

Projeto SICADI Módulo Contábil
Leiaute do Arquivo de Balancetes Contábeis
Planificação Contábil nº 5 (a partir de Janeiro/2010)
Versão 1.8 (Adequação periodicidade trimestral)

Elaboração	Aprovação
Nome: Afonso Bezerra Lima Neto	Nome: Mauricio de Aguirre Nakata
Gestor do Projeto (Dataprev)	Cliente
Data: __/__/__.	Data: __/__/__.

Histórico de Revisões

Data	Versão	Descrição	Autor
12/06/2009	1.0	Elaboração do documento	Marcelo Santos de Araújo (UDCE)
21/07/2009	1.1	Ajuste no layout	Leonardo Moura Leitão (UDCE)
10/08/2009	1.2	Alteração da cardinalidade do elemento “plano-beneficio”	Marcelo Santos de Araújo (UDCE)
20/11/2009	1.3	Ajuste na restrição do tipo simples “codigo-entidadeType”, “rubricaType” e “cnpbType”.	Marcelo Santos de Araújo (UDCE)
03/12/2009	1.4	Ajuste nos exemplos: alteração no nome do atributo “cnpb” do tipo complexo “plano-beneficioType”.	Marcelo Santos de Araújo (UDCE)
26/04/2011	1.5	Ajuste no Layout para inserção de um novo campo “Periodicidade” e modificação do campo “mes” para “mes_trimestre”	Luiz Soares (UDCE)
02/05/2011	1.6	Revisão do documento por pares.	Cristina Holanda (UDCE) Luiz Soares (UDCE)
03/05/2011	1.7	Mudança da tag <trimestreMes> por <mes>. Exclusão da tag <periodicidade> e criação da tag <trimestre>. Uso da Tag <Choice> para declaração dos campos mes e trimestre como de exclusão mútua.	Luiz Soares (UDCE)
20/05/2011	1.8	Revisão do cliente e do gestor do projeto.	Cristina Holanda (UDCE)

Índice Analítico

1.Orientações Gerais	5
1.1.Quem está obrigado a enviar o arquivo	5
1.2.Abrangência dos dados	5
1.3.Responsabilidade e frequência do envio dos arquivos	5
1.4.Vigência do leiaute	5
2.Nome do arquivo	5
3.Estrutura do XML	5
4.Explicando o Schema	9
4.1.Definição do Tipo Complexo – dados-contabeisType	9
4.2.Definição do Tipo Complexo – lancamentoType	12
4.3.Definição do Tipo Complexo – consolidadoType	14
4.4.Definição do Tipo Complexo – plano-beneficioType	14
4.5.Definição do Tipo Complexo – plano-gestao-administrativaType	15
4.6.Definição do Tipo Simples – valorMonetarioType	15
4.7.Definição do Tipo Simples – anoType	16
4.8.Definição do Tipo Simples – mesType	16
4.9.Definição do Tipo Simples – trimestreType	17
4.10.Definição do Tipo Simples – naturezaType	17
4.11.Definição do Tipo Simples – codigo-entidadeType	18
4.12.Definição do Tipo Simples – plano-contasType	18
4.13.Definição do Tipo Simples – rubricaType	18
4.14.Definição do Tipo Simples – cnpbType	19



PD-DATAPREV

Processo de Desenvolvimento de Software da Dataprev

SICADI Módulo Contábil – Leiaute do Arquivo de Balancetes Contábeis

5.Exemplo de um balancete (resumido) em XML

19

1. Orientações Gerais

1.1. Quem está obrigado a enviar o arquivo

Entidades Fechadas de Previdência Complementar (EFPC) que possuem registros patrimoniais. As EFPC que operam planos de assistência à saúde com registro na ANS deverão seguir as instruções e a planificação contábil da Agência Nacional de Saúde Suplementar - ANS.

1.2. Abrangência dos dados

O arquivo possui os registros contábeis das EFPCs, que devem estar em conformidade com a planificação contábil vigente na data base do arquivo.

1.3. Responsabilidade e frequência do envio dos arquivos

O envio do arquivo é de responsabilidade das EFPCs e a frequência é trimestral, podendo excepcionalmente ser mensal. O prazo legal para envio é até o último dia do mês subsequente ao trimestre de referência do arquivo, no entanto as EFPC poderão enviar seus arquivos em qualquer tempo, sujeitando-se às penalidades cabíveis.

