

CASO DE APLICACIÓN ALGORITMO MEJORADO DE BERNSTEIN PARA LA
NORMALIZACIÓN 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 Contenidos

1. INTRODUCCIÓN	3
2. CASO DE ESTUDIO.....	3
3. ESPECIFICACIÓN ESTRUCTURAS DE INFORMACIÓN	4
3.1 Cliente	4
3.2 Contacto	5
3.3 Vinculación.....	6
3.4 Dato Adicional.....	8
3.5 Estado Civil	8
3.6 Ciudad	10
3.7 Departamento	10
3.8 Programa de Vinculación	11
4. CONJUNTO DE ATRIBUTOS Y DEPENDENCIAS FUNCIONALES	11
5. NORMALIZACIÓN EN TERCERA FORMA NORMAL MEDIANTE EL ALGORITMO MEJORADO DE BERNSTEIN.....	14
6. CONCLUSIONES	14

1. INTRODUCCIÓN

Este documento presenta un caso de estudio (real) sobre la aplicación de síntesis de dependencias funcionales en tercera forma normal, a continuación se presentará el caso de estudio, su dominio, su universo de atributos y de dependencias funcionales.

2. CASO DE ESTUDIO

Una organización pretende implementar una solución tecnológica para la administración de clientes, parametrizable y en un modelo multiempresa, permitiendo la adaptación de nuevos negocios, que incluye la creación, modificación de clientes, vinculaciones y novedades individuales y masivas, garantizando la calidad del dato y su trazabilidad, en un esquema multicanal.

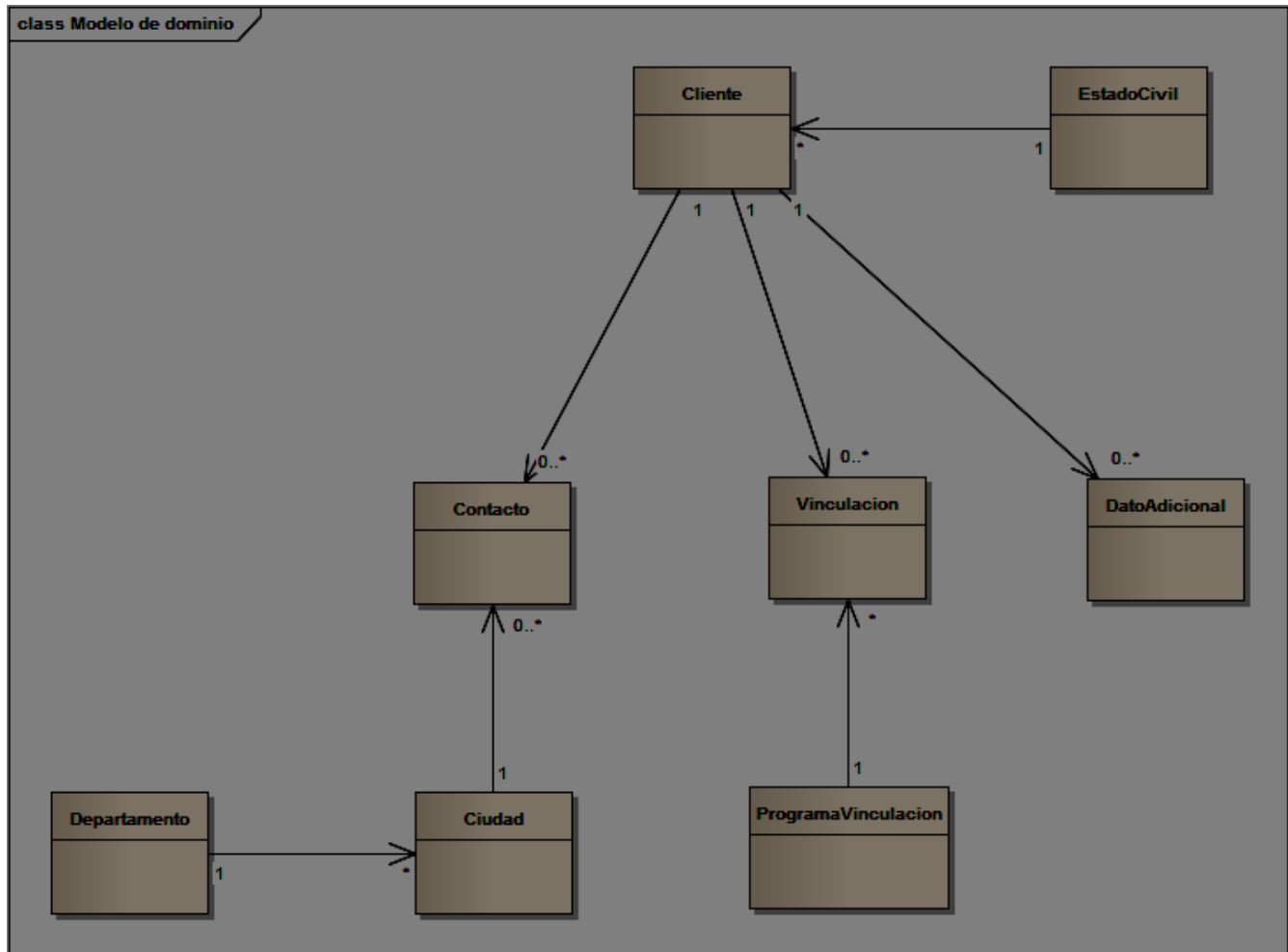


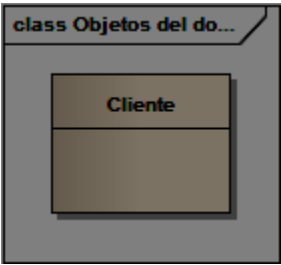
Figura 1. Modelo conceptual.

