UNIVERSIDADE DE BRASÍLIA

Faculdade do Gama

Sistemas de Banco de Dados 2

Tecnologias de Banco de Dados (TI-BD)

Bancos de Dados Semiestruturados e XML

Pedro Rodrigues Pereira, 17/0062686

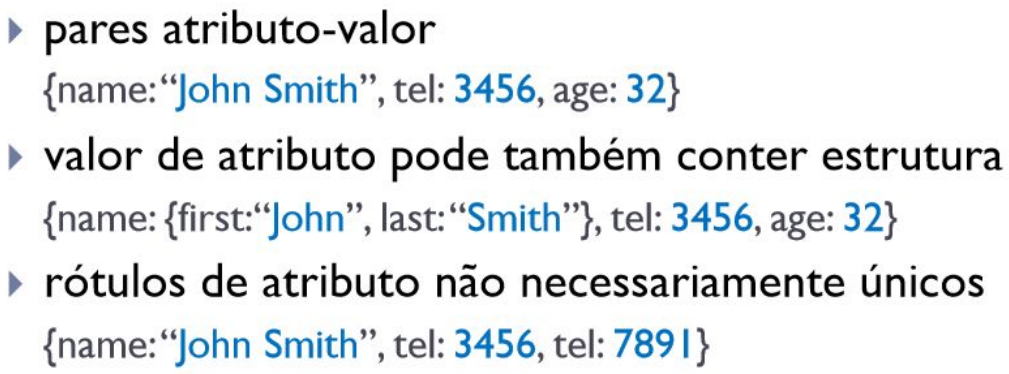
Brasília, DF

2019

Definição

Para a definição de dados semi-estruturados é necessário primeiro definir o que são dados estruturados, em sistemas de SGBDs(Sistema Gerenciador de Banco de Dados) comuns os dados possuem uma estrutura previsível, ou seja um esquema previamente definido, e as operações que são realizadas em cima destes dados seguem esse padrão. Em posse desse conceito é possível estabelecer que dados semi-estruturados são aqueles os quais não seguem um padrão bem definido ou que não possuem uma generalização desse padrão, como por exemplo um dado Web que pode variar desde de um texto simples sem divisões, até um documento extenso com vários títulos e subtítulos.

Um dado semi estruturado conta com uma abordagem de não obrigatoriedade de organização, suprindo a necessidade de alocar dados os quais não é possível prever seu esquema e/ou estrutura. Ou seja, um dado semi-estrutura, torna mais flexível a abordagem de organização deles. Na imagem a seguir vemos um exemplo de dado semi-estruturado, ou como a estrutura de um mesmo dado pode variar:



Ambos anseiam representar o mesmo dado e a mesma informação, mas são descritos de formas diferentes, ou seja mudando sua estrutura de acordo com a abordagem escolhida.

A característica de organizar dados com vários padrões torna a busca por esses dados muito complexa, já que não existe uma estrutura padrão a qual a

busca seria orientada. A busca por palavra-chave, uma das mais empregadas, não se faz eficiente nesse contexto e geraria um gargalo gigante, nisso é visto que as formas padrões de organizar dados estruturados não cobre uma forma eficiente de tratar dados semi-estruturados.

A estrutura complexa dos dados semi-estruturados prevê uma série de características que o tornam difuso dos dados comum, mas seguem um padrão de nunca serem complementamente sem estrutura e nem altamente estruturados, sempre ocupando o espectro mediano. Em uma comparação entre Dados estruturados e semiestruturados, os tópicos abordados sempre serão em relação a estrutura dos mesmos e a consequência que isso gera para a manipulação deles. Na tabela a seguir é possível enxergar essas diferenças:

Dados tradicionais	Dados semi-estruturados
Esquema predefinido	Nem sempre há um esquema predefinido
Estrutura regular	Estrutura irregular
Estrutura independente dos dados	Estrutura embutida no dado
Estrutura reduzida	Estrutura extensa
Estrutura fracamente evolutiva	Estrutura fortemente evolutiva
Estrutura prescritiva	Estrutura descritiva
Distinção entre estrutura e dado é clara	Distinção entre estrutura e dado não é clara

Em decorrência disso, a forma de organização o qual os dados semi estruturados são descritos é o XML(Extensible Markup Language), que se trata de uma linguagem de marcação para a web. o XML traz características que permitem a representação de dados semi-estruturados, já que em sua essência ele não depender de uma estrutura já definida, sendo flexível para que o esquema de um dado se adapte a necessidade. Em um sistema Web por exemplo é possível estabelecer o HTML como aquele de descreve a apresentação e o XML aquele que se responsabiliza pelo conteúdo, assim sendo um blog de notícias, teria em seu HTML, o corpo da notícia, enquanto o xml possuiria o autor, o título, o publicador e todas as coisas que diferem aquela notícia de qualquer outra.

O XML segue um padrão simples, de abrir e fechar “tags”, partindo da raiz do documento e realizando o processo de aninhamento, em que tudo que é aberto fecha anterior a aquilo que já estava aberto. Exemplo de dois funcionários

descritos em XML, é possível notar que não há uma estrutura sendo mantida, a forma como cada funcionário foi descrito é diferente:

```
<? xml version="1.0" ?>
<empregados>
  <empregado cod="E01" dept="D01">
    <nome>João</nome>
    <inicial-meio>S.</inicial-meio>
    <sobrenome>Santos</sobrenome>
  </empregado>
 <empregado cod="E02" dept="D01">
    <nome>Ana</nome>
    <sobrenome>Ferraz</sobrenome>
  </empregado>
</empregados>
```

Outro ponto importante para se levar em consideração em XML, é a não relevância da ordem de descrição dos dados, os atributos não são ordenados. No passo que o que define o atributo é a chave que o acompanha na tag corresponde a ele. No exemplo da imagem anterior não importa a ordem em que os funcionários foram cadastrados, geraria o mesmo resultado.

