

Análise da representação semântica de modelos de dados do formato JSON

João Batista da Silva Júnior¹, Paulo Caetano Silva¹

¹Programa de Pós-Graduação em Sistemas e Computação – Universidade Salvador, BA

joaojunior.mail@gmail.com, paulo.caetano@unifacs.br

Abstract. *The objective of this work was to evaluate the representation of semantic web data based on the JSON file format. This file format has been chosen because it reduces the amount of data trafficked in the network and has been widely used as Electronic Data Interchange for many of today's Web servers. To carry out the study, an experiment was carried out to evaluate the utilization of processing and memory resources of semantic JSON data models. As a result, it was concluded that JSON has a lot of potential to semantically represent models and structure of data and objects.*

Resumo. *O objetivo deste trabalho é avaliar a representação de dados da web semântica com base no formato de arquivo JSON. Este formato de arquivo tende a reduzir a quantidade de dados trafegados na rede e vem sendo bastante utilizado como intercâmbio eletrônico de dados em muitos dos servidores Web da atualidade. Para realização do trabalho, foi realizado um experimento que avaliou a utilização de recursos de processamento e memória dos modelos semânticos de dados JSON. Como resultado, concluiu-se que o JSON apresenta muito potencial para representar semanticamente modelos e estrutura de dados e objetos.*

1. Introdução

Com a evolução das tecnologias e a diversidade de sistemas e plataformas digitais, tornou-se comum a necessidade da comunicação entre sistemas, sejam eles desktops, web ou embarcados.

Diversos modelos e padrões de comunicação surgiram para suprir a necessidade do Intercâmbio Eletrônico de Dados (*Electronic Data Interchange – EDI*), que segundo Hutchison [Hutchison 1987] não se limita nas diferenças entre computadores e equipamentos, mas supera quaisquer necessidades de troca de mensagens entre diferentes sistemas informatizados, não implicando necessariamente em uma troca de mensagens em tempo real.

Turban et. Al. [Turban et al. 2005], define o EDI como uma comunicação

e movimentação eletrônica de documentos padronizados entre entidades e baseia-se no uso de padrões de mensagens, garantindo que todos os participantes da comunicação utilizem uma linguagem comum, de modo que todos os participantes possam interpretar ou escrever uma mensagem eletrônica. É importante que o destinatário da mensagem conheça sua estrutura e seja capaz de recebê-la e tratá-la automaticamente.

Devido ao crescimento do uso da internet e o número de dispositivos conectados no mundo [Cisco 2017], instituições como a Organização das Nações Unidas (ONU) conduziram estudos com o foco em criar uma padronização universal da troca de mensagens devido à multiplicidade de dialetos entre empresas e instituições.

Como resultado dos estudos realizados, desenvolveram-se, durante anos, diversos tipos de tecnologias que permitem essas comunicações, dentre elas está a Notação de Objeto Javascript (*Javascript Object Notation* – JSON).

Este artigo tem como objetivo analisar a aplicação de diferentes modelos de representação de dados no formato JSON e identificar as possibilidades de aplicação, servindo de orientação a pesquisas futuras que utilizem esta abordagem.

A fim de propiciar a obtenção do objetivo deste trabalho, foram definidas as seguintes etapas que permitiram o seu desenvolvimento:

1. *Pesquisar os modelos de dados relativos ao formato JSON;*
2. *Definir os modelos que serão abordados;*
3. *Elencar as principais características desses modelos;*
4. *Definir a estrutura do objeto a ser avaliado;*
5. *Definir as tecnologias responsáveis pela avaliação;*
6. *Avaliar os resultados.*

Este trabalho está dividido em 5 seções: A 2ª seção apresenta os principais conceitos necessários ao entendimento do trabalho: JSON, Web Semântica, RDF e sua aplicabilidade. Na 3ª seção é apresentada a metodologia para a realização do experimento, a 4ª seção discute os resultados obtidos por meio do experimento e, por último, na 5ª seção a conclusão deste trabalho é apresentada.

2. JSON e Web Semântica - Conceitos básicos

Com a necessidade da definição de um padrão para troca de mensagens, surgiu na década de 90 a Linguagem de Marcação Extensível (*eXtensible Markup Language* – XML), que segundo a *World Wide Web Consortium* (W3C), principal organização de padronização da *World Wide Web*, é a melhor linguagem de marcação para necessidades especiais [Group 2008a].

