

**EXPLORANDO LINGUAGENS DE MARCAÇÃO PARA
REPRESENTAÇÃO DE RELATÓRIOS DE INFORMAÇÕES
FINANCEIRAS**

PAULO CAETANO DA SILVA

MAIO - 2003

UNIVERSIDADE SALVADOR
PROGRAMA DE PÓS-GRADUAÇÃO EM REDES DE COMPUTADORES

**EXPLORANDO LINGUAGENS DE MARCAÇÃO PARA
REPRESENTAÇÃO DE RELATÓRIOS DE INFORMAÇÕES
FINANCEIRAS**

PAULO CAETANO DA SILVA

Dissertação apresentada à Universidade Salvador,
como parte das exigências do Curso de Mestrado
Profissional em redes de computadores, área de
concentração em Tecnologias Web e Aplicações
Distribuídas, para obtenção do título de “Mestre”

Orientador
Prof. Dr. Cesar Camilo Teixeira

SALVADOR
MAIO - 2003

Dedico este trabalho a meus pais, por despertarem em mim o gosto pelo estudo e a constante busca pelo conhecimento, a minha querida Ana Paula, por sua dedicação, compreensão e principalmente pelo exemplo de perseverança e a pequena Giovanna, por me dar esperança e incentivo para melhorar sempre.

Agradeço aos colegas do Departamento de Tecnologia da Informação, Alexandre Sattin, Fernando Freitas, Patrícia Martinelli e Suely Iwamoto, a José Carlos, na Gerência do Banco Central em Salvador, e a Denise Sande, no Departamento de Mercado Aberto, pelas valiosas contribuições que possibilitaram o desenvolvimento desta dissertação, ao amigos Pêpê, pela ajuda na conceituação dos termos econômicos/financeiros e Camilo Mussi, pelo incentivo para realização do curso, e aos colegas do mestrado pelas horas de estudo e bom convívio que se transformaram em amizade.

“O real poder da Web semântica será realizado quando pessoas criarem programas que colem o conteúdo de diversas fontes da Web, processem a informação e permutem os resultados com outros programas. A Web semântica promoverá esta sinergia, mesmo agentes que não forem projetados para trabalharem juntos poderão transferir dados entre si, desde que os dados contenham semântica”, Tim Berners-Lee, James Hendler & Ora Lassila - The Semantic Web

SUMÁRIO

	Página
1. Introdução	01
1.1 Contexto.....	01
1.2 Objetivo.....	04
1.3 Organização.....	04
2. Linguagens de Marcação para o intercâmbio de informações financeiras na Internet ...	08
2.1 Linguagens de Marcação	08
2.2 Linguagens financeiras de Marcação.....	10
2.3 XML: uma retrospectiva.....	10
2.4 OFX	14
2.5 FIXML e swiftML.....	15
2.6 STPML.....	17
2.7 FpML	17
2.8 FinXML	18
2.9 TAXML	19
2.10 XBRL.....	21
2.11 Comparando linguagens financeiras de marcação	22
3. XBRL – Extensible Business Reporting Language	25
3.1 Motivos para se usar XBRL.....	25
3.2 Criação de documentos XBRL.....	30
3.3 A estrutura XBRL.....	32
3.4 Documentos XBRL – os elementos e atributos	34
3.4.1 O documento de Taxonomia XBRL	34
3.4.1.1 O elemento linkbaseRef	36
3.4.1.2 Tipos de dados monetary e share	37
3.4.1.3 O elemento Element.....	37
3.4.2 Os documentos Linkbases.....	38
3.4.2.1 Elementos do tipo link estendido.....	39
3.4.2.1.1 Elementos do tipo locator	40
3.4.2.1.1.1 Links Label	41
3.4.2.1.1.2 Links Reference.....	42

3.4.2.1.2 Elementos do tipo arc	43
3.4.2.1.2.1 Links Calculation	44
3.4.2.1.2.2 Links Presentation	45
3.4.2.1.2.3 Links Definition	45
3.4.3 O documento XBRL	46
3.4.3.1 O elemento group.....	47
3.4.3.2 O elemento item.....	47
3.4.3.3 O elemento context (numericContext e nonNumericContext).....	47
3.4.3.3.1 O atributo id.....	48
3.4.3.3.2 O atributo precision.....	48
3.4.3.3.3 O atributo cwa.....	48
3.4.3.3.4 O subelemento entity.....	49
3.4.3.3.4.1 O subelemento identifier.....	49
3.4.3.3.4.2 O subelemento segment.....	49
3.4.3.3.5 O subelemento period.....	50
3.4.3.3.6 O subelemento unit.....	50
3.4.3.3.7 O subelemento scenario.....	50
3.4.3.4 O elemento tuple.....	50
3.5 Outros aspectos dos documentos XBRL	51
4. Estudo de caso: Fluxo de informações no Banco Central do Brasil	53
4.1 Modelo atual.....	53
4.1.1 Ambiente interno de intercâmbio de informações.....	54
4.1.2 Ambiente de intercâmbio de informações com as instituições financeiras	55
4.1.3 Sítio do Banco Central	58
4.2 Modelo proposto.....	58
4.3 O impacto da aplicação de XBRL no Banco Central do Brasil	61
4.4 Ambiente do DEMAB	63
5. Estudo de caso: Representando relatórios financeiros em XBRL	66
5.1 Definição do documento e extração de dados.....	66
5.2 Taxonomia.....	68
5.3 Linkbases	69
5.4 Documentos XBRL	73
5.5 Folha de estilo	74

5.6	Aplicações para extração e utilização da informação em XBRL.....	78
6.	Implantando XBRL no Banco Central	80
6.1	Metodologia	80
6.2	Sugestões.....	82
7.	Conclusão.....	86
7.1	Contribuições	86
8.	Glossário	88
8.1	Glossário de termos financeiros	88
8.2	Glossário de termos de informática	90
9.	Referências Bibliográficas	94
10.	Apêndice A - XBRL Instance Schema	98
11.	Apêndice B - XBRL Linkbase Schema	102
12.	Apêndice C - Linkbases	105
13.	Apêndice D - Taxonomia - arquivo bcb_taxonomia.xsd	111
14.	Apêndice E - Documento XBRL - arquivo bcb_bp_28_02_2001.xml	113
15.	Apêndice F - Definição de atributos Xlink - arquivo Xlink.xsd	116
16.	Apêndice G - Definição de elementos Xlink - arquivo Xl.xsd	117
17.	Apêndice H – Folha de estilo – arquivo bcb_Instview.xsl	119
19.	Apêndice I - Folha de estilo - arquivo bcb_TxCompress.xsl.....	125
20.	Apêndice J - Códigos Java e JSP das aplicações.....	129
21.	Apêndice L - Resumo dos elementos e atributos XBRL.....	135

RESUMO

Caetano da Silva, Paulo. Explorando linguagens de marcação para representação de relatórios de informações financeiras. 2003. 141 f. Dissertação (Mestrado Profissional em Redes de Computadores), Universidade Salvador, Salvador.

Palavras-chave: Informações financeiras, Web, Internet, linguagem de marcação, XML, XBRL

O atual processo de intercâmbio de informações financeiras ocorre, geralmente, utilizando-se sistemas proprietários ou por meio da publicação na Web, o que impõem necessidade de esforços para a conversão de formatos de representação dos dados.

Este trabalho foi desenvolvido com o propósito de explorar a tecnologia de linguagens de marcação, com a proposição de marcas específicas para a preservação semântica dos documentos como solução alternativa para a padronização da representação de informações para relatórios financeiros (demonstrativos de informações financeiras), possibilitando a qualquer ente interessado o processamento dos relatórios de forma eficiente e eficaz.

Neste estudo foram analisadas linguagens financeiras derivadas de XML (eXtensible Markup Language), já que este é um padrão amplamente aceito para representação de dados. Entre as linguagens pesquisadas para a área financeira, XBRL (eXtensible Business Markup Language) foi a que mais se adequou ao desenvolvimento deste trabalho, pois, ao contrário de outras que são orientadas à transação financeira, seu principal objetivo é fornecer a estrutura necessária para a representação de informações para relatórios financeiros.

Como demonstração da viabilidade e vantagens na utilização de XBRL, foi realizado um estudo de caso, em que o intercâmbio de informações financeiras no ambiente do Banco Central do Brasil é analisado em todas as suas variações, ou seja, no contexto do relacionamento com as instituições do sistema financeiro nacional, com o público usuário de suas próprias informações disponíveis na Internet e internamente entre seus departamentos. Desta forma pôde-se verificar que a adoção de XBRL como padrão para representação de relatórios financeiros poderá estruturar a informação disponível na Web e facilitar a extração de dados e o processamento direto entre aplicações.

ABSTRACT

Caetano da Silva, Paulo. Explorando linguagens de marcação para representação de relatórios de informações financeiras. 2003. 141 f. Dissertação (Mestrado Profissional em Redes de Computadores), Universidade Salvador, Salvador.

Keywords: Financials informations, Web, Internet, Markup language, XML, XBRL

The exchange of financial information happens, usually, through systems proprietors or Internet. These methods now impose need of great efforts for systems conversion and data extraction.

This work was developed with the purpose of exploring the markup languages technology, with the proposition of specific marks for the semantic preservation of the documents as alternative solution for the financial information reports representation, making possible any interested being your processing with minimal computational effort in an efficient and effective way.

In this study derived financial languages of XML(eXtensible Markup Language) were analyzed, since this is thoroughly a pattern accepted for representation of data. Among the languages researched for the financial area, XBRL (eXtensible Business Markup Language) was adapted better to the development of this work, because, unlike others that are guided to the financial transaction, its main objective is to supply the necessary structure for the financial reports representation.

As demonstration of the viability and advantages in the use of XBRL, a sample was accomplished, where the financial information exchange is analyzed in all variations of the Central Bank of Brazil environment, in other words, in the context of the relationship with the National Financial System institutions, with the public user of your own available information in Internet and internally among your departments. Then could be verified that the adoption of XBRL as pattern for representation of financial reports can structure the available information in the Web and to facilitate the extraction of data and the direct processing among applications.

LISTA DE FIGURAS, TABELAS e ABREVIATURAS

FIGURAS

	Pagina
Figura 1.1 - Fluxo de informação - Desperdício de tempo e custo com transformações	02
Figura 1.2 - Escopo dos documentos XBRL.....	03
Figura 1.3 - Fluxo de informação com XBRL: reduz redirecionamento da informação e custo de transformações sucessivas de formatos.	04
Figura 3.1 – Escopo dos documentos de taxonomia XBRL, em relação à XML Schema e XML.	31
Figura 3.2 - Processo de criação e utilização de documentos XBRL.....	32
Figura 3.3 – Documentos envolvidos na criação da Taxonomia.	39
Figura 3.4 - Relacionamento dos arquivos usados na construção do documento XBRL....	46
Figura 3.5 - Esquema de Validação dos Documentos.	51
Figura 4.1 - Ambiente atual de intercâmbio de informações financeiras entre o Sistema Financeiro Nacional e o Banco Central do Brasil.....	54
Figura 4.2 - Ambiente de intercâmbio de informações financeiras entre o Sistema Financeiro Nacional e o Banco Central do Brasil, usando XBRL.....	61
Figura 4.3 - Dados recebidos pelo Banco Central por tipo de fonte.	62
Figura 4.4 - Modo de recebimento de dados pelo Banco Central.	62
Figura 4.5 - Tipo de registro de dados no Banco Central.	63
Figura 4.6 - Atual ambiente de intercâmbio de informações financeiras no DEMAB.	64
Figura 4.7 - Ambiente de intercâmbio de informações financeiras no DEMAB, usando XBRL.....	65
Figura 5.1 - Fluxo do processo da transformação do documento XBRL em HTML.....	76
Figura 5.2 - Exibição do documento XBRL, com a transformação XSLT.....	77
Figura 5.3 - Exibição do documento XBRL sem a transformação XSLT.	77
Figura 5.4 – Processamento automatizado dos documentos XBRL.....	78
Figura 5.5 - Exibição do resultado da aplicação usando-se XBRL/DOM/JAVA/JSP.	79

TABELAS

Tabela 2.1 – Comparação entre linguagens de marcação para informações financeiras..	24
Tabela 4.1 – Documentos enviados ao Banco Central por meio do aplicativo PSTAW10.	57

ABREVIATURAS

CSS – Cascade StyleSheet

DEINF – Departamento de Tecnologia da Informação do Banco Central do Brasil

DEMAB – Departamento de Mercado Aberto do Banco Central do Brasil

DEPIN – Departamento de Operações das Reservas Internacionais do Banco Central
do Brasil

DOM – Document Object Model

DTD – Document Type Definition

EDI – Eletronic Data Interchange

FinXML - Financial XML

FIXML – Financial Information eXchange Markup Language

FpML – Financial Products Markup Language

HTML – Hypertext Markup Language

HTTP – Hypertext Transfer Protocol

IAS – International Accounting Standards

IBM – International Business Machines

ISO – International Organization for Standardization

IP – Internet Protocol

JSP – Java Server Pages

OFX – Open Financial Exchange

SGML – Standard Generalized Markup Language

SNA – System Network Architecture

SPB – Sistema de Pagamentos Brasileiro

SSL – Secure Sockets Layer

STPML – Straight Through Processing Markup Language

swiftML – Society for Worldwide Interbank Financial Telecommunication Markup
Language

TAXML – Tax eXtensible Markup Reporting Language

TCP – Transfer Control Protocol

XHTML – extensible HTML

XML – extensible Markup Language

XSLT – eXtensible Stylesheet Language Transformation

W3C – World Wide Web Consortium

WWW – World Wide Web

1. Introdução

1.1. Contexto

Informações financeiras geralmente são armazenadas em diferentes formatos de banco de dados, documentos texto, planilhas, etc. Indubitavelmente estas informações precisam ser traduzidas de um formato para outro, já que os mais diversos tipos de aplicações e sistemas computacionais podem utilizá-las como fonte de dados. Os dados que estão armazenados nos mais variados formatos podem ser distribuídos por meio de relatórios para diversos fins. Informações contidas nestes relatórios podem ser redirecionadas para outros relatórios (páginas da Internet, relatórios externos e internos às organizações, relatórios de aplicações financeiras, etc.), assumindo outros formatos. Portanto, para a reutilização das informações é necessário, em geral, um redirecionamento e transformação dos formatos dos dados destes relatórios. Todo esse processo de transformação da informação financeira de um formato para outro faz com que organizações despendam esforços excessivos, o que pode provocar um incremento de custos para a utilização de uma mesma informação.

O processo de transformação e redirecionamento da informação ocorre não apenas internamente em uma organização, mas, também, quando o destino da informação é uma outra organização. Por exemplo, um sistema pode extrair a informação de uma planilha eletrônica, produzida em um departamento de determinada instituição, traduzí-la para gerar um relatório no formato de um documento texto, que é enviado a outro departamento, o qual faz uso interno da informação e extrai parte desta e a transforma em um documento HTML (Hypertext Markup Language), para então ser publicada na Internet. Digamos então, que uma outra organização qualquer tenha interesse nessa informação e que trabalhe com sistemas e formatos de dados diferentes dos da origem. Será, portanto, necessário que se faça uma extração não automática da informação presente na página da Internet ou que se solicite da organização, que contém a fonte das informações, o envio dos dados no formato desejado, podendo a demanda ser atendida ou não.

Na figura 1.1 pode-se observar como uma informação pode ser traduzida diversas vezes, como no caso daquela armazenada em uma planilha eletrônica que é traduzida para um formato de publicação em uma página na Internet e para um formato de um documento texto. Com a informação formatada como um documento texto, pode-se então ser processada e transformada para um formato específico, como por exemplo o de um órgão fiscalizador. Esta mesma informação, quando recebida no órgão fiscalizador pode, por sua vez, ainda ser

transformada em um outro formato utilizado internamente, ou seja, ocorrem pelo menos três transformações sucessivas de formatos.

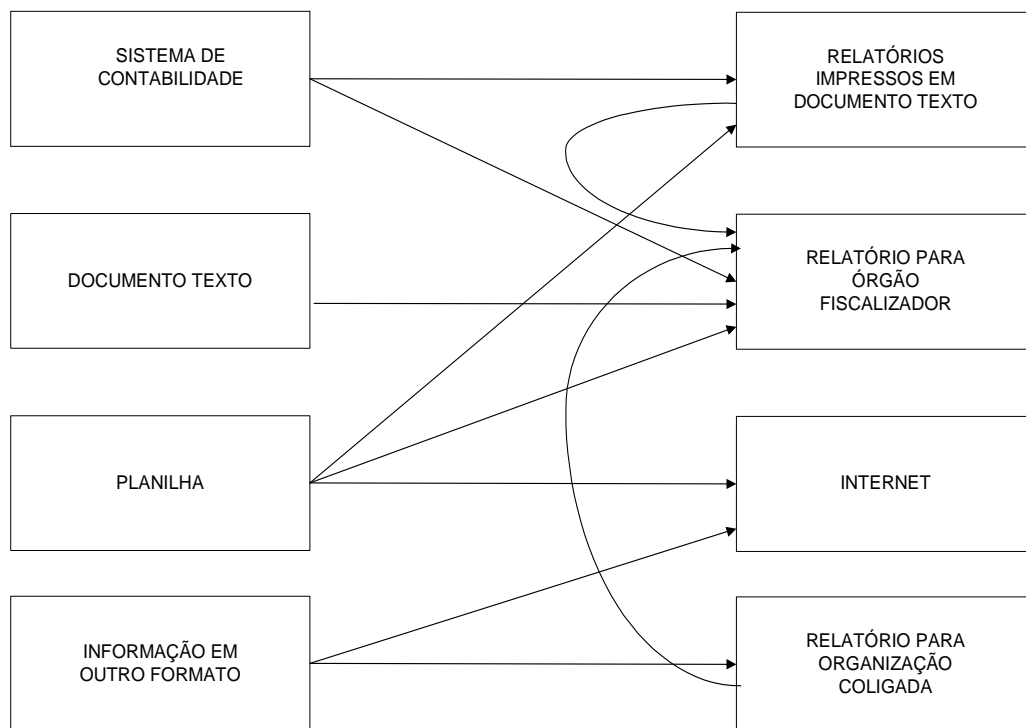


Figura 1.1 - Fluxo de informação - Desperdício de tempo e custo com transformações. [1]

Claramente, todo este processo de transformar a informação de um formato em outro pode introduzir muitas oportunidades de erro, requerer uma grande carga de trabalho e dispêndio de grande quantidade de recursos financeiros.

Um possível cenário para a redução da necessidade destas transformações é a utilização de uma linguagem padrão para o intercâmbio das informações financeiras. A conversão para um formato canônico poderá trazer benefícios como diminuição de custos, facilidade de extração e utilização dos dados etc. Com a padronização todo o processo de intercâmbio da informação fica facilitado, já que se todos os agentes interessados na informação usarem a mesma linguagem de marcação (*markup*) para representação dos dados, a necessidade de transformação de formatos será reduzida.

Neste cenário de padronização, o uso da Internet como meio de comunicação poderá trazer outros benefícios, como a democratização da informação, pois por ser a Internet largamente difundida e de baixo custo, o seu uso proporciona ampla publicidade da informação.

Como linguagem apropriada para a representação de dados, XML está a cada dia com sua utilização mais difundida, em razão das suas características que proporcionam facilidade de extração e utilização da informação. XML é uma metalinguagem que fornece estrutura para a criação de linguagens voltadas para o conteúdo da informação. Entre as linguagens estudadas, derivadas de XML para a área financeira, XBRL, um conjunto de regras que limita o escopo de XML para a disponibilização de relatórios financeiros, como pode ser observado pela figura 1.2, foi a que melhor se adequou ao objetivo deste trabalho. XBRL define uma estrutura que, com regras definidas por XML, permite a criação de linguagem própria para a construção de hiperdocumentos para a área financeira.

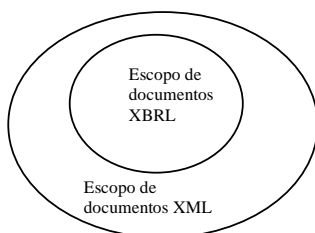


Figura 1.2 – Escopo dos documentos XBRL.

Usando XBRL como padrão, o processo de intercâmbio da informação financeira poderá tornar-se bastante simplificado, como pode ser visto pela figura 1.3, pois a modificação de formatos da informação original ocorre apenas uma vez para o formato XBRL e, então, a informação poderá ser reutilizada e distribuída automaticamente e de uma só vez para quaisquer outros formatos. Utilizando o mesmo exemplo anterior, onde os dados estão armazenados em uma planilha, pode-se realizar a conversão para o formato XBRL. E, por meio de folhas de estilo, ser produzido um documento texto ou de publicação na Web. Um órgão fiscalizador pode receber a informação em XBRL e armazená-la em seu banco de dados, alimentar seus sistemas, produzir seus relatórios, etc., com menos dispêndio de recursos, evitando-se excessivas conversões de formatos. Uma terceira parte interessada na informação disponível na Web, também, poderá usar a informação mais facilmente, pois se esta estiver representada usando XBRL, o seu processamento poderá ser facilitado, como será visto nos capítulos seguintes. Ou seja, sendo XBRL adotada como formato canônico para representação das informações financeiras, o percurso desta entre os diversos interessados ocorrerá de forma eficiente, com redução da necessidade de tradução seqüencial de formatos e, conseqüentemente, de erros e custos. Outra vantagem que incentiva a utilização de XBRL é que ela é um padrão aberto, ou seja, sua licença de uso é livre e sem custos.

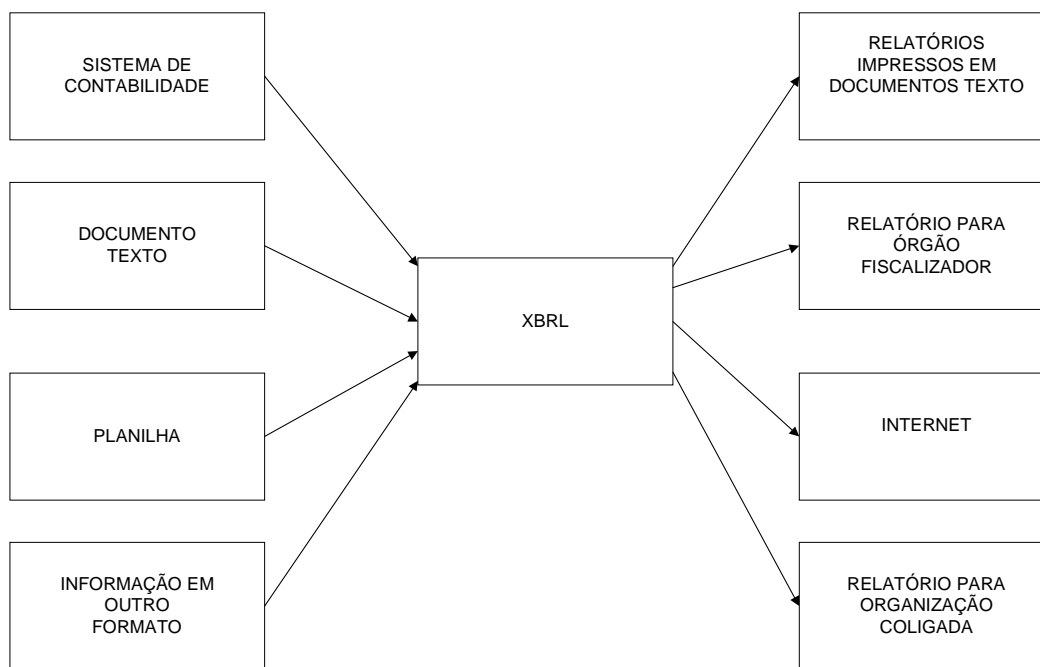


Figura 1.3 - Fluxo de informação com XBRL: reduz redirecionamento da informação e custo de transformações sucessivas de formatos [1].

1.2. Objetivo

O propósito desta dissertação é avaliar o uso de linguagens de marcação como possível forma de representação de dados para relatórios de informações financeiras, independente da plataforma computacional em que as informações estejam disponíveis. É proposta a utilização de XBRL por organizações que desejam disponibilizar suas informações financeiras na Web e para o intercâmbio de relatórios financeiros entre empresas parceiras ou com órgãos de governo.

Como estudo de caso foi utilizada uma situação específica do Banco Central do Brasil, com a proposição de uma taxonomia para representação de seus balanços patrimoniais no formato XBRL, para a comprovação da eficiência trazida com o seu uso. Neste estudo de caso, foi analisado o intercâmbio de informações financeiras no ambiente do Banco Central do Brasil em três vertentes, ou seja, no relacionamento com as instituições do sistema financeiro nacional, com o público usuário de suas informações disponíveis na Internet e internamente com o trânsito das informações entre seus departamentos.

1.3. Organização

Neste capítulo é feita uma explanação de como normalmente ocorre o intercâmbio de informações financeiras e como XBRL pode interferir no cenário da disponibilização de

relatórios de informações financeiras. Nos capítulos seguintes serão apresentados um pequeno histórico do surgimento das linguagens de marcação, comparação entre as linguagens derivadas de XML para a área financeira, detalhamento de XBRL e, por fim, o estudo de caso, em cuja aplicação prática o uso de XBRL demonstrou benefícios tais, que é vista como viável a sua adoção como padrão para o intercâmbio de relatórios de informações financeiras no ambiente computacional do Banco Central do Brasil.

No capítulo dois é feito um histórico das linguagens de marcação, com a apresentação de problemas que ocorrem hoje na Internet, partindo do surgimento de SGML (Standard Generalized Markup Language), uma metalinguagem que deu origem a outras linguagens de marcação, expondo-se as limitações de HTML, bem como a conveniência da utilização de XML, metalinguagem que proporcionou o surgimento de diversas linguagens de marcação específicas para determinada área do conhecimento, entre elas XBRL para a área financeira. É feita também uma descrição da origem das linguagens financeiras de marcação, de XML e de algumas de suas linguagens derivadas para a área financeira: FIXML (Financial Information eXchange Markup Language), swiftML (Society for Worldwide Interbank Financial Telecommunication Markup Language), STPML (Straight Through Processing Markup Language), FpML (Financial Products Markup Language), FinXML (Financial XML), OFX (Open Financial Exchange), TAXML (Tax eXtensible Markup Reporting Language) e XBRL (eXtensible Business Reporting Language).

No terceiro capítulo XBRL é tratada com mais detalhes. São destacados os seus benefícios, os motivos para usá-la, a sua área de abrangência e como deverá ser aplicada. Nesse capítulo é apresentada a definição da estrutura dos elementos XBRL usados para a elaboração da taxonomia, dos documentos *linkbases* e do documento XBRL. Na taxonomia são definidos os elementos a serem usados no documento XBRL, os *linkbases* descrevem os relacionamentos dos elementos criados na taxonomia e o documento XBRL é uma instância dos elementos criados na taxonomia, com seus valores e contextos.

O capítulo quatro discute como XBRL pode ser usada no contexto dos dados do Banco Central do Brasil, ou seja, na disponibilização de sua página na Web, no intercâmbio de informações com os agentes do Sistema Financeiro Nacional e na circulação da informação entre seus departamentos. No Banco Central existem duas unidades computacionais que funcionam de forma relativamente independente, o Departamento de Tecnologia da Informação (DEINF), que é responsável pelo sistema computacional de quase toda a organização, e o Departamento de Mercado Aberto (DEMAB) que possui um centro de

processamento independente do DEINF. Maior ênfase foi dada no capítulo quatro ao intercâmbio de informações no DEINF, porém também foi feita uma análise para o ambiente do DEMAB. Atualmente parte das informações prestada ao DEINF são por meio de uma rede SNA (System Network Architecture), utilizando o Sistema de Informações do Banco Central (Sisbacen), cuja plataforma computacional de grande porte é desenvolvida em linguagem de programação Natural e banco de dados Adabas. Este processo é custoso para as instituições do sistema financeiro, que são obrigadas a converter as informações presentes nos seus sistemas para o formato do Banco Central e manter estrutura física compatível com a rede SNA. Porém, também custa ao Banco Central, pois a utilização destas informações atualmente requer o desenvolvimento de programas em linguagem Natural, cuja estrutura de linguagem procedimental para computador de grande porte exige grande dispêndio de esforços para qualquer modificação ou manutenção, pois a reutilização de códigos é muito reduzida quando comparada com tecnologias de desenvolvimento de softwares mais modernas. A adoção de XBRL viabiliza a utilização de linguagens orientadas a objeto, como Java, cujos códigos desenvolvidos são reutilizáveis e portáteis a qualquer plataforma computacional, minimizando assim os custos de transformação da informação e manutenção de sistemas.

O ambiente computacional atual do Banco Central, que está sob controle do DEINF, requer diversas transformações de formatos para se utilizar modernas técnicas de Sistemas de Informação, como por exemplo *Datawarehouse*. No caso do *Datawarehouse* é necessário transpor todas as informações desejadas do banco de dados Adabas para o banco de dados relacional DB2, para que então se possam usar as ferramentas de gerenciamento da informação, existentes no mercado. Com o uso de XBRL, acredita-se que a transformação para os esquemas de banco de dados pode ser simplificada, pois a informação que for fornecida em XBRL poderá suprir diretamente os diversos bancos de dados existentes, independente da tecnologia destes, eliminando possibilidades de erros e facilitando a extração e utilização da informação.

O capítulo quinto demonstra a utilização prática de XBRL na disponibilização de documentos contábeis do Banco Central. Os documentos contábeis escolhidos foram balanços patrimoniais, por serem um típico documento que as instituições do sistema financeiro nacional são obrigadas a apresentar mensalmente ao Banco Central e por ser um tipo de documento base para qualquer análise financeira. Nos subitens desse capítulo é feita uma análise de cada elemento e atributos usados nos documentos XBRL construídos (taxonomia,

linkbases e o documento instanciado XBRL), fazendo-se, assim, uma complementação da discussão sobre a estrutura dos elementos XBRL iniciada no capítulo 3. Desta forma, foi desenvolvido um exemplo de uma taxonomia XBRL para representação do balanço patrimonial do Banco Central em hiperdocumentos, para que possam ser publicados na Web ou transitarem entre aplicações, exemplificando assim um determinado tipo de relatório de informação financeira. Também foram criadas uma folha de estilo, usando a linguagem XSLT (eXtensible Stylesheet Language Transformation) para publicação na Web, e uma aplicação, codificada em Java e DOM (Document Object Model), que extrai as informações do documento XBRL e armazena em um banco de dados, sendo esses dados posteriormente utilizados por outra aplicação para a criação de uma página da Web, desenvolvida em JSP (Java Server Pages). Assim, procurou-se demonstrar as vantagens da utilização de XBRL, não só no aspecto da apresentação visual para a Web, mas também para a criação de aplicações que possam processar as informações representadas nos hiperdocumentos.

No sexto capítulo é discutido o processo de implantação de XBRL para o intercâmbio de relatórios financeiros e sua publicação na Web. Além de ser sugerida uma metodologia e procedimentos a serem adotados pelo Banco Central.

Na Conclusão são feitas considerações sobre as vantagens do uso de XBRL para os usuários de informações financeiras e os benefícios esperados com a sua adoção pelo Banco Central.