1.4. Vigência do leiaute

O leiaute aqui discriminado deverá ser utilizado para envio de balancetes cujas competências sejam iguais ou superiores a janeiro de 2010. Balancetes com competências anteriores a janeiro de 2010, mesmo que enviados em atraso, deverão respeitar o leiaute XML vigente à época.

2. Nome do arquivo

Para as competências a partir de janeiro de 2011, o nome do arquivo com as informações contábeis deve seguir a regra de formação descrita abaixo:

- CodigoDaEntidade_Ano(Mes/Trimestre)(M/T).xml

3. Estrutura do XML

A estrutura do XML é definida por um arquivo conhecido como XML Schema. Esse arquivo define como o XML é esquematizado, ou seja, como os elementos (TAGs) são organizados dentro do documento.

XML Schema do Balancete Contábil (Completo)

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema
  targetNamespace="http://balancete.tiposxml.cgpc28.comum.spccontabil.dataprev.gov.br"
  elementFormDefault="qualified"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://balancete.tiposxml.cgpc28.comum.spccontabil.dataprev.gov.br">
  <xsd:complexType name="dados-contabeisType">
```

```
<xsd:sequence>
  <xsd:element name="codigo-entidade"
    type="codigo-entidadeType" maxOccurs="1" minOccurs="1">
  </xsd:element>
  <xsd:element name="plano-contas" type="plano-contasType"
    maxOccurs="1" minOccurs="1">
  </xsd:element>
  <xsd:element name="e-mail" type="xsd:string"
    maxOccurs="unbounded" minOccurs="0">
  </xsd:element>
  <xsd:choice maxOccurs="1" minOccurs="1">
    <xsd:element name="mes" type="mesType" maxOccurs="1"
      minOccurs="1">
    </xsd:element>
    <xsd:element name="trimestre" type="trimestreType" maxOccurs="1"
      minOccurs="1">
    </xsd:element>
  </xsd:choice>
  <xsd:element name="ano" type="anoType" maxOccurs="1"
    minOccurs="1">
  </xsd:element>
  <xsd:element name="consolidado" type="consolidadoType"
    maxOccurs="1" minOccurs="1">
  </xsd:element>
  <xsd:element name="plano-beneficio" type="plano-beneficioType"
    maxOccurs="unbounded" minOccurs="0">
  </xsd:element>
  <xsd:element name="plano-gestao-administrativa"
    type="plano-gestao-administrativaType" maxOccurs="1" minOccurs="1">
  </xsd:element>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="lancamentoType">
  <xsd:sequence>
    <xsd:element name="rubrica" type="rubricaType" maxOccurs="1"
      minOccurs="1">
    </xsd:element>
    <xsd:element name="saldo-inicial" type="valorMonetarioType"
      maxOccurs="1" minOccurs="1">
    </xsd:element>
    <xsd:element name="natureza" maxOccurs="1" minOccurs="1"
      type="naturezaType">
    </xsd:element>
    <xsd:element name="debito" type="valorMonetarioType"
      maxOccurs="1" minOccurs="1">
    </xsd:element>
    <xsd:element name="credito" type="valorMonetarioType"
      maxOccurs="1" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:simpleType name="valorMonetarioType">
  <xsd:restriction base="xsd:decimal">
    <xsd:fractionDigits fixed="true" value="2" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:element name="dados-contabeis" type="dados-contabeisType">
</xsd:element>