A XML trouxe à Web a possibilidade não só de trocar mensagens, mas a possibilidade de definir estruturas específicas para as mais diversas necessidades, porém para o uso desta tecnologia é necessário a criação de estruturas de texto que torne viável a comunicação rápida dos dados em redes com pouca capacidade de banda, sem exigir da rede uma maior quantidade de tráfego de dados. A Figura 1 apresenta um modelo simples da estrutura de dados de uma mensagem XML.

Com o passar do tempo a web e as suas necessidades mudaram, surgindo novos conceitos de usabilidade, web semântica, acessibilidade e experiência de usuário. O uso de comunicações assíncronas tornou-se quase que pré-requisito no desenvolvimento de sistemas, assim como manipulação de dados com a linguagem Javascript. Desta forma, o Javascript se tornou bastante forte com o uso do "Javascript Assíncrono e XML" (*Asynchronous Javascript and XML – AJAX*). Mas, a manipulação de arquivos XML para realização do *parsing* tornou-se custoso durante a etapa de codificação no processo de desenvolvimento de software.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<Movies>
  <movie>
    <name>The Matrix</name>
    <releaseYear>1999</releaseYear>
    <categories>
      <category>Science fiction</category>
      <category>Action</category>
    </categories>
  </movie>
</Movies>
```

Figura 1. Exemplo da estrutura de um arquivo XML

Com o surgimento da necessidade de uma linguagem comum que seja compacta e que exige uma menor quantidade de processamento, surgiu o JSON.

2.1. JSON

JSON é um formato de texto derivado da linguagem Javascript e é representado por uma coleção de objetos compostos por pares de chave-valor conforme definição da 8ª edição do ECMAScript Language Standard [International 2011], que é responsável pela definição da linguagem Javascript.

Segundo [T. Bray 2013], o JSON é uma estrutura de texto no formato de objeto Javascript e pode ser composto de subobjetos. Sua estrutura possibilita uma diversificada representação de dados, pois o mesmo suporta quatro tipos de dados primitivos e estrutura de subobjetos, como *strings*, *number*, *integer*, *boolean*, *object* e *array*. A Figura 2 representa um modelo de objeto JSON.