Por fim, foram elaborados apêndices, de “A” até “L”, onde são apresentados os códigos XBRL da taxonomia, dos *linkbases*, do documento XBRL e da folha de estilo, em XSLT. No apêndice “J” estão os códigos, em Java e JSP, das aplicações criadas para processarem os documentos XBRL. E, finalizando, no apêndice “L” foram criadas três tabelas com um resumo dos elementos, seus atributos e funções, usados em cada tipo de documento: taxonomia, *linkbases* e o documento XBRL.

2. Linguagens de Marcação para o intercâmbio de informações financeiras na Internet

2.1 Linguagens de Marcação

O crescimento do uso da Internet, que pôde ser observado na última década, pode ser parcialmente atribuído à utilização de padrões amplamente aceitos, como o protocolo de rede TCP/IP e a linguagem de marcação HTML.

Embora HTML tenha tido grande aceitação como linguagem de publicação eletrônica, ela apenas descreve como um navegador Web (*browser*) deve organizar o documento em uma página. Por ser uma linguagem preocupada apenas com o formato de apresentação da informação, HTML possui algumas limitações - uma delas é a dificuldade de se criar uma página na Web que funcione não apenas para enviar documentos a qualquer um que o solicite. Pessoas e companhias desejam páginas com informações estruturadas, de forma eficaz e eficiente, para que possam facilmente ser extraídas, armazenadas e processadas. HTML não foi projetada para tal tarefa [2], ou seja, não possui nenhum elemento (*tag*) que informe o significado do dado, apenas formata a apresentação. Uma outra limitação de HTML é a inflexibilidade, adicionar um novo elemento a HTML é um processo bastante burocrático, senão impossível, que pode ser mais custoso do que a sua própria necessidade.

Como dito por Jon Bosak e Tim Bray [2], a solução, teoricamente, é muito simples: o uso de elementos que digam o que é a informação e não como ela deve ser mostrada. O W3C – World Wide Web Consortium (consórcio internacional que busca promover padrões na Web) começou a preparar tal solução em 1996. Porém essa idéia não foi completamente original. Linguagens de marcação foram desenvolvidas primeiro pela IBM e depois implementadas em larga escala dentro de companhias editoriais, as quais precisavam de meios para marcar um documento de forma que pudessem ser apresentadas em vários modos diferentes. Estes "marcadores" evoluíram até 1986, quando, depois de décadas de trabalho, a Organização para Padronização Internacional (ISO) [3] aprovou um sistema para a criação da linguagem de marcação, chamada de Standard Generalized Markup Language, SGML, uma linguagem para descrever linguagens - uma metalinguagem, ou seja, um conjunto de regras que definem como outras linguagens devem trabalhar - a qual tem provado ser útil, desde então, em muitas aplicações. Codificando um documento em SGML é possível reutilizar o texto em muitos formatos diferentes. Os autores, no caso das empresas editoriais, poderiam escrever uma vez, etiquetar com SGML e exibir a informação em muitos formatos distintos.

Como uma metalinguagem, SGML fornece as regras globais e procedimentos que permitem uma variedade grande de aplicações existirem. No entanto, um grande problema que é possível identificar em SGML é que ela é muito genérica - repleta de características inteligentes, projetadas para uma época onde todo byte tinha que ser levado em consideração [2]. SGML é extensivamente usada na publicação de documentos, porém provou ser muito complexa para larga adoção e uso na Web.

Na tentativa de solucionar as limitações de HTML e SGML, XML foi projetada eliminando complexidade e tornando possível a criação de elementos que possam descrever e estruturar a informação. Desde que XML foi adotada como padrão, no início de 1998 pelo W3C, esta linguagem vem expandindo-se por todos os campos do conhecimento humano [2].

XML é uma linguagem derivada de SGML, projetada como resultado direto do ganho de conhecimento das experiências com SGML e HTML. XML foi criada removendo-se características de SGML para se chegar a uma metalinguagem mais dinâmica. XML consiste de regras que qualquer um pode seguir para criar uma linguagem de marcação. Da mesma maneira que HTML possibilitou a todo usuário de computador ler documentos na Internet, XML torna possível se criar linguagens que permitam aos computadores extrair e processar as informações contidas em documentos.

Então, as linguagens de marcação tiveram as suas origens no início dos anos 60 [4], quando a IBM percebeu que existiam muitos diferentes tipos de sistemas que não se comunicavam entre si. A solução foi criar uma linguagem de marcação generalizada, uma série de códigos com os quais todos os sistemas poderiam compartilhar e usar para se comunicar. Durante os anos, esta linguagem de marcação evoluiu para várias linguagens diferentes, porém relacionadas entre si:

- Standard Generalized Markup Language: SGML foi desenvolvida entre 1978 e 1986. É um padrão internacional robusto (ISO 8879) usado em documentos de várias indústrias, como de publicação, aeroespacial, manufatura pesada e farmacêutica.

- Hypertext Markup Language: HTML foi desenvolvida por Tim Berners-Lee no começo de 1989. Ele chamou seu sistema hipertexto de World Wide Web, o qual se tornou o serviço mais usado na Internet. Porém, como documentos de Web evoluíram de páginas de exibição estáticas para aplicações interativas, as limitações de HTML ficaram rapidamente aparentes, isto é, “não há nada no texto HTML para facilitar que outros programas entendam a estrutura e o conteúdo dos dados exibidos” [5].

- Extensible Markup Language: XML resolve muito dos problemas de SGML e HTML. O W3C a adotou como um padrão formal em fevereiro de 1998. Embora compatível com SGML, é muito menos complexa. XML pode incorporar códigos e trabalhar bem com grande variedade de programas e plataformas de computação. É particularmente satisfatória para a Web, pode ser vista e editada com qualquer editor de texto.

2.2 Linguagens financeiras de Marcação

Intercâmbio de Dados Eletrônicos (EDI) é a troca de dados empresariais em formatos padronizados. EDI esteve presente de uma forma ou de outra durante mais de 30 anos na relação entre parceiros comerciais, ajudando a abrir caminho para a inovação do comércio eletrônico. Bem antes da Internet, companhias estavam trocando dados eletronicamente usando transações construídas para cada tipo de indústria. Muitas companhias de automóvel, supermercado e indústrias eletrônicas alcançaram significantes economias e melhoraram seus processos comerciais usando EDI [6]. Como aplicações tradicionais de EDI pode-se citar: ordens de compra, faturas, ordens de pagamentos etc. Porém, o desenvolvimento de padrões e o uso difundido de computadores encorajaram o uso de EDI em muitas novas áreas, inclusive na administração, em serviços financeiros e em outras transações através da Internet.

Com a evolução do EDI e das linguagens de marcação para a Internet, o padrão de comércio eletrônico mudou consideravelmente, introduzindo novos modos de comercializar e permitindo a interação entre grupos que previamente não poderiam comercializar uns com os outros [6]. Desta maneira, o desenvolvimento e utilização de XML para o atual intercâmbio de informações comerciais e financeiras, que requer interoperabilidade de aplicações de softwares, formatos e protocolos apropriados, proporcionam a facilitação do processamento da informação dentro e fora das organizações, bem como entre usuários individuais e diferentes softwares [7]. Portanto, em razão das suas características, XML está se tornando a linguagem base para a criação de linguagens de marcação para a troca de informações financeiras e de comércio eletrônico.

2.3 XML: uma retrospectiva

Segundo Tim Berners-Lee [8], o conteúdo da Web hoje é composto, em sua maioria, por documentos para que pessoas possam ler, pouca coisa é feita para que computadores possam manipular os documentos. Atualmente, computadores podem analisar gramaticalmente páginas da Web para definir o seu *layout* (localização do cabeçalho,

formatação da fonte do texto, identificação de um link para outra página, etc.), mas em geral não conseguem processar o conteúdo da página.

Uma solução alternativa está sendo construída com o que está sendo chamado de Web semântica, que trará estrutura para o conteúdo das páginas, criando um ambiente onde softwares possam navegar de uma página a outra e executar tarefas automatizadas. A Web semântica não será separada da Web atual, mas sim uma extensão, na qual é dado um significado bem definido aos elementos que contêm a informação.

Para a Web semântica funcionar bem, computadores terão que ter acesso à estrutura da informação e ao conjunto de regras que possam conduzir a um processamento automatizado. Uma tecnologia importante para atender estes requisitos desta nova Web é XML. XML permite, sob certas regras, que elementos sejam criados. Programas, então, podem usar esses elementos de diversas maneiras, desde que, os programadores conheçam o que cada elemento significa e as regras usadas na sua criação. Ou seja, XML permite que usuários adicionem estrutura a seus documentos, mas quem for usá-los terá que saber o significado dessa estrutura para que possa criar os programas que irão processá-los. A solução para a questão do significado dos elementos é fornecida por uma coleção de informações chamada de ontologia. Segundo Falbo [9], “o termo ontologia foi adotado da Filosofia. Já há muito tempo, filósofos têm usado ontologias para tentar descrever domínios naturais (as coisas naturais do mundo) e a existência dos seres e coisas em si. Entretanto, na Inteligência Artificial, ontologias têm um caráter um pouco distinto. Na Filosofia, ontologias são usadas com o intuito de desvendar o significado das coisas no mundo, procurando descrever a natureza das coisas. Na Inteligência Artificial, por sua vez, ontologias são usadas para descrever domínios já consagrados, como Medicina, Engenharia e Direito, onde é possível saber o significado projetado das coisas. Assim, o que se busca com uma ontologia em Inteligência Artificial é firmar um acordo sobre o vocabulário do domínio de interesse, a ser compartilhado por agentes que conversam sobre ele”. Uma típica ontologia na Web é a taxonomia, que define elementos e a relação entre eles [8].

Charlie Hoffman em um de seus textos [4] faz a seguinte citação sobre XML: “Talvez ela seja mais facilmente descrita analisando o seu próprio nome, Extensible Markup Language” – *Linguagem de Marcação eXtensível*, ou seja, *markup* significa que etiquetas (*tags*) identificam a informação e XML é uma metalinguagem de marcação que permite a criação de linguagens com elementos que portam o significado dos dados neles contidos. Já a característica de extensibilidade aparece em razão de ser uma metalinguagem usada para definir, para domínios específicos, estrutura e semântica de outras linguagens de marcação.

Uma informação representada em XML pode ser usada para publicação na Web ou para ser processada por aplicações. Para que isso ocorra pode-se usar até três tipos básicos de documentos. O arquivo contendo os elementos XML que envolvem os dados e dois arquivos opcionais auxiliares. O arquivo principal provê os dados, outro inclui informação da estrutura dos elementos e o terceiro fornece as instruções de apresentação. Cada um desses tipos de documento pode ser descrito assim:

- arquivo de dados XML: Este contém a informação do marcador. Por exemplo, um documento de ordem de compra deve conter elementos que forneçam a informação sobre aquele pedido, como o número do cliente, nome do cliente, número da ordem de compra, etc;

- Schema (ou DTD): Descreve a estrutura dos elementos usados no arquivo XML. Antes de rascunhar uma nova linguagem XML, os projetistas têm que concordar em três aspectos: quais elementos serão permitidos, como os elementos podem aninhar um dentro do outro e como eles devem ser processados. Esses aspectos - o vocabulário da linguagem e sua estrutura - são definidos tipicamente em uma Definição de Tipo de Documento, DTD, ou em um documento XML Schema. O padrão XML não obriga os projetistas de linguagem a usarem DTDs ou Schemas, mas a maioria das novas linguagens provavelmente fará uso, porque o Schema (ou DTD) facilita aos programadores escreverem softwares que entendam e processem o marcador [2];

- folha de estilo: fornece a informação de apresentação dos dados que estão contidos no primeiro documento. Podem ser usadas folhas de estilo diferentes para atender a mídias diferentes. Folhas de estilo são opcionais e geralmente são desnecessárias para transferência entre sistemas de computador, já que nesse caso apresentação não é importante.

Outras tecnologias baseadas em XML fornecem grande flexibilidade aos documentos. XML Link, recentemente recomendada pelo W3C, e XML Pointer permitem que documentos apontem para partes de outros documentos. Para isto, usa-se um identificador para o documento ou para parte dele [5]. Os *Links* em XML são elementos que possuem um atributo *xml:link* e outros atributos especificando a natureza da ligação. Com este processo de ligação a definição do vínculo entre um elemento, pertencente a um documento, com um outro documento qualquer, pode ser definida externamente aos dois documentos. Assim, é possível se fazer a vinculação sem alterar os documentos. Essa vinculação é feita por meio dos documentos de *Linkbase*.

XML pode ser aplicado em muitos problemas empresariais - manuseando arquivos de formatos diferentes, integrando aplicações, em transações empresariais, na administração da

informação, etc. A maioria da informação empresarial existe em documentos desestruturados e em bancos de dados proprietários. Embora estes repositórios de dados possam estar conectados à Internet, a informação pertinente não é necessariamente facilmente acessível. Um padrão de informação compreensivo permite às pessoas e aos computadores facilmente procurar, organizar, movimentar, exibir e personalizar a informação. A estrutura de elementos de XML ajuda alcançar esta meta. Para a área financeira existem algumas boas razões para se usar XML:

- é uma linguagem de marcação que codifica toda a informação, fazendo-a acessível por qualquer programa e plataforma;
- pode fornecer a informação do significado do dado, como por exemplo, o valor do Ativo Circulante em um balanço patrimonial, de que maneira este valor é calculado e como deve ser apresentado;
- pode automatizar muito do trabalho dos auditores, que têm que manipular quantias vastas de dados, permitindo que sistemas diferentes se comuniquem mais facilmente uns com os outros, dando ao auditor mais tempo para o trabalho analítico;
- não é propriedade de qualquer companhia, é uma linguagem aberta e disponível para todos. Padrões abertos são críticos para adoção de uma tecnologia nova, já que usuários não se tornam refém de uma companhia de software;
- padroniza o processo de representação da informação, o que é uma necessidade para que todo relatório de informações financeiras possa ser facilmente processado, não importando qual programa ou plataforma o use;
- a necessidade de distribuição de cópias da mesma informação em múltiplos formatos: em relatório impresso, para a Web ou para um sistema proprietário. Normalmente este processo requer um esforço manual demorado para transformar a informação de um relatório impresso para um formato utilizável em um meio digital. Com o uso de XML as instruções poderiam automaticamente ser codificadas para cada formato, economizando tempo e reduzindo o fator de erro humano.

Basicamente, XML fornece a estrutura para que uma simples transação contenha uma informação complexa, como processar regras para os dados. É flexível e extensível o suficiente para satisfazer as mudanças das variáveis de negócio, sem requerer significativo retrabalho de desenvolvimento [4].

Algumas linguagens derivadas de XML para a área financeira têm surgido, podendo ser citadas FIXML (Financial Information eXchange Markup Language), swiftML (Society

for Worldwide Interbank Financial Telecommunication Markup Language), STPML (Straight Through Processing Markup Language), FpML (Financial Products Markup Language), FinXML (Financial XML), OFX (Open Financial Exchange), TAXML (Tax eXtensible Markup Reporting Language) e XBRL (eXtensible Business Reporting Language). Sendo que XBRL possui um escopo não transacional, ou seja, não está voltada para transações financeiras e comerciais, incluindo apenas relatórios financeiros e contemplando extensivamente detalhes na representação e uso de convenções contábeis, enquanto que TAXML está focada na representação de impostos e as outras linguagens aqui referenciadas em transações financeiras de diversas modalidades. Nos próximos tópicos deste capítulo será feita uma breve descrição dessas linguagens XML para a área financeira, com suas características e áreas de atuação, para definição da que melhor se adapte ao desenvolvimento deste trabalho.

2.4 OFX - Open Financial Exchange

OFX é uma especificação para a troca eletrônica de dados financeiros por meio da Internet. A utilização de OFX agiliza o processo de intercâmbio de dados entre companhias de serviços financeiros e seus clientes, em páginas transacionais. É uma solução para as indústrias de serviços financeiros simplificarem o modo de troca de informações com consumidores individuais e pequenos negócios. OFX suporta aplicações para serviços financeiros como, por exemplo, os serviços de transações bancárias, transferência de fundos, sincronização de dados para bancos, serviços de pagamentos e investimentos [10].

Usando-se OFX o problema de conectividade como um obstáculo para o crescimento de serviços financeiros e bancários on-line pode ser eliminado. OFX também permite a companhias de serviços financeiros escolherem as plataformas, processos e sistemas de computação que desejem trabalhar. Isto faz aumentar a competitividade e reduzir os custos [11].

OFX é um padrão com orientação cliente-servidor para a Internet, o qual conecta diretamente o cliente e o servidor da instituição financeira, empregando um modelo *request/response* (pedido/resposta), proporcionando, assim, uma completa sincronização de dados e recuperação de erros. Neste modelo, um ou mais pedidos podem ser enviados em um arquivo, o qual inclui uma assinatura do pedido e um ou mais pedidos para um serviço específico, como pode ser observado no exemplo que se apresenta a seguir. Um servidor da instituição financeira deverá processar todos os pedidos e retornar um arquivo de resposta.

Ambos os arquivos (pedido/resposta) são formatados usando OFX como gramática baseada em XML. Um exemplo de um arquivo request, em que é feito um pedido de autenticação de um usuário, é mostrado logo abaixo, onde elementos são usadas para identificar o usuário, a instituição e os demais elementos necessários para identificar e se fazer o pedido de autenticação, de forma semelhante é o arquivo response [12] :

```

<OFX>
  <SIGNONMSGSRQV1>
    <SONRQ>
      <DTCLIENT>19991029101000 </DTCLIENT>
      <USERID>123-45-6789</USERID>
      <USERPASS> My password</USERPASS>
      <LANGUAGE>ENG</LANGUAGE>
      <FI>
        <ORG>NCH</ORG>
        <FID>1001</FID>
      </FI>
      <APPID>MyApp</APPID>
      <APPVER>0500</APPVER>
    </SONRQ>
  </SIGNONMSGSRQV1>
  <BANKMSGSRQV1>
    <STMTTRNRQ>
      <TRNUID>1001</TRNUID>
      <STMTRQ>
        <BANKACCTFROM>
          <BANKID>121099999</BANKID>
          <ACCTID>999988</ACCTID>
          <ACCTTYPE>CHECKING</ACCTTYPE>
        </BANKACCTFROM>
        <INCTRAN>
          <INCLUDE>Y</INCLUDE>
        </INCTRAN>
      </STMTRQ>
    </STMTTRNRQ>
  </BANKMSGSRQV1>
</OFX>

```

<!--Início dos dados de request -->
 <!--início da assinatura do request-->
 <!-- Oct. 29, 1999, 10:10:00 am -->
 <!-- ID do usuário -->
 <!--Password (com criptografia SSL)-->
 <!--Idioma para o texto -->
 <!-- ID da Instituição receptora -->
 <!-- Nome do ID -->
 <!--ID da instituição -->
 <!--fim da assinatura -->
 <!-- primeiro request no arquivo -->
 <!-- início do request -->
 <!-- Identificação da conta -->
 <!--outro ID da instituição -->
 <!--número da conta-->
 <!--tipo da conta-->
 <!--fim do ID da conta-->
 <!-- início da transação include-->
 <!-- transação Include-->
 <!-- fim da transação include-->
 <!-- fim do request-->
 <!-- fim do primeiro request-->
 <!-- fim dos dados de request-->

Este padrão incorpora padrões abertos largamente aceitos, como XML - para formatação de dados, TCP/IP e HTTP - para transporte via Internet e SSL - para segurança. A licença de uso da especificação é gratuita, permitindo qualquer desenvolvedor projetar uma interface para seus sistemas, além de suportar diferentes plataformas.

2.5 FIXML - Financial Information eXchange Markup Language e swiftML – Society for Worldwide Interbank Financial Telecommunication Markup Language

FIXML foi desenvolvido em 1998 pela FIX Protocol Ltd, companhia responsável pelo protocolo de comunicação FIX (Financial Information Exchange) para a troca de informações financeiras, como um vocabulário XML para facilitar o intercâmbio de transações de *securities* em tempo real [13], [14].

swiftML foi criada pela SWIFT, que é uma cooperativa fornecedora de serviços de mensagens para bancos, *dealers*, investidores, para infra-estruturas de mercado de pagamentos, tesouraria e *securities*. swiftML opera, assim como FIXML, na área de *securities* [15], [16].

Esses dois protocolos de comunicação operam em domínios bastante próximos, FIX no domínio de execução do pré-negócio/negócio e SWIFT no pós-negócio das operações de *securities*. Por esta razão em julho de 2001, a companhia FIX Protocol Ltd, anunciou os planos de juntar os esforços com a Society for Worldwide Interbank Telecommunications (SWIFT), que estava trabalhando com sua própria iniciativa, também baseada em XML, a swiftML. A iniciativa consiste na criação de uma versão, baseada em XML, da norma ISO-15022, protocolo para mensagens seguras, sendo desenvolvido pela ISO [14]. Assim a junção das diferentes partes, FIX no domínio de execução do pré-negócio/negócio e SWIFT no pós-negócio, poderá facilitar a integração de todo o ciclo de vida da transação, favorecendo o processamento direto da operação (Straight Through Processing – STP). Recentemente foi anunciada a utilização desta iniciativa em um programa piloto [17].

FIXML representa em XML um *tag* do formato FIX. Esse *tag* FIXML é então envolvido dentro de um cabeçalho tradicional FIX, com o objetivo de minimizar o impacto em implementações FIX existentes, requerendo apenas um analisador XML e uma ferramenta FIX para a comunicação [13].

O exemplo seguinte demonstra uma operação típica de *securities* representada usando FIXML [13], ou seja, uma operação em que é dado um título em garantia de um empréstimo ou financiamento. Nele pode ser observado os elementos que representam o cabeçalho, *<header>*, e a própria transação de *securitie*, *<ApplicationMessage>*, com os elementos que a identificam:

```
<FIXML>
  <FIXMLMessage>
    <Header>
      <Sender>
        <CompID>VENDOR</CompID>
      </Sender>
      <OnBehalfOf>
        <CompID>CUSTOMER</CompID>
        <LocationID>BOSTONEQ</LocationID>
      </OnBehalfOf>
      .
      .
      .
      <SendingTime>20000907-09:25:58</SendingTime>
      <PossDupFlag Value="Y" />
    </Header>
    <ApplicationMessage>
      <Order>
        <ClOrdID>ORD_1</ClOrdID>
        .
        .
        .
      </Order>
    </ApplicationMessage>
  </FIXMLMessage>
</FIXML>
```

Assim como OFX está fortemente focado no mercado de varejo, ou seja, nas transações com os clientes bancários, o objetivo de FIXML/swiftML é o mercado entre instituições.

2.6 STPML - Straight Through Processing Markup Language

STPML foi criada pela Financial Models Company (FMC), com o apoio da Microsoft, concebida como um padrão capaz de suportar outros padrões de intercâmbio de dados dentro da indústria de *securities* para o processamento direto (STP), entre eles o SWIFT e o FIX.

A base para o grupo de mensagens definidas pelo STPML surgiu do modelo de ciclo completo de negócio da Global Straight Through Processing Association, criado pela SWIFT. Esta base foi expandida para outras mensagens de transações comuns e mensagens para intercâmbio de dados de mercado, baseado na Market Data Markup Language (MDML), criada pelas empresas Bridge e Microsoft [18], [19].

O documento Schema que define os elementos da STPML pode ser encontrado na sua página oficial [18]. No entanto, as características dessa linguagem são semelhantes às de FIXML e por isso, com o aprimoramento da FIXML, a STPML provavelmente deverá ser incorporada em uma nova versão deste padrão [19].

2.7 FpML - Financial Products Markup Language

FpML é um padrão, baseado em XML, para atividades de comércio eletrônico no campo de derivativos financeiros. O padrão é de licença livre e automatiza o fluxo de informação entre os parceiros do derivativo e a rede do cliente, independente do software subjacente ou infra-estrutura de hardware que apóiam as atividades relacionadas a estas transações. FpML é bastante útil quando a comunicação direta das descrições do comércio de derivativos e informações do ambiente entre duas empresas é desejada, permitindo a integração eletrônica de uma gama de serviços de comércio eletrônico e confirmações para especificação da análise de risco [20], [21].

Com a adoção de FpML a padronização de vários processos de negócio ocorre, como por exemplo:

- termos de negociação e estruturação da transação;
- execução e confirmação da transação;
- detalhes da comunicação da transação;
- transação de análise de risco e preço.

Uma operação típica de derivativos financeiros - *swap*, que é uma operação de troca de títulos baseados em indicadores financeiros diferentes entre duas instituições, *Chase* e *Barclays*, representadas em FpML pode ser vista no seguinte exemplo a seguir [20], onde é mostrado o cabeçalho e o tratamento de datas importantes à transação:

```

<FpML version = "3-0">
  ...
  <tradeHeader>
    <partyTradeIdentifier>
      <partyReference href = "#CHASE" />
      <tradeId tradeIdScheme = "http://www.chase.com/swaps/tradeid">TW9235</tradeId>
    </partyTradeIdentifier>
    <partyTradeIdentifier>
      <partyReference href = "#BARCLAYS" />
      <tradeId tradeIdScheme = "http://www.barclays.com/swaps/trade-id">SW2000</tradeId>
    </partyTradeIdentifier>
    <tradeDate>1994-12-12</tradeDate>
  </tradeHeader>
  <swap>
    <!-- Chase pays the floating rate every 6 months, based on 6M DEM-LIBOR-BBA, on an ACT/360
    basis -->
    <swapStream>
      <payerPartyReference href = "#CHASE" />
      <receiverPartyReference href = "#BARCLAYS" />
      <calculationPeriodDates id = "floatingCalcPeriodDates">
        <effectiveDate>
          <unadjustedDate>1994-12-14</unadjustedDate>
          .
          .
          .
        </effectiveDate>
        <terminationDate>
          <unadjustedDate>1999-12-14</unadjustedDate>
          <dateAdjustments>
            <businessDayConvention>MODFOLLOWING</businessDayConvention>
            <businessCenters id = "primaryBusinessCenters">
              <businessCenter>DEFR</businessCenter>
            </businessCenters>
          </dateAdjustments>
        </terminationDate>
        .
        .
        .
      </calculationPeriodDates>
      <paymentDates>
        <calculationPeriodDatesReference href = "#floatingCalcPeriodDates" />
        <paymentFrequency>
          .
          .
          .
        </paymentFrequency>
        <payRelativeTo>CalculationPeriodEndDate</payRelativeTo>
        <paymentDatesAdjustments>
          .
          .
          .
        </paymentDatesAdjustments>
      </paymentDates>
      <resetDates id = "resetDates">
        .
        .
        .
      </resetDates>
      <calculationPeriodAmount>
        .
        .
        .
      </calculationPeriodAmount>
    </swapStream>
    <!-- Barclays pays the 6% fixed rate every year on a 30E/360 basis -->
    <swapStream>
      .
      .
      .
    </swapStream>
  </swap>
</trade>
<party id = "CHASE">
  <partyId>CHASUS33</partyId>
</party>
<party id = "BARCLAYS">
  <partyId>BARCGB2L</partyId>
</party>
</FpML>

```

2.8 FinXML - Financial XML

FinXML é uma linguagem baseada na estrutura de XML, desenvolvida para suportar um padrão para intercâmbio de dados dentro de Mercados de Capital. FinXML age como um padrão para aplicação de intercâmbio cruzado de informações, permitindo que uma instituição

financeira comunique os detalhes de transações financeiras em formato eletrônico. Como tal, FinXML pode ser usada como a base para processamento direto e análise de risco dentro de uma instituição financeira. Desta forma, FinXML sobrepõe esforços de desenvolvimento de soluções para o intercâmbio eletrônico de transações financeiras como o STPML e FIXML [22], [23], [24].

FinXML é uma importante especificação para interoperabilidade de sistemas em mercados de capital. A FinXML foi desenvolvida em conjunto com padrões definidos pela ISDA (International Swaps and Derivatives Association, Inc.), que já são extensamente usados na indústria financeira. FinXML é interoperável com protocolos financeiros existentes como FIX e SWIFT. Além disso, FinXML é compatível com padrões externos de indústria como o BizTalk da Microsoft e o cXML da Ariba que suportam comércio eletrônico e EDI.

FinXML é baseada em uma estrutura XML, dentro da qual podem ser definidos vocabulários para mercados de capital, onde aplicações podem ser desenvolvidas para processar as informações por eles representadas. Como tal, FinXML representa a base de uma solução completa para comércio eletrônico dentro de uma instituição financeira e entre bancos e os seus clientes.

Vocabulários definidos baseados em FinXML suportam um grande grupo de elementos e atributos que representam transações financeiras, dados de referência, dados de mercado, pagamentos, determinações e confirmações. FinXML é aplicável a vários produtos financeiros, como taxa de juros, câmbio, derivativos, *commodities*, empréstimos e depósitos, mercado futuro e de opções.

Assim como OFX, FinXML é uma linguagem para interação entre usuários e serviços financeiros. Entretanto, ao contrário de OFX, cujo campo de atuação são as operações bancárias mais corriqueiras, FinXML fornece suporte para transações de produtos estruturados, que são complexos e requerem linguagens e sistemas significativamente mais sofisticados.

2.9 TAXML - Tax eXtensible Markup Reporting Language

TaXML é uma iniciativa dedicada ao projeto e desenvolvimento de um padrão para troca de informações, baseado em XML, específicas para o domínio de impostos.

A grande quantidade de participantes e inerente complexidade do processo de submissão de impostos incitou o desenvolvimento do padrão que poderá ajudar os contribuintes, agências de governo, provedores de solução, etc., estabelecendo um

vocabulário para intercâmbio de informações dirigido a este domínio. TaXML é definida pela colaboração de partes interessadas, como os contribuintes, vendedores de software e provedores de solução, agências de governo e instituições financeiras [25], [26].

TaXML está focalizada na representação de elementos de informação de impostos e suas interdependências. Os elementos usados podem ser de outros padrões e a definição de estruturas pode variar para tipos de imposto diferentes, tipos de mensagens, jurisdições, etc.

Fases diferentes da submissão de imposto como avaliação, coleta, informação e pagamento, requerem diferentes informações a serem agregadas e transmitidas. TaXML pode representar todas estas fases de uma maneira uniforme, simplificando o processo, aumentando a automatização e minimizando erros e ambigüidades.