3. ESPECIFICACIÓN ESTRUCTURAS DE INFORMACIÓN

A continuación se detallan las estructuras y los campos que se encuentran dentro del modelo conceptual

detallado de información.

3.1 Cliente



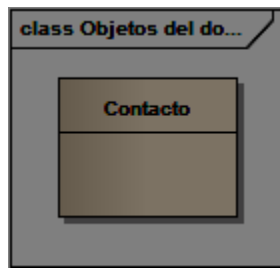
Estructura que contiene la información básica del cliente. Esta estructura contempla los datos básicos para personas y para empresas.

Es considerado como datos básicos, aquella información que tiene la mayoría de los clientes.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción
PK	IdCliente	Numérico	Si	Si	Numero de autenticación del cliente. La creación de este número es definida por negocio.
	Tipo identificación	Numérico	Si	No	Tipo Identificación
	Numero identificación	Alfanumérico	Si	No	Número de identificación
	Estado civil	Numérico	Si	No	Estado civil
	Primer nombre	Alfanumérico	Si	No	Nombre del cliente
	Segundo nombre	Alfanumérico	No	No	Nombre del cliente o razón social
	Primer apellido	Alfanumérico	Si	No	Primer apellido
	Segundo apellido	Alfanumérico	Si	No	Segundo apellido
	Fecha nacimiento	Fecha	Si	No	Fecha de nacimiento
	Genero	Alfanumérico	Si	No	Género
	Estado	Bool	Si	No	Estado de Activación

3.2 Contacto

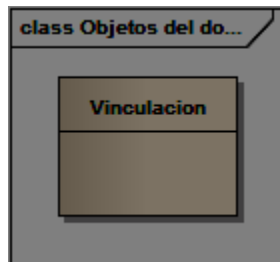


Esta estructura permite almacenar los datos de contacto del cliente. Los datos de contacto es la información que se le solicita al cliente para poder establecer un contacto posterior con él.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción
PK	Id datos contacto	Númérico	Si	Si	Identificador numérico autoincremental de la estructura
	Direccion	Alfanumérico	Si	No	Dirección o de contacto
	Barrio	Alfanumérico	No	No	Barrio donde se encuentra ubicada la dirección
	Código zona	Númérico	No	No	Código de Zona
FK	Id ciudad	Númérico	Si	No	Llave foránea con Ciudad
	Área direccion	Númérico	Si	No	Código de Área
	Telefono	Númérico	No	No	Teléfono
	Extensión	Númérico	No	No	Extensión asociada al teléfono
	Celular	Númérico	No	No	Número de teléfono móvil
	Correo electronico	Alfanumérico	Si	No	Correo electrónico
FK	IdCliente	Númérico	Si	Si	Llave foránea con Cliente

3.3 Vinculación

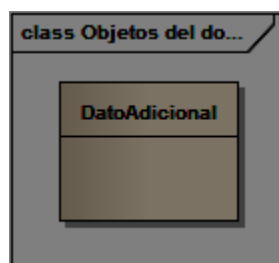


Esta estructura almacena la vinculación que tiene un cliente a través de un producto de negocio.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción
PK	Id vinculación	Numérico	Si	Si	Identificador numérico autoincremental de la estructura
FK	Id programa	Numérico	No	No	Llave foránea con la estructura Programa de Vinculación
FK	Id cliente	Numérico	Si	Si	Llave foránea con Cliente
	Fecha ingreso empresa	Fecha	Si	No	Fecha en que ingreso el cliente a la empresa
	Fecha vinculación	Fecha	Si	No	Fecha de la vinculación
	Fecha inicio vigencia	Fecha	Si	No	Fecha inicio de vigencia
	Fecha fin vigencia	Fecha	Si	No	Fecha final de la vigencia
	Estado vinculación	Numérico	Si	No	Llave foránea con estado de vinculación
	Fecha retiro	Fecha	Si	No	Fecha de retiro
	Causal retiro	Numérico	Si	No	Causal de Retiro
	Salario	Numérico	Si	No	Salario del Cliente, el cual sirve para determinar el estrato para la vinculación
	Estrato	Numérico	Si	No	Valor que se define por la cantidad de salarios mínimos legales mensuales vigentes SMLMV que representa el salario del cliente