<xsd:simpleType name="anoType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="2010"></xsd:minInclusive>
    <xsd:maxInclusive value="2100"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="mesType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"></xsd:minInclusive>
    <xsd:maxInclusive value="12"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="trimestreType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"></xsd:minInclusive>
    <xsd:maxInclusive value="4"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="naturezaType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="DV"></xsd:enumeration>
    <xsd:enumeration value="CR"></xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="codigo-entidadeType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"></xsd:minInclusive>
    <xsd:maxInclusive value="99999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="plano-contasType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="5"></xsd:minInclusive>
    <xsd:maxInclusive value="999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

```
<xsd:simpleType name="rubricaType">
  <xsd:restriction base="xsd:long">
    <xsd:minInclusive value="1000000000"></xsd:minInclusive>
    <xsd:maxInclusive value="9999999999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="cnpbType">
  <xsd:restriction base="xsd:long">
    <xsd:minInclusive value="1000000000"></xsd:minInclusive>
    <xsd:maxInclusive value="9999999999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="consolidadoType">
  <xsd:sequence>
    <xsd:element name="lancamento" type="lancamentoType"
      maxOccurs="unbounded" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="plano-gestao-administrativaType">
  <xsd:sequence>
    <xsd:element name="lancamento" type="lancamentoType"
      maxOccurs="unbounded" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="plano-beneficioType">
  <xsd:sequence>
    <xsd:element name="lancamento" type="lancamentoType"
      maxOccurs="unbounded" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="cnpb" type="cnpbType" use="required"></xsd:attribute>
</xsd:complexType>
</xsd:schema>
```


4. Explicando o Schema

4.1. Definição do Tipo Complexo – dados-contabeisType

A linha `<xsd:complexType name="dados-contabeisType">` define um tipo complexos (elemento) chamado `dados-contabeisType`. Esse elemento possui outros elementos como veremos a seguir.

A linha `<xsd:sequence>` define que os elementos no arquivo XML devem obedecer a ordem dos elementos definidos dentro de `<xsd:sequence>...</xsd:sequence>`.

XML Schema/XML Válido

```
<xsd:complexType name="dados-contabeisType">
  <xsd:sequence>
    <xsd:element name="codigo-entidade"
      type="codigo-entidadeType" maxOccurs="1" minOccurs="1">
    </xsd:element>
    <xsd:element name="plano-contas" type="plano-contasType"
      maxOccurs="1" minOccurs="1">
    </xsd:element>
    <xsd:element name="e-mail" type="xsd:string"
      maxOccurs="unbounded" minOccurs="0">
    </xsd:element>
    <xsd:element name="mes" type="mesType" maxOccurs="1"
      minOccurs="1">
    </xsd:element>
    <xsd:element name="trimestre" type="trimestreType" maxOccurs="1"
      minOccurs="1">
    </xsd:element>
    <xsd:element name="ano" type="anoType" maxOccurs="1"
      minOccurs="1">
    </xsd:element>
    <xsd:element name="consolidado" type="consolidadoType"
      maxOccurs="1" minOccurs="1">
    </xsd:element>
    <xsd:element name="plano-beneficio"
      type="plano-beneficioType" maxOccurs="unbounded" minOccurs="0">
    </xsd:element>
    <xsd:element name="plano-gestao-administrativa"
      type="plano-gestao-administrativaType" maxOccurs="1" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="dados-contabeis" type="dados-contabeisType">
</xsd:element>
```

```
<dados-contabeis
xmlns="http://balancete.tiposxml.cgpc28.comum.spccontabil.dataprev.gov.br">
```

```

<codigo-entidade>12345</codigo-entidade>
<plano-contas>5</plano-contas>
<e-mail>email_da_entidade@dominio.com.br</e-mail>
<trimestre>2</trimestre>
<ano>2011</ano>
<consolidado>
...
</consolidado>
<plano-beneficio cnpb="1987654321">
...
</plano-beneficio>
<plano-gestao-administrativa>
...
</plano-gestao-administrativa>
</dados-contabeis>

```

Os sub-elementos do elemento **dados-contabeisType**:

1. Nome: **codigo-entidade**
Tipo: **codigo-entidadeType** (comentado posteriormente)
Ordem: **Primeiro elemento**
Um elemento (**obrigatório**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

```

<xsd:element name="codigo-entidade" type="codigo-entidadeType"
  maxOccurs="1" minOccurs="1">
</xsd:element>

<codigo-entidade>12345</codigo-entidade>

```

2. Nome: **plano-contas**
Tipo: **plano-contasType** (comentado posteriormente)
Ordem: **Segundo elemento**
Um elemento (**obrigatório**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

```

<xsd:element name="plano-contas" type="plano-contasType"
  maxOccurs="1" minOccurs="1">
</xsd:element>

<plano-contas>5</plano-contas>

```

3. Nome: **e-mail**
Tipo: **xsd:string**
Ordem: **Terceiro elemento**
Zero ou mais elementos (**opcional**). (**maxOccurs="unbounded" minOccurs="0"**)