Esta é uma especificação em desenvolvimento, porém já pode ser observado como deverá ser a estrutura de um documento TAXML observando-se os elementos no trecho de código a seguir [25], que descrevem detalhes sobre o contribuinte, autoridade e demais informações relativas ao imposto:

```
<?xml version="1.0" encoding="UTF-8" ?>
<TaxReturn
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="TaxReturn_TaXML.xsd" type="String"
  purpose="Original">
  <taxpayer.details>
    <TaxpayerId type="String">Text</TaxpayerId>
    <taxpayerParty type="String" description="String">
      <person.details>
        <person.name>
          <FormattedName type="presentation">String</FormattedName>
          .
          .
          .
        </person.name>
        <person.gender.code>Text</person.gender.code>
        <birth.date>Text</birth.date>
      </person.details>
    </taxpayerParty>
    <taxpayerAddress>
      <address.type.code>Text</address.type.code>
      .
      .
      .
    </taxpayerAddress>
  </taxpayer.details>
  <TaxAuthority type="String">
    <taxAuthority.identifier>Text</taxAuthority.identifier>
    <taxAuthority.name>Text</taxAuthority.name>
  </TaxAuthority>
  .
  .
  .
  <Preparer type="String" description="String">
    <person.details>
      <person.name>
        <FormattedName type="presentation">String</FormattedName>
        .
        .
        .
      </person.name>
      <person.gender.code>Text</person.gender.code>
      <birth.date>Text</birth.date>
    </person.details>
  </Preparer>
  <FilingDate>1967-08-13</FilingDate>
  <FilingTime>14:20:00-05:00</FilingTime>
</TaxReturn>
```

2.10 XBRL - eXtensible Business Reporting Language

XBRL é uma especificação em conformidade com a recomendação XML 1.0. XBRL é uma linguagem especificamente desenvolvida para a área de relatórios financeiros.

XBRL foi desenvolvida para a preparação e intercâmbio de dados financeiros, tendo como objetivo fornecer uma estrutura baseada em XML para uso na criação, intercâmbio e análise de demonstrações financeiras, especificamente para a área contábil, incluindo, mas não se limitando a demonstrativos contábeis, análises de auditoria, etc [27]. Podendo, também, ser utilizada para a disponibilização de taxas de câmbio, de juros, ou até mesmo publicação de notas explicativas ou normativos por partes de agências governamentais.

XBRL teve sua origem no início de 1998, quando se começou a investigar a utilização de XML para o intercâmbio eletrônico de relatórios de informações financeiras. Inicialmente chamada de XFRML (eXtensible Financial Reporting Markup Language) e posteriormente XBRL, em meados de 1999 com a criação do consórcio para o desenvolvimento da especificação. Em julho de 2000, foi lançada oficialmente a sua primeira versão. A partir daí, com a experiência adquirida e com o surgimento de novas tecnologias, como XML Schema e XML Link, aperfeiçoamentos ocorreram, os quais resultaram no lançamento da segunda versão em dezembro de 2001[28].

XBRL define uma estrutura na qual a comunidade interessada em representar seus relatórios financeiros pode definir seus próprios elementos. Isto ocorre porque um fato financeiro que ocorre em uma indústria, por exemplo, na área bancária, pode não ocorrer em outro tipo de indústria. Assim, XBRL fornece a estrutura para criação de elementos para a representação personalizada de informações financeiras.

Um exemplo típico de um documento XBRL pode ser visto a seguir [28], com a definição dos *namespaces* [29], espaços de nomes, onde estão definidos os elementos e seus relacionamentos, e dos elementos que irão representar o relatório financeiro, no caso um demonstrativo de balanço patrimonial. O documento também apresenta um elemento *<footnoteLink>*, baseado na tecnologia XML LINK [30], que cria um arco denominado “nota de rodapé” para expressar os efeitos que não podem ser estruturados no documento, por exemplo uma fusão de empresas, e, por fim, um elemento, *<numericContext>*, que define o contexto em que o documento é inserido:

```

<?xml version="1.0"?>
<group xmlns=http://www.xbrl.org/2001/instance"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ias="http://www.iasb.org.uk/xbrl/2001-08-16/"
  xmlns:link=http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:iso4217=http://www.iso.org/4217
  xsi:schemaLocation="http://www.iasb.org.uk/xbrl/2001-08-16/ ias.xsd">
  <ias:bs>
    <ias:asset-cce-net numericContext="c1">100</ias:asset-cce-net>
    <ias:asset-rec-net numericContext="c1">700</ias:asset-rec-net>
    . . .
    <ias:eq-total id="f3" numericContext="c1">1100</ias:eq-total>
    <ias:liab-total id="f2" numericContext="c1">2600</ias:liab-total>
  </ias:bs>
  <link:footnoteLink xlink:type="extended" xlink:role="Merger" xlink:title="1">
    <link:footnote xlink:type="resource" xlink:label="footnotel" xlink:title="1"
      link:role="standard" xml:lang="en">Including the effects of the
      merger.</link:footnote>
    <link:footnote xlink:type="resource" xlink:label="footnotel" xlink:title="1"
      xlink:role="standard" xml:lang="fr">Y compris les effets de la
      fusion.</link:footnote>
    <link:loc xlink:type="locator" xlink:label="fact1" xlink:href="#f1"/>
    <link:loc xlink:type="locator" xlink:label="fact1" xlink:href="#f2"/>
    <link:loc xlink:type="locator" xlink:label="fact1" xlink:href="#f3"/>
    <link:footnoteArc xlink:from="fact1" xlink:to="footnotel"
      xlink:title="fact1-footnotel"
      xlink:arcrole="http://www.xbrl.org/linkprops/arc/fact-footnote"
      xlink:show="replace" xlink:actuate="onRequest"/>
    <link:footnoteArc xlink:from="footnotel" xlink:to="fact1"
      xlink:title="footnotel-fact1"
      xlink:arcrole="http://www.xbrl.org/linkprops/arc/footnote-fact"
      xlink:show="replace" xlink:actuate="onRequest"/>
  </link:footnoteLink>
  <numericContext id="c1" precision="18" cwa="true">
    <entity>
      <identifi er scheme="http://www.un.org/">Example plc</identifi er>
      <segment/>
    </entity>
    <period>
      <instant>2001-08-16</instant>
    </period>
    <unit><measure>iso4217:EUR</measure></unit>
    <scenario name="Actual values">
      <ias:scenarioType>actual</ias:scenarioType>
    </scenario>
  </numericContext>
</group>

```

2.11 Comparando as linguagens financeiras de markup

Das linguagens citadas, todas baseadas em XML, observou-se que algumas são voltadas à representação e ao processamento das transações financeiras, enquanto XBRL tem como propósito a representação de relatórios financeiros:

- OFX está fortemente focada nas transações entre o cliente e a instituição financeira, em um modelo *request/response*;

- FIXML, desenvolvida para representação do protocolo FIX, está orientada à indústria de *securities* e SwiftML, incorporando-se à FIXML para o processamento completo das transações de *securities*;

- STPML possibilita a inclusão de outros padrões de intercâmbio de dados dentro do ambiente de *securities* para o processamento direto (Straight Through Processing), entre eles o SWIFT e o FIX;

- FpML é um padrão para transações de comércio eletrônico no campo de derivativos financeiros, cujo foco inicial foi em *swaps*, mercado futuro e câmbio, porém todas as categorias de derivativos já são incorporadas a esta linguagem;

- FinXML é uma linguagem desenvolvida para intercâmbio de dados dentro de Mercados de Capital, permitindo uma comunicação eletrônica bem estruturada dos detalhes da transação financeira. Assim é possível ser usada como base para o processamento direto e análise de risco;

- TAXML é um padrão focado em todo o processo de submissão de impostos como avaliação, coleta, informação e pagamento. Por esta razão pode incorporar outros padrões, que atuam no ambiente das transações financeiras, para compor o imposto.

Percebe-se que os padrões citados acima, além de terem características transacionais, são em sua maioria concorrentes, atuando na mesma área. Por exemplo, FinXML engloba as áreas de atuação dos outros padrões, exceto de OFX, cujo campo de atuação é o relacionamento entre o cliente e seu banco, e TAXML, que é orientado às transações relativas a impostos.

Em contraste aos padrões expostos acima, o escopo de XBRL não está na relação de transações financeiras. XBRL é uma linguagem peculiar, pois propicia o intercâmbio de informações em relatórios financeiros. Pode-se dizer que sua atuação é na área de *e-reporting*, pois fornece condições para a preparação, publicação em uma variedade de formatos, extração de forma confiável e automaticamente fazer o intercâmbio de demonstrações financeiras, especificamente na área de auditoria contábil. Na tabela 2.1 é feita uma comparação, mostrando as características de cada linguagem analisada.

Portanto, das tecnologias pesquisadas para a área financeira, XBRL foi a que apresentou melhores características para a disponibilização de relatório de informações financeiras. As outras especificações aqui citadas, além de serem orientadas à transação, são específicas para determinada área financeira, como derivativos, mercados de capital, impostos etc. Desta forma, em razão do escopo de XBRL ser no intercâmbio de informações baseadas em relatórios financeiros, este padrão é assumido como o de melhor aplicabilidade no desenvolvimento deste trabalho.

No capítulo seguinte XBRL será tratada com maiores detalhes, analisando-se as vantagens na sua utilização para a representação de relatórios financeiros e a estrutura dos documentos.

Tabela 2.1 – Comparação entre linguagens de marcação para informações financeiras

	APLICAÇÕES	CARACTERÍSTICAS
OFX	- Intercâmbio de informações financeiras entre bancos e clientes (serviços de transações bancárias, transferência de fundos, sincronização de dados para bancos, serviços de pagamentos e investimentos)	- Padrão com orientação cliente-servidor , cujo principal aspecto é uma conexão direta entre o cliente e o servidor da instituição financeira, empregando um modelo request/response, o que caracteriza uma completa sincronização de dados e recuperação de erro. - Há uma comunidade grande e crescente de corretoras, bancos, dealers, companhias de software, processadores e provedores de solução de tecnologia apoiando e implementando OFX
FIXML	- Transações de <i>securities</i> - Representa em XML o protocolo FIX	- Desenvolvido em 1998 pela FIX Protocol Ltd, companhia responsável pelo protocolo FIX de comunicação para informações financeiras. - Tendência para juntar os esforços com a SWIFT para criar uma versão baseada em XML da ISO-15022 (protocolo de mensagens seguras), FIX no domínio de execução do pré-negócio/negócio e SWIFT no pós-negócio, favorecendo o Straight Through Processing.
swiftML	- Opera na área de <i>securities, dealer, compensação, liquidação, etc.</i> - Representa em XML o protocolo SWIFT	- Criada pela SWIFT, organização responsável pelo protocolo swift de comunicação para informações financeiras - Padrões de negócio definido em UML e sintático em XML
STPML	- Processamento direto dentro da indústria de <i>securities</i>	- Modelo de ciclo completo de negócios da Global Straight Through Processing Association da SWIFT. - Suporta outros padrões como o FIX e SWIFT, solucionando problemas de sobreposição. - Tendência de ser incorporada em uma nova versão do FIXML
FpML	- Intercâmbio de informações para derivativos financeiros. - Foco em <i>swaps</i> , mercado futuro, câmbio.	- Baseado numa estrutura XML desenvolvida para suportar um único padrão universal para troca de dados dentro do mercado de derivativos - Para a transação inclui termos de negociação e estruturação; execução e confirmação; detalhes da comunicação; análise de risco e preço.
FinXML	- Intercâmbio de informações para mercados de capital.	- Desenvolvida junto com padrões definidos pelo ISDA (International Swaps and Derivatives Association, Inc.), que já estão sendo extensamente usados na indústria financeira. - Interoperável com protocolos financeiros existentes, como FIX e SWIFT e padrões externos de indústria, como o BizTalk de Microsoft e o cXML de Ariba. - Envolvimento da SUN e Chase Manhattan. - Suporta uma grande variedade de produtos financeiros inclusive taxa de juros, câmbio, derivativos, <i>commodities</i> , empréstimos e depósitos, e mercados futuros e de opções.
TAXML	- processo de submissão de impostos - foco na representação da informação de imposto e suas interdependências	- Os elementos usados podem ser de outros padrões. - Definição de estruturas pode variar para tipos de imposto diferentes, tipos de mensagens e jurisdições. - Engloba fases diferentes da submissão de imposto como avaliação, coleta, informação e pagamento.
XBRL	- Intercâmbio de relatórios financeiros, principalmente na área contábil.	- O escopo do XBRL inclui relatórios financeiros e contempla detalhes na representação e uso de convenções contábeis, é orientado a relatórios, o que o distingue dos outros trabalhos que são orientados a transações. - Suporta outros padrões internacionais de transações financeiras e outros idiomas que não sejam o inglês. - Permite análises automatizadas, recebimento da informação no formato de preferência para o estilo específico da análise. - A Microsoft e grandes grupos da área financeira e governamental, espalhados pelo mundo, fazem parte do consórcio para adotar e implementar XBRL - Discussões ocorrem com outros grupos de especificações XML na área financeira: OAG(Open Applications Group), OMG (Object Management Group), FpML (Financial Products Markup Language), finXML (Fianancial XML), OFX/IFX (Open Financial Exchange) e ebXML (e-business XML).

3. XBRL – Extensible Business Reporting Language

3.1 Motivos para se usar XBRL

O crescimento da Internet não só está mudando a maneira como as companhias conduzem seus negócios, mas também promete alterar o modo como elas comunicam o desempenho de seus negócios. Com investidores exigindo acesso instantâneo às informações de negócios e agentes governamentais reguladores que pressionam por maior lisura e transparência destas informações, a efetiva informação dos relatórios financeiros na Web será fundamental em um futuro próximo. Disponibilidade de informação poderá determinar quais companhias se beneficiam do mercado com perspicácia e quais estão ferindo os seus princípios.

Um dos benefícios ainda não realizados pela Internet é a habilidade para automatizar informações de negócio. Estruturação de dados para relatórios financeiros ajudará a incrementar a eficiência e baixar os custos e riscos para companhias. Assim como *e-commerce*, o *e-reporting* requer padrões de tecnologia com os quais participantes possam trocar dados eletronicamente, tanto internamente como externamente às organizações. XBRL é um padrão para representar e intercambiar relatórios de informação financeira, incluindo demonstrações contábeis, informações para agências governamentais reguladoras e para investidores.

Normalmente os dados provenientes de relatórios de informações financeiras são reentrados para interpretação em aplicações de computador. XBRL poderá solucionar este problema permitindo que o conteúdo do relatório seja criado uma vez e distribuído de uma só vez pela Web e entre aplicações. A adoção de XBRL como padrão para representação de informações financeiras facilitará a criação e distribuição de relatórios financeiros por qualquer entidade.

Por ser uma tecnologia XML voltada para a disponibilização de relatórios financeiros, XBRL poderá representar o vocabulário padrão para o intercâmbio de relatórios de informações financeiras. Companhias poderão usar o padrão XBRL para desenvolver diversos tipos de relatórios financeiros, como por exemplo [31]:

- .XBRL para demonstrações financeiras;
- .XBRL para livro-razão;
- .XBRL para prestação de informações de Impostos;
- .XBRL para órgãos fiscalizadores e reguladores;
- .XBRL para relatórios de linhas de crédito;
- .XBRL para indicadores econômicos;

.XBRL para relatórios de risco;

XBRL permite aos usuários obterem melhorias significantes na utilização de dados eletrônicos, proporcionando capacidade para exibição dos dados e fornecendo a informação sobre o contexto em que foram criados. A habilidade para exibir e simultaneamente revelar o contexto da informação financeira é uma das motivações primárias de XBRL.

Em XBRL são adicionados elementos a cada segmento de informação financeira. Este modelo de identificação fornece ao computador o significado dos dados. Assim, dois problemas significantes para os usuários e preparadores de demonstrações financeiras podem ser resolvidos, fornecendo uma preparação e extração eficiente dos dados financeiros. Com a utilização de XBRL a informação financeira precisa apenas de uma entrada de dados, para, então, ser distribuída em múltiplos formatos, reduzindo o risco de erros pela eliminação do sucessivo redirecionamento da informação. Por exemplo, tradicionalmente uma informação pode ser enviada como um documento em um formato de qualquer editor de texto, os quais apesar da conveniência para o leitor, para um computador torna-se uma tarefa não trivial a extração da informação contida nestes formatos. Porém se a informação estiver em XBRL, uma aplicação poderá facilmente decifrar a informação e colocá-la no formato desejado, já que XBRL é uma linguagem projetada para representar dados em elementos que descrevem o significado da informação.

XBRL oferece vários benefícios como independência de tecnologia, interoperabilidade, preparação eficiente de relatórios financeiros e facilidade de extração dos dados.

Os aspectos chaves da tecnologia XBRL são (a) contexto, (b) acesso universal e (c) extensibilidade [31].

(a) Contexto:

No âmbito de domínio/jurisdição, existe a informação em qualquer documento XBRL dos padrões de contabilidade do país para o qual este foi preparado [31], por exemplo, o IAS (*International Accounting Standards*), o US GAAP (*US Generally Accepted Accounting Principles*), padrões da Espanha, Reino Unido, Alemanha, Brasil, etc. Supondo uma aplicabilidade universal de XBRL, a noção de se ter vários padrões contábeis, principalmente em uma economia globalizada, poderá tender a convergir para um único. Desta forma, os custos de conversão entre padrões, que não são pequenos, poderão ser diminuídos, apenas por se dispor da informação representada em elementos e de acordo com o modelo conceitual de cada padrão contábil.

Uma outra vantagem, no que diz respeito ao contexto, é que relatórios poderão ser facilmente moldados à sua utilização, por exemplo, relatórios financeiros que são fornecidos ao governo podem ser completamente diferentes da informação de interesse dos investidores.

(b) Acesso universal:

Sendo uma tecnologia baseada em XML, as informações em XBRL podem ser produzidas para qualquer tipo de mídia - Web, impressão ou e-book. Podendo ser exportada ou importada por qualquer sistema financeiro/contábil.

A utilização de XBRL poderá transformar a produção impressa de relatórios para documentos digitais. Então, novos negócios e serviços poderão surgir com características como integridade, permissões para uso das informações, uso em diversos tipos de mídia, personalização, rastreamento da informação, rápida distribuição, etc.

(c) extensibilidade:

Um dos benefícios de XBRL é ser extensível, isto é, XBRL permite a criação de novos elementos para descrever novos e imprevisos campos de mensagens. Isto significa que novos elementos de dados personalizados podem ser criados, contanto que eles adiram à especificação XBRL e ao padrão de elementos definidos pelo Consorcio W3C. Por exemplo, no Brasil a classificação da atividade econômica é dividida em três setores: primário, secundário e terciário, e dentro destes há subdivisões. Isto implica que personalizações devem ocorrer, a depender da área de atuação da indústria, ou seja, o setor bancário tem características diferentes do setor de manufatura, o que implicará em elementos distintos.

A utilização de XBRL possibilitará uma comunicação mais eficiente, porém, o controle da revelação da informação ainda será feito pelo contador, ou outro produtor do relatório financeiro. Ou seja, XBRL não prescreve como, por exemplo, ativos são determinados, somente etiqueta a informação para que possa ser processada por aplicações. XBRL não dita como se faz o relatório financeiro. Por exemplo, uma companhia pode querer criar o seu próprio conjunto de elementos, para relatórios internos. Com XBRL, o departamento financeiro pode personalizar a informação para cada executivo, para cada métrica, para cada análise desejada. Se a informação tiver que ser exportada para um ambiente externo, como para investidores, o departamento financeiro da empresa poderá analisar a informação interna e informar somente o que for interessante ser informado. Toda a informação poderá estar em XBRL, mas como qualquer tecnologia, as pessoas é que são responsáveis pela forma como ela é usada.

No setor público a utilização de XBRL poderá trazer benefícios a economia do país, ou seja, se as informações prestadas pelos agentes financeiros da economia para as agências

governamentais forem em XBRL, as análises de risco, comparação de desempenhos, evolução dos parâmetros de análise da economia, poderão ser feitas de forma mais eficiente e simples. E se toda a informação financeira transitada na economia estiver em XBRL, tanto internamente às organizações, como para as informações externas, certamente o funcionamento da economia do país será bem mais transparente do que ocorre hoje, o que, sem dúvida, proporcionará maior credibilidade às instituições e conseqüentemente maior investimento, pois quando os setores público e privado tornam-se mais transparente e eficiente, a burocracia, mau gerenciamento e fraude decrescem.

Com a globalização dos setores privado e público, a responsabilidade do setor público tem crescido, requerendo um papel mais importante de organizações globais de monitoramento como FMI (Fundo Monetário Internacional), BIS (Banco de Compensações Internacionais) e BIRD (Banco Mundial), o que certamente, implicará a exigência de que os países em desenvolvimento adotem sistemas e controles que estejam alinhados com a capacidade da Internet, proporcionando uma maior transparência nas finanças do país. E, quando uma economia é mais transparente, a estabilidade econômica é favorecida, dando maior credibilidade ao país e facilitando, em caso de necessidade, o suporte financeiro, inclusive com menores taxas de juros. Os países emergentes, que já possuem infra-estrutura de telecomunicações para adotarem qualquer tecnologia para a Internet, poderão com a adoção de XBRL tornar suas economias mais ágeis e transparentes, facilitando, assim, o aporte de novos recursos. No entanto, os benefícios provenientes da aplicação de XBRL à economia de um país só terão fortes impactos positivos se ambos os setores da economia, público e privado, cada um dentro de seu domínio e responsabilidades, estiverem sintonizados com a adoção desta nova tecnologia.

O uso de XBRL deverá proporcionar economia de escala, pois grupos com interesses comuns poderão compartilhar juntos os custos fixos de desenvolvimento de padrões técnicos e de aplicações. Os fornecedores de software já estão desenvolvendo uma grande gama de produtos complementares que usem padrões comuns, como XML e XBRL. A padronização traz inúmeras vantagens, como por exemplo, a do custo de manutenção e atualizações que são tipicamente baixos, já que um grande grupo de peritos técnicos poderá estar disponível, e a redução do custo de treinamento, pois o mecanismo de aprendizado no trabalho é facilitado pelo grande número de pessoas que serão conhecedoras do padrão. Este é o processo pelo qual a experiência específica e o conhecimento do padrão por parte dos usuários contribuem para o desenvolvimento pelo fabricante de software das capacidades técnicas do padrão.

Com XBRL pode-se fazer um processamento direto entre todas as partes envolvidas, facilitando a obtenção dos dados e processando a informação entre sistemas incompatíveis. Processamento direto entre tecnologias diferentes (hardware, software) requer a transmissão eletrônica automatizada e livre de erros. Isto pode ser alcançado se as partes falarem o mesmo idioma, em outras palavras, requer um padrão de mensagem para transações dentro de uma companhia e agregação da informação em relatórios para serem transmitidos para agentes externos seguindo esse mesmo padrão.

Apesar de XBRL ser uma tecnologia recente é possível prever que será, em pouco tempo, largamente difundida em função de alguns aspectos, tais como:

- a preparação de documentos para a área financeira se tornará mais fácil, já que a informação poderá ser importada dos sistemas de contabilidade existentes e de outros sistemas que contêm dados para relatórios financeiros. Em função da facilidade herdada de XML de nomear elementos, um programa de computador poderá encontrar os dados desejados, extraindo-os ou adicionando-os aos elementos de um documento XBRL;
- os usuários não estarão mais vinculados a um formato de dados proprietário, o que facilita, por exemplo, a mudança de um determinado pacote de aplicações para outro. Por ser XBRL uma especificação gratuita, aberta e não proprietária, o que fará o usuário manter seus dados em um formato proprietário será a qualidade do software;
- pressões do mercado que usa informações financeiras deverão crescer no sentido de reduzir custos de análise. Com XBRL o dueto tempo-custo para extração e análise da informação será reduzido. Parte da análise da informação poderá ser executada por computadores, pois os dados possuirão significados. Restando, assim, aos analistas, maior tempo para as análises mais significativas e que não dependem de processamento de máquina, o que fará com que o custo global diminua.

A utilização de XBRL fornece vantagens em cada etapa na cadeia de provisão da informação financeira, tendo como beneficiários os produtores e consumidores de relatórios financeiros: contadores, auditores, analistas financeiros, investidores, credores, fornecedores de tecnologia, os executivos de finanças, consultores e pesquisadores financeiros, órgãos reguladores e governamentais, etc. Ou seja, o uso de XBRL poderá ajudar cada membro da cadeia da informação financeira, proporcionando vantagens como [32]:

- auditores: XBRL permite aos auditores focalizarem esforços no trabalho de análise da informação, reduz os erros de transformação de formatos dos dados e aumenta a velocidade na qual a informação é acessível;

- companhias que preparam relatórios empresariais e demonstrações financeiras: estas companhias podem aumentar eficiência e precisão na preparação de relatórios empresariais e demonstrações financeiras porque eles são criados uma vez e publicados como relatórios impressos ou em locais da Web, intercambiados em relatórios corporativos internos ou submetidos a órgãos reguladores;
- publicadores de informações financeiras: coleta de dados mais eficiente reduz custos operacionais associados à inerente transformação de formatos da fonte da informação, reduzindo erros, permitindo que estes grupos concentrem-se em agregar valor aos dados;
- analistas, investidores e órgãos reguladores: usuários de relatórios financeiros beneficiam-se com novas possibilidades de análises automatizadas e atualização mais freqüente da informação. Estes grupos também se favorecem com o recebimento da informação em um formato eletrônico reutilizável;
- vendedores independentes de software: qualquer produtor de software que administra informação financeira pode usar XBRL para formatação de seus dados para exportação e importação, aumentando seu potencial de interoperabilidade com outras aplicações financeiras e analíticas. XBRL também abre possibilidades de novos produtos de análise da informação em relatórios financeiros;
- consultores de informática: essa comunidade pode descobrir oportunidades novas com a adoção de XBRL. Estas oportunidades incluem integração de dados e de aplicações, como também personalização de programas analíticos e estudo do impacto que XBRL pode causar na Tecnologia da Informação da organização.

Um aspecto relevante é que o impacto de XBRL nos processos de informação financeira não deverá ser significante, pois XBRL facilita a interoperabilidade entre sistemas, o que significa que companhias não terão que substituir sistemas existentes.

3.2 Criação de documentos XBRL

Para a implementação de XBRL é necessária uma coalizão de usuários, tantos dos que fornecem a informação como dos que dela se utilizam. Esta coalizão deverá definir os elementos XBRL que descreverão os conceitos financeiros. Nenhum padrão na verdade será usado dentro de um mercado a menos que se ajuste às necessidades da maioria ou de todos os usuários.

A estrutura dos elementos XBRL é definida em um documento de taxonomia. A taxonomia, que é um documento baseado em elementos e atributos XML Schema, é a

biblioteca dos termos financeiros usados na preparação dos relatórios financeiros. Assim como XML Schema é uma linguagem XML para descrever e restringir o conteúdo de documentos XML [33], a taxonomia restringe a linguagem XML Schema para descrever documentos XBRL. Na figura 3.1 é possível observar graficamente o escopo da taxonomia em relação à XML Schema, ou seja, ela é um subconjunto de XML Schema, assim como este é de XML.

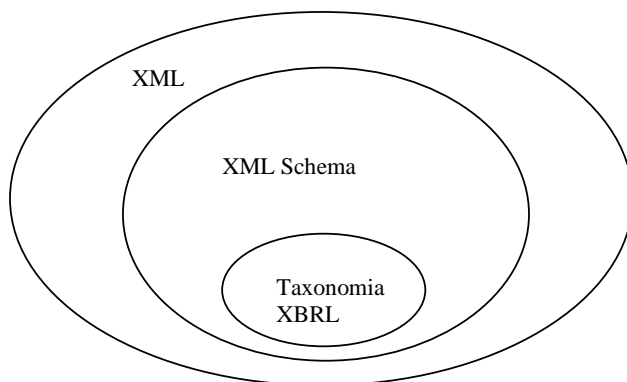


Figura 3.1 – Escopo dos documentos de taxonomia XBRL, em relação à XML Schema e XML.

Uma taxonomia contém estrutura, *labels*, relações matemáticas, ordem de apresentação e outras características para cada elemento. A taxonomia pode ser dividida em vários documentos, um chamado de taxonomia, para criar os elementos e definir sua estrutura, e outros para definir os relacionamentos entre os elementos, chamados de *Linkbases*.

Para iniciar um processo de criação de um documento XBRL deve-se selecionar uma taxonomia, ou criar a própria taxonomia, ou estender uma existente para atender aos requisitos específicos, ou ainda usar uma ou mais taxonomias para a criação de um documento. Para se criar uma taxonomia é necessário se determinar quais os fatos financeiros que se deseja expressar e a hierarquia entre estes fatos, a qual é expressa em um relacionamento hierárquico pai - filho.

Uma vez definida a taxonomia a ser usada, deve-se criar o documento, que contém os valores dos dados referentes ao fato financeiro que se deseja representar. As informações destes valores podem ser provenientes de um banco de dados, de outros documentos XBRL, de documentos de qualquer outra linguagem derivada de XML, ou mesmo de entradas manuais. Criado o documento XBRL, este pode ser validado na taxonomia, por meio de uma aplicação, da própria Internet ou de rotinas próprias, verificando se os elementos usados estão contidos na taxonomia e de acordo com a estrutura lá definida. Se a informação do documento

XBRL precisar ser distribuída em vários formatos, é necessário se criar folhas de estilos para expressá-la nos formatos desejados.

Considerando uma determinada companhia que usa determinada conta contábil, não prevista na taxonomia que esteja sendo por ela usada, um novo elemento para essa conta pode ser adicionado e, assim, se personalizar a taxonomia. Uma vez estando a taxonomia personalizada, o novo elemento poderá fazer parte do documento XBRL. Isto permite aos usuários codificarem em XBRL a informação específica de determinada companhia.

A figura 3.2 mostra o processo de criação e utilização de um documento XBRL [1].

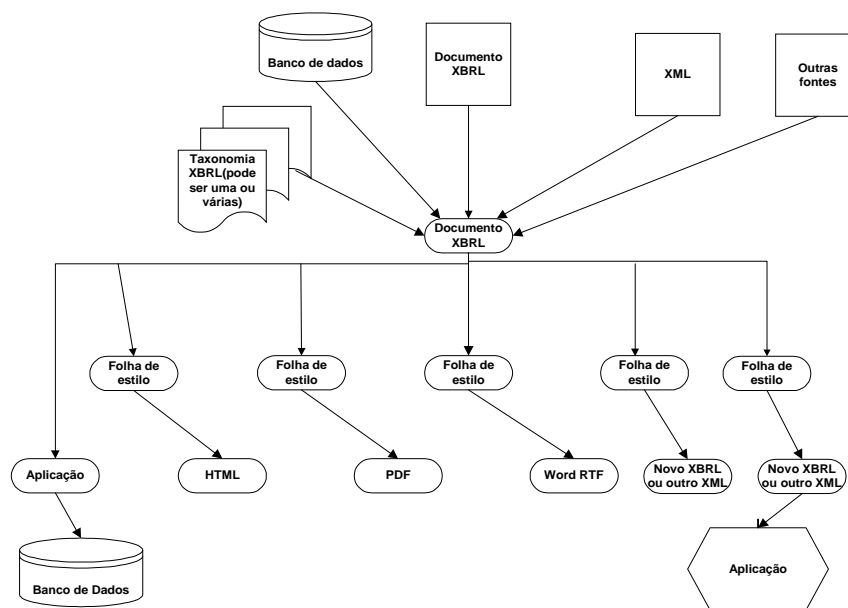


Figura 3.2 - Processo de criação e utilização de documentos XBRL.

3.3 A estrutura XBRL

Como já foi dito, XBRL fornece flexibilidade e extensibilidade na criação de relatórios financeiros, porém não define os elementos necessários para o intercâmbio da informação financeira. Esta responsabilidade é da comunidade de usuários que cria a sua própria taxonomia. A comunidade é quem determina quais as informações necessárias e qual o nível de detalhe exigido.