3.4 Dato Adicional

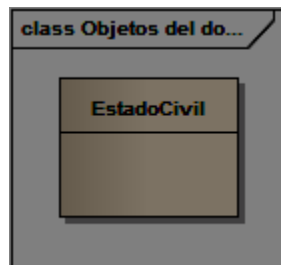


Esta estructura almacena datos adicionales de clientes.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción
PK	Id dato adicional	Númerico	Si	Si	Identificador numérico autoincremental de la estructura
FK	Id cliente	Númerico	Si	No	Llave foránea con la estructura Cliente
	Valor	Alfanumérico	Si	No	Valor del Requisito para el Cliente
	Fecha inicio vigencia	Númerico	Si	No	Fecha inicial de vigencia
	Fecha fin vigencia	Númerico	Si	No	Fecha de final de vigencia

3.5 Estado Civil

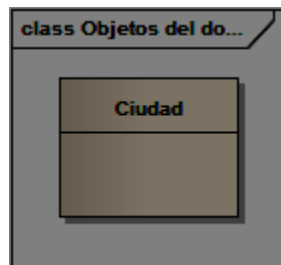


Estructura que almacena el listado de los posibles estados civiles de un Cliente.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción	Observaciones
PK	Id estado civil	Númerico	Si	Si	Identificador numérico autoincremental de la estructura	
	Descripción	Alfanumérico	Si	No	Se almacenan los estados civiles	

3.6 Ciudad

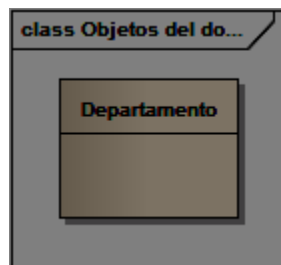


Estructura que almacena el listado de ciudades.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción
PK	Id ciudad	Numérico	Si	Si	Número dado por el DANE
	Descripción	Alfanumérico	Si	No	Ciudad
FK	Id departamento	Numérico	Si	No	Llave foránea a la estructura Departamento

3.7 Departamento

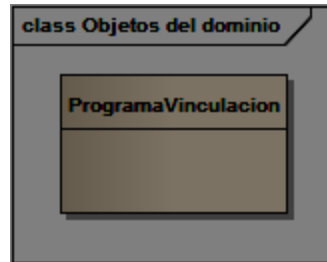


Estructura que almacena el listado de departamentos.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción
PK	Id departamento	Numérico	Si	Si	Número dado por el DANE
	Descripción	Alfanumérico	Si	No	Departamento

3.8 Programa de Vinculación



Estructura que almacena los tipos de vinculación, los tipos de vinculación se determinan de acuerdo a si el Cliente ya ha tenido o no una relación con el negocio para el plan por el cual se vincula.

Los campos definidos para esta estructura son:

Tipo de Llave	Nombre atributo propuesto	Tipo de dato	Dato Obligatorio	Dato Único	Descripción
PK	Id tipo vinculación	Numérico	Si	Si	Identificador numérico autoincremental de la estructura
	Descripción	Alfanumérico	Si	No	Se almacenan los tipos de clientes

4. CONJUNTO DE ATRIBUTOS Y DEPENDENCIAS FUNCIONALES

El siguiente archivo contiene el universo de atributos definidos y las dependencias funcionales, este archivo es el insumo de la aplicación de normalización en tercera que utiliza una versión mejorada del algoritmo de Bernstein.

```
<root>
  <attributes>
    <!-- Cliente -->
    <attribute>idCliente</attribute>
    <attribute>tipoIdentificacion</attribute>
    <attribute>numeroIdentificacion</attribute>
    <attribute>estadoCivil</attribute>
    <attribute>primerNombre</attribute>
    <attribute>segundoNombre</attribute>
    <attribute>primerApellido</attribute>
    <attribute>segundoApellido</attribute>
    <attribute>fechaNacimiento</attribute>
    <attribute>estadoCliente</attribute>
    <!-- Datos contacto -->
    <attribute>idContacto</attribute>
    <attribute>direccion</attribute>
    <attribute>barrio</attribute>
    <attribute>codigoZona</attribute>
    <attribute>ciudadID</attribute>
```

```

        <attribute>area</attribute>
        <attribute>telefono</attribute>
        <attribute>extension</attribute>
        <attribute>celular</attribute>
        <attribute>email</attribute>
        <!-- Vinculacion -->
        <attribute>idVinculacion</attribute>
        <attribute>fechaIngresoEmpresa</attribute>
        <attribute>fechaVinculacion</attribute>
        <attribute>fechaInicioVigenciaVinculacion</attribute>
        <attribute>estadoVinculacion</attribute>
        <attribute>fechaRetiroVinculacion</attribute>
        <attribute>causalRetiroVinculacion</attribute>
        <attribute>salario</attribute>
        <attribute>estrato</attribute>
        <!-- Dato adicional -->
        <attribute>idDatoAdicional</attribute>
        <attribute>valor</attribute>
        <attribute>fechaInicioVigenciaDatoAdicional</attribute>
        <attribute>fechaFinVigenciaDatoAdicional</attribute>
        <!-- Estado Civil -->
        <attribute>idEstadoCivil</attribute>
        <attribute>descripcionEstadoCivil</attribute>
        <!-- Ciudad -->
        <attribute>idCiudad</attribute>
        <attribute>descripcionCiudad</attribute>
        <!-- Departamento -->
        <attribute>idDepartamento</attribute>
        <attribute>descripcionDepartamento</attribute>
        <!-- Programa Vinculacion -->
        <attribute>idProgramaVinculacion</attribute>
        <attribute>decripcionVinculacion</attribute>
    </attributes>
</fds>

    <!-- Cliente -->
    <fd>
        <left>
            <attribute>idCliente</attribute>

        </left>
        <right>
            <attribute>tipoIdentificacion</attribute>
            <attribute>numeroIdentificacion</attribute>
            <attribute>estadoCivil</attribute>
            <attribute>primerNombre</attribute>
            <attribute>segundoNombre</attribute>
            <attribute>primerApellido</attribute>
            <attribute>segundoApellido</attribute>
            <attribute>fechaNacimiento</attribute>
            <attribute>estadoCliente</attribute>
        </right>
    </fd>
    <!-- Contacto -->
    <fd>
        <left>
            <attribute>idContacto</attribute>
        </left>