XML Schema/XML Válido

```

<xsd:element name="e-mail" type="xsd:string"
  maxOccurs="unbounded" minOccurs="0">

```

<code></xsd:element></code>
<code><e-mail>email_da_entidade@dominio.com.br</e-mail></code>

4. Nome: **mes**
 Tipo: **mesType** (comentado posteriormente)
 Ordem: **Quarto elemento**
 Um elemento (**opcional condicional**). (**maxOccurs="1" minOccurs="1"**)

Nome: **trimestre**
 Tipo: **trimestreType** (comentado posteriormente)
 Ordem: **Quinto elemento**
 Um elemento (**opcional condicional**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

<code><xsd:choice maxOccurs="1" minOccurs="1"></code>
<code> <xsd:element name="mes" type="mesType" maxOccurs="1"</code>
<code> minOccurs="1"></code>
<code> </xsd:element></code>
<code> <xsd:element name="trimestre" type="trimestreType" maxOccurs="1"</code>
<code> minOccurs="1"></code>
<code> </xsd:element></code>
<code></xsd:choice></code>
<code><mes>2</mes></code>
Ou
<code><trimestre>2</trimestre></code>

5. Nome: **ano**
 Tipo: **anoType** (comentado posteriormente)
 Ordem: **Sexto elemento**
 Um elemento (**obrigatório**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

<code><xsd:element name="ano" type="anoType" maxOccurs="1" minOccurs="1"></code>
<code></xsd:element></code>
<code><ano>2011</ano></code>

6. Nome: **consolidado**
 Tipo: **consolidadoType** (comentado posteriormente)
 Ordem: **Sétimo elemento**
 Um elemento (**obrigatório**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

<code><xsd:element name="consolidado" type="consolidadoType"</code>
<code> maxOccurs="1" minOccurs="1"></code>
<code></xsd:element></code>
<code><consolidado></code>
<code>...</code>
<code></consolidado></code>

7. Nome: **plano-beneficio**
Tipo: **plano-beneficioType** (comentado posteriormente)
Ordem: **Oitavo elemento**
Zero ou mais elementos (**opcional**). (**maxOccurs="unbounded" minOccurs="0"**)

XML Schema/XML Válido

```
<xsd:element name="plano-beneficio"
  type="plano-beneficioType" maxOccurs="unbounded" minOccurs="0">
</xsd:element>

<plano-beneficio cnpb="1987654321">
...
</plano-beneficio>
<plano-beneficio cnpb="2987654321">
...
</plano-beneficio>
```

8. Nome: **plano-gestao-administrativa**
Tipo: **plano-gestao-administrativaType** (comentado posteriormente)
Ordem: **Nono elemento**
Zero ou um elemento (**Obrigatório**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

```
<xsd:element name="plano-gestao-administrativa"
  type="plano-gestao-administrativaType" maxOccurs="1" minOccurs="1">
</xsd:element>

<plano-gestao-administrativa>
...
</plano-gestao-administrativa>
```

4.2. Definição do Tipo Complexo – **lancamentoType**

A linha **<xsd:complexType name="lancamentoType">** define um tipo complexo (elemento) chamado **lancamentoType**. Esse elemento possui outros elementos como veremos a seguir.

A linha **<xsd:sequence>** define que os elementos no arquivo XML devem obedecer a ordem dos elementos definidos dentro de **<xsd:sequence>...</xsd:sequence>**.

XML Schema/XML Válido

```
<xsd:complexType name="lancamentoType">
  <xsd:sequence>
    <xsd:element name="rubrica" type="rubricaType" maxOccurs="1"
      minOccurs="1">
    </xsd:element>
    <xsd:element name="saldo-inicial" type="valorMonetarioType"
      maxOccurs="1" minOccurs="1">
    </xsd:element>
    <xsd:element name="natureza" maxOccurs="1" minOccurs="1"
      type="naturezaType">
    </xsd:element>
```

<pre> <xsd:element name="debito" type="valorMonetarioType" maxOccurs="1" minOccurs="1"> </xsd:element> <xsd:element name="credito" type="valorMonetarioType" maxOccurs="1" minOccurs="1"> </xsd:element> </xsd:sequence> </xsd:complexType> </pre>
<pre> <lancamento> <rubrica>1000000000</rubrica> <saldo-inicial>1390032854.16</saldo-inicial> <natureza>DV</natureza> <debito>141229276.17</debito> <credito>137734648.59</credito> </lancamento> </pre>