Existem razões para que a comunidade usuária da informação seja responsável pela definição da taxonomia, uma delas é que para uma informação que vá transitar externamente em uma companhia, o nível de detalhe é mais sumarizado do que aquela que circulará internamente, logo a companhia é quem deverá determinar quais elementos a serem usados. Outra razão é que organizações têm fatos financeiros diferentes, países têm legislações

específicas para suas finanças, com regulamentações contábeis diferentes. Certamente não seria possível, ou extremamente difícil e vagaroso, se criar uma única taxonomia para atender a todas as particularidades de todos os fatos financeiros no mundo. Por isso XBRL delega esta função à comunidade usuária, definindo apenas a estrutura para a criação das taxonomias, a qual especificará os elementos e suas características para determinada comunidade. XBRL é extremamente flexível, porém esta flexibilidade deve estar dentro de certos parâmetros definidos na sua especificação.

O usuário, baseado nas suas necessidades, define os elementos e constrói os documentos que ele quer disponibilizar. Entretanto, XBRL torna estes documentos previsíveis, já que outros usuários podem saber como eles foram construídos, pois a especificação XBRL é quem dita como o documento deve ser criado. O fato de se saber como o documento e seus elementos foram elaborados, facilita extremamente a reutilização da informação e a adição de qualquer novo elemento. Ou seja, o uso de XBRL permite a reusabilidade e extensibilidade da informação.

A especificação XBRL descreve como elementos e atributos são organizados e estruturados para a criação de documentos e de suas taxonomias. Ela fornece a direção para se construir o documento, se todos usam essa mesma direção, então se pode estar certo de que o documento é consistente para ser usado por qualquer usuário que tenha acesso à taxonomia.

Então, a representação de um relatório financeiro em XBRL é constituída por três tipos de documentos:

- o próprio documento XBRL, que contém os dados do relatório financeiro;
- o documento de taxonomia, que descreve os elementos usados no documento XBRL, os quais representam os fatos financeiros, ou seja, a taxonomia é o vocabulário e dicionário dos termos usados;
- os documentos *linkbases* – que expressam os relacionamentos entre os elementos definidos na taxonomia, estes documentos são estruturados de acordo com a recomendação XML *Link* do W3C;

Para estruturar os relacionamentos de forma mais precisa, facilitando manutenção e adição de relacionamentos entre novos elementos, os documentos *linkbases* são divididos em cinco: *calculation*, *presentation*, *definition*, *reference* e *label*, cujas funções podem ser expressas como se seguem:

- *Links Relation (definition, calculation, e presentation)*, os quais expressam as relações entre os elementos da taxonomia, ou seja, quem é o pai, quem é o filho; como os valores

dos elementos filhos se relacionam para fornecer o do pai (adicionando ou subtraindo); e como os elementos devem ser ordenados para a apresentação;

- *Link label*, o qual expressa um nome para o elemento na taxonomia, que pode estar associada a vários idiomas, ou seja, um elemento pode possuir diversos *labels*, um para cada idioma;
- *Link reference*, que é utilizado para se fazer uma referência, por exemplo, a uma literatura normativa.

Os principais blocos de construção de um documento XBRL são expressos por dois outros documentos: o XBRL Instance Schema e o XBRL Linkbase Schema, que estão, respectivamente, nos apêndices A e B. Esses dois documentos, juntamente com a taxonomia, criada a partir deles, podem ser usados por aplicações para ajudar a validar documentos.

O XBRL Instance Schema descreve como documentos XBRL e a taxonomia são criados e o XBRL Linkbase Schema define os elementos *linkbases* que estruturarão os relacionamentos dos elementos criados na taxonomia.

3.4 Documentos XBRL - os elementos e atributos

Os documentos descritos previamente - taxonomia, *linkbases* e o documento XBRL trabalham juntos para descreverem e tornarem a informação disponível. A seguir será feita uma descrição de cada um destes documentos, com seus principais elementos e atributos. Nem todos os elementos e atributos presentes na especificação XBRL foram aqui analisados, apenas os mais importantes para uma compreensão melhor de XBRL e para a discussão do estudo de caso, que será vista nos capítulos seguintes. No apêndice L encontra-se um resumo tabulado dos elementos e seus atributos, definidos na especificação XBRL, necessários para compor cada tipo de documento (taxonomia, *linkbases* e documento XBRL).

3.4.1 O documento de Taxonomia XBRL

Um documento de taxonomia consiste de um vocabulário de fatos financeiros, ou seja, é uma coleção de fatos financeiros. Esta coleção de fatos consiste de uma lista das definições de novos elementos, com seus atributos e tipos de dados, além das relações entre esses elementos ou destes com elementos de outras taxonomias. Uma taxonomia é definida usando um vocabulário XML Schema e uma série de elementos Xlink que implementam um grupo de *linkbases*.

Uma taxonomia contém e descreve as seguintes informações para cada fato financeiro:

- *Name* – nome do fato, como “AtivoCirculante”;

- *Type* – tipo de dado do fato, como *monetary* ou *share*;
- *Documentation* – uma descrição do fato financeiro;
- *To* – nome de um elemento para o qual um arco está sendo direcionado;
- *From* – nome do elemento onde um arco está sendo originado;
- *Weight* – indica uma relação matemática do elemento filho com o pai, se seu valor será completamente, ou parcialmente, adicionado ou subtraído a outros para fornecer o valor do pai;
- *Order* – indica a ordem de apresentação dos elementos filhos, por exemplo, 2 indica que o elemento será o segundo a ser apresentado no documento;
- *Label* – um nome para o fato, pode-se ter vários labels, cada um em um idioma diferente;

Cada documento de taxonomia deve obrigatoriamente importar o documento XBRL Schema Instance, que define os elementos abstratos *item* e *tuple*, para serem usados no atributo *substitutionGroup* na definição dos elementos, e os tipos de dados básicos, para a definição dos elementos na taxonomia. Para realizar isso é necessário usar o elemento padrão *import* do XML Schema, que também serve para referenciar outras taxonomias XBRL. Os tipos de dados básicos dos elementos podem ser o *monetary*, que serve para representar um valor financeiro, ou o *share*, que representa ações. O documento XBRL Schema Instance, por sua vez, importa o documento XBRL Schema Linkbase, que define o vocabulário *linkbase*. Em geral, vários outros *schemas* poderão ser referidos por meio de declarações de *namespaces*, como o *schema* para XML Schema ou uma auto-referência para a taxonomia em construção.

Na página seguinte é apresentado um trecho da taxonomia criada para o estudo de caso, que será apresentado em capítulo posterior, a taxonomia completa está no apêndice D:


```

<schema
  targetNamespace=" http://www.bcb.gov.br"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:link="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  xmlns:xbrli="http://www.xbrl.org/2001/instance"
  xmlns:bcb=" http://www.bcb.gov.br " elementFormDefault="qualified">
  <annotation>
  <appinfo>
    <link:linkbaseRef xlink:type="simple" xlink:href="bcb_presentation.xml"
      xlink:actuate="onRequest"
      xlink:role="http://www.xbrl.org/linkprops/linkRef/presentation"
      xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbase">
      <xhtml:p>Links para relacionamentos de apresentação</xhtml:p>
    </link:linkbaseRef>
  </appinfo>
  </annotation>
  <!-- Import de XBRL instance schema (exigido em toda taxonomia XBRL) -->
  <import namespace="http://www.xbrl.org/2001/instance" schemaLocation="xbrl-instance.xsd"/>
  <!-- Definição de elementos -->
  <element name="bp" id="bp" type="xbrli:tupleType" substitutionGroup="xbrli:tuple"/>
  <element name="ativo" id="ativo" type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item">
    <annotation>
      <documentation>Seção que contém os ativos do balanço patrimonial.
    </documentation>
    </annotation>
  </element>
  <!-- Os elementos abaixo foram adicionados para declarar a diretoria da instituição -->
  <element name="diretores" type="string" substitutionGroup="link:part">
  </element>
  <!-- O elemento abaixo foi adicionado para declarar o elemento scenario usado no documento
  XBRL -->
  <element name="scenarioType">
    <simpleType>
      <restriction base="string">
        <enumeration value="final"/>
        <enumeration value="preliminar"/>
      </restriction>
    </simpleType>
  </element>
  <!-- O elemento descrição foi adicionado para ser usado no documento XBRL -->
  <element name="descricao" type="string" substitutionGroup="link:part"/>
  <!-- O elemento abaixo foi adicionado para ser usado como elemento de referência -->
  <element name="referencia" type="string" substitutionGroup="link:part"/>
</schema>

```

Este é um modelo para uma taxonomia, contendo *imports*, *namespaces* e *links*.

Uma característica fundamental do XBRL é a extensibilidade de taxonomias. Uma taxonomia deve ser extensível para acomodar virtualmente qualquer relatório financeiro, podendo inclusive ser construída de forma a referir-se a outras taxonomias. A seguir serão descritos os principais elementos para composição de uma taxonomia.

3.4.1.10 elemento *linkbaseRef*

A especificação Xlink fornece um caminho padrão para encontrar os *linkbases*, onde serão definidos os relacionamentos entre os elementos. O elemento *linkbaseRef* é usado no documento de taxonomia para encontrar os *linkbases* dos cinco tipos de *links* estendidos utilizados por XBRL. A definição deste elemento está presente no documento XBRL Linkbase Schema (apêndice B), que define os *linkbases* com os seguintes atributos:

O atributo **xlink:type** tem o valor fixo “simple” e deve ser explicitado em qualquer elemento *linkbaseRef*, para que aplicações (como as baseadas em XSL e DOM) possam encontrar as informações contidas no elemento. Este atributo faz parte da estrutura de outros elementos da especificação e os seus valores também são fixos e com esta mesma função.

O atributo **xlink:href** contém uma URI que aponta para um *linkbase* particular, ou seja, ele indica onde está o arquivo que contém os links que definem o relacionamento entre os elementos.

O atributo **xlink:role** identifica o tipo de link estendido contido no *linkbase*. Os valores padrão definidos na especificação XBRL para os tipos de links estendidos são:

```
http://www.xbrl.org/linkprops/linkRef/presentation
http://www.xbrl.org/linkprops/linkRef/calculation
http://www.xbrl.org/linkprops/linkRef/definition
http://www.xbrl.org/linkprops/linkRef/label
http://www.xbrl.org/linkprops/linkRef/reference
```

O atributo **xlink:arcrole** tem o valor “http://www.w3.org/1999/xlink/properties/linkbase”, que é fixo, definido por Xlink, e explicitado em qualquer elemento *linkbaseRef*, para que aplicações saibam que esse elemento é uma ligação para um *linkbase*.

Os outros atributos *title*, *show* e *actuate* de um elemento Xlink serão discutidos adiante, juntamente com os elementos do tipo de *link* estendido.

3.4.1.2 Tipos de dados *monetary* e *share*

A especificação XBRL define o tipo de dado “monetary”, que é uma especialização do tipo decimal de XML Schema. A taxonomia, que contém elementos numéricos que devem ser interpretados como valores monetários, deve obrigatoriamente usar este tipo de dado. Se um elemento possui o tipo monetário, o conteúdo do elemento *Unit*, no documento XBRL onde o elemento está representado, deve ser o nome da moeda designada na norma ISO 4217. No caso do Brasil, cuja moeda é o Real, o valor para este elemento deve ser BRL.

Da mesma forma, XBRL define o tipo de dado “share” para representar quantitativos de ações. As definições para esses tipos de dados estão no apêndice A.

3.4.1.3O elemento *Element*

O elemento *Element* contém os atributos *name*, *substitutionGroup*, o qual deve ter como valor *item* ou *tuple*, e *type*, que pode ser *monetary* ou *share*. Um elemento deve ser único dentro de uma taxonomia, isto é, na taxonomia determinado nome deve referir-se apenas a um elemento. Porém, não há impedimento para que o mesmo nome denomine outro elemento em outra taxonomia, ainda que as duas sejam usadas em um mesmo documento XBRL, a diferenciação será feita pelo prefixo do *namespace*.

Para uma melhor integração com Xlink, todo elemento pode ter um atributo opcional *id*, cujo conteúdo deve ser igual ao conteúdo do atributo *name*. A razão para isto, é que o valor deste atributo *id* poderá ser usado para que o elemento seja referenciado no elemento de tipo *Locator* de um *linkbase*.

3.4.2 Os documentos *Linkbases*

Fatos financeiros geralmente possuem uma hierarquia ou relacionamentos entre si. Esta hierarquia é expressa em uma taxonomia usando *links*. Por exemplo, Balanço Patrimonial é um pai que possui dois filhos: Ativo e Passivo. Ativo possui como filhos: Circulante, Realizável a Longo Prazo, etc. Os conceitos de XML Link são aplicados aqui para expressar essa relação de hierarquia.

Em XBRL, os *linkbases* são usados para definir os elementos que fazem a ligação dos elementos XBRL e determinar qual o tipo de ligação (*relation*, *reference* ou *label*) é efetuada.

Como já anteriormente dito, há cinco tipos de *links* estendidos usados em XBRL. Estes *links* podem estar no documento de taxonomia ou em documentos separados e devem conter o elemento XBRL *linkbase* como elemento raiz.

Para validar os documentos *linkbase* é necessário usar outros *schemas* que implementam a especificação Xlink. Esses *schemas* são o *xlink.xsd* (apêndice F), o qual define os atributos a serem usados na definição dos elementos *xlink*, que estão definidos no *schema xl.xsd* (apêndice G).

A figura 3.3 abaixo mostra como todos esses documentos estão relacionados para a criação da taxonomia:

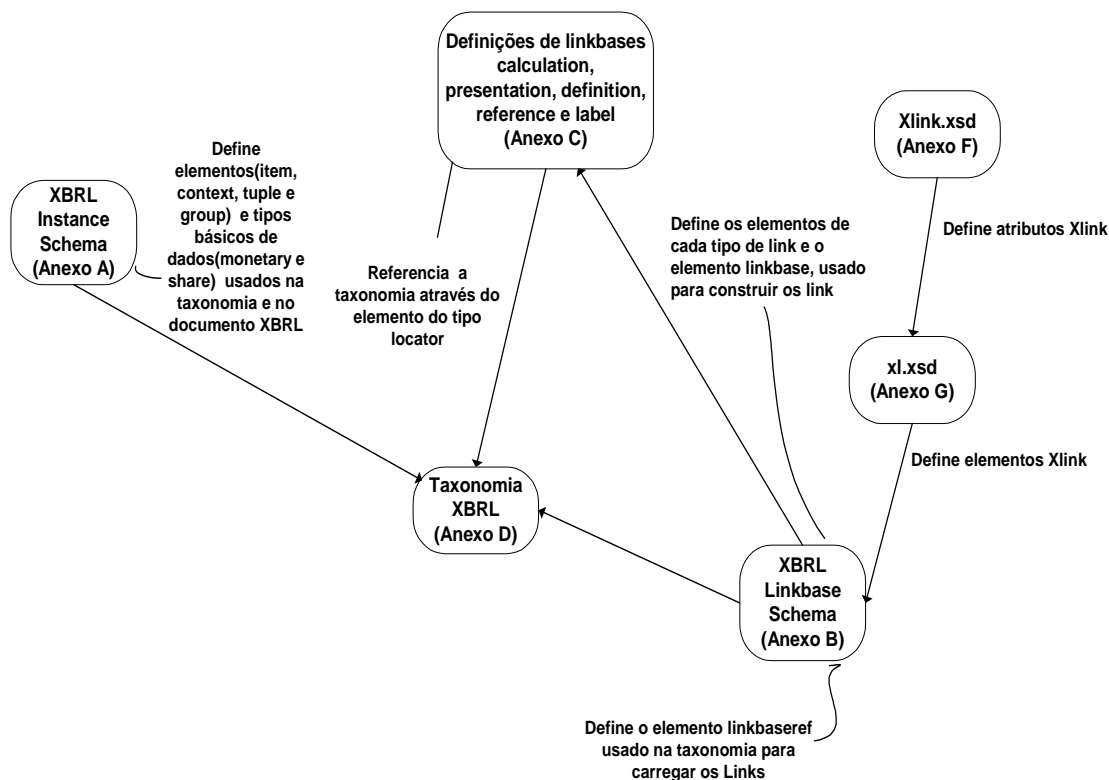


Figura 3.3 – Documentos envolvidos na criação da Taxonomia

Nos próximos subitens será feita uma descrição dos elementos usados para a construção dos *linkbases* XBRL.

3.4.2.1 Elementos do tipo link estendido

Há cinco tipos de elementos de *link* estendido na taxonomia XBRL:

- *calculationLink* – contém os elementos *loc* e *calculationArc*;
- *definitionLink* - contém os elementos *loc* e *definitionArc*;
- *presentationLink* - contém os elementos *loc* e *presentationArc*;
- *labelLink* - contém os elementos *loc*, *label* e *labelArc*;
- *referenceLink* - contém os elementos *loc*, *reference* e *referenceArc*;

O atributo **xlink:type** nesses cinco elementos tem o valor “extended”, para definir que este é um *link* estendido.

O atributo **xlink:role** para esses cinco elementos possui uma lista de valores padrão, que informam qual o tipo do relatório que está sendo representado:

```

http://www.xbrl.org/linkprops/extended/balanceSheet
http://www.xbrl.org/linkprops/extended/incomeStatement
http://www.xbrl.org/linkprops/extended/statementOfComprehensiveIncome
http://www.xbrl.org/linkprops/extended/statementOfStockholdersEquity
http://www.xbrl.org/linkprops/extended/cashFlows
  
```

Isto facilita a pesquisa feita por aplicações, evitando a leitura de todos os links, se apenas determinado grupo de documentos é o desejado. Porém, outros valores podem ser criados pelos autores da taxonomia, na medida das suas necessidades.

3.4.2.1.1 Elementos do tipo *locator*

Uma característica comum de todos os *links* estendidos é o uso do elemento de localização *loc*, para indicar, através do atributo *href*, a taxonomia onde o elemento está definido. A definição deste elemento está no *schema* xl.xsd (apêndice G). Um exemplo deste elemento pode ser visto em `<loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#ativo" xlink:label="bcb_ativo" xlink:title="ativo" />`.

No exemplo acima, o atributo **xlink:href** contém uma URI, que é composta pelo nome de um arquivo, a taxonomia, e o valor do atributo *id* do elemento, atributo incluído na definição do elemento, como dito anteriormente, com o propósito de constituir este elo de ligação. O formato da URI é o seguinte: “arquivo de taxonomia#id do elemento”.

O atributo **xlink:type** tem o valor fixo “locator”, para informar que há um recurso remoto endereçado pelo atributo **xlink:href**.

O atributo **xlink:label** identifica o elemento *loc*, para ser referenciado pelo elemento do tipo *arc*, que será visto adiante.

O atributo **xlink:role**, não usado no exemplo, serve como diferenciador funcional para os elementos do tipo *locator*, ou seja, ele aponta para a raiz de uma árvore de elementos na taxonomia. Taxonomias poderão conter uma ou mais árvores de elementos e cada tipo de *link* poderá ter seu próprio grupo de árvores. Distinguindo a raiz das árvores, uma aplicação poderá mais facilmente descobrir a informação. Portanto, o atributo pode ter valor semelhante a `xlink:role="http://www.xbrl.org/linkprops/locator/root"`.

O atributo opcional **xlink:title** pode ser usado para conter informações que facilitem o entendimento a usuários do *link* ou para que aplicações utilizem para um propósito específico, como por exemplo emitir uma mensagem de status (ex: Vá para o label *bcb_ativo_br*).

3.4.2.1.1.1 Links *Label*

Uma das características fundamentais da internacionalização de XBRL é que embora cada taxonomia defina um único grupo de elementos que representa um conjunto de conceitos financeiros, o *label* - uma *string* usada para apresentar o nome do conceito – é declarado separadamente com uma indicação do idioma que se está usando, por meio do atributo XML

lang. Assim, um determinado conjunto de fatos financeiros poderia ser apresentado em qualquer idioma selecionado pelo usuário.

Os elementos *label* aparecem dentro do elemento de *link* estendido *labelLink*. O conteúdo de *label* pode ser uma simples *string* ou elementos de XHTML [28], os quais devem ser restritos a elementos de texto.

O atributo **xlink:type** do elemento tem o valor fixo “resource”, para informar que este elemento é um recurso local.

O atributo **xlink:role** é opcional e pode distinguir *labels* por sua intenção de uso. Dois valores padrão para este atributo são:

```
http://www.xbrl.org/linkprops/label/standard
http://www.xbrl.org/linkprops/label/total
```

No estudo de caso estes valores serão usados para distinguir os elementos (*total*) cujo valores são o resultado da soma dos valores de outros elementos (*standard*).

Exemplo de *labels* e definições dos links estão no apêndice C, o fragmento de código seguinte exemplifica um destes links:

```
<linkbase
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbase xbrl-linkbase.xsd">
  <labelLink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/BalancoPatrimonial">
  <!--labels definidos aqui.-->
    <label xlink:type="resource" xlink:label="bcb_bp_br" xlink:title="bcb_bp_br"
      xlink:role="http://www.xbrl.org/linkprops/label/total" xml:lang="br">BALANÇO
      PATRIMONIAL</label>
  <!--Definições das localizações dos elementos-->
    <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#bp"
      xlink:label="bcb_bp" xlink:title="bp" />
  <!--Definições dos arcos de ligação entre elementos e seus labels-->
    <labelArc xlink:type="arc" xlink:from="bcb_bp" xlink:to="bcb_bp_br"
      xlink:show="embed" xlink:actuate="onRequest" xlink:title="Vá para o label bcb_bp_br"
      xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-label"/>
      . . .
  </labelLink>
</linkbase>
```

O elemento *label* contém o texto do *label*, que nomeia o elemento da taxonomia, referenciado pelo elemento *loc*, no idioma definido pelo atributo *XML:lang*, e os dois elementos *arc* permitem a navegação bi-direcional entre o elemento e o *label*. Os atributos *show*, *actuate* e *title* permitem que softwares baseados em XLink naveguem no *linkbase*.

3.4.2.1.1.2 Links Reference

Elementos *Reference* permitem taxonomias XBRL referenciar as definições de conceitos financeiros dentro da literatura financeira e contábil. Referências podem estar subdivididas em partes, como capítulo, parágrafo, artigo, etc. Portanto, um subelemento abstrato *part* é criado, pela especificação XBRL, para essa função, ou seja, taxonomias podem

definir elementos que pertencem ao grupo de substituição *part*, para serem usados no documento XBRL, já que sendo *part* um elemento baseado no tipo abstrato, de XML Schema, não pode aparecer no documento. A definição dos elementos *part* e *reference* está no documento *schema* XBRL-linkbase.xsd (apêndice B).

Este tipo de informação define a referência para a literatura de um conceito financeiro. O exemplo abaixo mostra a utilização deste tipo de *link*. No estudo de caso foram criados elementos de referência para os diretores da instituição, cujos nomes são representados no documento XBRL, e um elemento *referencia* para a literatura normativa, que é definido no *linkbase Reference* (apêndice C).

A definição dos elementos na taxonomia pode ser observada abaixo:

```
<element name="diretores" type="string" substitutionGroup="link:part">
  <element name="presidente" type="string" substitutionGroup="link:part"/>
  <element name="diretorDipec" type="string" substitutionGroup="link:part"/>
  .
  .
  .
</element>
<!-- O elemento abaixo foi adicionado para ser usado como elemento de referência -->
  <element name="referencia" type="string" substitutionGroup="link:part"/>
```

A referência bibliográfica é definida pelo elemento “*bcb:referencia*” no documento

linkbase:

```
<linkbase
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  .
  .
  .
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbase xbrl-linkbase.xsd"
  <referencelink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/BalancoPatrimonial">
<!-- Referências definidas aqui -->
  <reference xlink:type="resource" xlink:label="bcb_bp_ref" xlink:title="bcb_bp_ref">
    <bcb:referencia>As demonstrações contábeis são elaboradas de acordo com a legislação
    aplicável ao Banco Central, com destaque para a Lei 4.320/64, Lei 4.595/64 (alterada
    pelo Decreto-Lei 2.376/87), Decreto-Lei 278/67, Lei 7.862/89, Medida Provisória
    2.179/2001 e a Lei Complementar 101/2000 (Lei de Responsabilidade Fiscal). Em
    conformidade com essa legislação, o Banco Central apura resultado e elabora balanços
    semestrais em junho e dezembro de cada ano, bem como balancetes nos demais meses.
    </bcb:referencia>
  </reference>
<!-- Definições das localizações do elemento-->
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#bp" xlink:label="bcb_bp"
    xlink:title="bp"/>
<!--Definições dos arcos de ligação entre o elemento e sua referência-->
  <referenceArc xlink:type="arc" xlink:from="bcb_bp" xlink:to="bcb_bp_ref" xlink:show="embed"
    xlink:actuate="onRequest" xlink:title="Vá para a referência bcb_bp_ref"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-reference"/>
  <referenceArc xlink:type="arc" xlink:from="bcb_bp_ref" xlink:to="bcb_bp"
    xlink:show="replace"
    xlink:actuate="onRequest" xlink:title="Vá para o elemento bcb_bp"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/reference-element"/>
  </referencelink>
</linkbase>
```

O elemento *reference* contém a referência e os dois elementos *arc* permitem a navegação bi-direcional entre o elemento da taxonomia e o elemento de referência vinculado a ele.

E no documento XBRL o elemento “*bcb:diretores*” é usado para expressar os nomes da diretoria da organização:

```
<bcb:diretores nonNumericContext="c2">
  <bcb:presidente nonNumericContext="c2">Arminio Fraga Neto</bcb:presidente>
  .
  .
  <bcb:chefedeafi nonNumericContext="c2">Jefferson Moreira - CRC: DF 7333</bcb:chefedeafi>
</bcb:diretores>
```

3.4.2.1.2 Elementos do tipo *arc* (links calculation, presentation e definition)

Elementos do tipo *arc* ligam os recursos referenciados pelos atributos *from* e *to*. Esses atributos contêm os valores do atributo *xlink:label* de um elemento do tipo *locator* ou do tipo *resource* (no caso dos elementos *labelArc* e *referenceArc*, vistos acima) dentro de um mesmo *link* estendido. Esses elementos do tipo *arc* têm os seguintes atributos:

O atributo **xlink:type** tem o valor fixo “arc”, para informar que é um elemento que fornecerá regras para a travessia entre os recursos dos *links* que estão participando do arco.

O atributo **xlink:show** tem os valores *embed* ou *replace*. O uso de *embed* é para as situações onde a origem do recurso está ligada a um elemento que está em um arquivo diferente, como no caso do arco cuja origem está em um elemento da taxonomia e o destino é um elemento *label* presente no *labellink*. E *replace* para arcos cuja origem esteja no mesmo arquivo do arco. Esse atributo é melhor entendido analisando-se o subelemento *labelArc* de um *Link label*.

O atributo **xlink:actuate** tem o valor fixo “*onRequest*”. Nenhum destes *links* são considerados mandatórios, nem precisam ser atravessados no momento em que o arquivo é carregado (*onLoad*). Este atributo deve ser explicitado em qualquer elemento *arc*.

O atributo **xlink:to** deve ter no seu conteúdo um valor que deve ser o mesmo do atributo *xlink:label* de um elemento *locator* ou *resource*, contido no mesmo elemento de *link* estendido do elemento de tipo *arc*.

O atributo **xlink:from** é semelhante ao **xlink:to**.

O atributo **xlink:arcrole** deve ter como valor uma *string*. Em uma navegação bi-direcional, como de um arco A para B e vice-versa, este atributo tem a função de informar a direção em que o link está indo, como por exemplo, “de pai para filho” ou “de filho para pai”.

Nos cinco links estendidos de XBRL os valores padrão são os seguintes:

xlink:arcrole de calculationArc, definitionArc, presentationArc	Significado
http://www.xbrl.org/linkprops/arc/child-parent	O arco é de filho para pai.
http://www.xbrl.org/linkprops/arc/parent-child	O arco é de pai para filho.
http://www.xbrl.org/linkprops/arc/dimension-element	O arco é de um elemento para outro equivalente
http://www.xbrl.org/linkprops/arc/element-dimension	O arco é de um elemento para outro equivalente

xlink:arcrole de labelArc	Significado
http://www.xbrl.org/linkprops/arc/label-element	O arco é do <i>label</i> para o elemento.
http://www.xbrl.org/linkprops/arc/element-label	O arco é do elemento para o <i>label</i> .
xlink:arcrole de referenceArc	Significado
http://www.xbrl.org/linkprops/arc/reference-element	O arco é de uma referência para um elemento.
http://www.xbrl.org/linkprops/arc/element-reference	O arco é de um elemento para uma referência.
http://www.xbrl.org/linkprops/arc/actual-element	O arco é do conteúdo da literatura para o elemento.
http://www.xbrl.org/linkprops/arc/element-actual	O arco é do elemento para o conteúdo da literatura.

3.4.2.1.2.1 Links Calculation

O elemento *calculationlink* define como os elementos estão relacionados um com os outros em uma adição matemática simples. O elemento *calculation* é um elemento *arc* de XML Link.

Os atributos obrigatórios *from* e *to* contêm os valores do atributo *label* do elemento *locator*, já explicados anteriormente. Os elementos *locator* têm que existir no mesmo *link* estendido do elemento *calculationArc*, que está definido no elemento *calculationlink*. Essa restrição é válida para todos os tipos de *links*.

O atributo **weight** indica qual o multiplicador a ser aplicado em um valor de um elemento, quando este contém valores numéricos que serão usados pela hierarquia da taxonomia. Ou seja, um valor "1.0" significa que 100% do valor numérico do elemento será uma parcela aditiva da soma que resultará no valor do elemento pai. Um valor de "-1.0" significa que 100% do valor numérico do elemento será subtraído como parcela da soma que resultará no valor do elemento pai. Se o valor do atributo *arcrole* for *dimension-element* ou *element-dimension*, o valor de *weight* deverá ser ignorado. Um trecho deste tipo de *link* usado no estudo de caso é visto a seguir:

```
<calculationlink xlink:type="extended"
  xlink:role="http://www.xbrl.org/linkprops/extended/balanceSheet">
  ...
<!--Definições dos arcos-->
<!-- Relacionamento de filho para pai -->
  <calculationArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo"
    xlink:to="bcb_ativo" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_ativo" weight="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
<!-- Relacionamento de pai para filho -->
  <calculationArc xlink:type="arc" xlink:from="bcb_ativo"
    xlink:to="bcb_atv_circ_real_lprazo" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_atv_circ_real_lprazo" weight="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/parent-child"/>
  ...
</calculationlink>
```

No exemplo acima há uma relação de cálculo entre `bcb_ativo` e `bcb_atv_circ_real_lprazo`, onde `bcb_atv_circ_real_lprazo` contribui com todo seu valor para o valor de `bcb_ativo`.

3.4.2.1.2.2 Links Presentation

O elemento *presentationlink* define como os elementos estão relacionados uns com os outros, para definir uma ordem de apresentação no documento, numa típica relação hierárquica pai-filho. O elemento *presentation* é um elemento *arc* de XML Link.