```

{
  "image": {
    "width": 800,
    "height": 600,
    "title": "JSON-
logo",
    "thumbnail": {
      "url": "http://www.json.org/img/json160.gif",
      "height": 160,
      "width": 160
    },
    "animated": false,
    "IDs": [116, 943,
234]
  }
}

```

Figura 2. Exemplo da estrutura de um arquivo JSON

Diferente da XML, o JSON utiliza uma estrutura de arquivos composta de propriedade-valor e permite uma visível redução de *bytes* ao arquivo, possibilitando o intercâmbio de dados com uma menor exigência de banda.

2.2. Web Semântica

Criada em 1989 por Tim Berners-Lee [Berners-Lee 2017], a Web surgiu para proporcionar um ambiente digital de informação e seus dados possuem significados bem definidos, facilitando a comunicação entre pessoas e seus meios computacionais.

No entanto, o crescimento da Web ocorreu de forma exponencial e descentralizada [Souza e Alvarenga 2004]. Isso dificultou a inserção de dados com informações semanticamente definidas, obrigando o meio receptor conhecer o contexto e tratar os respectivos dados conforme previa configuração para sua interpretação.

Para suprir esta necessidade, surgiu o conceito de Web Semântica, que é uma extensão da Web e visa agregar significados aos dados, melhorando a sua compreensão e interpretação baseados em um contexto. Deste modo, retira-se a necessidade de o agente computacional possuir uma prévia configuração para o conteúdo e possibilita generalização. A Web Semântica permite que os mecanismos de pesquisa compreendam o significado das informações e proporcionem uma melhor experiência ao usuário. De modo geral, é um modelo tecnológico que partilha o significado das informações globalmente entre as máquinas.

Embora o termo Semântica tenha como descrição "o estudo do significado das palavras", Guiraud & Mascarenhas [Guiraud and Mascarenhas 1972] questionam e distinguem a semântica em três principais problemas:

1. a ordem dos problemas psicológicos, que relaciona os estados fisiológicos e psíquicos dos interlocutores nos processos de comunicação de signos;
2. a ordem dos problemas lógicos, que estabelece as relações dos signos com a realidade no processo de significação;
3. a ordem dos problemas linguísticos, que estabelece a natureza e as funções dos vários sistemas de signos.

No que tange a Web Semântica, a segunda definição é a que melhor à representa, pois associa significados aos documentos da Web por meio de metadados descritivos. Segundo a W3C [Group 2008b], a abordagem semântica para documentos web permite que sejam criados vocabulários e escrito regras para o gerenciamento de dados. Para isso, existem algumas tecnologias e conceitos que permitem a sua aplicabilidade:

- Vínculo de Dados: Para que os significados dos dados sejam compreendidos pelos agentes computacionais é importante a existência de pelo menos um repositório de dados na Web, disponível em um formato padrão, acessível e gerenciável pelas ferramentas da Web Semântica. O W3C disponibiliza tecnologias para vínculo dos dados, por exemplo, o *Resource Description Framework* – RDF. Estas tecnologias permitem a integração e raciocínio dos dados em grande escala e baseiam-se essencialmente na acessibilidade e incorporação dos dados vinculados em múltiplos níveis de complexidades.

- Ontologia: Ontologias são responsáveis por definir, classificar e relacionar os conceitos necessários para representar dados em um contexto, permitindo caracterizar possíveis relacionamentos e definir possíveis restrições em sua utilização. As ontologias apoiam a integração de dados, de modo a ajudar a identificar ambiguidades de relacionamentos e possíveis relacionamentos entre os dados.

- Consulta: Em programação, as consultas têm por finalidade recuperar as informações dos dados semânticos através de um conjunto de tecnologias e protocolos. A metodologia necessária para a recuperação dos dados se vale do tipo de vínculo de dados definido. Assim como nos bancos de dados relacionais e os *no-SQL*, a consulta de dados precisa de uma linguagem que permita retornar os metadados definidos no vínculo de dado, normalmente essa linguagem é representada usando o RDF como formato de dados.

- Inferência: Instrumento que possibilita a melhoria da qualidade da integração de dados na Web, permitindo a descoberta de novos relacionamentos e analisando automaticamente as informações dos dados, assim como gerenciar seus significados. Técnicas de inferência também ajudam na descoberta de inconsistências dos dados integrados.

- Aplicação vertical: Representa a aplicação genérica de áreas específicas que possibilitam: apoiar os procedimentos das operações, melhorar suas eficiências e proporcionar melhores experiências de usuários. As aplicações verticais podem apresentar casos de uso específicos e às vezes não triviais.

