

MANUAL DE USUARIO NORMALIZADOR EN TERCERA FORMA NORMAL

FABIO JULIAN BERNAL
LUIS GUILLERMO CORREA
CRISTHIAN LEONARDO TOLOSA

GRUPO 2

UNIVERSIDAD DISTRITAL FRANCISCO JOSÉ DE CALDAS
ESPECIALIZACIÓN EN INGENIERÍA DE SOFTWARE
FACULTAD DE INGENIERÍA
BASES DE DATOS
BOGOTÁ
2014

Tabla de contenido

1. INTRODUCCION.....3

2. OBJETIVO.....3

3. NORMALIZADOR EN TERCERA FORMA NORMAL BERNSTEIN.....3

4. REQUERIMIENTOS TÉCNICOS.....3

5. ARCHIVOS DE ENTRADA.....3

6. EJECUTAR APLICACIÓN.....7

1. INTRODUCCION

Este documento pretende ser una guía de interacción entre el usuario y el sistema normalizador en tercera forma normal.

2. OBJETIVO

Especificar qué módulos y funcionalidades cubre y cuáles son las características de los usuarios que interactúan con el software. Explicar brevemente la estrategia de navegación y cubrimiento en este manual.

3. NORMALIZADOR EN TERCERA FORMA NORMAL BERNSTEIN

El software es una implementación del algoritmo de Bernstein para el síntesis de dependencias funcionales permitiendo normalizar en tercera forma normal un conjunto de relaciones a partir de un conjunto de dependencias funcionales.

4. REQUERIMIENTOS TÉCNICOS

Para poder ejecutar el software es necesario contar con Java Runtime Environment 1.6 o superior.

5. ARCHIVOS DE ENTRADA

El normalizador en tercera forma normal utiliza archivos en formato XML para la entrada de datos. A continuación se describe el esquema XSD y un archivo de ejemplo XML.

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
```

```

xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="root">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="attributes">
          <xs:complexType>
            <xs:sequence>
              <xs:element type="xs:string" name="attribute" maxOccurs="unbounded"
minOccurs="1"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="fds">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="fd" maxOccurs="unbounded" minOccurs="1">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="left">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element type="xs:string" name="attribute"
maxOccurs="unbounded" minOccurs="1"/>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                    <xs:element name="right">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element type="xs:string" name="attribute"
maxOccurs="unbounded" minOccurs="1"/>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

Tabla 1. Esquema XSD archivo de entrada.

```

<root>
  <attributes>
    <attribute>A</attribute>
    <attribute>B</attribute>
    <attribute>C</attribute>
    <attribute>D</attribute>
    <attribute>E</attribute>
    <attribute>F</attribute>

```

```

    <attribute>G</attribute>
    <attribute>I</attribute>
    <attribute>J</attribute>
    <attribute>K</attribute>
    <attribute>L</attribute>
    <attribute>M</attribute>
    <attribute>N</attribute>
    <attribute>P</attribute>
    <attribute>Q</attribute>
    <attribute>R</attribute>
    <attribute>S</attribute>
    <attribute>T</attribute>
</attributes>
<fds>
  <fd>
    <left>
      <attribute>A</attribute>
      <attribute>B</attribute>
    </left>
    <right>
      <attribute>C</attribute>
    </right>
  </fd>
  <fd>
    <left>
      <attribute>A</attribute>
    </left>
    <right>
      <attribute>D</attribute>
      <attribute>E</attribute>
      <attribute>F</attribute>
      <attribute>G</attribute>
    </right>
  </fd>
  <fd>
    <left>
      <attribute>E</attribute>
    </left>
    <right>
      <attribute>G</attribute>
    </right>
  </fd>
  <fd>
    <left>
      <attribute>F</attribute>
    </left>
    <right>
      <attribute>D</attribute>
      <attribute>J</attribute>
    </right>
  </fd>
  <fd>
    <left>
      <attribute>G</attribute>
    </left>
    <right>
      <attribute>D</attribute>