O atributo **order**, que deve conter um valor de um número inteiro não negativo, indica como são ordenados os elementos irmãos para apresentação dentro do elemento pai. Seu valor *default* é "1". Uma aplicação é em princípio livre para ignorar este atributo, e se vários elementos irmãos têm o mesmo valor para *order*, então a apresentação será dependente da aplicação. Um trecho deste tipo de *link* usado no estudo de caso é visto a seguir:

```
<presentationlink xlink:type="extended"
  xlink:role="http://www.xbrl.org/linkprops/extended/balancoPatrimonial">
  . . .
  <presentationArc xlink:type="arc" xlink:from="bcb_ativo" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest" xlink:title="Apresentação: Vá para bcb_bp"
    order="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <presentationArc xlink:type="arc" xlink:from="bcb_passivo" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest" xlink:title="Apresentação: Vá para bcb_bp"
    order="2"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  . . .
</presentationlink>
```

Há uma relação de apresentação entre `bcb_ativo`, `bcb_passivo` e `bcb_bp`, onde `bcb_ativo` é o primeiro filho de `bcb_bp` e `bcb_passivo` é o segundo filho.

3.4.2.1.2.3 Links Definition

O elemento *definitionlink* define uma relação hierárquica pai-filho dos elementos. O elemento *definitionArc* é um elemento *arc* de XML Link. Os atributos obrigatórios *from* e *to* e o elemento *locator* são semelhantes aos seus homônimos anteriores. Um exemplo deste link pode ser observado a seguir:

```
<definitionlink xlink:type="extended"
  xlink:role="http://www.xbrl.org/linkprops/extended/balanceSheet">
  ...
  <definitionArc xlink:type="arc" xlink:from="bcb_ativo" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_bp"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <definitionArc xlink:type="arc" xlink:from="bcb_passivo" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_bp"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  ...
</definitionlink>
```

Há uma relação de definição (pai e filho) entre *bcb_ativo*, *bcb_passivo* e *bcb_bp*. O exemplo da definição deste e dos outros tipos de *links* estão no apêndice C.

3.4.3 O documento XBRL

Os documentos XBRL devem estar de acordo com as regras gerais e estruturas especificadas no documento da especificação XBRL [28].

A sintaxe núcleo para documentos XBRL é definida usando o documento XBRL Instance Schema (apêndice A). Os elementos definidos neste documento são *item*, *context*, *tuple* e *group*, que são utilizados para especificar os elementos XBRL na taxonomia e construir os documentos XBRL que representam os relatórios financeiros.

A figura 3.4, na próxima página, mostra a relação de todos documentos envolvidos na criação de um documento XBRL:

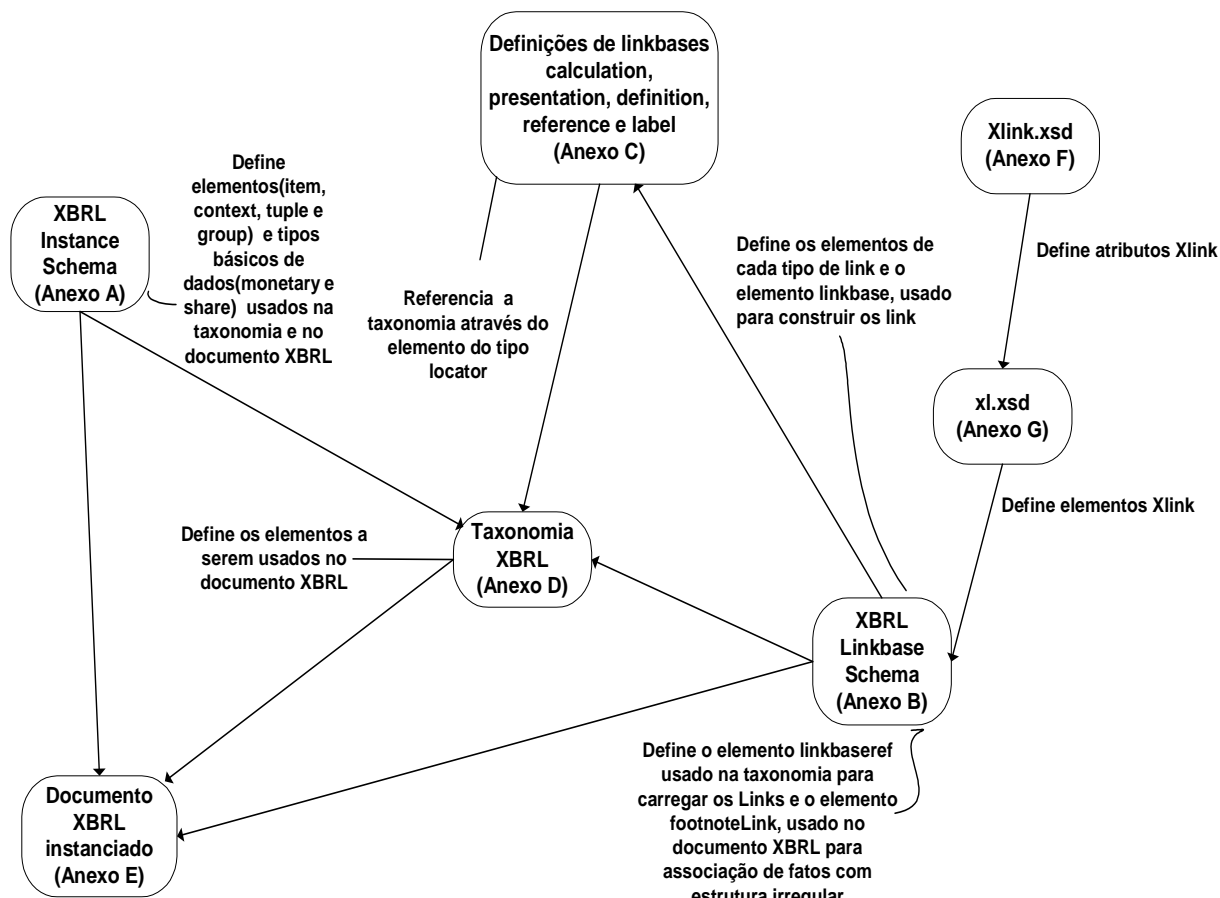


Figura 3.4 - Relacionamento dos arquivos usados na construção do documento XBRL.

3.4.3.1 O elemento *group*

Este é um elemento *container* genérico do vocabulário XBRL. É usado como elemento raiz nos documentos. Sua definição é encontrada no XBRL Instance Schema (apêndice A).

Um exemplo de sua utilização está no documento XBRL, contido no apêndice E, onde o uso do *group*, como elemento raiz, informa os *namespace* onde os elementos estão definidos, o atributo *xsi:schemaLocation* informando onde está a taxonomia e os elementos definidos na taxonomia com os seus valores.

3.4.3.2 O elemento *item*

Um item representa um único fato ou medida de negócio. Em XML Schema, *item* está definido como um elemento do tipo abstrato, o que significa que nunca aparecerá no texto de um documento XBRL. Já os elementos do documento de taxonomia para poderem aparecer em um documento XBRL, ou devem ser membros do grupo de substituição *item*, ou devem ser membros de um grupo de substituição que seja originalmente baseado em *item*, ou seja, os elementos definidos na taxonomia herdam a estrutura de *item*. Por exemplo, o elemento *ativo* foi definido na taxonomia (apêndice D), como sendo do tipo *monetaryItemType* e pertencente ao grupo de substituição *item*, o que faz com que seja obrigatória a presença do atributo *numericContext* de *monetaryItemType* na representação deste elemento no documento XBRL:

```
<element name="ativo" id="ativo" type="xbrli:monetaryItemType" substitutionGroup="xbrli:item">
```

Como exemplo da representação do elemento em um documento XBRL pode-se ter:

```
<bcb:ativo numericContext="c1">270190618</bcb:ativo>
```

, o que significa que o valor do ativo no contexto numérico chamado “c1” é R\$ 270.190.618,00.

Os atributos *numericContext* e *nonNumericContext*, especificados na estrutura do tipo *monetaryItemType*, são um identificador para o elemento *context*, que será visto adiante, que carrega o metadado do elemento. Um elemento XBRL tem que sempre conter um atributo *context* (*nonNumericContext* ou *numericContext*), para fazer referência a um elemento *context*.

3.4.3.3 O elemento *context* (*numericContext* e *nonNumericContext*)

O elemento *context*, que na realidade são os elementos *numericContext* e *nonNumericContext*, é o armazenador dos metadados nos conteúdos de seus atributos e

elementos, os quais juntos fornecem o contexto necessário para se entender um fato financeiro. O elemento *nonNumericContext* contém metadados e atributos relacionados a fatos textuais, enquanto *numericContext* contém metadados e atributos relacionados a fatos numéricos. O elemento *context* deve estar de acordo com a definição presente no documento XBRL Schema Instance (apêndice A).

Esse elemento foi usado no estudo de caso da seguinte forma:

```
<numericContext id="c5" precision="10" cwa="true">
  <entity>
    <identifier scheme="http://www.bcb.com.br/">Banco Central do Brasil</identifier>
    <segment/>
  </entity>
  <period>
    <instant>30-04-2001</instant>
  </period>
  <unit>
    <measure>iso4217:BRL</measure>
  </unit>
  <scenario name="Balanco Patrimonial">
    <bcbscenarioType>final</bcbscenarioType>
  </scenario>
</numericContext>
```

A seguir será discutido cada atributo e sub elemento dos elementos *context*.

3.4.3.3.1 O atributo *id*

Este atributo é obrigatório. O conteúdo deve começar com um caractere alfa. O atributo *id* é usado para identificar um contexto de forma que este possa ser referenciado pelos elementos, por meio dos atributos *numericContext* e *NonNumericContext*. Ex: c1.

3.4.3.3.2 O atributo *precision*

É um inteiro que informa a precisão aritmética da medida, válido apenas para contextos numéricos. Softwares diferentes podem necessitar níveis diferentes de precisão para os números que eles produzem. O atributo *precision* permite a qualquer autor de taxonomias declarar a precisão numérica da informação. Por exemplo, um valor 10 para esse atributo corresponde a uma precisão de dez casas decimais.

3.4.3.3.3 O atributo *cwa*

O atributo *cwa* é um tipo boleano. Se *cwa*="true", então o 'closed world assumption' é válido para o grupo de elementos deste contexto. Ou seja, aplicações podem assumir que a informação fornecida é definitiva. Se *cwa*="false", uma aplicação não deve calcular nenhum valor baseado na informação fornecida. Este atributo só é válido para contextos numéricos.

3.4.3.3.4 O subelemento *entity*

O elemento *entity* registra para qual organização (de negócio, governo, departamento, individual, etc.) o fato financeiro é válido. O elemento *entity*, que deve obrigatoriamente estar dentro do elemento *context*, contém um elemento *identifier*, obrigatório, e múltiplos elementos, opcionais, *segment*.

3.4.3.3.4.1 O subelemento *identifier*

Um subelemento *identifier* especifica um sistema para identificação de entidades de negócio e um identificador particular dentro do sistema. Uma entidade de negócio não precisa ser uma entidade de toda a corporação, pode ser uma subsidiária, uma divisão, ou até individual: qualquer unidade para a qual há uma transação financeira.

O atributo *scheme*, do subelemento *identifier*, contém o namespace de identificação, que fornece uma estrutura para se referenciar a entidade. O conteúdo do elemento deve ser uma *string*, que é um identificador válido dentro do namespace referenciado pelo atributo *scheme*.

3.4.3.3.4.2 O subelemento *segment*

A especificação XBRL não pode incluir todos os possíveis tipos de áreas de negócio na sua estrutura, especialmente aquelas que podem ser pertinentes para propósitos financeiros internos. Então, o elemento *segment* é usado como um recipiente para elementos adicionais, que podem ser utilizados por preparadores de documentos para identificar com mais detalhes uma área de negócio. Um exemplo [28] da utilização deste subelemento pode ser visto a seguir:

```
<group xmlns="http://www.xbrl.org/2001/instance"
xmlns:my="http://www.someCompany.com/scenarios" xmlns:other="http://www.example.com/">
  <nonNumericContext id="c1" >
    <entity>
      <!--required content -->
      <identifier scheme="www.dnb.com">1234567890</identifier>
      <!-- optional content -->
      <segment>
        <my:state>NJ</my:state>
      </segment>
    </entity>
    <!--conteúdo obrigatório -->
    <period>
      <instant>2001-17-12</instant>
    </period>
    <!-- optional content -->
    <scenario>
      <other:budgeted/>
    </scenario>
  </nonNumericContext>
</group>
```

O subelemento *segment* foi usado para indicar que o fato financeiro relata uma operação da companhia no estado de New Jersey, no dia 17 de dezembro de 2001.

3.4.3.3.5 O subelemento *period*

Todo fato financeiro aplica-se a um momento particular ou intervalo de tempo. O elemento *period* é subdividido em subelementos que são usados para representações diversas de data:

- instant - Define um momento exato no tempo;
- forever - Um elemento para representar “sempre”;
- startDate, duration - Um período começando numa data e com uma duração;
- duration, endDate - Um período com uma duração e data fim especificada;
- startDate, endDate - Um período com datas início e fim especificadas.

3.4.3.3.6 O subelemento *unit*

Unit especifica o padrão usado para a medida do elemento. O padrão ISO4217 para designação de moedas deve ser usado neste atributo.

No elemento *Unit* o prefixo do atributo deve ser previamente definido nas declarações dos namespaces.

3.4.3.3.7 O subelemento *scenario*

Fatos financeiros podem ser informados como *actual*, *budgeted*, *restated*, *pro forma*, etc. Em propósitos internos, pode haver até uma maior variedade de metadados adicionais, os quais os preparadores de documentos queiram associá-los com os elementos. O elemento *scenario*, opcional, permite incluir elementos adicionais para este propósito.

Os elementos *segment* e *scenario* têm exatamente a mesma estrutura, porém são usados para propósitos diferentes. *segment* é usado para especificar uma parte de uma entidade de negócio, enquanto *scenario* é usado para documentar determinada situação de um grupo de elementos.

3.4.3.4 O elemento *tuple*

Enquanto muitos dos fatos financeiros podem ser entendidos independentemente, alguns fatos são dependentes uns dos outros para a sua própria compreensão, especialmente se estão sendo informadas ocorrências múltiplas do fato. O elemento *tuple*, assim como o elemento *item*, é abstrato, portanto, elementos específicos de outras taxonomias podem estar

no grupo de substituição *tuple* e *tuples* pode conter *itens* e outros *tuples*. No exemplo [28], mostrado a seguir, pode ser observada a utilização deste elemento, onde em uma informação para a administração de uma companhia, cada nome de empregado deve estar associado ao cargo:

- Na taxonomia:

```
<schema xmlns=http://www.w3.org/2001/XMLSchema"
xmlns:xbrli="http://www.xbrl.org/2001/instance" >
  <element name="managementInformation" substitutionGroup="xbrli:tuple"/>
  <element name="managementName" type="xbrli:stringItemType"
    substitutionGroup="xbrli:item"/>
  <element name="managementTitle" type="xbrli:stringItemType"
    substitutionGroup="xbrli:item"/>
</schema>
```

- No linkbase:

```
<linkbase xmlns=http://www.xbrl.org/2001/XLink/xbrllinkbase"
xmlns:xlink="http://www.w3.org/1999/xlink" >
  <definitionArc xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"
    xlink:from="managementName" xlink:to="managementInformation"/>
  <definitionArc xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"
    xlink:from="managementTitle" xlink:to="managementInformation"/>
</linkbase>
```

- No documento XBRL:

```
<group xmlns=http://www.xbrl.org/2001/instance"
xmlns:ci="http://www.xbrl.org/us/gaap/ci/2001/us-gaap-ci-2001">
  <ci:managementInformation>
    <ci:managementName nonNumericContext="c1">Bill Ballmer</ci:managementName>
    <ci:managementTitle nonNumericContext="c1">Big Kahuna</ci:managementTitle>
  </ci:managementInformation>
</group>
```

3.5 Outros aspectos dos documentos XBRL

A validação de um documento contra o XBRL *Schema* é esperada, mesmo que algumas aplicações possam vir a ter a capacidade para processar um arquivo de dados XBRL sem fazer referência a qualquer taxonomia, normalmente a interpretação e o processamento de qualquer elemento XBRL são relativos ao conteúdo de uma taxonomia. Da mesma forma, é desejável a validação XML Schema de todas as taxonomias que um documento faz referência.

A figura 3.5 abaixo mostra como os documentos XBRL podem ser validados:

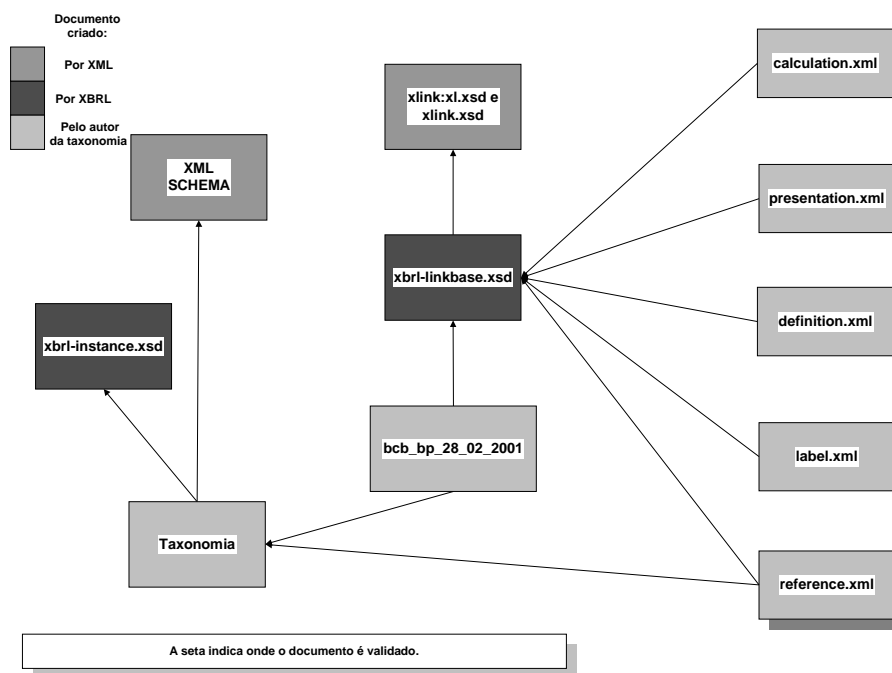


Figura 3.5 - Esquema de Validação dos Documentos

Há muitas aplicações que requerem que a transmissão de informações financeiras seja feita com segurança, com uma ênfase particular em integridade de dados e confiabilidade (conduzindo ao uso de meios de criptografia). XBRL não provê nenhum destes mecanismos, já que seu objetivo está na representação do conteúdo financeiro dentro de um acordo de formato; é assumido que como qualquer outro bloco de dados, integridade de dados pode ser aumentada com correção de erro, por criptografia, com assinatura de chave privada, ou outro mecanismo qualquer. Estes mecanismos estão todos fora do escopo de XBRL.

Um estudo mais detalhado de XBRL pode ser feito pela leitura de sua especificação, que é encontrada no sítio oficial do consórcio XBRL [28].

No próximo capítulo será feito um estudo de caso da aplicabilidade de XBRL, tomando-se como exemplo o ambiente de intercâmbio de informações financeiras do Banco Central do Brasil, pois nesta instituição além de ser possível se desenhar vários cenários para a utilização de XBRL, pode-se conduzir ao uso nas instituições do sistema financeiro e em outros tipos de organizações.

4. Estudo de caso: Fluxo de informações no Banco Central do Brasil

Como estudo de caso para se avaliar a aplicação de XBRL na representação de demonstrativos financeiros, foi selecionada uma situação específica do Banco Central do Brasil. Uma das razões para a escolha deste exemplo é que sendo um órgão de governo, regulador e fiscalizador, o Banco Central poderá impor a utilização de XBRL aos agentes do sistema financeiro nacional, que estão sob sua tutela, para a prestação das informações que por determinação legal são obrigados a lhe informar. Desta forma, a sua utilização poderá ser difundida nas instituições financeiras, o que poderá formar uma tendência de difusão do uso de XBRL por todos os outros segmentos da economia. Uma outra razão para a análise do uso de XBRL neste contexto é que por ser o autor desta dissertação funcionário do Banco Central, poderá haver maior facilidade de implantação de XBRL no ambiente computacional da instituição.

Então, caso a aplicação de XBRL no ambiente do Banco Central venha ocorrer, espera-se que a sua utilização seja rapidamente difundida, pois, se as instituições financeiras forem obrigadas a fornecer suas informações para o Banco Central em XBRL, o uso interno nessas organizações dessa linguagem de marcação tenderá a ser impulsionado.

A utilização de XBRL no Banco Central do Brasil poderá ser aplicada em três ambientes. O primeiro é o ambiente interno onde transitam as informações provenientes das instituições do sistema financeiro nacional e as informações produzidas internamente; o segundo é o ambiente de intercâmbio de informações entre as instituições financeiras e o Banco Central; e o terceiro ambiente é o sítio do Banco Central na Internet, com as informações que são disponibilizadas para o público usuário da Web. Todos esses ambientes estão interligados, já que uma mesma informação pode percorrer todos os três cenários. Nos itens seguintes, será feita uma análise de cada ambiente e de como XBRL poderá ser aplicado de forma a oferecer vantagens com a sua utilização.

Entre os diversos departamentos do Banco Central, o DEMAB possui um ambiente computacional que funciona de maneira autônoma com relação ao DEINF, o qual é responsável por toda a estrutura tecnológica dos outros departamentos, por isso nos próximos quatro itens será analisado a estrutura dos ambientes do DEINF, para em seguida ser analisado o ambiente computacional do DEMAB.

4.1 Modelo Atual

O atual modelo do fluxo das informações financeiras, no DEINF, nos três ambientes mencionados acima são descritos a seguir e poderão ser visualizadas pela figura 4.1.

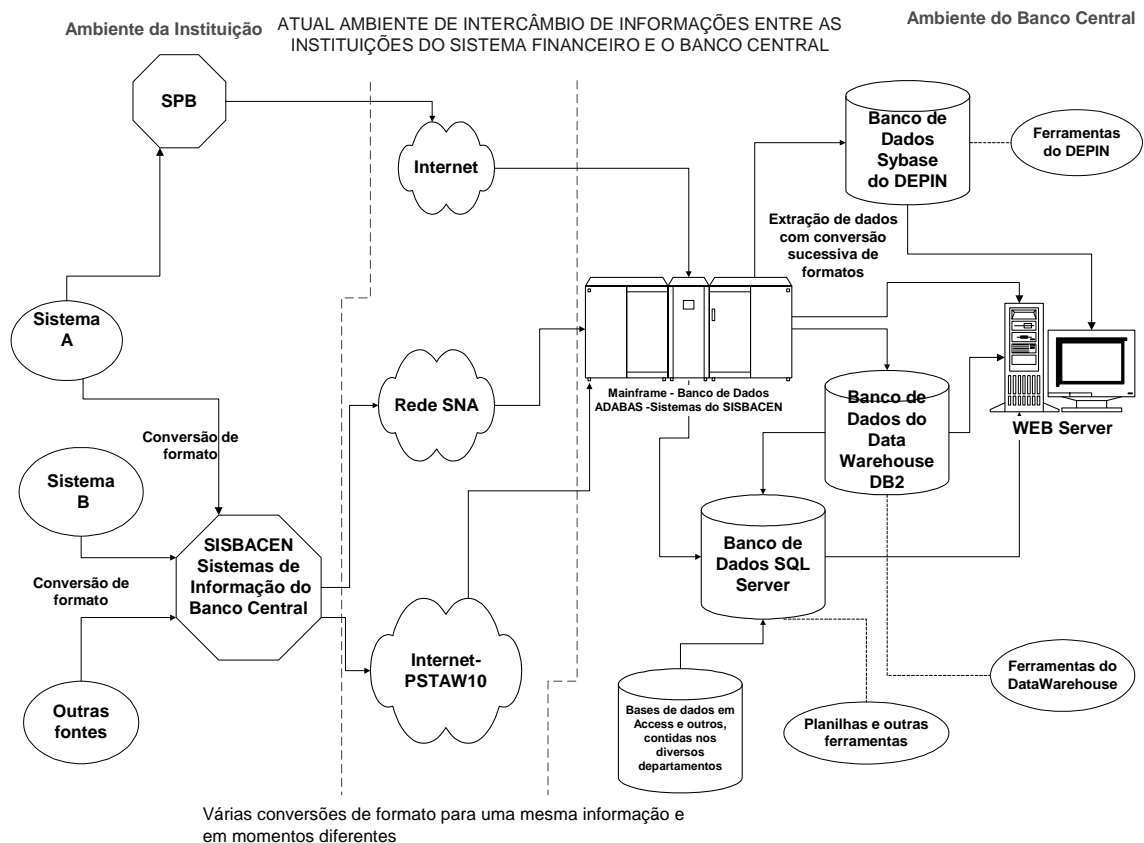


Figura 4.1 – Ambiente atual de intercâmbio de informações financeiras entre o Sistema Financeiro Nacional e o Banco Central do Brasil

4.1.1 Ambiente interno de intercâmbio de Informações

O Banco Central possui uma infra-estrutura computacional composta por computadores IBM de grande porte, com as plataformas OS/390, VM, Sun Solaris, IBM AIX e Windows, banco de dados Adabas, DB/2 e SQL Server, linguagem de programação Natural, Cobol, SQL e cerca de 4.000 estações de trabalho de computadores PC interligadas por redes Windows 2000. Nas gerências regionais e nos departamentos centrais são desenvolvidos sistemas estruturados em plataformas de pequeno porte, redes locais, Windows, Access, VisualBasic, etc. O DEPIN (Departamento de Operações das Reservas Internacionais) faz uso de uma rede para a mesa de operações, baseada na linguagem PowerBuilder e no banco de dados Sybase. O DEINF disponibiliza para os usuários ferramentas para consultas, emissão de relatórios e extração de dados baseadas em recursos do Natural/Adabas e um Datawarehouse com recursos do DB/2 e da ferramenta DSS da Microstrategy.

As informações contidas no DB/2 para o DataWarehouse são provenientes do Adabas e de outras fontes, como planilhas, fontes externas, etc. O DEINF, também, coloca à

disposição dos usuários informações para consulta em um banco de dados SQL Server, cuja fonte são os bancos de dados ADABAS, DB/2, Access e planilhas Excel.

Atualmente, o DEINF está migrando a plataforma de desenvolvimento de sistemas para um ambiente com tecnologia de Orientação a Objetos, aplicado à Web, utilizando Java como linguagem de programação.

Neste contexto interno a aplicação de XBRL como linguagem padrão para representação dos dados financeiros poderá contribuir, juntamente com a nova tecnologia baseada em Java, para integrar as diversas plataformas existentes no Banco Central.

4.1.2 Ambiente de intercâmbio de Informações com as Instituições Financeiras

O Banco Central do Brasil possui uma relação de intercâmbio de informações com as instituições do mercado financeiro que estão sob sua supervisão, cujo trâmite ocorre por uma das seguintes maneiras:

- usando rede com o protocolo SNA (System Network Architecture) da IBM para conexão com o SISBACEN – SISTEMA DE INFORMAÇÕES DO BANCO CENTRAL DO BRASIL, que é um complexo informacional administrado pelo Banco Central do Brasil destinado ao tratamento, armazenamento e recuperação online de dados e informações, com atualização em tempo real. O complexo SISBACEN, cuja plataforma computacional é composta por computador de grande porte, usando a linguagem de programação Natural e banco de dados ADABAS, incorpora uma sofisticada rede de teleinformática que interliga o Banco Central às instituições do sistema financeiro nacional, órgãos do governo federal e dos poderes legislativo e judiciário e entidades estatais;
- por meio da Internet para o acesso ao SISBACEN em duas categorias: o acesso Público e o acesso Pessoa Física e Institucional. O acesso Público permite a qualquer pessoa consultar informações de interesse geral divulgadas pelo Banco Central usando o SISBACEN, como os normativos do Banco Central, taxas diversas, cotações de moedas, indicadores econômico-financeiros, resultados de leilões de câmbio entre outras informações. O acesso às transações com conteúdo de interesse particular (acompanhamento de processos, situação na Central de Riscos, Registro Declaratório dos Investimentos Externos Diretos, Registros de Operações Financeiras, etc.), encontram-se no acesso Pessoa Física e Institucional e exigem senha individual. Para ter disponível esse acesso, emulando um terminal de computador de grande porte IBM-3270, os usuários utilizam uma VPN (Virtual Private Network), que usa o protocolo TELNET para estabelecer uma conexão on-line com o computador de grande porte, onde executam o software PASCW10, desenvolvido para esse fim. O PASCW10 pode ser

obtido junto às representações regionais do Banco Central ou por meio do seu próprio sítio na Internet. A opção de acesso Institucional via Internet está disponível apenas para o uso das transações do SISBACEN relativas ao Registro Declaratório Eletrônico dos Investimentos Externos Diretos (RDE-IED), ao Cadastro de Inadimplentes e à Central de Risco de Crédito. Este acesso é permitido para as empresas receptoras do investimento externo direto (e/ou pelo(s) representante(s) no País dessas empresas e dos investidores não-residentes), cooperativas de crédito e servidores do Banco Central. As demais instituições financeiras autorizadas a funcionar pelo Banco Central continuam a utilizar a rede SNA;

- pelo uso da Internet, com as ferramentas Connect Direct, da Sterling Software, e PSTAW10, desenvolvido internamente, para o intercâmbio de arquivos de documentos texto;
- e, por fim, usando-se tecnologia de mensagens XML assinadas digitalmente, para o trânsito de informações na Internet, para o Sistema de Pagamentos Brasileiro (SPB) por meio do software MQ-Series da IBM e para a Central de Risco com as ferramentas Connect Direct e PSTAW10.

Com exceção das informações para o SPB e Central de Risco os demais sistemas têm entrada de dados para o SISBACEN via rede SNA ou via PSTAW10, entre eles estão os sistemas de Contabilidade, Pessoal, Patrimônio etc. Especificamente para as transferências de informações via PSTAW10, os documentos que podem ser transferidos aos computadores do Banco Central são os que estão na tabela 4.1 [34] seguinte:

DOCUMENTO	Conteúdo/Assunto
0001	Envio de Informações de Patrimônio Líquido
0002	Declaração do Censo de Capitais Estrangeiros
0585	Cédulas de Financiamento Agrícola – RECOR
1001	Devedores do Setor Público – CADIN
1010 1020 1030	CADIP - Dados Cadastrais – Cadastro da Dívida Pública CADIP - Dados de Movimento CADIP - Informações Mensais
2010	Informações de Consórcios
2020	Exposição Cambial
2030	Exposição Cambial
2116 2126	Fundo Garantidor de Crédito – FGC
2211	Informações sobre Operações de Crédito
2310 2320	Risco de Taxa de Juros Prefixada
2510	Fundos – CVM
3010	Devedores/Central de Risco , Substituição de Informações
3020 3026 3030	Central de Risco de Crédito
4010 4016 4020 4026 4040 4150 4510 4046 4110 4350 4500	Plano Contábil das Instituições do Sistema Financeiro Nacional – COSIF
4033	Estatísticas Bancárias Internacionais
5011	Participações das Instituições - Res. 2674
5816	Cartão de Crédito Internacional – emitido no país
5817	Cartão de Crédito Internacional – emitido no exterior

Tabela 4.1 – Documentos enviados ao Banco Central por meio do aplicativo PSTAW10.