Para o desenvolvimento deste trabalho, adotou-se como ferramenta de vínculo de dados o Framework de Descrição de Recursos (*Resource Description Framework* – RDF). A próxima seção abordará os principais conceitos sobre RDF.

2.3. Resource Description Framework - RDF

Aprovado e recomendado pela W3C em 1999 como um modelo padrão para intercâmbio de dados na Web. O RDF possui recursos que facilitam a fusão de dados e suporta a injeção de metadados, ampliando a estrutura dos dados e adicionado significado aos dados por meio de relações externas a estrutura, que são especificações que proporcionam a possibilidade da expansão de conteúdo devido a sua estrutura generalizada e independente.

Ao longo dos anos, a comunidade W3C vem apresentando várias tecnologias que são derivadas do RDF, dentre elas, a mais atual, lançada em 2015, é a *Linked Data Plat- form* – LDP [Speicher et al. 2015].

A LDP é uma plataforma que utiliza o RDF para vincular, editar, excluir e criar metadados à uma estrutura de dados web. Desta forma é possível vincular significados aos dados, gerando um sentido semântico para eles. Para que seja possível a implementação da LDP, é necessário a utilização de *Uniform Resource Identifier* – URI que é uma sequência de caracteres que tem por finalidade identificar um recurso na rede.

No contexto deste trabalho, foi adotado o *JSON for Linking Data* – JSON-LD, que é uma definição estrutural da representação dos dados no formato JSON que utiliza a tecnologia LDP. A próxima seção apresentará o tema JSON-LD e seus modelos de dados.

2.4. JSON-LD

JSON-LD significa *Javascript Object Notation* para vinculação de dados e tem como objetivo de ligar dados à Web, este vínculo é realizado de tal forma que ele pode estar interligado entre diferentes sites, permitindo que os sites possam se referenciar a dados de outros sites. O JSON-LD é uma extensão do JSON que proporciona uma melhor utilidade aos desenvolvedores web, designers ou qualquer pessoa que precise transmitir informações entre navegadores e servidores web.

2.4.1. Funcionamento do JSON-LD

Para melhor compreensão do funcionamento do JSON-LD, é necessário primeiramente compreender o funcionamento da Web. Atualmente, para que um usuário acesse um conteúdo na Web, ele utiliza um navegador (Google Chrome, Internet Explorer, Mozilla Firefox, etc.) que tem como objetivo a interpretação de documentos na Web. Estes documentos são normalmente implementados na linguagem HTML e ficam disponíveis em servidores web que retornam o conteúdo do documento quando solicitado por um utilizador. Estes documentos podem comportar em sua estrutura imagens, textos, vídeos, áudio e ferramentas de

hiperlink que tem como objetivo vincular um documento a outro documento.

Os *hiperlinks* permitem que o usuário navegue em outro documento, inclusive um documento que esteja em outro servidor, que por sua vez pode possuir outros *hiperlinks* que também podem ter várias ligações para outros documentos.

Normalmente os dados pertencentes aos respectivos documentos são cadastrados por meio de Sistemas Gerenciadores de Conteúdo (*Content Management System* – CMS), o que proporciona a inserção, alteração e exclusão dos dados nos documentos de forma dinâmica.

Neste contexto, pode-se dividir a Web em dois grupos:

- **Front-End:** É basicamente representado pelos elementos visuais compostos no documento Web, que normalmente é composto por HTML e Javascript, e permite a leitura e transmissão de dados no formato JSON por meio de requisições Web.
- **Back-End:** Componente destinado a intercomunicação entre agentes computacionais e que, geralmente, transmite as informações dos dados ao Front-End no formato de JSON. Este também permite expressar os dados para os navegadores e máquinas com legibilidade.

Como abordado anteriormente, o JSON é recomendado pela W3C como uma solução para o intercâmbio de dados eletrônicos [T. Bray 2013], permitindo facilidade no seu manuseio tanto para os desenvolvedores web quanto para os servidores web. O problema é o que acontece quando o recebimento dos dados advém de múltiplos servidores, mas com dados completamente distintos e possuindo a mesma estrutura. Como exemplo desta situação, imaginemos um serviço que recebe dados de dois diferentes sites, no entanto, um dos sites disponibiliza dados com informações de um produto, enquanto o outro, informações de um serviço de manutenção de notebooks. Se os dados possuírem a mesma estrutura e o serviço que os está consumindo precisar misturá-los, o objeto final de leitura terá ambiguidade nos dados. As Figuras 3 e 4 apresentam os modelos JSON conforme descrição.