Os sub-elementos do elemento `lancamentoType`:

1. Nome: **rubrica**
 Tipo: **rubricaType** (comentado posteriormente)
 Ordem: **Primeiro elemento**
 Um elemento (**Obrigatório**). (`maxOccurs="1" minOccurs="1"`)

XML Schema/XML Válido

<pre> <xsd:element name="rubrica" type="rubricaType" maxOccurs="1" minOccurs="1"> </xsd:element> </pre>
<pre> <rubrica>1000000000</rubrica> </pre>

2. Nome: **saldo-inicial**
 Tipo: **valorMonetarioType** (comentado posteriormente)
 Ordem: **Segundo elemento**
 Um elemento (**Obrigatório**). (`maxOccurs="1" minOccurs="1"`)

XML Schema/XML Válido

<pre> <xsd:element name="saldo-inicial" type="valorMonetarioType" maxOccurs="1" minOccurs="1"> </xsd:element> </pre>
<pre> <saldo-inicial>1390032854.16</saldo-inicial> </pre>

3. Nome: **natureza**
 Tipo: **naturezaType** (comentado posteriormente)
 Ordem: **Terceito elemento**
 Um elemento (**Obrigatório**). (`maxOccurs="1" minOccurs="1"`)

XML Schema/XML Válido

<pre> <xsd:element name="natureza" maxOccurs="1" minOccurs="1" type="naturezaType"> </xsd:element> </pre>

```
<natureza>DV</natureza>
```

4. Nome: **debito**
Tipo: **valorMonetarioType** (comentado posteriormente)
Ordem: **Quarto elemento**
Um elemento (**Obrigatório**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

```
<xsd:element name="debito" type="valorMonetarioType"
maxOccurs="1" minOccurs="1">
</xsd:element>
```

```
<debito>141229276.17</debito>
```

5. Nome: **credito**
Tipo: **creditoType** (comentado posteriormente)
Ordem: **Quinto elemento**
Um elemento (**obrigatório**). (**maxOccurs="1" minOccurs="1"**)

XML Schema/XML Válido

```
<xsd:element name="credito" type="valorMonetarioType"
maxOccurs="1" minOccurs="1">
</xsd:element>
```

```
<credito>137734648.59</credito>
```

4.3. Definição do Tipo Complexo – **consolidadoType**

A linha `<xsd:complexType name="consolidadoType">` define um tipo complexo (elemento) chamado **consolidadoType**. Esse elemento possui um ou mais elementos do tipo **lançamentoType**.

XML Schema/XML Válido

```
<xsd:complexType name="consolidadoType">
  <xsd:sequence>
    <xsd:element name="lançamento" type="lançamentoType"
      maxOccurs="unbounded" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```
<consolidado>
  <lançamento>...</lançamento>
  <lançamento>...</lançamento>
  // Outros lançamentos...
</consolidado>
```

4.4. Definição do Tipo Complexo – **plano-beneficioType**

A linha `<xsd:complexType name="plano-beneficioType">` define um tipo complexo (elemento) chamado `plano-beneficioType`. Esse elemento possui um ou mais elementos do tipo `lançamentoType`.

A linha `<xsd:attribute name="cnpb" type="cnpbType" use="required"/>` define o atributo chamado `cnpb` e o seu valor deve ser do tipo simples `cnpbType`.

XML Schema/XML Válido

```
<xsd:complexType name="plano-beneficioType">
  <xsd:sequence>
    <xsd:element name="lançamento" type="lançamentoType"
      maxOccurs="unbounded" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
  <xsd:attribute name="cnpb" type="cnpbType" use="required"/>
</xsd:complexType>

<plano-beneficio cnpb="1987654321">
  <lançamento>...</lançamento>
  <lançamento>...</lançamento>
  // Outros lançamentos...
</plano-beneficio>
```

4.5. Definição do Tipo Complexo – `plano-gestao-administrativaType`

A linha `<xsd:complexType name="plano-gestao-administrativaType">` define um tipo complexo (elemento) chamado `plano-gestao-administrativaType`. Esse elemento possui um ou mais elementos do tipo `lançamentoType`.