Como pode ser observado pela tabela acima, os documentos que podem ser transmitidos por meio do PSTAW10 são em sua grande maioria documentos financeiros. Portanto, é neste contexto, transmissão de informações via rede SNA e PSTAW10, que XBRL pode ser aplicado para o intercâmbio de informações entre o Banco Central e as instituições financeiras, já que seu foco principal é a representação de relatórios de informações financeiras, principalmente para a área contábil, podendo, entretanto, ser estendido para outros tipos de relatórios financeiros devido à sua característica de extensibilidade.

4.1.3 Sítio do Banco Central

O sítio do Banco Central do Brasil na Internet contém inúmeras informações úteis ao mercado financeiro brasileiro, às instituições acadêmicas e de estudos econômicos, aos consultores financeiros e econômicos, etc. Entre elas podem-se citar os demonstrativos contábeis, as taxas de câmbio, séries históricas de informações econômico-financeiras, normativos emitidos pelo Banco Central, etc.

Atualmente o sítio do Banco Central na Internet é construído baseado em linguagem HTML, com elementos que especificam estritamente a forma de apresentação. Os elementos HTML não carregam informação semântica, o que torna o acesso, extração e processamento das informações, por qualquer tipo de aplicação, muito trabalhoso e difícil. Esses problemas poderão ser solucionados com a representação das informações financeiras usando a linguagem XBRL, que permite que aplicações possam ser construídas para extrair e processar os dados. Desta forma, os usuários das informações contidas no sítio do Banco Central, poderão construir suas próprias aplicações para o processamento das informações que lhes interessam, tornando o manuseio dos dados mais acessível e democratizando o uso da informação.

4.2 Modelo Proposto

Este modelo de comunicação, para os três contextos acima descritos contém problemas bastante conhecidos, como por exemplo:

- pouca flexibilidade, já que as soluções de integração entre sistemas existentes são em geral específicas, o que provoca lentidão no seu desenvolvimento e baixa portabilidade e interoperabilidade entre aplicações;
- necessidade de adaptação dos sistemas por parte das instituições para prestarem informações no formato desejado pelo Banco Central;
- dificuldade para extração e processamento da informação;
- grande desprendimento de esforço computacional e humano.

Então, a utilização de XBRL como padrão para o intercâmbio de dados poderá trazer vantagens significativas não só no intercâmbio de informações entre as instituições financeiras e o Banco Central do Brasil, mas também entre seus departamentos. Por exemplo, um departamento 'A' usa um determinado banco de dados para armazenar comportamentos das instituições, enquanto o departamento B usa um outro banco de dados para processar informações dessas mesmas instituições. Os dois departamentos possivelmente têm interesses

comuns e suas transações e comportamentos estudados se complementam para melhor análise do perfil das instituições. O processo tradicional de exportar dados de ambos os sistemas, usando um programa de conversão para traduzí-los para um formato específico e produzir um relatório, funciona bem, porém, este processo é relativamente lento e, por causa da transformação de formatos, freqüentemente requer muito trabalho. Com XBRL a necessidade de converter formatos não é removida, porém, a conversão de dados é efetuada automaticamente e de uma vez para os formatos desejados, diminuindo a possibilidade de erro por sucessivas transformações de formato dos dados.

Com uma taxonomia estabelecida, cada dado de cada sistema pode ser identificado, comparado e empacotado em um documento XBRL, que pode ser formatado para apresentação usando uma linguagem de transformação que converta o documento XBRL no formato desejado, por exemplo, XSLT. Uma vez criado o ambiente de intercâmbio de informações usando XBRL, este poderá ser aplicado em situações semelhantes de diversos tipos de bancos de dados e sistemas. Ou seja, o problema de integração de sistemas com XBRL é mais facilmente resolvido.

Entre outras vantagens que XBRL pode trazer ao Banco Central é possível citar:

- a padronização, isto é, um padrão independente de fabricante, como XBRL, facilita todo o processo de utilização da informação. No ambiente interno do Banco Central a padronização do formato de representação da informação poderá facilitar o intercâmbio de informações entre os departamentos, disponibilizando a informação mais prontamente;
- informação estruturada e elementos XBRL preservam informação contextual e semântica sobre as partes do documento;
- informação necessita ser preparada de forma que visualmente seja legível ao usuário, mesmo que o processo de transferência ocorra somente entre máquinas, em algum momento uma pessoa estará nele envolvido e XBRL torna isso factível;
- XBRL democratiza a informação, alcançando uma quantidade de usuários que qualquer outra tecnologia não consegue, já que permite a estruturação da informação na Web, que utiliza protocolo de comunicação TCP/IP, favorecendo a interconexão de qualquer tipo de computador, de qualquer porte, qualquer fabricante, com qualquer sistema operacional e arquitetura.
- usando a Intranet e representando as informações em XBRL, os departamentos poderão disponibilizar mais facilmente suas informações para que outro departamento faça uso de forma automatizada, podendo inclusive descobrir novas utilidades para as aplicações,

simplesmente por estarem as informações disponíveis em um formato que facilita o processamento pelo usuário;

- aceitação e suporte do formato de um documento é importante quando um grande público alvo está envolvido (como no caso da Internet), neste caso ferramentas para leitura e manipulação do formato devem estar disponíveis. XBRL, por ser baseada em XML, já oferece uma grande gama destas ferramentas;

- documentos XBRL podem transitar entre aplicações baseadas em diversos protocolos já estabelecidos, como o HTTP. O Banco Central possui ambientes que são incompatíveis, como Unisys, Windows, MVS, etc. Com a informação em XBRL, a facilidade da sua extração e processamento poderá facilitar a interoperabilidade entre estes ambientes;

- dados representados em XBRL possuem seus metadados no documento de taxonomia e linkbases, o que facilita o entendimento da informação;

- XBRL permite associação entre elementos através de *hyperlinks*, que permitem construir poderosas aplicações com *links* multi-direcionais e *links* externos (onde a informação do *link* é mantida fora do documento); Por exemplo, *links* podem ser criados entre a conta Reserva Bancária de cada instituição financeira e relatórios que demonstram a solidez da instituição. Uma aplicação então pode buscar o saldo da Reserva Bancária e trazê-lo para os documentos que representem esses relatórios.

- transformação sucessiva de formatos é eliminada, reduzindo a quantidade de erros e trabalho de conversão. Atualmente, no Banco Central, informações de diversas fontes estão sendo colhidas e disponibilizadas para os usuários em um banco de dados SQL Server. Para que isso ocorra, o usuário tem que fazer um pedido ao DEINF, que elabora um programa para extração da informação no banco de dados onde ela está armazenada e a disponibiliza no SQL Server. O usuário, então, poderá criar seu programa para usar a informação, ou usar aplicativos como os da Microsoft, Access e Excell, que extraem a informação diretamente no SQL Server. Recebendo, o Banco Central, a informação em XBRL, esta poderá ser distribuída automaticamente e de uma só vez para os diversos banco de dados, inclusive o SQL Server;

Aspectos positivos na utilização de tecnologias derivadas de XML (XBRL, DOM, XSLT, etc.) são por serem de uso livre e pelo fato de o Banco Central estar adotando Java como a linguagem padrão para desenvolvimento de seus sistemas, facilitando, portanto, uma melhor integração na aplicação dessas tecnologias, minimizando problemas de portabilidade e interoperabilidade e proporcionando facilidades na reutilização de códigos e manutenção de sistemas.

A inserção de XBRL no contexto do Banco Central produz graficamente o ambiente mostrado na figura 4.2 seguinte:

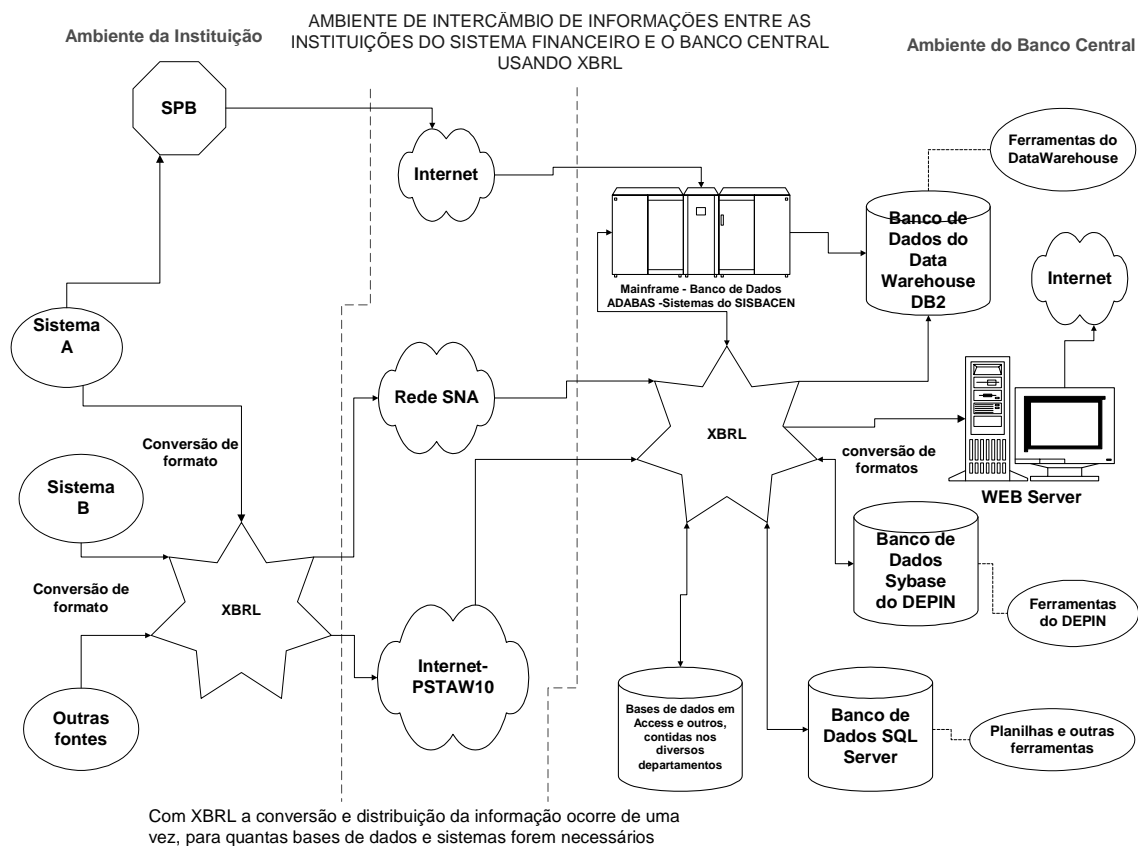


Figura 4.2 – Ambiente de intercâmbio de informações financeiras entre o Sistema Financeiro Nacional e o Banco Central do Brasil, usando XBRL

Com a visualização gráfica fornecida pela figura 4.2, pode ser observado que a utilização de XBRL simplifica o processo de distribuição da informação financeira no contexto do Banco Central.

4.3 O impacto da aplicação de XBRL no Banco Central

As informações recebidas pelo Banco Central possuem na sua grande maioria como fonte de dados as instituições financeiras, como ilustrado pela figura 4.3 [34]. Informações que constituem, de um modo geral, relatórios financeiros.

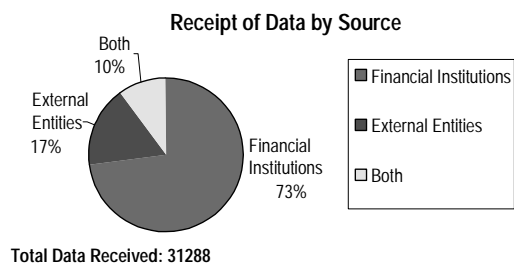


Figura 4.3 – Dados recebidos pelo Banco Central por tipo de fonte.

O recebimento das informações pelo Banco Central por meio do SISBACEN e PSTAW10 contempla, pela observação da figura 4.4 [34], 79% dos dados.

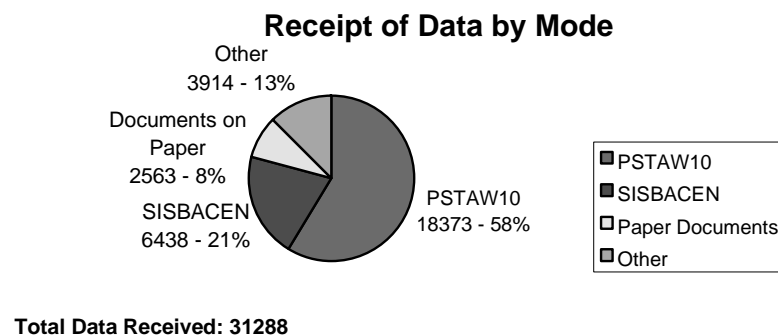
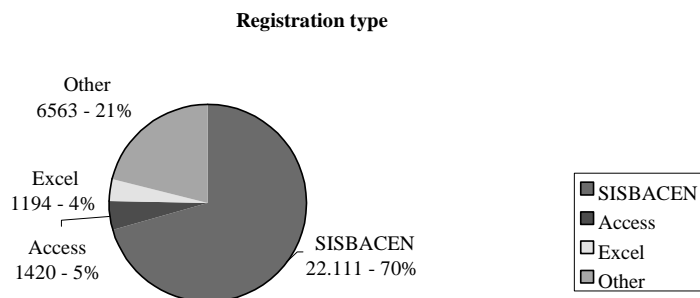


Figura 4.4 – Modo de recebimento de dados pelo Banco Central.

Pela análise das duas figuras acima, percebe-se que a utilização de XBRL para representar as informações financeiras causará um forte impacto no ambiente de intercâmbio de informações financeiras do Banco Central, já que pelo menos 73% dos dados são provenientes das instituições financeiras e estes são transferidos de forma expressiva (79%) usando os sistemas SISBACEN e PSTAW10 e, como visto nos subitens anteriores, este é um dos contextos em que XBRL pode ser inserido.

Um outro aspecto que demonstra o impacto que a adoção de XBRL poderá causar, é pela análise do local onde os dados são registrados. A figura 4.5 [34] indica que a maior parte da informação é registrada usando o SISBACEN, que sofre, no ambiente interno do Banco, conversão para os outros formatos existentes. Com XBRL, a distribuição destas informações pode ser feita instantaneamente, minimizando a necessidade de conversão sucessiva de formatos.



Total Data Received: 31185

Figura 4.5 – Tipo de registro de dados no Banco Central.

Portanto, sendo a maioria das informações financeiras no Banco Central provenientes das instituições financeiras (73%), que utilizam para transmissão os sistemas SISBACEN e PSTAW10 (79 %) e com os registros efetuados majoritariamente no SISBACEN (70%), a adoção de XBRL, para representação destas informações, seria refletida significativamente em seu ambiente computacional.

4.4 Ambiente do DEMAB

O DEMAB trabalha integrado ao SPB, usando XML para troca de mensagens com as instituições financeiras e alguns subsistemas do DEINF, usando a chamada RSFN, ou seja, Rede do Sistema Financeiro Nacional. Rede privada de dados com protocolo TCP/IP, criada com a finalidade de suportar o tráfego de mensagens XML entre as instituições financeiras titulares de conta Reservas Bancárias, as câmaras e os prestadores de serviços de compensação e de liquidação, a Secretaria do Tesouro Nacional - STN e o Banco Central do Brasil, no âmbito do SPB [34]. Neste ambiente que envolve o DEMAB, além das mensagens XML do SPB, há também uma troca diária de arquivos de dados via FTP com o DEINF e eventual com o sistema financeiro.

Além do SPB, o DEMAB se comunica com as instituições e o mercado via terminais virtuais do mainframe, disponíveis para aquelas instituições que não fazem parte do SPB (corretoras, distribuidoras de valores, etc.) e por meio da Web. A maioria das informações disponíveis na Web são provenientes do mainframe, cujos dados são convertidos para o formato HTML e exibidos na página do SELIC na Rede de Telecomunicações para o Mercado (RTM), administrada pela Associação Nacional das Instituições do Mercado Financeiro (Andima). O SELIC - Sistema Especial de Liquidação e Custódia é um sistema eletrônico que processa o registro, a custódia e a liquidação financeira das operações

realizadas com títulos públicos, sistema administrado em conjunto pelo DEMAB e pela ANDIMA.

Internamente ao DEMAB, o banco de dados do mainframe Unisys supre com informações diversos sistemas (por exemplo de contabilidade) e outros bancos de dados de plataforma baixa. Existe também o banco de dados Sybase que atende as operações do SELIC, com aplicativos próprios de contabilidade, controle, etc. As informações que são armazenadas no Sybase são provenientes do Unisys e de mensagens do SPB.

O ambiente computacional do DEMAB pode ser visualizado na figura 4.6.

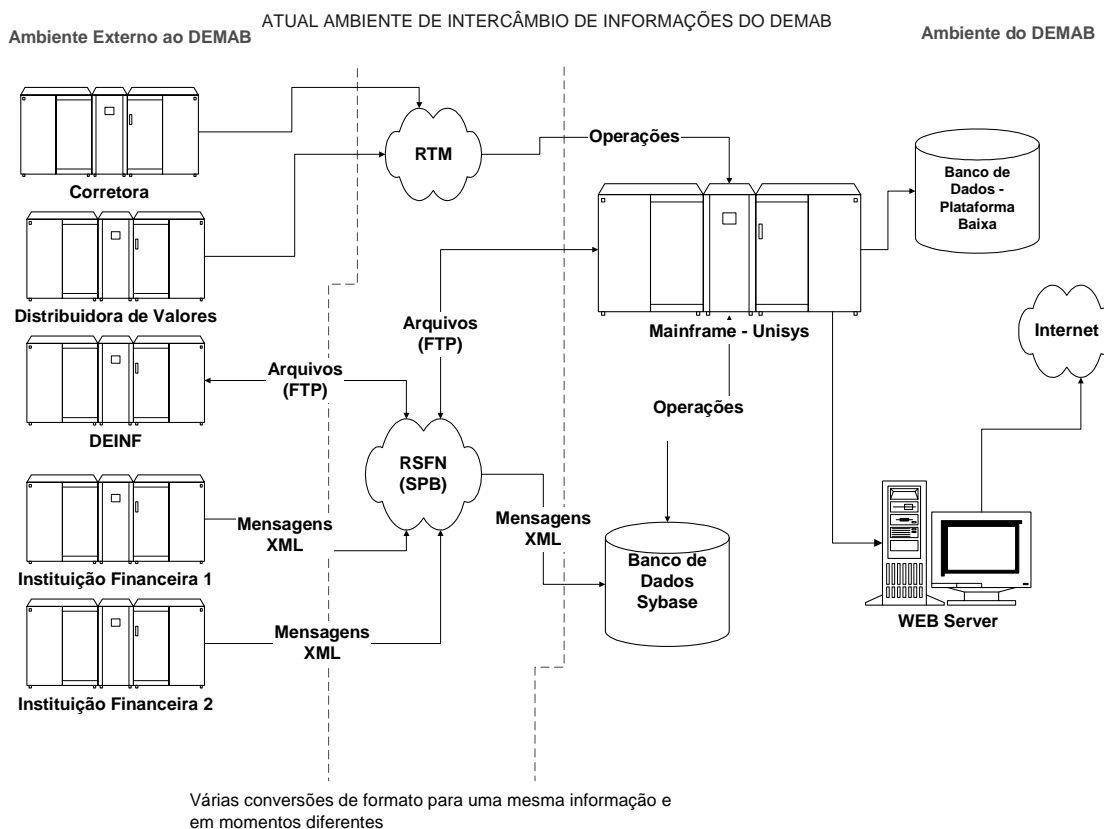


Figura 4.6 – Atual ambiente de intercâmbio de informações financeiras no DEMAB

Por consequência, o uso de XBRL no DEMAB poderá trazer benefícios semelhantes aos anteriormente mencionados, quando aplicado ao ambiente do DEINF. Para o intercâmbio dos relatórios financeiros, principalmente os contábeis, a aplicação de XBRL no ambiente do DEMAB pode ser graficamente visualizada pela figura 4.7.

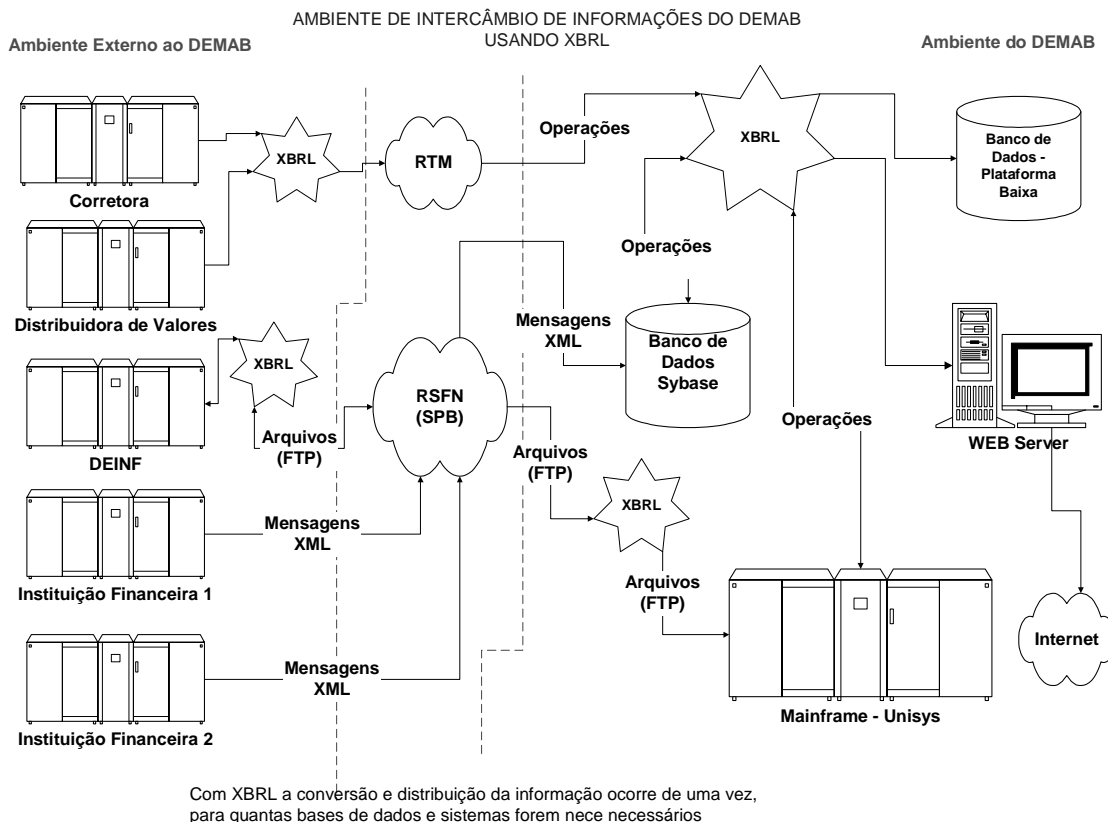


Figura 4.7 – Ambiente de intercâmbio de informações financeiras no DEMAB, usando XBRL.

Para comprovação da aplicabilidade de XBRL e das suas vantagens em relação ao cenário atual das informações financeiras no Banco Central, serão utilizados balanços contábeis disponíveis no seu sítio na Internet [34] como fonte para o exemplo. No capítulo seguinte são apresentados esses documentos no formato XBRL, a taxonomia, os *linkbases*, bem como a folha de estilo para apresentação na Web e uma aplicação que extrai as informações contidas no documento e armazena em um banco de dados, as quais posteriormente são usadas para gerar uma nova página da Internet, com informações úteis para análise financeira. Este exemplo, apesar de apenas representar as informações de um tipo de relatório financeiro que o Banco Central disponibiliza em seu sítio na Internet, também será útil para se avaliar, por analogia, como poderá vir a ser a forma como as instituições financeiras poderão prestar as informações que são obrigadas a fornecer ao Banco Central e como a informação financeira poderá circular internamente nos diversos tipos de organizações.

5. Estudo de caso: representando relatórios financeiros em XBRL

Nos capítulos anteriores foi visto que XBRL apresenta-se como uma boa alternativa ao Banco Central para a representação e para o intercâmbio de relatório de informações financeiras. Sendo uma linguagem apropriada tanto para a estruturação das informações contidas no sítio na Internet, como para o recebimento das informações enviadas a este órgão pelos agentes do sistema financeiro e para a integração entre os sistemas internos existentes.

Como comprovação das facilidades que podem ser trazidas por XBRL é apresentado na seqüência um exemplo de sua utilização com a construção de uma página que disponibiliza demonstrativos contábeis do Banco Central em um navegador Web e com a extração e processamento das informações contidas nesta página. Este exemplo poderá servir de parâmetro para a análise de como as instituições financeiras poderão enviar suas informações ao Banco Central e como poderá ser disponibilizada as suas informações financeiras na Web. Para isso foram construídos um exemplo de taxonomia, dos documentos *linkbases*, do documento XBRL e duas folhas de estilo, baseada na tecnologia XSLT, que transforma o documento XBRL em HTML, para publicação na Web. Para demonstração de como essa informação representada em XBRL pode ser extraída e utilizada foram desenvolvidas duas aplicações. A primeira, usando a linguagem Java e DOM, teve como função extrair os dados presentes no documento XBRL e armazená-los em um banco de dados SQL Server. A segunda aplicação, desenvolvida com JSP, executa uma consulta a esses dados armazenados e processa-os gerando índices analíticos financeiros que são utilizados para a criação de uma outra página para ser apresentada em um navegador Web.

5.1 Definição do documento e extração dos dados

Para apresentação da utilização de XBRL foram usados os dados presentes em três balanços patrimoniais do Banco Central, disponíveis em sua página na Internet. As informações ali prestadas foram analisadas para a definição dos elementos, da sua estrutura, dos seus relacionamentos e para a criação dos documentos XBRL.

O processo de extração destas informações ocorreu de forma não automatizada, já que as páginas estão codificadas com HTML. Porém para o carregamento dos dados das diversas fontes para uma página XBRL, pode-se usar as mesmas técnicas automatizadas já utilizadas pelo Banco Central, para publicação na Web no atual formato. Nos parágrafos seguintes é feita uma descrição de como esse processo ocorre.

A maior parte dos aplicativos do Banco Central usados para suprir informações para o seu sítio na Internet acessa o banco de dados Adabas através do embires Brunir (ferramenta da

SoftwareAG) que se divide praticamente em duas partes distintas. Uma parte é instalada no *mainframe* e fica continuamente sendo executada, esperando e respondendo chamadas do cliente. A outra parte, cliente, é instalada no servidor Web. O EntireX atua como um *middleware* fazendo a ponte entre a aplicação e os dados no banco de dados do *mainframe*, o Adabas, usando o protocolo de rede TCP/IP.

No servidor Web, o EntireX pode ser utilizado de duas formas, uma é usando linguagens de programação, como ASP, VB, JSP, JAVA, DELPHI, que instanciam um objeto do tipo EntireX, que com seus métodos e propriedades proporcionam o acesso aos dados. A outra forma é por meio do software iXpress, também da SoftwareAG, que trabalha com EntireX e que foi a primeira ferramenta utilizada pelo Banco Central para extração de dados do Adabas, ainda hoje sendo utilizada nos principais aplicativos (taxas de câmbio, busca de normativos, conversão de valores e moedas, etc). Seu uso atualmente está sendo reduzido em razão da possibilidade de utilização do EntireX com outras linguagens, apesar da vantagem de se poder fazer *cache* diferenciado, ou seja, fazer separadamente *cache* de cada página, diminuindo o acesso ao banco de dados.

No *mainframe* existe um *job* que fica em execução indefinidamente e que aguarda uma chamada do cliente. Com a requisição do cliente é passado ao *job* alguns parâmetros, entre eles o nome do subprograma que deve ser executado e que faz o acesso aos dados utilizando a regra de negócio especificada. Como resultado do trabalho do *job* são devolvidos ao cliente os parâmetros que contêm os dados. No cliente, esta resposta é formatada de acordo com regras pré-definidas para associar elementos HTML aos dados. Geralmente os elementos são criados no lado cliente por vários motivos, entre eles o tamanho limitado à 30Kb (no *mainframe*) da área que contém os parâmetros.

Então, por exemplo, usando como linguagem de programação ASP, um objeto do EntireX é instanciado, sendo executado um método que faz a chamada e passagem de parâmetros para o *mainframe*. No *mainframe*, estes parâmetros geram a execução de um subprograma, este subprograma extrai os dados, que são devolvidos ao ASP, que formata a *string* de texto, incluindo os elementos HTML, e entrega o resultado ao servidor Web e, portanto, ao usuário.

Este é o atual processo de carregamento das informações para o sítio, o qual pode-se com pequenas adaptações ser usado para a construção de documentos XBRL

5.2 Taxonomia

Um documento XBRL de taxonomia está baseado na linguagem XML Schema, onde o autor da taxonomia define os elementos a serem usados. No estudo de caso, a taxonomia criada está no documento `bcb_taxonomia.xml`, apêndice D. A seguir será feita uma discussão, já iniciada no capítulo 3, de cada elemento usado na criação da taxonomia, cujo entendimento pode ser facilitado com a consulta às tabelas do apêndice L.

O elemento *schema*, é o elemento raiz da taxonomia, definido em XML Schema, que contém as definições dos elementos XBRL, com seus atributos, que serão usados pelo documento XBRL. O atributo *targetNamespace* do elemento *schema* informa que os elementos definidos na taxonomia pertencem a este *namespace*, ou seja, documentos que usarem estes elementos deverão fazê-lo usando como referência a URI: "`www.bcb.gov.br`".

Os outros *namespaces* são usados para identificar os elementos definidos por XML Schema, XML Link, XHTML e XBRL:

```
<schema
  targetnamespace="www.bcb.gov.br"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:link="http://www.xbrl.org/2001/XLink/xbrlLinkbases"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  xmlns:xbrli="http://www.xbrl.org/2001/instance"
  xmlns:bcbs="http://www.bcb.gov.br/bcb_taxonomia.xsd"
  elementFormDefault="qualified">
<annotation>
  <appinfo>
    <link:linkbasesRef xlink:type="simple"
      xlink:href="bcbs_presentation.xml"
      xlink:actuate="OnRequest"
      xlink:role="http://www.xbrl.org/linkprops/linkRef/presentation"
      xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbases">
    <xhtml:p>Links para relacionamentos de apresentação</xhtml:p>
    </link:linkbasesRef>
  </appinfo>
</annotation>
  . . .
```

O *namespace* `xmlns:bcbs="http://www.bcb.gov.br/bcb_taxonomia.xsd"` associa o prefixo "bcbs" com a URL onde se encontra a taxonomia, para que documentos XBRL e outros documentos possam referenciar os elementos definidos nesta taxonomia.