```
{
  "name": "Smartphgne XX",
  "homepage": "http://www.domain.com",
  "email": "shop@domain.com",
  "telephone": [ "+55-00-00000-0000" ]
}
```

Figura 3. Modelo JSON para um produto

O JSON-LD surgiu para tentar resolver este problema de ambiguidade e possibilitar o compartilhamento de múltiplos dados com seus respectivos contextos, especificando-os através de *links*.

```

{
  "name": "SOSNotebooks",
  "homepage": "http://www.domain.com",
  "email":
  "sosnotebooks@domain.com",
  "telephone": [ "+55-11-11111-
1111 " ]
}

```

Figura 4. Modelo JSON para um serviço de manutenção e notebooks

O JSON-LD permite que os dados sejam vinculados e referenciados por outros servidores por meio do JSON. Assim como os documentos web que utilizam o *hiperlink* do HTML para realizar ligações entre diversos documentos, os dados do JSON-LD são também referenciados por meio de *links* [Group 2014b], dando significados aos dados. A Figura 5 exemplifica um modelo de dados JSON-LD.

```

{
  "@context": "http://json-
ld.org/context/person.jsonld", "@id":
  "person/John",
  "name": "John",
  "homepage": "http://www.domain.com",
  "email": "john@domain.com",
  "telephone": [ "+55-00-00000-
0000 " ]
}

```

Figura 5. Exemplo da estrutura de um arquivo JSON-LD

É observado que o JSON-LD representa os dados juntamente com seu contexto, permitindo a junção de vários objetos de dados sem criar ambiguidade na estrutura final. Assim, o JSON-LD propõe a publicação, por meio da RDF, de uma forma de expressar dados vinculados e a reutilização destes vínculos, incluindo todas as informações referentes a outros objetos para seus respectivos dados, fornecendo semântica. Deste modo, o JSON-LD realiza a marcação semântica dos dados para serem consumidos por motores de busca, sites, redes sociais, etc.

O JSON-LD possui três camadas fundamentais para seu pleno funcionamento:

1. Definir um contexto ao documento;
2. Definir os identificadores ao documento;
3. Utilizar as terminologias disponíveis nas referências do contexto.

Destarte, os usuários recebem os retornos dos conteúdos que atendem as suas necessidades, pois mesmo visualizado no conteúdo dos documentos web, os textos, imagens, i.e., cada conteúdo, incorpora aos dados um vínculo e sentido dentro de um contexto. No entanto, para que os vínculos sejam realmente

contextualizados de forma generalizada, é necessário que as descrições dos contextos sejam identificadas globalmente.

Para este trabalho, foi utilizado o vocabulário desenvolvido pela comunidade W3C para identificação global, denominado de Schema.org [Guha et al. 2016]. A próxima seção abordará a metodologia adotada para o desenvolvimento deste trabalho.

3. Metodologia

Esta seção tem por finalidade apresentar os procedimentos realizados para avaliação dos dados JSON-LD, para tal, foram definidas as etapas para definição de modelos, das tecnologias, do ambiente de testes e, por fim, definição do teste. A seguir detalha-se cada uma dessas etapas.

3.1. Definição dos Modelos

Nesta etapa foram encontrados alguns modelos de dados que são compreendidos como JSON-LD. Foram identificados dois tipos de modelos representativos: *JSON-LD Context* e *JSON-LD Expanded*. As Figuras 6 e 7 ilustram os modelos dos objetos utilizados neste trabalho.

```
{
  "@context" :
  "http://schema.org/Person", "name":
  "Barack Obama",
  "givenName": "Barack",
  "familyName": "Obama",
  "jobTitle": "44th President of the United States"
}
```

Figura 6. Objeto no modelo JSON-LD Context