```

```

        <right>
            <attribute>direccion</attribute>
            <attribute>barrio</attribute>
            <attribute>codigoZona</attribute>
            <attribute>ciudadID</attribute>
            <attribute>area</attribute>
            <attribute>telefono</attribute>
            <attribute>extension</attribute>
            <attribute>celular</attribute>
            <attribute>email</attribute>
        </right>
    </fd>
    <!-- Vinculacion -->
    <fd>
        <left>
            <attribute>idVinculacion</attribute>
        </left>
        <right>
            <attribute>fechaIngresoEmpresa</attribute>
            <attribute>fechaVinculacion</attribute>
            <attribute>fechaInicioVigenciaVinculacion</attribute>
            <attribute>estadoVinculacion</attribute>
            <attribute>fechaRetiroVinculacion</attribute>
            <attribute>causalRetiroVinculacion</attribute>
            <attribute>salario</attribute>
            <attribute>estrato</attribute>
        </right>
    </fd>
    <!-- Dato adicional -->
    <fd>
        <left>
            <attribute>idDatoAdicional</attribute>
        </left>
        <right>
            <attribute>valor</attribute>
            <attribute>fechaInicioVigenciaDatoAdicional</attribute>
            <attribute>fechaFinVigenciaDatoAdicional</attribute>
        </right>
    </fd>
    <!-- Estado civil -->
    <fd>
        <left>
            <attribute>idEstadoCivil</attribute>
        </left>
        <right>
            <attribute>descripcionEstadoCivil</attribute>
        </right>
    </fd>
    <!-- Ciudad -->
    <fd>
        <left>
            <attribute>idCiudad</attribute>
            <attribute>descripcionCiudad</attribute>
        </left>
        <right>
            <attribute>idDepartamento</attribute>
            <attribute>descripcionDepartamento</attribute>
        </right>
    </fd>

```

```

        </right>
    </fd>
</fds>
</root>

```

5. NORMALIZACIÓN EN TERCERA FORMA NORMAL MEDIANTE EL ALGORITMO MEJORADO DE BERNSTEIN

A continuación se muestra el resultado de la ejecución del algoritmo mejorado de Bernstein mediante la aplicación desarrollada.

⑩ Cálculo de dependencias necesarias

FD: {idEstadoCivil-->descripcionEstadoCivil, idProgramaVinculacion-->decripcionVinculacion, idCliente-->tipoIdentificacion, idCliente-->numeroIdentificacion, idCliente-->estadoCivil, idCliente-->primerNombre, idCliente-->segundoNombre, idCliente-->primerApellido, idCliente-->segundoApellido, idCliente-->fechaNacimiento, idCliente-->estadoCliente, idContacto-->direccion, idContacto-->barrio, idContacto-->codigoZona, idContacto-->ciudadID, idContacto-->area, idContacto-->telefono, idContacto-->extension, idContacto-->celular, idContacto-->email, idVinculacion-->fechaIngresoEmpresa, idVinculacion-->fechaVinculacion, idVinculacion-->fechaInicioVigenciaVinculacion, idVinculacion-->estadoVinculacion, idVinculacion-->fechaRetiroVinculacion, idVinculacion-->causalRetiroVinculacion, idVinculacion-->salario, idVinculacion-->estrato, idDatoAdicional-->valor, idDatoAdicional-->fechaInicioVigenciaDatoAdicional, idDatoAdicional-->fechaFinVigenciaDatoAdicional, idCiudad-->descripcionCiudad-->idDepartamento, idCiudad-->descripcionCiudad-->descripcionDepartamento}