Exemplo utilizando a imagem anterior na Linha 3:

<empregado cod=”E01” dept=”D01”>

<empregado dept=”D01” cod=”E01”>, as duas formas resultam no mesmo.

Objetivos

Os dados semi estruturados visam a representação de dados mistos, ou de estrutura variável, onde não é possível prever o esquema em que ele se enquadra. Num contexto atual a maior contribuição dos dados semi estruturados está na Web, onde a sua forma de trabalhar torna possível a indexação desses dados e o uso posterior deles entre outras aplicações.

Os bancos de dados semi estruturados tem grande relevância em contextos web, já que dados web se enquadram num contexto de estrutura maleável ou heterogênea. Modelos de dados para Bds tradicionais não cobrem essa característica, logo se faz necessário o uso dos bancos de dados semi estruturados através de grafos direcionados rotulados.

Vantagens

- Estrutura variável, não exige uma estrutura padrão dos dados, uma vantagem perante o banco relacional que se limita a uma estrutura já definida pelo projetista do banco.
- Estrutura evolucionária, num banco de dados semi estruturados os valores podem ser mudados com frequência, diferente do banco relacional que devido ao seu sistema mais rígido de relações tende a tornar o processo mais trabalho e a base menos evolucionária.
- Definição à posteriori, a estrutura é extraída após os dados, o que torna mais simples a definição do banco, sem exigir um projeto de extenso da estrutura do banco como o banco relacional.

Desvantagens

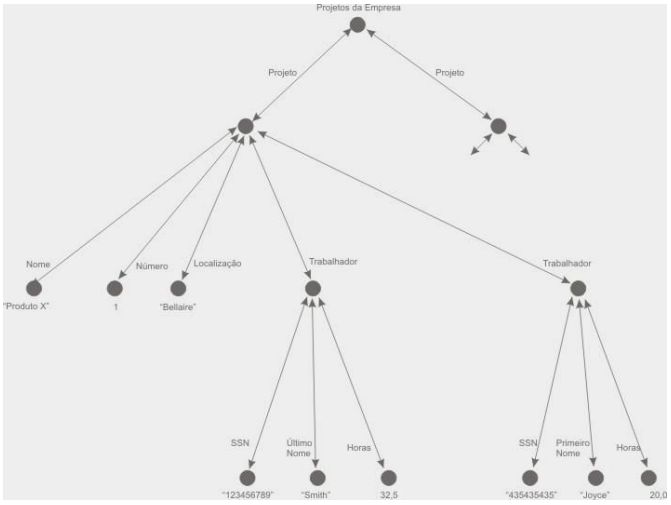
- Graças a característica que também o destaca, a qual é a flexibilidade de dados, ele sofre uma ressalva, a qual é a dificuldade que isso gera no retorno dos dados. A pesquisa em uma estrutura heterogênea é custosa e frente a um volume de dados grande se torna quase impossível, diante disso, o XML prevê uma solução e uma forma de busca dos dados, o XML-QL.

  As pesquisas utilizando XML-QL podem extrair os dados de um arquivo XML e retornar um novo arquivo XML com o que foi encontrado perante a busca. Assim como sql retorna uma nova tabela com o resultado, o XML-QL retorna um novo arquivo com os dados manipulados. Exemplo de pesquisa básica utilizando XML-QL:

```
CONSTRUCT <result> {
  WHERE<artigo>
            <nomea>Heuser</>
            <título>$t</>
            <resumo>$r</>
          </> IN "www.a.b.c/bib.xml"
  CONSTRUCT <artigo>
              <título>$t</>
              <resumo>$r</>
              </> }
```

Em um contexto de bancos relacionais, a pesquisa devido à estrutura heterogênea é prejudicada quando comparado ao banco relacional, onde a estrutura dos dados é previsível.

- Distinção entre estrutura e dados não é clara, num banco de dados semi estruturados, a distinção entre os dados e a estrutura não se faz clara, uma vez que um dado pode possuir um novo nó de atributos a partir dele.



Essa característica se difere do Banco de dados relacional, onde a estrutura é previsível e evita o processamento de recursos para tratamento dessa nova estrutura.

- A estrutura em dados semi estruturados está embutida nos dados, enquanto nos sistemas relacionais tradicionais a estrutura possui independência dos dados, o que gera uma visualização mais fácil.

Aplicações

- Sedna: O sedna se trata de um SGBD XML nativo open source, o mesmo utiliza a linguagem XQUERY para consulta e atualização dos dados. Como uma aplicação open source, o código do mesmo pode ser encontrado no github: https://github.com/sedna/sedna. Esse SGBD é gratuito.

- eXist XLM: eXist XLM se trata de um SGBD que faz uso da sintaxe XPath, o SGBD é open source e o código do mesmo pode ser encontrado no github:https://github.com/eXist-db/exist. Esse SGBD é gratuito.

Bibliografia

CAVALCANTI, Daniel. Dados Estruturados e Semiestruturados. Acesso em 09 de setembro de 2019. Disponível em: https://danielcavalcanti.com.br/home/dados-estruturados-e-semiestruturados/

MELLO, Ronaldo dos Santos. Dados Semi-Estruturados .Instituto de Informática (II) Universidade Federal do Rio Grande do Sul (UFRGS).

WALMSLEY, Priscilla. XQuery: Search Across a Variety of XML Data (English Edition).

BRAGANHOLO, Vanessa.Curso baseado em mini-cursos apresentados no SBBD. Autores: Carlos Heuser, Carina Dorneles e Vanessa Braganholo.(UFF).