```
{
  "http://schema.org/Person": {
    "http://schema.org/name" : [{ "@value" : "Barack Obama"
    }], "http://schema.org/givenName" : [{ "@value" :
    "Barack" }], "http://schema.org/familyName" : [{
    "@value" : "Obama" }], "http://schema.org/jobTitle" :
    [{ "@value" : "44th President of
    the United States" }]
  }
}
```

Figura 7. Objeto no modelo JSON-LD Expanded

3.2. Definição das Tecnologias

Nesta etapa, foram definidas as tecnologias necessárias para realização de uma análise comparativa do uso de recursos dos modelos JSON-LD Context e JSON-LD Expanded (i.e., uso de processamento e de memória RAM). Para isso, foram pesquisadas ferramentas e tecnologias em comunidades de desenvolvimento de software e definidas para este trabalho as seguintes tecnologias:

- **Servidor Web:** Responsável por disponibilizar os recursos necessários para executar os serviços requeridos para a realização do teste. O servidor foi preparado para executar o Apache Software Foundation 2.2, com o PHP 5.6, para executar a linguagem de desenvolvimento web PHP e o Sistema Gerenciador de Banco de Dados (SGBD) MySQL Server 5.5, para persistir os dados;
- **Web Service:** Tem como finalidade enviar, receber e tratar os dados provenientes dos dados enviados por um transmissor de requisições HTTP. O Web Service foi concebido com a utilização do Framework PHP Laravel, apropriado para o desenvolvimento de aplicações Web, utilizando o padrão de arquitetura Model-View- Controller (MVC). A escolha do Laravel foi em função de sua popularidade, vasta documentação e disponibilidade dos módulos de gerenciamento de conteúdos semânticos RDF;
- **Transmissor de Dados:** Aplicação responsável pela transmissão de dados através de requisições HTTP. Para esta finalidade foi utilizada a aplicação Postman devido a sua popularidade e a possibilidade de realizar múltiplos tipos de requisições;
- **Ferramenta de Teste:** Responsável pela realização de testes funcionais e medição de desempenho. Foi utilizado como ferramenta o Apache JMeter 3.2, pois é um software *open source* e possui uma comunidade ativa de desenvolvedores e usuários.

3.3. Definição do Ambiente

A definição do ambiente em que o teste será realizado pode impactar diretamente nos seus resultados, pois o resultado pode variar conforme a capacidade de processamento de uma máquina computacional. Para este trabalho foi utilizado um Macbook Pro de 13 polegadas, com processador 2,4GHz Intel Core i5, 3ª geração, com Turbo Boost de até 3 GHz, memória RAM de 8GB DDR3 de 1666MHz, Placa de Vídeo Integrada Intel de 512MB, armazenamento SSD de 256GB e sistema operacional macOS Sierra versão 10.12.5.

3.4. Definição do Teste

Para a realização dos testes, foram utilizados os modelos JSON-LD (Figura 6 e Figura 7) e realizadas 1000 requisições simultâneas ao *Web Service*. Para isso, utilizou-se o Apache JMeter, que é um software que tem como objetivo realizar testes de aplicações web¹.

¹ <http://jmeter.apache.org>

Para que fosse possível realizar o teste em ambiente controlado, foi deixado em execução no sistema operacional apenas o servidor Web que retorna os respectivos modelos JSON-LD (Figura 6 e Figura 7), o software Apache JMeter e as aplicações nativas do sistema operacional que não podiam ser encerradas em função das restrições do sistema operacional.

Os testes para cada modelo JSON-LD apresentados neste trabalho foram realizados de forma consecutiva e não simultânea. Isto foi feito para que a execução de um teste não interferisse nos resultados de outro teste. A próxima seção apresenta os resultados advindos dos testes realizados.

4. Resultados

A realização dos testes para realização da comparação entre os modelos JSON-LD Context e JSON-LD Expanded foram divididos em 2 etapas apresentadas nas próximas seções.

4.1. Teste Bruto

Nesta etapa os testes foram realizados sem nenhum tratamento dos dados, fazendo a leitura de todos os dados de forma unitária dos vínculos. A Tabela 1 apresenta os resultados obtidos.

Tabela 1. Resultados do teste bruto

	JSON-LD Context	JSON-LD Expanded
Total de dados	1000	1000
Tempo de execução	126s	187s
Uso médio de memória	54MB	113MB
Uso médio de	67%	48%

4.2. Teste com Correções

Nesta etapa foram implementados as boas práticas recomendadas pela W3C [Group 2014a], que referem-se ao uso de cache e definição do tipo de mídia de contexto para os pacotes enviados². A Tabela 2 apresenta os resultados obtidos.

Tabela 2. Resultados do teste com correções

	JSON-LD	JSON-LD
Total de dados	1000	1000
Tempo de execução	12s	30s
Uso médio de memória	63MB	78MB
Uso médio de	53%	27%

Baseado nos dados obtidos nos testes foi possível inferir alguns

² <https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html>