⑩ Eliminación de elementos extraños

FD: {idEstadoCivil-->descripcionEstadoCivil, idProgramaVinculacion-->decripcionVinculacion, idCliente-->tipoIdentificacion, idCliente-->numeroIdentificacion, idCliente-->estadoCivil, idCliente-->primerNombre, idCliente-->segundoNombre, idCliente-->primerApellido, idCliente-->segundoApellido, idCliente-->fechaNacimiento, idCliente-->estadoCliente, idContacto-->direccion, idContacto-->barrio, idContacto-->codigoZona, idContacto-->ciudadID, idContacto-->area, idContacto-->telefono, idContacto-->extension, idContacto-->celular, idContacto-->email, idVinculacion-->fechaIngresoEmpresa, idVinculacion-->fechaVinculacion, idVinculacion-->fechaInicioVigenciaVinculacion, idVinculacion-->estadoVinculacion, idVinculacion-->fechaRetiroVinculacion, idVinculacion-->causalRetiroVinculacion, idVinculacion-->salario, idVinculacion-->estrato, idDatoAdicional-->valor, idDatoAdicional-->fechaInicioVigenciaDatoAdicional, idDatoAdicional-->fechaFinVigenciaDatoAdicional, idCiudad-->descripcionCiudad-->idDepartamento, idCiudad-->descripcionCiudad-->descripcionDepartamento}

⑩ Eliminación de redundancias

FD: {idEstadoCivil-->descripcionEstadoCivil, idProgramaVinculacion-->decripcionVinculacion, idCliente-->tipoIdentificacion, idCliente-->numeroIdentificacion, idCliente-->estadoCivil, idCliente-->primerNombre, idCliente-->segundoNombre, idCliente-->primerApellido, idCliente-->segundoApellido, idCliente-->fechaNacimiento, idCliente-->estadoCliente, idContacto-->direccion, idContacto-->barrio, idContacto-->codigoZona, idContacto-->ciudadID, idContacto-->area, idContacto-->telefono, idContacto-->extension, idContacto-->celular, idContacto-->email, idVinculacion-->fechaIngresoEmpresa, idVinculacion-->fechaVinculacion, idVinculacion-->fechaInicioVigenciaVinculacion, idVinculacion-->estadoVinculacion, idVinculacion-->fechaRetiroVinculacion, idVinculacion-->causalRetiroVinculacion, idVinculacion-->salario, idVinculacion-->estrato, idDatoAdicional-->valor, idDatoAdicional-->fechaInicioVigenciaDatoAdicional, idDatoAdicional-->fechaFinVigenciaDatoAdicional, idCiudad-->descripcionCiudad-->idDepartamento, idCiudad-->descripcionCiudad-->descripcionDepartamento}

⑩ Agrupamiento por llaves

FD: {idEstadoCivil-->descripcionEstadoCivil, idProgramaVinculacion-->decripcionVinculacion, idCliente-->tipoIdentificacionnumeroIdentificacionestadoCivilprimerNombresegundoNombreprimerApellidosegundoApellidofechaNacimientoestadoCliente, idContacto-->direccionbarriocodigoZonaciudadIDareatelefonoextensioncelularemail, idVinculacion-->fechaIngresoEmpresafechaVinculacionfechaInicioVigenciaVinculacionestadoVinculacionfechaRetiroVinculacioncausalRetiroVinculacionsalarioestrato, idDatoAdicional-->valorfechaInicioVigenciaDatoAdicionalfechaFinVigenciaDatoAdicional, idCiudaddescripcionCiudad-->idDepartamentodescripcionDepartamento}

⑩ Determinar la super llave

Super llave:

idEstadoCivildProgramaVinculacionidClienteidContactoidVinculacionidDatoAdicionalidCiudaddescripcionCiudad

⑩ Definición de subesquemas

R1(idEstadoCivil, descripcionEstadoCivil)

R2(idProgramaVinculacion, decripcionVinculacion)

R3(idCliente, tipoIdentificacion, numeroIdentificacion, estadoCivil, primerNombre, segundoNombre, primerApellido, segundoApellido, fechaNacimiento, estadoCliente)

R4(idContacto, direccion, barrio, codigoZona, ciudadID, area, telefono, extension, celular, email)

R5(idVinculacion, fechaIngresoEmpresa, fechaVinculacion, fechaInicioVigenciaVinculacion, estadoVinculacion, fechaRetiroVinculacion, causalRetiroVinculacion, salario, estrato)

R6(idDatoAdicional, valor, fechaInicioVigenciaDatoAdicional, fechaFinVigenciaDatoAdicional)

R7(idCiudad, descripcionCiudad, idDepartamento, descripcionDepartamento)

R8(idEstadoCivil, idProgramaVinculacion, idCliente, idContacto, idVinculacion, idDatoAdicional, idCiudad, descripcionCiudad)

En la siguiente imagen, se puede observar el proceso de normalización por parte de la aplicación.