XML Schema/XML Válido

```
<xsd:complexType name="plano-gestao-administrativaType">
  <xsd:sequence>
    <xsd:element name="lançamento" type="lançamentoType"
      maxOccurs="unbounded" minOccurs="1">
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<plano-gestao-administrativa>
  <lançamento>...</lançamento>
  <lançamento>...</lançamento>
  // Outros lançamentos...
</plano-gestao-administrativa>
```

4.6. Definição do Tipo Simples – `valorMonetarioType`

A linha `<xsd:simpleType name="valorMonetarioType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `valorMonetarioType`.

A linha `<xsd:restriction base="xsd:decimal">` define que:

- O tipo `valorMonetarioType` é um decimal.

- O tipo `xsd:decimal` definido no XML usa como separador de decimais o ponto.

A linha `<xsd:fractionDigits fixed="true" value="2"/>` define que:

- O número decimal deve ter duas casas decimais.

Exemplo: 123456.89

XML Schema

```
<xsd:simpleType name="valorMonetarioType">
  <xsd:restriction base="xsd:decimal">
    <xsd:fractionDigits fixed="true" value="2" />
  </xsd:restriction>
</xsd:simpleType>
```

4.7. Definição do Tipo Simples – anoType

A linha `<xsd:simpleType name="anoType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `anoType`.

A linha `<xsd:restriction base="xsd:integer">` define que:

- O tipo `anoType` é um inteiro

A linha `<xsd:minInclusive value="2010"></xsd:minInclusive>` define que:

- O valor deve ser maior ou igual a 2010.

A linha `<xsd:maxInclusive value="2100"></xsd:maxInclusive>` define que:

- O valor deve ser menor ou igual a 2100.

XML Schema

```
<xsd:simpleType name="anoType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="2010"></xsd:minInclusive>
    <xsd:maxInclusive value="2100"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

4.8. Definição do Tipo Simples – mesType

A linha `<xsd:simpleType name="mesType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `mesType`.

A linha `<xsd:restriction base="xsd:integer">` define que:

- O tipo `mesType` é um inteiro.

A linha `<xsd:minInclusive value="1"></xsd:minInclusive>` define que:

- O valor deve ser maior ou igual a 1.

A linha `<xsd:maxInclusive value="12"></xsd:maxInclusive>` define que:

- O valor deve ser menor ou igual a 12.

XML Schema

```
<xsd:simpleType name="mesType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"></xsd:minInclusive>
    <xsd:maxInclusive value="12"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

4.9. Definição do Tipo Simples – trimestreType

A linha `<xsd:simpleType name="trimestreType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `trimestreType`.

A linha `<xsd:restriction base="xsd:integer">` define que:

- O tipo `trimestreType` é um inteiro.

A linha `<xsd:minInclusive value="1"></xsd:minInclusive>` define que:

- O valor deve ser maior ou igual a 1.

A linha `<xsd:maxInclusive value="4"></xsd:maxInclusive>` define que:

- O valor deve ser menor ou igual a 4.

XML Schema

```
<xsd:simpleType name="trimestreType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"></xsd:minInclusive>
    <xsd:maxInclusive value="4"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

4.10. Definição do Tipo Simples – naturezaType

A linha `<xsd:simpleType name="naturezaType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `naturezaType`.

A linha `<xsd:restriction base="xsd:string">` define que:

- O tipo `naturezaType` é um string.

A linha `<xsd:enumeration value="DV"></xsd:enumeration>` define que:

- Os valores desse elemento devem fazer parte da enumeração definida.
- Podemos ver abaixo que os possíveis valores são: DV e CR.

XML Schema

```
<xsd:simpleType name="naturezaType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="DV"></xsd:enumeration>
    <xsd:enumeration value="CR"></xsd:enumeration>
  </xsd:restriction>
```

```
</xsd:simpleType>
```

4.11. Definição do Tipo Simples – `codigo-entidadeType`

A linha `<xsd:simpleType name="codigo-entidadeType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `codigo-entidadeType`.

A linha `<xsd:restriction base="xsd:integer">` define que:

- O tipo `codigo-entidadeType` é um inteiro.

A linha `<xsd:minInclusive value="1"></xsd:minInclusive>` define que:

- O valor deve ser maior ou igual a 1.

A linha `<xsd:maxInclusive value="99999"></xsd:maxInclusive>` define que:

- O valor deve ser menor ou igual a 99999.

XML Schema

```
<xsd:simpleType name="codigo-entidadeType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="1"></xsd:minInclusive>
    <xsd:maxInclusive value="99999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

4.12. Definição do Tipo Simples – `plano-contasType`

A linha `<xsd:simpleType name="plano-contasType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `plano-contasType`.