O atributo *elementFormDefault*, definido por XML Schema, é usado para informar como deve ser processada a informação do *namespace*, neste caso todo elemento deve estar escrito de forma qualificada, ou seja, com o prefixo do *namespace* ao qual pertence.

Os elementos *annotation* e *appinfo* são elementos opcionais usados pelo autor da taxonomia para inserir informações específicas para aplicações, neste caso informando quais *linkbases* estão disponíveis na taxonomia.

Os elementos *linkbasesRef* dão a direção de onde encontrar os *linkbases* (*calculation*, *presentation*, *definition*, *label* e *reference*). Eles diferem entre si pela informação do arquivo

onde se encontram e pelo atributo *role*, que informa o tipo do *linkbase*. Os atributos pertencentes a este elemento foram previamente discutidos no capítulo 3.

O elemento *xhtml* não pertence às definições de XBRL, apenas foi colocado para exemplificar como estes podem ser adicionados aos documentos XBRL.

O elemento *element* define os elementos XBRL que serão usados no documento XBRL, como por exemplo:

```
<element name="ativo" id="ativo" type="xbrli:monetaryItemType" substitutionGroup="xbrli:item">
```

Os atributos dos elementos têm as seguintes funções:

- *name* - atribui um nome a um elemento XBRL;
- *id* - este atributo deve ser igual ao atributo name. Ele permite que aplicações possam mais facilmente encontrar o elemento;
- *type* - informa o tipo de dado do elemento;
- *substitutionGroup* - este atributo informa que o elemento pertence àquele grupo de substituição e contém as características do elemento de substituição.

O elemento *scenarioType* pode ser usado para caracterizar um determinado cenário de um fato financeiro, nesse caso foram definidos dois cenários – “final” e “preliminar”, caracterizando assim em que estágio se encontra a informação:

```
<element name="scenarioType">
  <simpleType>
    <restriction base="string">
      <enumeration value="final"/>
      <enumeration value="preliminar"/>
    </restriction>
  </simpleType>
</element>
```

Os próximos elementos foram definidos para serem usados na denominação dos diretores da instituição e responsáveis pela emissão do relatório representado no documento XBRL:

```
<element name="diretores" type="string" substitutionGroup="link:part">
  <element name="presidente" type="string" substitutionGroup="link:part"/>
  <element name="diretorDipec" type="string" substitutionGroup="link:part"/>
  <element name="diretorDinor" type="string" substitutionGroup="link:part"/>
  <element name="diretorDirad" type="string" substitutionGroup="link:part"/>
  <element name="diretorDipom" type="string" substitutionGroup="link:part"/>
  <element name="diretorDifis" type="string" substitutionGroup="link:part"/>
  <element name="diretorDirex" type="string" substitutionGroup="link:part"/>
  <element name="diretorDifip" type="string" substitutionGroup="link:part"/>
  <element name="chefedeafi" type="string" substitutionGroup="link:part"/>
</element>
```

5.3 Linkbases

Linkbases Label - A seguir é feita uma descrição dos elementos usados para compor o *linkbase Label*, definidos no documento *bcbl_label.xml*, apêndice C. O elemento raiz de um *linkbase* XBRL é um elemento *linkbases*:

```
<linkbases
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbases"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbases xbrl-linkbases.xsd">
  <labellink xlink:type="extended"
xlink:role=http://www.xbrl.org/linkprops/extended/BalancoPatrimonial>
  . . .
```

O *namespace default*, definido pela especificação XBRL, para esse elemento é `xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbases"`, o que significa que quando elementos sem prefixo aparecerem neste documento, pertencerão a este *namespace*.

O atributo *xsi:schemaLocation* fornece a localização do documento XML Schema que descreve os elementos de construção do *linkbase*. A primeira parte do atributo, antes do espaço, mostra a URI onde ele está localizado, a segunda parte informa o documento que especifica os elementos pertencentes a este *namespace*. A definição deste atributo ocorre na especificação XML Schema.

O elemento *labellink* é um *link* estendido que contém elementos do tipo *arc*, *resource* e *locator*, os quais definem os relacionamentos entre os elementos e seus *labels*. O atributo *xlink:type* é sempre *extended* e o atributo *xlink:role* informa qual o tipo de relatório financeiro este documento representa, no caso um balanço patrimonial.

Um elemento *label* do tipo *resource*, exemplificado a seguir, filho de *labellink*, é a fonte do texto do *label*. Este fornece um *label* ao elemento do tipo *arc*, especifica o idioma e o texto do *label*:

```
<label xlink:type="resource" xlink:label="bcb_ativo_br" xlink:title="bcb_ativo_br"
  xlink:role="http://www.xbrl.org/linkprops/label/total" xml:lang="br">ATIVO TOTAL</label>
```

O elemento *locator*, mostrado a seguir, tem como propósito fornecer a localização do elemento a que o *label* se refere:

```
<loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#ativo" xlink:label="bcb_ativo"
xlink:title="ativo"/>
```

Seus atributos, já comentados no capítulo 3, são:

- *xlink:type* - sempre *locator*;
- *xlink:href* - este contém uma expressão XPOINTER que identifica o elemento que se deseja fornecer o *label*, o que significa que o elemento no arquivo *bcb_taxonomia.xsd* possui o atributo *id*, cujo valor é *ativo*;
- *xlink:label* - um *label* do elemento especificado no atributo *href*. Este atributo *xlink:label* pode ser usado por um elemento do tipo *arc* para se conectar a outro elemento dentro do mesmo elemento estendido *labellink*;
- *xlink:title* - atributo opcional.

Elementos *labelArc* são elementos do tipo *arc*. Eles especificam ligações entre elementos por referência aos valores do atributo *xlink:label* do elemento *label*. Possuem, também, outras informações que indicam a direção do arco (*label* para o elemento ou vice-versa), sendo possível se criar *links* em ambas as direções. Cada direção deve possuir seu próprio elemento do tipo *arc*.

```
<labelArc xlink:type="arc" xlink:from="bcb_ativo" xlink:to="bcb_ativo_br" xlink:show="embed"
  xlink:actuate="onRequest" xlink:title="Vá para o label bcb_ativo_br"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-label"/>
<labelArc xlink:type="arc" xlink:from="bcb_ativo_br" xlink:to="bcb_ativo" xlink:show="replace"
  xlink:actuate="onRequest" xlink:title="Vá para o elemento bcb_ativo"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/label-element"/>
```

Os atributos para este elemento são:

- *xlink:type* - sempre *arc*;
- *xlink:from* - a origem do *link*;
- *xlink:to* - o destino do *link*;
- *xlink:show* - pode ser *embed* ou *replace*. Um *label* usa *embed* se o arco é do elemento para o *label*, já que a origem do arco está em um elemento que está em um arquivo diferente daquele onde o *linkbase* está localizado, e *replace* se for o inverso;

- *xlink:arcrole* - informa qual a direção do *link*;
- *xlink:title* - contém informação adicional sobre o significado do arco.

Linkbases Presentation - Os elementos usados para a construção do *linkbase Presentation* foram definidos em *bcb_presentation.xml*, apêndice C. Este elemento *linkbases*, como todos os outros, é definido pela especificação XBRL e contém elementos de *link* estendidos que descrevem as relações de apresentação entre os elementos definidos no documento de taxonomia.

Os *namespaces* pertencentes a este *linkbase* são semelhantes àqueles contidos no *linkbase Label*, possuindo as mesmas funções.

O elemento *presentationlink* é um *link* estendido que contém elementos do tipo *arc* e *locator*. O atributo *role* tem como função agrupar, para facilitar a pesquisa por aplicações, os elementos do *link* por tipo de documento:

```
<presentationlink xlink:type="extended"
  xlink:role="http://www.xbrl.org/linkprops/extended/BalancoPatrimonial">
```

Os elementos *locator* têm a mesma função dos seus homônimos no *labelLink*.

Os elementos *presentationArc* são semelhantes aos *labelArc*, diferindo apenas no valor dos atributos *xlink:show*, que deve ser sempre *replace* para apresentação no mesmo local do recurso inicial, *xlink:arcrole* e pela inclusão do atributo *xlink:order*, que sugere a ordem que o elemento deve ser apresentada, *order=1* significa que este deve ser o primeiro filho a ser

mostrado, *order=2* o segundo e assim sucessivamente. Porém qualquer aplicação tem a liberdade de apresentar os dados na ordem desejada:

```
<presentationArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo" xlink:to="bcb_ativo"
  xlink:show="replace" xlink:actuate="onRequest" xlink:title="Apresentação: Vá para
  bcb_ativo" order="1" xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
```

O elemento acima traz a seguinte informação: "para propósitos de apresentação, o elemento *bcb_atv_circ_real_lprazo* é o primeiro filho de *bcb_ativo*".

Linkbases Calculation e Definition - Os elementos usados para definir os *linkbases Calculation* e *Definition* foram definidos nos documentos *bcb_calculation.xml* e *bcb_definition.xml*, apêndice C. Estes *linkbases* possuem características, nos seus elementos e atributos, semelhantes ao *Presentation*, porém com algumas particularidades.

A função do *linkbases Calculation* é definir as relações de cálculo entre os elementos XBRL definidos na taxonomia. Estas relações são expressas através dos elementos *calculationArc*, onde o atributo *weight* define quanto do valor do elemento filho deve ser adicionado, ou subtraído, aos valores de seus elementos irmãos para formar o valor do elemento pai (por exemplo, *weight=1* significa que todo o valor do elemento filho deve ser adicionado na soma que resultará no valor do elemento pai e *weight=-1* a totalidade do valor deve ser subtraída).

Já o *linkbase Definition* carrega a relação hierárquica entre os elementos da taxonomia, na definição do elemento pai e seus elementos filhos, por meio do atributo *arcrole*.

```
<calculationArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo" xlink:to="bcb_ativo"
  xlink:show="replace" xlink:actuate="onRequest" xlink:title="Cálculo: Vá para bcb_ativo"
  weight="1" xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>

<definitionArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo" xlink:to="bcb_ativo"
  xlink:show="replace" xlink:actuate="onRequest" xlink:title="Definição: Vá para
  bcb_ativo"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
```

Linkbases Reference - os elementos usados para construir o *linkbases Reference* foram definidos no documento *bcb_reference.xml*, apêndice C. Este *linkbase* tem a mesma estrutura dos antecedentes e seus elementos de *link* estendido descrevem o relacionamento entre referências e os elementos definidos na taxonomia.

Da mesma forma que no elemento *labelLink*, o elemento *reference* possui um elemento do tipo *resource*, além dos elementos *locator* e *arc*, que também estão presentes nos *linkbases Relations (Calculation, Definition e Presentation)*.

O elemento *reference* pode conter subelementos, definidos na taxonomia, os quais podem especificar um material de referência, no exemplo em questão foi incluído o elemento *referencia* cujo conteúdo indica quais as leis que obrigam a publicação do balanço

patrimonial. Outros tipos de referências podem ser utilizados, por exemplo para comentários ou notas explicativas.

O elemento *referenceArc* no documento fornece a seguinte informação: "o elemento *bcb_bp* tem um *link* para a referência especificada pelo *label bcb_bp_ref*":

```
<referenceArc xlink:type="arc" xlink:from="bcb_bp" xlink:to="bcb_bp_ref" xlink:show="embed"
  xlink:actuate="onRequest" xlink:title="Vá para a referência bcb_bp_ref"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-reference"/>
<referenceArc xlink:type="arc" xlink:from="bcb_bp_ref" xlink:to="bcb_bp" xlink:show="replace"
  xlink:actuate="onRequest" xlink:title="Vá para o elemento bcb_bp"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/reference-element"/>
```

5.4 Documentos XBRL

Foram criados três documentos XBRL, *bcb_bp_28_02_2001.xml*, *bcb_bp_31_03_2001.xml* e *bcb_bp_30_04_2001.xml*, estando o primeiro disponível no apêndice E, com as informações necessárias para a disponibilização do balanço patrimonial do Banco Central do Brasil em três momentos diferentes. Neste exemplo foi usada uma folha de estilo XSLT para a transformação do documento XBRL em HTML e publicação por meio de um navegador Web.

O primeiro elemento do documento XBRL é o elemento *group* que contém todos os elementos e contextos necessários para a definição do relatório financeiro, este é o elemento raiz nos termos de XML.

Os *namespaces* especificados neste documento são semelhantes aos dos documentos de taxonomia e *linkbases*.

O elemento *bcb:bp* é definido dentro da taxonomia como um tipo "*tuple*". Este elemento serve para especificar que os elementos por ele contidos pertencem a um determinado tipo de relatório, no caso, Balanço Patrimonial.

```
<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="bcb_InstView.xsl"?>
<group
  xmlns="http://www.xbrl.org/2001/instance"
  . . .
  xsi:schemaLocation="www.bcb.gov.br bcb_taxonomia.xsd">
  <bcb:bp numericContext="c1">
<bcb:ativo numericContext="c1">270190618</bcb:ativo>
<bcb:atv_circ_real_lprazo numericContext="c1">261376808</bcb:atv_circ_real_lprazo>
<bcb:atv_ext numericContext="c1">81072891</bcb:atv_ext>
  . . .
```

Os elementos filhos de *bcb:bp* relatam os fatos financeiros, como por exemplo:

```
<bcb:ativo numericContext="c1">270190618</bcb:ativo>
```

As outras informações para este fato, como unidade da moeda, precisão, período, etc., são definidas pela associação do valor do atributo "*numericContext*" com o correspondente elemento *numericContext*, presente neste mesmo documento. Ou seja, com esta associação

pode-se entender que a entidade Banco Central do Brasil em 28 de fevereiro de 2001 possuía um ativo no seu balanço patrimonial no valor de R\$270.190.618,00.

O elemento *numericContext* contém detalhes das informações sobre quantidades numéricas especificadas nos elementos que possuem o atributo *numericContext="c1"*.

```
<numericContext id="c1" precision="10" cwa="true">
  <entity>
    <identifier scheme="http://www.bcb.com.br/">Banco Central do
      Brasil</identifier>
    <segment/>
  </entity>
  <period>
    <instant>28-02-2001</instant>
  </period>
  <unit>
    <measure>iso4217:BRL</measure>
  </unit>
  <scenario name="Balanço Patrimonial">
    <bcbscenarioType>final</bcbscenarioType>
  </scenario>
</numericContext>
```

Da mesma forma que o elemento acima descreve valores de fatos numéricos, o elemento *nonNumericContext* descreve valores de fatos não numéricos, como os dos elementos *bcb:presidente*, *bcb:diretores*, etc. Os subelementos e atributos presentes neste elemento são aqueles mesmos do *numericContext*, com exceção dos que fazem referência a valores numéricos.

5.5 Folha de Estilo

Um navegador Web que compreenda a notação XML, simplesmente irá mostrar a estrutura do documento XML no formato de um arquivo texto. No entanto, pode-se usar uma aplicação para lidar com os arquivos texto e exibir o documento em outra forma desejada [36].

Existem alguns recursos para se apresentar um documento XML em um navegador Web de forma que não apareça a sua estrutura, mas o seu conteúdo em um formato visual aprazível:

- associando a um documento CSS, cujo processo ocorre de forma semelhante ao de associar um arquivo CSS a um documento HTML;
- transformando com XSL, uma linguagem que permite especificar de forma precisa como a saída deve ser;
- transformando em HTML, o que pode ser conseguido usando XSLT. Na verdade, o objetivo inicial da XSLT era transformar documentos XML em XSL. Porém, com a sua definição, XSLT pôde transformar arquivos XML em HTML, ou arquivos de texto comum ou arquivos de texto com estruturas não definidas em XML [36];

- Usando linguagens de programação como Java, C ou C++, que trazem o inconveniente de serem mais difíceis de se trabalhar do que a XSLT [36], que foi projetada especificamente para este fim, gerando, desta forma, códigos de transformação menores.

A transformação do documento XML em HTML, usando-se XSLT, pode ocorrer tanto no servidor Web, como no lado do cliente. No primeiro caso, o documento XML é entregue a um processador XSLT para convertê-lo em um HTML, de acordo com as instruções do documento XSLT. O navegador Web ao receber o documento HTML, não sabe se o documento original é um XML, nem precisa saber sobre XSLT ou como transformar o XML original. Apenas exibe o documento HTML. Quando a transformação ocorre no lado do cliente, o navegador Web recebe os documentos XML e XSLT e aplica o segundo sobre o primeiro para gerar o documento HTML, tendo o navegador Web que ter um processador XSLT [36]. Neste estudo de caso foi aplicada esta última solução, já que para o estudo desenvolvido nessa dissertação é irrelevante o método de transformação usado.

Em razão da sua versatilidade e do seu grande potencial, a tecnologia XSLT foi escolhida para transformar em HTML os documentos exemplos do Balanço Patrimonial, escritos em XBRL. O código XSLT foi desenvolvido baseado no código disponível no grupo público de discussões XBRL [37]. O código original sofreu diversas alterações para adaptá-lo à realidade do exemplo, com a utilização de *linkbases*, que originariamente não foram utilizados, inclusão de novas variáveis e código, para que a formatação do documento produzido em XBRL ficasse de forma que se assemelhasse o máximo possível ao documento HTML, publicado no sítio do Banco Central na Internet.

O fluxo do processo da transformação do documento XBRL em HTML, que pode ser observado pela figura 5.1, ocorre pela aplicação no documento XBRL da folha de estilo *bcb_Instiview.xsl* (apêndice H), que inclui uma outra folha de estilo, *bcb_TxCompress.xsl* (apêndice I).

A folha de estilo *bcb_TxCompress.xsl* quando aplicada a um documento de taxonomia, produz uma árvore resultante com as informações sobre a taxonomia e seus *linkbases*. As informações contidas nos documentos *linkbases* são encontradas por meio dos elementos *linkbasesRef*, contidos na taxonomia, que informam as localizações destes documentos. A folha de estilo *bcb_TxCompress.xsl* deve ser usada em conjunto com *bcb_InstView.xsl*, que é aplicada no documento XBRL. Desta forma é feita uma validação dos elementos contidos no documento XBRL contra a árvore resultante da aplicação da folha de estilo *bcb_TxCompress.xsl* na taxonomia.

A transformação ocorre na folha de estilo `bcb_InstView.xsl` usando a árvore resultante das informações obtidas na taxonomia e *linkbases*, que é passada por meio do elemento `xsl:include`, que inclui `bcb_TxCompress.xsl`. Estes códigos XSLT, quando aplicados ao documento codificado com a segunda versão de XBRL, fornece uma tabela HTML contendo informações dos elementos. Um maior entendimento dos códigos pode ser conseguido com a leitura dos comentários neles contidos, presentes nos apêndices I e H.

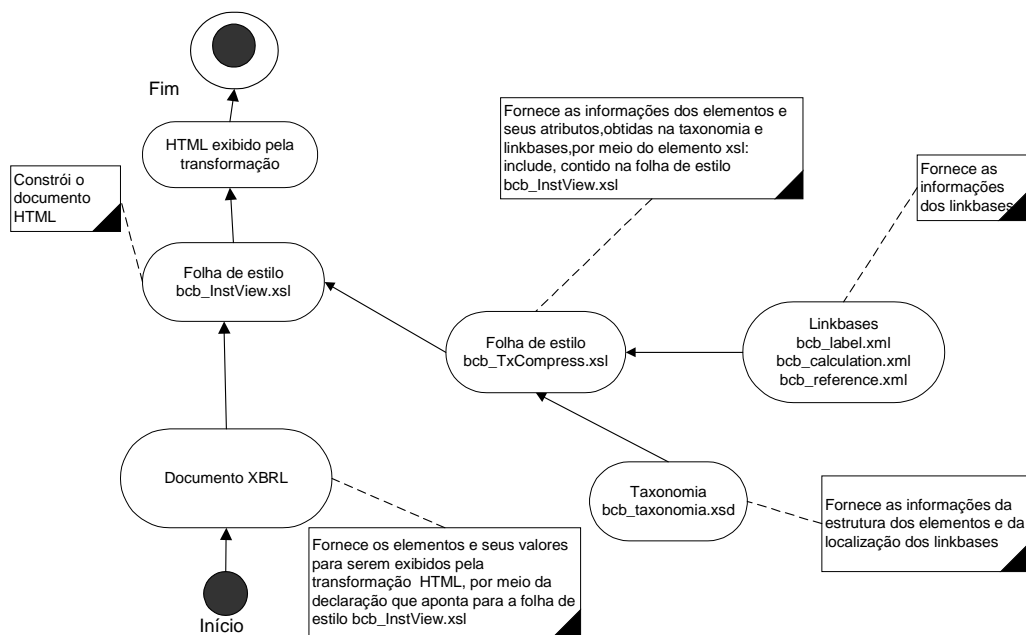


Figura 5.1 - Fluxo do processo da transformação do documento XBRL em HTML

O resultado da transformação do documento XBRL, `bcb_bp_28_02_2001.xml`, feito pelos códigos XSLT, `bcb_InstView.xsl` e `bcb_TxCompress.xsl`, em um documento HTML, produz pelo navegador Web a exibição mostrada na figura 5.2.

cinco *linkbases* e duas folhas de estilo), estes são feitos apenas uma vez e aplicados a qualquer documento XBRL que esteja em conformidade com a taxonomia.

5.6 Aplicações para extração e utilização da informação em XBRL

Para demonstrar como pode ser feita a extração da informação de um documento XBRL foram criadas duas aplicações desenvolvidas com as tecnologias DOM, Java e JSP.

A primeira aplicação faz a extração das informações contidas no documento XBRL e armazena em um banco de dados relacional, SQL Server, por meio da classe *LendoXML.java*, que extrai as informações do documento e invoca a classe *AtualizaTabela.java*, para armazenar os valores no banco de dados. Os códigos dessas aplicações encontram-se no apêndice J.

A outra aplicação foi criada para, com a classe *AcessaBancoDados.java*, obter as informações previamente armazenadas no banco de dados e, com auxílio do código JSP, calcular indicadores de análises financeira, para, então, se apresentar graficamente em uma página da Web.

O processo de utilização automatizada das informações contidas nos documentos XBRL pode ser observado pela figura 5.4.

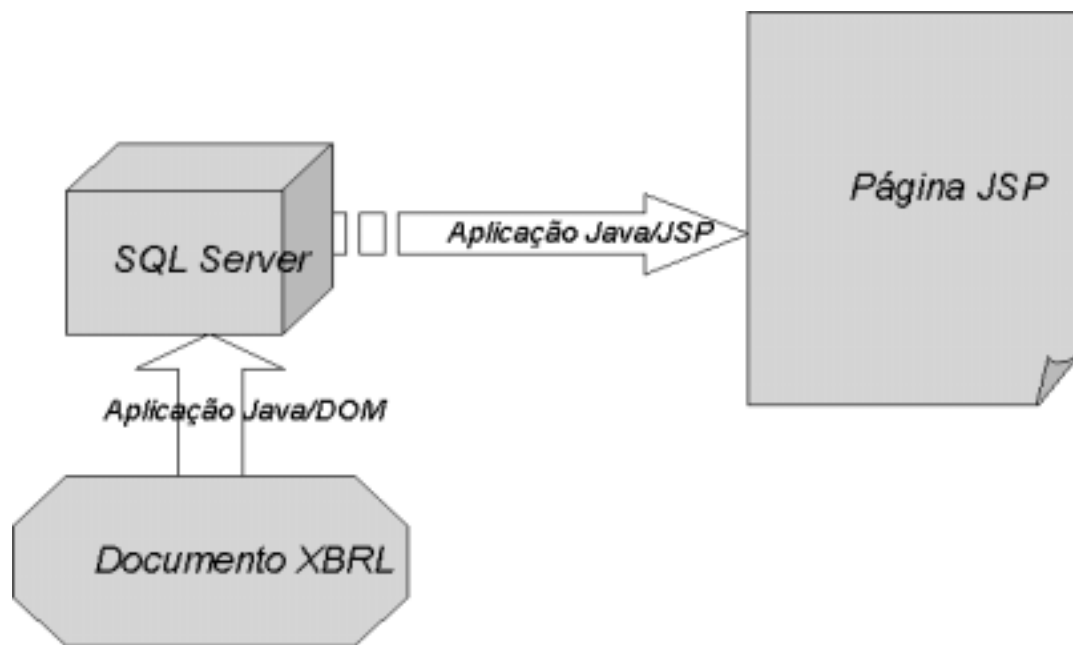


Figura 5.4 – Processamento automatizado dos documentos XBRL.

Desta forma, pôde ser constatada que com simples aplicações é possível extrair a informação codificada em XBRL, armazená-la em um banco de dados e processá-las para em

seguida representar o resultado graficamente, algo que não é trivial, para não se dizer quase impossível, na forma em que é apresentada no atual sítio do Banco Central na Internet.

A figura 5.5 seguinte mostra o resultado obtido com estas aplicações.

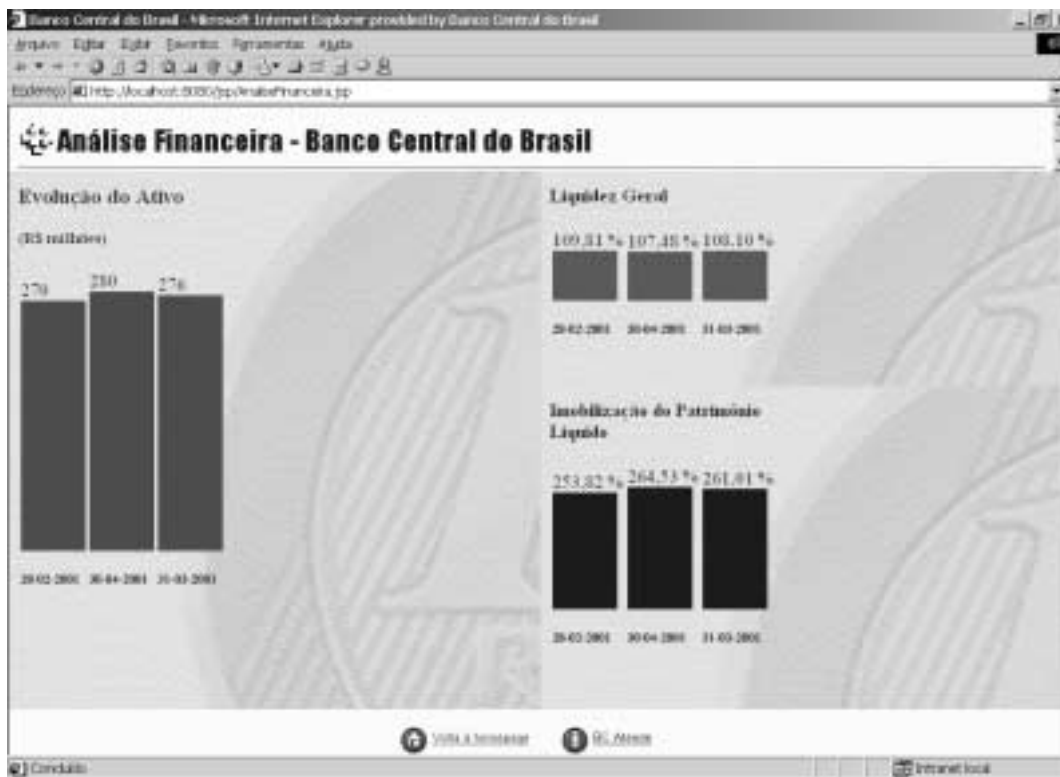


Figura 5.5 - Exibição do resultado da aplicação usando-se XBRL/DOM/JAVA/JSP

É importante ressaltar que pode-se usar outras técnicas para a extração da informação contida nos documentos XBRL, que podem ser definidas em função de fatores como volume de dados contidos nos documentos XBRL, limitação de memória na utilização de DOM, etc. Porém, é importante ressaltar que a análise dos fatores deve ser procedida em cada situação específica e que com o exemplo apresentado foi possível verificar como a utilização da linguagem de marcação XBRL possibilita a preservação semântica dos dados e facilita o processamento da informação financeira na representação de relatórios financeiros, sejam para aqueles que são disponibilizados na Internet, na Intranet, ou para os que são fornecidos para as instituições governamentais ou entre companhias parceiras.

6. Implantando XBRL no Banco Central

Antes de se iniciar a discussão de como implantar o uso de XBRL no Banco Central, é conveniente ressaltar as dificuldades encontradas no desenvolvimento deste trabalho, para que se possa avaliar a metodologia que deverá ser usada pelo Banco Central.

O processo de aprendizado e conhecimento da especificação XBRL requer um conhecimento prévio de padrões XML, como XML Schema, XML Link, XML Namespace, XML Path e XML Pointer. Sem um domínio destas tecnologias não é possível um completo entendimento de XBRL. Além do conhecimento tecnológico, é fundamental que se tenha um razoável entendimento da área de negócio em que XBRL atua, ou seja, o campo financeiro/contábil. Esse conhecimento é necessário para que melhor se entenda a aplicabilidade da especificação.

Para nomear os elementos que irão representar os fatos financeiros é preciso definir um padrão. Na elaboração da taxonomia, para o estudo de caso, ficou-se em dúvida se deveria ser usado os nomes dos conceitos financeiros, como “AtivoCirculanteRealizavelLongoPrazo”, ou abreviações. Como os nomes dos conceitos financeiros são geralmente longos, optou-se por nomear os elementos com abreviações que sintetizassem o nome do conceito, por exemplo, “atv” para representar “ativo”, “pssv” para “passivo” e nos nomes compostos, as abreviações foram separadas por símbolo de sublinhado “_”. Ficando, então, “atv_circ_real_lprazo” para *ativo circulante realizável a longo prazo*.

É também importante estar atento à hierarquia dos fatos financeiros para a construção dos *linkbases*, para que se possa definir corretamente quem é o elemento pai, quem é o elemento filho, a ordem de apresentação no documento e a relação matemática dos elementos filhos para compor o valor do elemento pai.

No subitem seguinte será apresentada uma sugestão de metodologia a ser aplicada no processo de implantação do uso de XBRL para representação de documentos financeiros no Banco Central, a qual poderá servir de referência para outras organizações.

6.1 Metodologia

A metodologia para implementação de uma nova tecnologia em um ambiente organizacional apresenta diversas técnicas que podem ser aplicadas. A escolha de uma delas varia de acordo com as ferramentas e o ambiente a ser utilizado.

O guia metodológico proposto abaixo explica, de forma concisa, como planejar, construir e implementar um ambiente computacional para o intercâmbio de informações

financeiras, usando XBRL como formato de representação dos dados. Foram sugeridas seis fases para o processo de implantação de XBRL:

1 - Planejamento

Proporciona uma estratégia para a implementação de XBRL. O planejamento deve contemplar aspectos, tais como:

- treinamento sobre a especificação XBRL;
- definição da equipe para cada fase de desenvolvimento do projeto;
- levantamento de prioridades;
- análise das ferramentas existentes no mercado para a solução de implementação;
- definição detalhada do roteiro da metodologia;
- refinamento e homologação dos objetivos e escopo do projeto;
- definição de como será feito o acompanhamento do projeto;
- criação de um comitê para apoiar, validar e acompanhar o projeto durante o seu desenvolvimento;
- definição da infra-estrutura de ambientes de desenvolvimento, homologação e produção.