Normalizador Bernstein

Resultado de la Normalización

Atributos: A: {idCliente, tipoIdentificacion, numeroIdentificacion, estadoCivil, primerNombre, segundoNombre}

Dependencias: {idDepartamento-->descripcionDepartamento, idProgramaVinculacion-->descripcionVinculacion}

1. Calculo de dependencias elementales

{idDepartamento-->descripcionDepartamento, idProgramaVinculacion-->descripcionVinculacion, idCiudad-->descripcionCiudad, idDepartamento-->descripcionDepartamento}

2. Eliminar elementos extraños

FD: {idEstadoCivil-->descripcionEstadoCivil, idProgramaVinculacion-->descripcionVinculacion, idCliente-->tipoIdentificacion, idCliente-->numeroIdentificacion}

3. Eliminar dependencias innecesarias

{idDepartamento-->descripcionDepartamento, idProgramaVinculacion-->descripcionVinculacion, idCiudad-->descripcionCiudad, idDepartamento-->descripcionDepartamento}

4. Agrupar dependencias por llave

{idDepartamento-->descripcionDepartamento, idProgramaVinculacion-->descripcionVinculacion, idCiudad-->descripcionCiudad, idDepartamento-->descripcionDepartamento}

5. Super llave

idEstadoCivilidProgramaVinculacionidClienteidContactoidVinculacionidDatoAdicionalidCiudaddescripcionCiudad

SubEsquemas

R7(idCiudad, descripcionCiudad, idDepartamento, descripcionDepartamento)

R8(idEstadoCivil, idProgramaVinculacion, idCliente, idContacto, idVinculacion, idDatoAdicional, idCiudad, descripcionCiudad)

Regresar

6. CONCLUSIONES

De acuerdo al resultado propuesto por el normalizador en las diferentes etapas que se han planteado se puede concluir lo siguiente:

- El ejercicio propuesto ya se encuentra en una forma normal superior o igual a la tercera forma normal por lo cual en las diferentes fases del programa no se presenta ningún cambio a las dependencias funcionales a pesar de aplicar el algoritmo de Bernstein.
- Los esquemas propuestos como salida final del programa son bastante similares a los que se muestran en este documento aunque se puede observar una organización de los esquemas de acuerdo a como se propone a través del algoritmo de Bernstein.
- Se agrega como paso adicional al programa un proceso llamado Hallar Super Llave que consiste en hallar un conjunto de atributos que definan todo el conjunto de dependencias propuestas, este proceso se realiza haciendo uso de la tabla de Izquierdos-Medios-Derechos para definir los atributos primos y no primos y cuales atributos deben aparecer en una super llave.
- Debido a que no fue posible evidenciar con el dominio propuesto un proceso de cambio de las dependencias funcionales haciendo uso del algoritmo de Bernstein se encuentra un ejemplo navegando en internet con un ejercicio propuesto y su resultado. A continuación se muestra el desarrollo de este a través del Normalizador y del resultado mostrado en el documento.

El ejemplo descrito se puede ver en el siguiente documento pero adicional es descrito de la misma forma acá para manejo del documento:

<http://cs.uns.edu.ar/~gis/tdbd/Archivos/Practicos/EjemplodependenciasfuncionalesydescomposicionTDBD.pdf>

Supongamos que deseamos mantener información sobre los cursados de alumnos de una universidad. Los datos que se desean mantener sobre los cursados incluyen

ESQUEMA_CURSADOS (MateriaCódigo, MateriaNombre, DptoCódigo, DptoNombre, AlumnoLU, AlumnoApellido, AlumnoNombre, AlumnoDomicilio, AlumnoTipoDoc, AlumnoNroDoc, Cuatrimestre, AñoAcadémico, CursadoResultado)

Donde:

Atributo	Descripción	Mapeo a Letra
MateriaCódigo	Código de materia. Es el atributo que identifica a una materia.	A
MateriaNombre	Nombre de la materia.	B
DptoCódigo	Código de departamento académico. Cada departamento tiene su propio código.	C
DptoNombre	Nombre del departamento.	D
AlumnoLU	Número de libreta de un alumno. A cada alumno se le asigna un único número de libret	E
AlumnoApellido	Apellido del alumno.	F
AlumnoNombre	Nombre del alumno.	G
AlumnoDomicilio	Domicilio del alumno.	
AlumnoTipoDoc	Tipo de documento. Puede asumir uno de los siguientes valores: DNI, LC, CI o Pasaporte.	I
AlumnoNroDoc	Número de documento	J
Cuatrimestre	Cuatrimestre lectivo. Asume los valores: 1 y 2	K
AñoAcadémico	Año académico.	L
CursadoResultado	Resultdo con que se califica el cursado. Puede tener asociado uno de los siguientes valores: "Aprobado", "Desaprobado", "Ausente", "Sin Calificar".	M

Para el esquema del inciso anterior se plantearon las siguientes restricciones o dependencias funcionales que se espera satisfagan todas las instancias de R:

Dependencias funcionales

1. MateriaCódigo → MateriaNombre, DptoCódigo, DptoNombre
2. DptoCódigo → DptoNombre
3. AlumnoLU → AlumnoApellido, AlumnoNombre, AlumnoDomicilio, AlumnoTipoDoc, AlumnoNroDoc
4. AlumnoTipoDoc, AlumnoNroDoc → AlumnoApellido, AlumnoNombre, AlumnoDomicilio,