```

```

        <attribute>I</attribute>
    </right>
</fd>
<fd>
    <left>
        <attribute>D</attribute>
        <attribute>M</attribute>
    </left>
    <right>
        <attribute>N</attribute>
        <attribute>P</attribute>
    </right>
</fd>
<fd>
    <left>
        <attribute>D</attribute>
    </left>
    <right>
        <attribute>M</attribute>
    </right>
</fd>
<fd>
    <left>
        <attribute>L</attribute>
    </left>
    <right>
        <attribute>D</attribute>
    </right>
</fd>
<fd>
    <left>
        <attribute>P</attribute>
        <attribute>R</attribute>
    </left>
    <right>
        <attribute>S</attribute>
    </right>
</fd>
<fd>
    <left>
        <attribute>P</attribute>
        <attribute>Q</attribute>
        <attribute>R</attribute>
    </left>
    <right>
        <attribute>S</attribute>
        <attribute>T</attribute>
    </right>
</fd>
<fd>
    <left>
        <attribute>D</attribute>
    </left>
    <right>
        <attribute>K</attribute>
        <attribute>L</attribute>
    </right>

```

```
</fd>
</fds>
</root>
```

Tabla 2. Archivo XML de entrada de ejemplo.

6. EJECUTAR APLICACIÓN

Los ejemplos aquí mostrados se basan en una máquina con sistema operativo Windows pero puede ser fácilmente extrapolado a cualquier sistema operativo con JRE instalado.

Abra una consola de comandos, cambie el directorio a la carpeta **dist** de esta solución.

```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) 2012 Microsoft Corporation. Todos los derechos reservados.
PS C:\Windows\system32> cd C:\Users\guillermo\Documents\ESPECIALIZACION\BD1\NormalizadorBernstein\dist
PS C:\Users\guillermo\Documents\ESPECIALIZACION\BD1\NormalizadorBernstein\dist> _
```

Lanze el siguiente comando para ejecutar la aplicación:

```
java -jar normalizador rutaArchivoXmlEntrada.xml
```

Reemplace rutaArchivoXmlEntrada.xml con la ruta relativa o absoluta de su archivo xml de entrada.

```
\dist> java -jar normalizador.jar universal.xml
```

La aplicación implementa paso a paso el algoritmo de Bernstein para realizar la normalización en tercera forma normal, como se muestra a continuación.

```
Carga de archivo con el dominio, conjunto de atributos y conjunto de dependencias funcionales
A: <A, B, C, D, E, F, G, I, J, K, L, M, N, P, Q, R, S, T>
FD: <AB->C, A->DEFG, E->G, F->DJ, G->DI, DM->NP, D->M, L->D, PR->S, PQR->ST, D->KL>

Paso 1: Calcular dependencias elementales
FD: <AB->C, E->G, D->M, L->D, PR->S, A->D, A->E, A->F, A->G, F->D, F->J, G->D, G->I, DM->N, DM->P, PQR->S, PQR->T, D->K, D->L>

Paso 2: Eliminar atributos extraños
FD: <AB->C, E->G, D->M, L->D, PR->S, A->D, A->E, A->F, A->G, F->D, F->J, G->D, G->I, D->N, D->P, PR->S, PQR->T, D->K, D->L>

Paso 3: Eliminar dependencias innecesarias
FD: <AB->C, E->G, D->M, L->D, A->E, A->F, F->D, F->J, G->D, G->I, D->N, D->P, PR->S, PQR->T, D->K, D->L>

Paso 4: Agrupar dependencias por llave
FD: <AB->C, E->G, D->MNPKL, L->D, A->EF, F->DJ, G->DI, PR->S, PQR->T>
PS C:\Users\guillermo\Documents\ESPECIALIZACION\BD1\NormalizadorBernstein\dist> _
```