreiras para a publicação adequada dos dados e que a ferramenta seja um modelo para futuras soluções.

Referências

- [1] LÓSCIO, B.F., FILHO, F.W. “Web Semântica: Conceitos e Tecnologias”(2010)
- [2] Lóscio, B. F., Oliveira, M. I. S., Bittencourt, I. I., 2015. Publicação e Consumo de Dados na Web: Conceitos e Desafios. Dados na Web, Minicurso SBB, 2015.
- [3] ANTONIOU, G.; HARMELEN, F. V. A Semantic Web Primer. Cambridge 2008.
- [4] BERNERS-LEE, T., HENDLER, J., LASSILA O. (2001) “The Semantic Web. ScientificAmerican”,v. 284, n. 5, p. 34-43
- [5] LIMA, J.C., CARVALHO, C.L. “Uma visão da Web Semântica”. Universidade Federal de Goiás 2008.
- [6] ALMEIDA, B.A. “Uma introdução ao XML, sua utilização na internet e alguns conceitos complementares”. Universidade Federal de Minas Gerais 2004.
- [7] Möller, K. (2013). Lifecycle models of datacentric systems and domains: The abstract data lifecycle model. Semantic Web 4, 1, 6788.
- [8] BIZER, C., HEATH, T., e Berners-LEE, T. Linked data the story so far. Semantic Services, Interoperability and Web Applications: Emerging Concepts, 205227.(2009)
- [9] CUNHA, DANUSA R. B.; LÓSCIO, B. F, SOUZA, D. Linked Data: da Web de Documentos para a Web de Dados. Recuperado em 17 de novembro de 2016, do <<http://die.ufpi.br/ercemapi2011/minicursos/MC2.pdf> >.
- [10] The Open Definition. Recuperado em 15 de novembro de 2016 do: <<http://opendefinition.org>>
- [11] The Three Laws of Open Government Data. Recuperado em 15 de novembro de 2016 do: <<http://eaves.ca/2009/09/30/three-law-of-open-government-data/>>.
- [12] O que são dados abertos? Recuperado em 15 de novembro de 2016, do<<http://dados.gov.br/dados-abertos/>>
- [13] Dados Abertos Conectados. Recuperado em 8 de novembro de 2016, do <<http://ceweb.br/livros/dados-abertos-conectados/>>

[14] Lóscio, B. F., Burle, C., Calegari, N. Data on the Web Best Practices. W3C Second Public Working Draft (2015). Recuperado em 17 de novembro de 2016, do <<http://www.w3.org/TR/dwbp/>>

[15] Oliveira, F. E., OpenAcademy: Uma Ferramenta de Gestão Acadêmica com Dados Abertos Conectados na Universidade Federal de Pernambuco. Universidade Federal de Pernambuco(2017).

[16] Melo, P. T. R. G., Geração de Dados Abertos sobre Discentes da UFPE. Universidade Federal de Pernambuco(2017).