AlumnoLU

5. AlumnoLU, MateriaCódigo, DptoCódigo, AñoAcadémico, Cuatrimestre →CursadoResultado

6. AlumnoTipoDoc, AlumnoNroDoc, MateriaCódigo, DptoCódigo, AñoAcadémico, Cuatrimestre →CursadoResultado

Estas mismas dependencias funcionales se representan en letras, de acuerdo a la tabla antes definida, de la siguiente manera:

1. A → BCD

2. C → D

3. E → FGHIJ

4. IJ → EFGH

5. ACEKL → M

6. IJACKL → M

A continuación se muestra el nuevo ejemplo que se está analizando bajo el esquema definido de XML para el Normalizador:

```
<root>
  <attributes>
    <attribute>MateriaCódigo</attribute>
    <attribute>MateriaNombre</attribute>
    <attribute>DptoCódigo</attribute>
    <attribute>DptoNombre</attribute>
    <attribute>AlumnoLU</attribute>
    <attribute>AlumnoApellido</attribute>
    <attribute>AlumnoNombre</attribute>
    <attribute>AlumnoDomicilio</attribute>
    <attribute>AlumnoTipoDoc</attribute>
    <attribute>AlumnoNroDoc</attribute>
    <attribute>Cuatrimestre</attribute>
    <attribute>AñoAcadémico</attribute>
    <attribute>CursadoResultado</attribute>
  </attributes>
  <fds>
    <fd>
      <left>
        <attribute>MateriaCódigo</attribute>
      </left>
      <right>
        <attribute>MateriaNombre</attribute>
        <attribute>DptoCódigo</attribute>
        <attribute>DptoNombre</attribute>
      </right>
    </fd>
    <fd>
      <left>
        <attribute>DptoCódigo</attribute>
```



```

        </left>
        <right>
            <attribute>DptoNombre</attribute>
        </right>
    </fd>
    <fd>
        <left>
            <attribute>AlumnoLU</attribute>
        </left>
        <right>
            <attribute>AlumnoApellido</attribute>
            <attribute>AlumnoNombre</attribute>
            <attribute>AlumnoDomicilio</attribute>
            <attribute>AlumnoTipoDoc</attribute>
            <attribute>AlumnoNroDoc</attribute>
        </right>
    </fd>
    <fd>
        <left>
            <attribute>AlumnoTipoDoc</attribute>
            <attribute>AlumnoNroDoc</attribute>
        </left>
        <right>
            <attribute>AlumnoApellido</attribute>
            <attribute>AlumnoNombre</attribute>
            <attribute>AlumnoDomicilio</attribute>
            <attribute>AlumnoLU</attribute>
        </right>
    </fd>
    <fd>
        <left>
            <attribute>AlumnoLU</attribute>
            <attribute>MateriaCódigo</attribute>
            <attribute>DptoCódigo</attribute>
            <attribute>AñoAcadémico</attribute>
            <attribute>Cuatrimestre</attribute>
        </left>
        <right>
            <attribute>CursadoResultado</attribute>
        </right>
    </fd>
    <fd>
        <left>
            <attribute>AlumnoTipoDoc</attribute>
            <attribute>AlumnoNroDoc</attribute>
            <attribute>MateriaCódigo</attribute>
            <attribute>DptoCódigo</attribute>
            <attribute>AñoAcadémico</attribute>
            <attribute>Cuatrimestre</attribute>
        </left>
        <right>
            <attribute>CursadoResultado</attribute>
        </right>
    </fd>
</fds>
</root>

```

A continuación se muestra el resultado de la ejecución del algoritmo mejorado de Bernstein mediante la aplicación desarrollada.

⑩ Cálculo de dependencias necesarias

FD: {DptoCódigo-->DptoNombre, AlumnoLUMateriaCódigoDptoCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, AlumnoTipoDocAlumnoNroDocMateriaCódigoDptoCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, MateriaCódigo-->MateriaNombre, MateriaCódigo-->DptoCódigo, MateriaCódigo-->DptoNombre, AlumnoLU-->AlumnoApellido, AlumnoLU-->AlumnoNombre, AlumnoLU-->AlumnoDomicilio, AlumnoLU-->AlumnoTipoDoc, AlumnoLU-->AlumnoNroDoc, AlumnoTipoDocAlumnoNroDoc-->AlumnoApellido, AlumnoTipoDocAlumnoNroDoc-->AlumnoNombre, AlumnoTipoDocAlumnoNroDoc-->AlumnoDomicilio, AlumnoTipoDocAlumnoNroDoc-->AlumnoLU}

⑩ Eliminación de elementos extraños

FD: {DptoCódigo-->DptoNombre, AlumnoLUMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, AlumnoTipoDocAlumnoNroDocMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, MateriaCódigo-->MateriaNombre, MateriaCódigo-->DptoCódigo, MateriaCódigo-->DptoNombre, AlumnoLU-->AlumnoApellido, AlumnoLU-->AlumnoNombre, AlumnoLU-->AlumnoDomicilio, AlumnoLU-->AlumnoTipoDoc, AlumnoLU-->AlumnoNroDoc, AlumnoTipoDocAlumnoNroDoc-->AlumnoApellido, AlumnoTipoDocAlumnoNroDoc-->AlumnoNombre, AlumnoTipoDocAlumnoNroDoc-->AlumnoDomicilio, AlumnoTipoDocAlumnoNroDoc-->AlumnoLU}