A linha `<xsd:restriction base="xsd:integer">` define que:

- O tipo `plano-contasType` é um inteiro.

A linha `<xsd:minInclusive value="5"></xsd:minInclusive>` define que:

- O valor deve ser maior ou igual a 5.

A linha `<xsd:maxInclusive value="999"></xsd:maxInclusive>` define que:

- O valor deve ser menor ou igual a 999.

XML Schema

```
<xsd:simpleType name="plano-contasType">
  <xsd:restriction base="xsd:integer">
    <xsd:minInclusive value="5"></xsd:minInclusive>
    <xsd:maxInclusive value="999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

4.13. Definição do Tipo Simples – `rubricaType`

A linha `<xsd:simpleType name="rubricaType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `rubricaType`.

A linha `<xsd:restriction base="xsd:long">` define que:

- O tipo `rubricaType` é um numérico (long).

A linha `<xsd:minInclusive value="1000000000"></xsd:minInclusive>` define que:

- O valor deve ser maior ou igual a 1000000000.

A linha `<xsd:maxInclusive value="9999999999"></xsd:maxInclusive>` define que:

- O valor deve ser menor ou igual a 9999999999.

XML Schema

```
<xsd:simpleType name="rubricaType">
  <xsd:restriction base="xsd:long">
    <xsd:minInclusive value="1000000000"></xsd:minInclusive>
    <xsd:maxInclusive value="9999999999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

4.14. Definição do Tipo Simples – `cnpbType`

A linha `<xsd:simpleType name="cnpbType">` define um tipo simples (elemento sem atributos e sem sub-elementos) chamado `cnpbType`.

A linha `<xsd:restriction base="xsd:long">` define que:

- O tipo `cnpbType` é um numérico (long).

A linha `<xsd:minInclusive value="1000000000"></xsd:minInclusive>` define que:

- O valor deve ser maior ou igual a 1000000000.

A linha `<xsd:maxInclusive value="9999999999"></xsd:maxInclusive>` define que:

- O valor deve ser menor ou igual a 9999999999.

XML Schema

```
<xsd:simpleType name="cnpbType">
  <xsd:restriction base="xsd:long">
    <xsd:minInclusive value="1000000000"></xsd:minInclusive>
    <xsd:maxInclusive value="9999999999"></xsd:maxInclusive>
  </xsd:restriction>
</xsd:simpleType>
```

5. Exemplo de um balancete (resumido) em XML

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<dados-contabeis
xmlns="http://balancete.tiposxml.cgpc28.comum.spccontabil.dataprev.gov.br">
  <codigo-entidade>12345</codigo-entidade>
```

```
<plano-contas>5</plano-contas>
<e-mail>email_da_entidade@dominio.com.br</e-mail>
<trimestre>2</trimestre>
<ano>2011</ano>
<consolidado>
  <lancamento>
    <rubrica>1000000000</rubrica>
    <saldo-inicial>1390032854.16</saldo-inicial>
    <natureza>DV</natureza>
    <debito>141229276.17</debito>
    <credito>158346.14</credito>
  </lancamento>
</consolidado>
<plano-beneficio cnpb="1987654321">
  <lancamento>
    <rubrica>1000000000</rubrica>
    <saldo-inicial>0.00</saldo-inicial>
    <natureza>DV</natureza>
    <debito>4310.38</debito>
    <credito>0.00</credito>
  </lancamento>
</plano-beneficio>
<plano-gestao-administrativa>
  <lancamento>
    <rubrica>1000000000</rubrica>
    <saldo-inicial>1443870.63</saldo-inicial>
    <natureza>DV</natureza>
    <debito>141224965.79</debito>
    <credito>158346.14 </credito>
  </lancamento>
</plano-gestao-administrativa>
</dados-contabeis>
```