resultados. Quanto às principais aplicações, pôde-se verificar que nas duas propostas buscou-se avaliar a melhor prática para o uso do vínculo de dados com o JSON-LD. Portanto, foi identificado que com a implantação das recomendações de boas práticas, os resultados foram obtidos em menor tempo e com um menor consumo de processamento e memória. Entretanto, mesmo com o JSON-LD Context obtendo uma menor exigência de hardware, foi identificado que o JSON-LD Expanded disponibiliza ao agente interpretador uma semântica independente, pois permite uma personalização da estrutura de dados mais detalhada no próprio objeto, ou seja, independente de outro objeto. No entanto, este modelo exige uma maior quantidade de bytes trafegados na rede.

Portanto, acredita-se que mesmo com o JSON-LD Expanded possibilitando uma maior abrangência dos contextos dos dados, o uso do JSON-LD Context é uma melhor opção quando o objetivo é vincular dados semânticos a um objeto com o menor consumo de dados e recurso computacional.

5. Conclusão

Este artigo teve com proposta a análise da apresentação semântica de modelos de dados do formato JSON com o objetivo de que fossem identificadas as possibilidades de aplicação, servindo de orientações a trabalhos futuros que utilizem esta abordagem.

Para que o objetivo fosse alcançado, foram pesquisados os modelos de dados do formato JSON e que tivessem contexto semântico. Em seguida, foram definidos os modelos de dados que seriam avaliados e identificadas as suas principais características. Também foram definidos o ambiente de avaliação e os formatos de dados que seriam avaliados e, por fim, realizado o experimento e analisado seus resultados. A partir dessas definições, pôde-se chegar aos seguintes resultados:

- O RDF contribui para a Web atual por meio de um padrão de troca de mensagens com descrição de informações relacionadas ao contexto;
- O uso de vínculo de dados possibilita uma melhor compreensão dos dados quando se incorpora múltiplos dados com a mesma estrutura;
- O JSON é uma tecnologia que permite em sua estrutura definir diversos modelos e esquemas de documentos Web.

Conclui-se, portanto, que embora este trabalho tenha se limitado ao pouco conteúdo na literatura sobre o tema proposto, foram abordados os aspectos conceituais disponíveis nas documentações das tecnologias apresentadas. Outra limitação se dá no ambiente da realização do teste que foi controlado e também na velocidade de conexão com a internet que foi um fator não controlado, por estar em um *Link* de internet compartilhado, variando entre 26 à 29 Megabits por segundo. No entanto, é possível identificar resultados que podem orientar futuras pesquisas que abordem este tema.

Além disso, percebe-se que este trabalho aborda um tema que pode ser utilizado por se tratar de tecnologias que possibilitam o reconhecimento de conteúdo e seus significados, principalmente no que se trata de mecanismos de buscas. É

relevante a necessidade do compartilhamento de conhecimento e informação dos modelos que definem os dados, pois quanto maior as contribuições à tecnologia, mais precisa, rápida, eficiente e uniforme ela se torna.

Como perspectiva para trabalhos futuros, sugere-se a realização de testes diversificados com os modelos apresentados e a realização de teste em múltiplos ambientes com diferentes condições de trabalho, analisando os resultados e identificando possíveis melhorias para a tecnologia JSON-LD, ou ainda, a criação de tecnologias para suprir necessidades que possam ser encontradas.

Referências

- [Berners-Lee 2017] Berners-Lee, T. (2017). Biography of the web creator. [https:// www.w3.org/People/Berners-Lee/](https://www.w3.org/People/Berners-Lee/).
- [Cisco 2017] Cisco (2017). Cisco visual networking index Global mobile data traffic fore- cast update. 11:35.
- [Group 2014a] Group, J.-L. C. (2014a). Building json-ld apis: Best practices. <https://json-ld.org/spec/latest/json-ld-api-best-practices/>.
- [Group 2014b] Group, J.-L. C. (2014b). Json-ld – json for linking data. [https:// json-ld.org/](https://json-ld.org/).
- [Group 2008a] Group, W. C. (2008a). Extensible markup language (xml). <https://www.w3.org/XML/>.
- [Group 2008b] Group, W. C. (2008b). Semantic web - w3c. <https://www.w3.org/standards/semanticweb/>.
- [Guha et al. 2016] Guha, R. V., Brickley, D., and Macbeth, S. (2016). Schema. org: Evolution of structured data on the web. *Communications of the ACM*, 59(2):44–51.
- [Guiraud and Mascarenhas 1972] Guiraud, P. and Mascarenhas, M. E. (1972). *A semântica*. PhD thesis, Difusão Européia do Livro.
- [Hutchison 1987] Hutchison, G., D. C. L. (1987). Electronic data interchange. *Browse Journals & Magazines*, 1(4):16–20.
- [International 2011] International, E. (2011). Ecmascript language specification edition 5.1.
- [Souza and Alvarenga 2004] Souza, R. R. and Alvarenga, L. (2004). A web semântica e suas contribuições para a ciência da informação. *Ciência da Informação*, 33(1).

[Speicher et al. 2015] Speicher, S., Arwe, J., and Malhotra, A. (2015). Linked data platform

1.0. W3C Recommendation, February, 26.

[T. Bray 2013] T. Bray, E. (2013). Rfc 7159 - the javascript object notation (json) data interchange format. 3:16.

[Turban et al. 2005] Turban, E., Rainer, R. K., and Potter, R. E. (2005). Administração de tecnologia da informação: teoria e prática. Elsevier.