⑩ Eliminación de redundancias

FD: {DptoCódigo-->DptoNombre, AlumnoTipoDocAlumnoNroDocMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, MateriaCódigo-->MateriaNombre, MateriaCódigo-->DptoCódigo, AlumnoLU-->AlumnoTipoDoc, AlumnoLU-->AlumnoNroDoc, AlumnoTipoDocAlumnoNroDoc-->AlumnoApellido, AlumnoTipoDocAlumnoNroDoc-->AlumnoNombre, AlumnoTipoDocAlumnoNroDoc-->AlumnoDomicilio, AlumnoTipoDocAlumnoNroDoc-->AlumnoLU}

⑩ Agrupamiento por llaves

FD: {DptoCódigo-->DptoNombre, AlumnoTipoDocAlumnoNroDocMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, MateriaCódigo-->MateriaNombreDptoCódigo, AlumnoLU-->AlumnoTipoDocAlumnoNroDoc, AlumnoTipoDocAlumnoNroDoc-->AlumnoApellidoAlumnoNombreAlumnoDomicilioAlumnoLU}

⑩ Determinar la súper llave

MateriaCódigoAñoAcadémicoCuatrimestreAlumnoLU

⑩ Definición de subesquemas

R1(DptoCódigo, DptoNombre)
R2(AlumnoTipoDoc, AlumnoNroDoc, MateriaCódigo, AñoAcadémico, Cuatrimestre, CursadoResultado)
R3(MateriaCódigo, MateriaNombre, DptoCódigo)
R4(AlumnoLU, AlumnoTipoDoc, AlumnoNroDoc)
R5(AlumnoTipoDoc, AlumnoNroDoc, AlumnoApellido, AlumnoNombre, AlumnoDomicilio, AlumnoLU)
R6(MateriaCódigo, AñoAcadémico, Cuatrimestre, AlumnoLU)

En la siguiente imagen, se puede observar el proceso de normalización por parte de la aplicación.

Normalizador Bernstein

Resultado de la Normalización

Atributos: {Nombre, AlumnoDomicilio, AlumnoTipoDoc, AlumnoNroDoc, Cuatrimestre, AñoAcadémico, CursadoResultado}

Dependencias: {AlumnoTipoDocAlumnoNroDocMateriaCódigoDptoCódigoAñoAcadémicoCuatrimestre-->CursadoResultado}

1. Calculo de dependencias elementales

FD: {DptoCódigo-->DptoNombre, AlumnoLUMateriaCódigoDptoCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, AlumnoTipoDocAlumnoNroDocMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado}

2. Eliminar elementos extraños

FD: {DptoCódigo-->DptoNombre, AlumnoLUMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, AlumnoTipoDocAlumnoNroDocMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado}

3. Eliminar dependencias innecesarias

FD: {DptoCódigo-->DptoNombre, AlumnoTipoDocAlumnoNroDocMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, MateriaCódigo-->MateriaNombre}

4. Agrupar dependencias por llave

FD: {DptoCódigo-->DptoNombre, AlumnoTipoDocAlumnoNroDocMateriaCódigoAñoAcadémicoCuatrimestre-->CursadoResultado, MateriaCódigo-->MateriaNombre}

5. Super llave

MateriaCódigoAñoAcadémicoCuatrimestreAlumnoLU

SubEsquemas

Subesquemas:
R1(DptoCódigo, DptoNombre)
R2(AlumnoTipoDoc, AlumnoNroDoc, MateriaCódigo, AñoAcadémico, Cuatrimestre, CursadoResultado)
R3(MateriaCódigo, MateriaNombre, DptoCódigo)

Regresar

A continuación se muestran los esquemas propuestos en el nuevo ejemplo:

L1(MateriaCódigo,MateriaNombre,DptoCódigo)

L2(DeptoCódigo,DptoNombre)

L3(AlumnoLU,AlumnoApellido,AlumnoNombre,AlumnoDomicilio,AlumnoTipoDoc,AlumnoNroDoc)

L4(MateriaCódigo,AlumnoLU,Cuatrimestre,AñoAcadémico,CursoResultado)

Cabe resaltar que el nuevo ejemplo propuesto se encuentra en BCNF, que es una adecuación de la 3ra forma normal y que no es trabajada por el algoritmo de Bernstein que utiliza el Normalizador construido.

Realizando una comparativa con los esquemas propuestos por el Normalizador construido se encuentra lo siguiente:

- El esquema R1 y el esquema L2 son idénticos.
- El esquema R3 y el esquema L1 son idénticos.
- Es posible eliminar el esquema R4 ya que el esquema R5 lo contiene.
- El esquema R5 es idéntico en atributos al esquema L3 pero con llaves primarias diferentes.
- La super llave que se halla con el Normalizador se encuentra contenida en L4, esta es una práctica propuesta para que no exista perdida de información al momento de agrupar las tablas y querer formar todo el universo del dominio.