2 - Treinamento

Refere-se à capacitação dos desenvolvedores na especificação XBRL e nas ferramentas disponíveis.

Qualquer implementação muda a rotina de uma organização. Além da novidade da nova tecnologia, tem-se que considerar o impacto nos processos e funções das áreas envolvidas.

O que determina o sucesso da implementação é indubitavelmente, a capacitação das pessoas em assimilar a nova tecnologia e aprender a operá-la corretamente. Somente assim se obterá todos os potenciais benefícios da tecnologia.

A formação da figura do multiplicador de conhecimento poderá ser a peça chave no processo de capacitação, com a responsabilidade de disseminar às demais pessoas da organização a necessidade de se usar XBRL, como implantá-la e quais os benefícios que ela poderá trazer. Assim como capacitar outros agentes multiplicadores do conhecimento técnico.

3 - Confecção da fase lógica

. definição de uma política de extração de dados para publicação na Internet – na realidade, essa política de extração de dados pode ser a mesma já utilizada atualmente para a publicação das páginas em HTML no sítio do Banco Central;

. definição de uma política de segurança – também pode-se usar as mesmas técnicas e políticas de segurança que hoje são aplicadas ao Sistema de Pagamentos Brasileiro, em razão

das similaridades deste sistema com XBRL, que usa como linguagem de marcação XML e a Internet como meio de transmissão das informações;

- . definição de uma política de recebimento das informações do sistema financeiro nacional – aqui, também, já existe uma política de recebimento, pois as informações que poderão ser representadas em XBRL, hoje já são enviadas em um outro formato. No entanto, é necessário que se defina como será a fase de transição para o formato XBRL. Deve-se levar em consideração a possibilidade de se utilizar a transmissão das informações nos dois formatos, em XBRL e no existente, até que os efeitos inerentes a uma transição estejam estabilizados;

- . definição de uma política de validação dos documentos XBRL – a sistemática de validação de documentos no Sistema de Pagamentos Brasileiro também pode ser aplicada neste caso, para validação dos documentos XBRL na taxonomia e linkbases.

4 - Construção dos documentos básicos XBRL

- . definição do padrão de nomenclatura dos elementos;
- . definição dos elementos;
- . construção da taxonomia e linkbases.

5 - Teste

Este passo tem como finalidade testar e aperfeiçoar o processo de elaboração, recebimento e validação de um documento XBRL.

6 - Implantação

Deve ser definida uma estratégia que vise minimizar os impactos naturais causados pelo processo de implantação de uma nova tecnologia. O acompanhamento da implementação pelos envolvidos no projeto é fundamental para o sucesso do projeto. Eventuais ajustes poderão ocorrer nessa fase.

Uma implementação bem sucedida depende fundamentalmente do comprometimento das pessoas envolvidas no projeto. São elas que levarão uma instituição a mudar de um estágio para o outro. Portanto, é fundamental a liderança em um processo de implementação de uma nova tecnologia, tendo o líder o papel de comprometer os colaboradores com os objetivos do projeto.

6.2 Sugestões

Para se implantar XBRL no Banco Central é necessário que algumas medidas sejam adotadas para cada finalidade no seu ambiente de intercâmbio de informações financeiras.

Inicialmente deve-se decidir quais as informações, tanto as que serão publicadas na Web, como as que serão recebidas das instituições financeiras, deverão vir a ser representadas em XBRL. Nem todas as informações que o Banco Central publica em seu sítio na Internet, nem as que lhe são enviadas pelos agentes do sistema financeiro, estão dentro do escopo de XBRL. Por exemplo, normativos emitidos pelo Banco Central estariam mais adequadamente representados em um outro formato que represente o tipo de normativo (resolução, carta-circular, circular, etc.), data de publicação, numeração, etc, podendo ser inclusive derivado de XML. Outro exemplo seria as informações do departamento de pessoal que não impactem qualquer resultado nos relatórios financeiros, estas não devem estar representadas em XBRL. Portanto, é necessário se fazer a seleção das informações que serão representadas em XBRL e a definição das prioridades de implementação.

Para publicação das informações no seu sítio na Internet, é preciso que uma taxonomia seja criada com elementos que representem as informações que serão fornecidas em XBRL. A definição destes elementos poderá inicialmente ser feita a partir do plano contábil do próprio Banco Central, para publicação dos seus demonstrativos contábeis. Em seguida devem-se definir os elementos para aquelas outras informações que não constam neste plano contábil, como taxas de câmbio, séries econômicas, etc e que produzem outros tipos de relatórios financeiros. É conveniente que equipes sejam formadas com pessoal da área tecnológica e especialistas das áreas que geram e fornecem a informação, para que a criação dos elementos aconteça de forma consistente.

Com a disponibilização na Web de documentos XBRL, deve-se, também, estar disponível a taxonomia e linkbases para que os usuários possam dela dispor para identificar as relações financeiras contidas nos documentos e realizarem o processamento automático destas informações.

Para o intercâmbio de informações financeiras com o sistema financeiro nacional o Banco Central deve aproveitar a experiência da implantação do Sistema de Pagamentos Brasileiro, pela razão de nele ter sido usado tecnologia XML, no que diz respeito a todos os aspectos da implementação (definição de elementos, requisitos de segurança, etc.) e no que tange o relacionamento com as instituições financeiras, Febraban (Federação Brasileira de Bancos), etc. Da mesma forma que para definição da taxonomia para a disponibilização das informações na Web, equipes devem ser montadas para criação da taxonomia que representará o COSIF - Plano Contábil das Instituições do Sistema Financeiro Nacional. O COSIF foi criado com a edição da Circular 1.273, em 29 de dezembro de 1987, com o objetivo de unificar os diversos planos contábeis existentes à época e uniformizar os

procedimentos de registro e elaboração de demonstrações financeiras. Aplicando-se XBRL ao COSIF e solicitando o fornecimento das informações financeiras neste formato, o acompanhamento, análise, avaliação do desempenho e controle das instituições integrantes do sistema financeiro nacional, certamente, poderá ser facilitado. Em seguida à implementação de XBRL para representar os relatórios derivados do COSIF, poder-se-á analisar quais outros relatórios financeiros fornecidos pelas instituições financeiras poderão ser representados por XBRL.

Um órgão regulador com poder mandatário, como o Banco Central, pode impor às entidades que operam sob sua tutela o fornecimento das informações desejadas no formato do seu sistema de arquivamento eletrônico, porém, é conveniente que todos os envolvidos participem do projeto XBRL, para que a implantação desta tecnologia ocorra sem percalços e com impactos negativos minimizados.

XBRL foi criada por um grande consórcio internacional composto por mais de 170 organizações da área financeira para desenvolver um método consistente e comum de intercâmbio de informações e relatórios financeiros. Entre estas organizações estão bancos, agentes reguladores, auditores, analista financeiros, etc., dos quais como órgãos governamentais pode-se citar o Banco Central Alemão, Deutsche Bundesbank, e o Departamento do Tesouro Americano [27, 38]. Então, é sugerido que o Banco Central do Brasil venha a se incorporar neste esforço mundial de especificação e implantação de XBRL, pois sua participação poderá trazer grandes benefícios, como:

- acesso a todas as taxonomias, especificações e documentação em fase de desenvolvimento;
- treinamento e uso gratuito de ferramentas XBRL em desenvolvimento;
- intercâmbio de informações com especialista em XBRL, o que poderá facilitar sua implementação;
- estar informado de planos de implementação e benefícios trazidos por XBRL ao redor do mundo;
- participar de grupos de discussão, onde experiências e troca de idéias são compartilhados;
- envolver-se em futuros desenvolvimentos.
- descobrir novas oportunidades de processar informações financeiras e adicionar valor aos relatórios financeiros.

Para a adoção de XBRL é prudente o contato com organismos que já implementaram XBRL, para aprendizado e troca de experiências de como ocorreu a implantação desta tecnologia. Entre órgãos de governo, o “Australian Prudential Regulatory Agency” (APRA) foi o primeiro a anunciar, no final de outubro de 2001, que XBRL estaria sendo usada para coleta de informações de aproximadamente 11.000 fundos, seguradoras e bancos, todos obrigados a fornecer-lhe informações financeiras de forma regular [39]. Além disso, a APRA agrega dados para o “Australian Bureau of Statistics” e o “Reserve Bank of Australia”, reduzindo esforços de processamento e evitando duplicação de informações para aquelas organizações que são obrigadas a se relacionar com todos os três órgãos governamentais. Este é um contato a ser feito, pois devido a similaridade de funções entre esses órgãos e o Banco Central do Brasil muitas informações úteis sobre o processo de adoção de XBRL poderão ser agregadas, como, dificuldades encontradas, método de criação da taxonomia e documentos XBRL, requisitos de segurança no intercâmbio da informação, etc.

Outras diversas organizações e corporações estão aderindo ao uso de XBRL, o que ressalta a importância de participação no consórcio formado para sua especificação, por exemplo, o Departamento do Tesouro Norte Americano sugeriu a adoção de XBRL por todas as agências do seu governo federal [38].

Em qualquer lugar onde organizações trocam informações de desempenho financeiro e empresarial, particularmente na Internet, XBRL apresenta uma oportunidade para melhorar o processo, se adotado como padrão de representação para os dados. Fontes de informação sobre XBRL se encontram no sítio oficial de XBRL [27], que inclui demonstrações, documentos e apresentações, páginas de ferramentas de vendedores e provedores de serviço, participação em eventos XBRL ou simplesmente o contato com alguém envolvido em algum projeto XBRL.

7. Conclusão

Nessa dissertação foi apresentada uma abordagem para a representação de relatórios financeiros de forma que o intercâmbio dos relatórios entre aplicações possa ocorrer de maneira simples e direta, eliminando-se a necessidade de transformações sucessivas de formatos e fornecendo a possibilidade de estruturação da informação financeira na Web.

A partir do estudo de caso efetuado pode-se concluir que a aplicação de XBRL como padrão para representação de dados de relatórios financeiros pode conduzir a um intercâmbio eficiente destes relatórios, facilitando as análises e tomadas de decisões necessárias para a atividade organizacional, ou seja, essa solução pode conduzir a melhorias nos processos de análise da informação financeira, diminuir custos, facilitar a interação das informações entre os departamentos das organizações, proporcionar maior interoperabilidade entre sistemas financeiros e possibilitar a estruturação das informações financeiras na Web, de forma que possam ser extraídas e usadas mais facilmente. Enfim, é uma abordagem que pode contribuir significativamente com a gestão da informação financeira.

7.1 Contribuições

O estudo realizado demonstrou que a aplicação de XBRL como padrão para a representação de relatórios financeiros poderá proporcionar as seguintes facilidades:

- uma boa interoperabilidade entre sistemas e plataformas computacionais, já que sendo adotada como padrão para representação dos dados, a necessidade de conversões sucessiva de formatos para os diversos ambientes é eliminada;
- uma boa forma de se distribuir a informação, pois estando representada em XBRL, a informação poderá facilmente e de maneira direta, ser distribuída para as diversas bases de dados ou formatos de relatórios;
- a estruturação da informação financeira na Web, tornando mais democrático o acesso às informações contidas nos relatórios, já que um grande público terá a informação disponível em um formato que facilita a extração e o processamento por aplicações, que poderão ser construídas pelos próprios usuários;

Com o estudo de caso pôde-se inferir que a utilização de XBRL como padrão no ambiente do Banco Central do Brasil poderá trazer melhorias significativas no seu processo de uso da informação financeira:

- A distribuição dos relatórios recebidos das instituições financeiras poderá ocorrer automaticamente para os diversos formatos de dados, diminuindo redundâncias e transformações sequenciais de formatos;
- A integração intra-departamental poderá ser melhorada, pois com um padrão para representação dos dados financeiros, certamente a informação que estiver, ou for produzida, em um departamento poderá mais facilmente torna-se disponível para as análises de outros departamentos;
- A estruturação das informações financeiras no sítio do Banco Central, facilitará para o público a extração dos dados lá contidos.

A adoção de XBRL no ambiente computacional de informações do Banco Central, poderá desencadear um processo de mudanças nas instituições financeiras para que elas também adotem nos seus ambientes internos o mesmo padrão. Tal fato, além de beneficiar as instituições com as vantagens já citadas, poderá favorecer todo o contexto financeiro do país, pois se todas as instituições adotarem XBRL como padrão de comunicação, os mecanismos de análise e checagem das informações serão facilitados, proporcionando agilidade, redução de custos e transparência ao sistema financeiro nacional.

8. Glossário

8.1 Glossário de Termos Financeiros

Análise de risco - toda e qualquer análise que se faça sobre o comportamento de uma empresa, título, ativo, moeda, segmento empresarial, país, no presente e no futuro, com base na experiência passada e com base nas expectativas futuras, com o objetivo de saber quais os riscos que estão implícitos, para a partir daí traçarem-se cenários. Tudo isso com vistas a saber se os compromissos poderão ser honrados futuramente. Os analistas de mercado fazem isso para saber se as empresas sobreviverão caso o preço das mercadorias caia até um certo patamar, ou se os acionistas vão ganhar ou perder a partir de uma revisão nos lucros de uma companhia aberta, ou ainda se os países conseguirão honrar suas dívidas a partir de uma administração de um novo governo.

Ativo Circulante – Ativos financeiros de alta liquidez, ou seja, liquidez de curto e médio prazo.

Câmbio - Negociação de moedas de países diversos.

Compensação - termo relacionado às tradicionais câmaras de compensação, que se constituem em um local para a liquidação financeira das operações de cheques. No Brasil o Banco do Brasil é o banco que tem a delegação do Banco Central para compensar os cheques transacionados na economia. A compensação de cheques é uma função típica de bancos centrais. Por exemplo, o Bradesco recebe em seus caixas milhares de cheques do Banco do Brasil, do Itaú, do Sudameris etc. Por sua vez os demais bancos recebem cheques do Bradesco. Então, diariamente todos os bancos se reúnem, sob a coordenação de um deles (Banco do Brasil), para compensarem, com os créditos que possuem, os débitos que sofreram no dia. As diferenças apuradas, são compensadas (a crédito ou a débito) na conta de Reserva Bancária da instituição no Banco Central.

Commodities - São mercadorias com apreciação em bolsas de mercadorias, com grande aceitação comercial, atingindo inclusive cotação internacional, portanto são exportáveis. São padronizáveis a uma medida comum por possuírem especificações comuns. São exemplos de commodities: ouro, minério de ferro, soja, milho, boi/gado, chips de computador, aços, petróleo etc.

Dealer - dealer é um agente autorizado a comercializar certo produto. No mercado de câmbio, o Banco Central possui seus dealers, previamente autorizados. O Brasil certamente possui mais de 200 bancos, mas somente cerca de trinta são dealers do Banco Central no mercado de câmbio. Somente esses dealers têm acesso aos leilões cambiais do Banco Central.

A partir do momento que adquirem os dólares no Banco Central, eles poderão vender a todo o mercado.

Derivativo - são os ativos que derivam de outros ativos. Por exemplo, uma operação de CDB em reais, um ativo em reais, pode derivar para um ativo em dólar, isso é um derivativo. Toda e qualquer operação de swap é um derivativo.

Mercados de capital - São os mercados onde são lançados e transacionados títulos e valores mobiliários. Exemplo: mercado acionário, mercado futuro, de opções, as bolsas de mercadorias, bolsas de valores, etc.

Mercados futuros - são os mercados nos quais os agentes fazem suas apostas futuras e cuja liquidação das posições sempre se dará em data futura. Os mercados futuros possuem implícita a possibilidade de grandes perdas ou ganhos, pois as apostas que são feitas no presente poderão ser ganhadoras ou perdedoras no momento de sua liquidação. A operação de swap é uma operação de mercado futuro.

Mercados de opções - é o mercado que negocia direito de compra ou venda. Por exemplo, quem compra uma opção tem um direito. Se a opção for de compra, o direito é de comprar. Se a opção for de venda, o direito é de vender. O comprador de uma opção de compra (venda), ou titular dessa opção, tem o direito de comprar (vender) uma certa quantidade de ações, a um preço pré-fixado até uma determinada data. O vendedor (o lançador da opção) de uma opção de compra fica com a obrigação de vendê-la. O titular tem sempre os direitos e o lançador as obrigações. O mercado de opções negocia esses direitos de compra ou de venda, mas não as ações diretamente.

Mercado de varejo - é o mercado de pequenas operações financeiras, geralmente dominado pelas pessoas físicas e pequenas empresas. Exemplificando, o Bradesco é um banco de varejo, possui várias agências espalhadas pelo país fazendo pequenos negócios. Já o Opportunity é um banco de atacado, realizando poucos e grandes negócios.

Securities - todo e qualquer título que seja dado em garantia de um empréstimo ou financiamento. Quando se diz que uma dívida foi “securitizada”, significa que ela possui títulos que respaldam sua liquidação.

Securities conversíveis - significa que, se no vencimento, os títulos dados em garantia não tiverem liquidez, poderão ser convertidos noutro ativo que possua liquidez, isso é previamente acertado. No ato da contratação o credor sabe que está obtendo *securities* conversíveis. Exemplo: uma empresa lança no mercado debêntures de R\$ 10 milhões, conversíveis em ações. Suponha que o prazo de resgate das debêntures seja 2010. Se em 2010

a empresa não tiver os R\$ 10 milhões para resgatar a dívida, as debêntures serão convertidas em ações da própria empresa.

Swaps - swap significa troca, ou seja, determinado cliente aplica R\$1.000,00 em um CDB do Banco Itaú, com vencimento em 60 dias, com valor de resgate de R\$1.030,00. Alguns dias após a operação, este cliente decide que vai viajar ao exterior em dezembro, e quando retornar terá uma fatura do cartão de crédito em dólares, vencendo um dia após o vencimento do CDB. Como terá um compromisso em dólares e possui um título em reais e para evitar surpresa, solicita ao Itaú uma operação de swap, assinando um contrato, que deverá ser registrado na BM&F (Bolsa de Mercadorias e Futuros), onde o cliente informa que tem um CDB em reais, com rendimento de 1,5% ao mês, e que deseja trocá-lo por um rendimento de variação cambial mais 0,2% ao mês, remuneração essa que será garantida pelo Itaú. Portanto, ele poderá viajar sossegado, pois com o swap feito, o dólar pode encarecer ou baratear, mas o que importa é que terá o exato montante em reais, suficiente e equivalente para liquidar a dívida em dólares, no vencimento. Essas operações de swap não são de varejo, e sim de atacado. Os agentes econômicos, respaldados por uma bolsa de mercadorias e de futuros, poderão fazer operações de swap trocando os mais diversos índices, (IGP-M x Dólar; pré-fixado x pós-fixado, TR x variação do preço do café), de acordo com suas conveniências, e desde que haja outra ponta querendo o inverso. Nessas operações está sempre implícito um sentimento de aposta. Ambas as pontas sempre acreditam que poderão ter perdas com os ativos que possuem em mãos e que estariam mais satisfeitas se fizessem um swap. O importante é que as condições da troca tragam mais tranquilidade às partes. Mas fica claro que sempre haverá uma ponta ganhadora e outra perdedora.

8.2 Glossário de Termos de Informática

API – Application Programming Interface, conjunto de instruções que permite um aplicativo ter acesso a funções de outro aplicativo.

Aplicativo – programa que é executado em uma máquina cliente.

Atributo – itens adicionais de informação que são acrescentados aos elementos.

Browser – programa usado para ter acesso aos recursos da Internet, também conhecido como navegador.

Cliente/Servidor – Filosofia de funcionamento de aplicações, na qual a informação existe dispersa (pelos servidores), em vez de estar concentrada centralmente, onde é passível de ser consultada pelos “clientes”, lógica e geograficamente dispersos

CSS – Cascade StyleSheet, conjunto de regras de estado que regem como os elementos e marcadores HTML e XML são apresentados por um navegador Web.

DOM – Document Object Model, uma abordagem W3C para processar um documento XML, na qual o documento inteiro é armazenado na memória como uma árvore hierárquica, tornando mais fácil o acesso e manipulação dos elementos contidos no documento.

DTD – Document Type Definition, uma especificação XML para um documento que organiza elementos estruturais e define marcadores, de modo que possam ser usados para criar outros documentos.

e-book – livro eletrônico.

e-commerce – comércio eletrônico.

e-reporting – relatório eletrônico.

EDI – Electronic Data Interchange. Formato usado para a troca de dados que apenas pode ser usado entre dois sistemas compatíveis. Os formatos EDI são mais difíceis de escrever que o XML, não podendo ser transmitidos através do HTTP como o XML.

Elemento – consiste da marcação e do seu texto. Um componente do documento.

Folha de estilo – um documento que usa marcação para fornecer informação sobre a estrutura e conteúdo de um outro documento. Também fornece informação sobre o estilo e apresentação do documento.

FTP - File Transfer Protocol. Protocolo de transferência de arquivos.

Hipertexto – Designa a ação de poder escolher um link numa determinada página de informação e assim obter uma outra página de informação

HTTP – Hypertext Transfer Protocol, protocolo para comunicação entre um servidor Web e um navegador Web.

Internet – O maior conjunto de redes de dados do mundo, tendo em comum apenas a utilização do protocolo TCP/IP, também designada por rede de redes.

Java – linguagem de programação orientada a objetos, desenvolvida pela SUN MICROSYSTEMS, usada para desenvolver aplicações para a Web.

Job – programa que é executado em computador de grande porte.

JSP – Java Server Page, API Java que permite combinar HTML e código Java em um único documento, para criar páginas Web dinâmicas.

Linguagem de marcação (markup) - uma maneira formal de marcar um documento ou coleção de documentos digitais usando elementos (tags) que indicam a estrutura do documento ou dados sobre o seu conteúdo. Permite também introduzir informação acerca do modo de processamento e visualização dos objetos e texto.

Link – uma conexão ou inter-relacionamento entre dois ou mais objetos de dados ou porções de objetos de dados.

Linkbase – documento que agrupa definições de links em documentos XML, construindo assim bases de links.

Link estendido – um link que é armazenado em um arquivo externo e permite expressar relacionamentos entre mais do que dois recursos.

Mainframe – computador de grande porte.

Markup – o mesmo que tag.

Namespace – Uma especificação XML que descreve um padrão para construir documentos que incluem componentes de domínios diferentes.

Servidor Web – ver Cliente/Servidor

SSL – Secure Sockets Layer, protocolo de segurança - um sistema padrão da Internet que permite codificar, ou seja, transformar em códigos todas as informações.

Tag – códigos de endereçamento que permitem que um programa de computação possa encontrar e processar informação neles contidos, sua representação é feita por um nome identificador entre sinais de maior e menor (<, >).

TCP/IP – Transmission Control Protocol/Internet Protocol, conjunto de protocolos de comunicações desenvolvido para suportar ambientes de rede mistos, como a Internet.

Unicode – conjunto de caracteres que usa padrões de 16 bits para representar mais de 65.000 caracteres.

UML – Unified Modeling Language, notação padrão para projetos orientados a objetos.

URI – Universal Resource Identifier, conjunto de nomes ou endereços eletrônicos que formam uma cadeia de caracteres que referenciam uma determinada localização de um recurso na Internet.

URL – É uma espécie de identificador universal que define uma dada localização de um recurso na Internet. Um URL é um endereço Internet que informa como e onde se localiza um determinado arquivo na WWW.

XHTML – eXtensible HTML, Especificação do HTML 4, reescrita com as regras do XML.

XML Link – linguagem de link de XML que permite se criar links em documentos XML.

XML Pointer – linguagem de ponteiro XML que permite designar recursos que especificam localizações em documentos ou recursos.

XML SCHEMA – semelhante a DTD, com mais recursos.

XSL – eXtensible Stylesheet Language, uma linguagem de folha de estilo que é personalizada para XML.

XSLT – eXtensible Stylesheet Language Transformation, linguagem para transformação de documentos XML em documentos XSL, HTML, documento texto ou outro XML.

Web – Termo pelo qual é conhecida a WWW (World Wide Web)

WWW – World Wide Web, rede virtual de informação, composta por um conjunto de máquinas denominadas servidores WWW que usam HTTP e interpretam informação armazenada em formato HTML. É uma grande teia de informação multimídia em hipertexto.

9. Referências Bibliográficas

1. Hoffman, C. & Strand, C. **XBRL ESSENTIALS**. New York: American Institute of Certified Public Accountants, 2001.
2. Bosak, Jon & Bray, Tim. **XML and the second generation web** May 1999. Disponível em <<http://www.sciam.com/article.cfm?articleID=0008C786-91DB-1CD6-B4A8809EC588EEDF&pageNumber=1&catID=2>>. Acesso em outubro/2000.
3. Webber, David; Peat, Bruce; Mikula, Norbert & Marchal, Benoit. **Guidelines for using XML for electronic data interchange**. Editor: Martim Bryan, The SGML Centre. 1998. Disponível em: <<http://www.xmlmedi-group/xmledigroup/guide.htm>>. Acesso em setembro/2000.
4. Hoffman, Charles; Kurt, Cristhopher & Koreto, Richards. **The XML Files**. Journal of Accountants Article. Disponível em: <<http://www.aicpa.org/pubs/jofa/may1999/hoffman.htm>>. Acesso em outubro/2001.
5. Abiteboul, Serge; Buneman, Peter & Suciu, Dan. **Gerenciado Dados Na Web**. Rio de Janeiro: Editora Campus, 2000.
6. Andrew, Richard; Martin Didier; Birbeck, Marck & others. **Professional XML**. WroxPress. 2000. cap.12.
7. Webber, David. **Introducing XML/EDI Frameworks**. Electronics Market Journal. 1998. Disponível em <<http://www.xmlmedi.org>>. Acesso em setembro/2000.
8. Berners-Lee, Tim; Hendler, James & Lassila, Ora. **The Semantic Web**. Scientific American.com, 17/05/2001. Disponível em: <<http://www.sciam.com/article.cfm?articleID=00048144-10D2-1C70-84A9809EC588EF21>>. Acesso em: outubro/2002.
9. Falbo, Ricardo Almeida. **Integração de conhecimento em um ambiente de desenvolvimento de software**. 1998. 205 f. Tese (Doutorado em Ciências em Engenharia de Sistemas e Computação) - Coordenação dos Programas de Pós-Graduação de Engenharia da Universidade Federal do Rio de Janeiro, Rio de Janeiro.
10. Sítio da OFX - Open Financial eXchange. Disponível em: <www.ofx.net> Acesso em outubro/2001.
11. **FAQ's – OFX**. Disponível em: <www.ofx.net/ofx/ab_faq.asp> Acesso em outubro/2001.
12. **Specification – OFX**. Disponível em: <www.ofx.net/ofx/spec> Acesso em outubro/2001.
13. Sítio da FIXML. Disponível em: <www.fixprotocol.org>. Acesso em novembro/2002

14. **<XML> on Wall Street – How XML is being applied in todays Financial Markets - FIXML.** Disponível em: <www.lighthouse-partners.com/xml/proj_fixml.htm>. Acesso em novembro/2002
15. **<XML> on Wall Street – How XML is being applied in todays Financial Markets - SWIFTXML.** Disponível em: <www.lighthouse-partners.com/xml/proj_swiftml.htm>. Acesso em novembro/2002
16. **SWIFTML for Business Message.** November 05, 2002. Disponível em <<http://xml.coverpages.org/swiftml.html>>. Acesso em dezembro/2002.
17. Gartmore and Drkw. **Go Live on SwiftnetFix with Fix Allocations.** Disponível em <http://www.swift.com/index.cfm?item_id=42281>. Acesso em dezembro/2002.
18. Sítio da STPML. Disponível em: <www.stpml.org>. Acesso em novembro/2002
19. **<XML> on Wall Street – How XML is being applied in todays Financial Markets - STPML.** Disponível em: <www.lighthouse-partners.com/xml/proj_stpml.htm>. Acesso em novembro/2002
20. Sítio da FpML. Disponível em: <www.fpml.org>. Acesso em novembro/2002
21. **<XML> on Wall Street – How XML is being applied in todays Financial Markets - FpML.** Disponível em: <www.lighthouse-partners.com/xml/proj_fpml.htm>. Acesso em novembro/2002
22. **FinXML Solutions.** Disponível em: <www.finxml.org/solutions/solutions.asp>. Acesso em novembro/2002
23. **FinXML - 'The Digital Language for Capital Markets'.** Disponível em: <xml.coverpages.org/finXML.html>. Acesso em novembro/2002
24. **<XML> on Wall Street – How XML is being applied in todays Financial Markets - FinXML.** Disponível em: <www.lighthouse-partners.com/xml/proj_finxml.htm>. Acesso em novembro/2002.
25. Sítio da TAXML. Disponível em: <www.taxml.org>. Acesso em novembro/2002
26. Cover, Robin. **XML Markup Languages for Tax Information. The XML cover pages hosted by OASIS.** February 18, 2002. Disponível em <<http://xml.coverpages.org/xmlandtaxes.html>>. Acesso em novembro/2002.
27. **XBRL International, Inc.** Sítio da XBRL. Disponível em: www.xbrl.org. Acesso em janeiro/2001.
28. **Extensible Business Reporting Language (XBRL) 2.0 Specification 2001-12-14.** Disponível em: <<http://www.xbrl.org/TR/2001/xbrl-2002-02-04.doc>>. Acesso em dezembro de 2001.

29. **Namespaces in XML - World Wide Web Consortium 14-January-1999.**
Disponível em: < <http://www.w3.org/TR/1999/REC-xml-names-19990114>>. Acesso em janeiro 2001.
30. **XML Linking Language (XLink) Version 1.0 - W3C Proposed Recommendation 20 December 2000.** Disponível em: < <http://www.w3.org/TR/2000/PR-xlink-20001220/>>. Acesso em janeiro 2001.
31. Coffin, Zachary. **Introducing XBRL: Reducing Time, Cutting Cost and Enabling Better Analysis for Financial Managers.** Disponível em <<http://groups.yahoo.com/group/xbml-public/files/>>. Acesso em: abril de 2001
32. Coffin, Zachary. **THE TOP TEM EFFECTS OF XBRL: THE FUTURE OF INTERNET REPORTING.** Prentice Hall, 2001. cap. 16.
33. Vlist, Eric Vander. **Introduction to XML Schemas.** Disponível em <http://www.xml.com/pub/a/2000/11/29/schemas/part1.htm>. Acesso em janeiro/2003.
34. Sítio do Banco Central do Brasil. Disponível em www.bcb.gov.br.
35. **Relatório de Análise de Contexto Institucional. Projeto Catálogo de Dados/Informações.** Disponível na Intranet do Banco Central do Brasil.
36. Fung, Khun Yee. **XSLT Interagindo com XML e HTML.** Rio de Janeiro: Editora Ciência Moderna Ltda, 2001.
37. Bruey, Don. Arquivos **InstView.xml** e **TxCompress.xml**. Disponível em <http://groups.yahoo.com/group/xbml-public>. Acesso em Janeiro 2002
38. **XBRL usage suggested for all u.s. federal government agencies in report by the joint financial management improvement program (JFMIP) which is lead by the department of treasury, general accounting office, office of management and budget and the office of personnel management.** Disponível em: <<http://www.xbrl.org/PressRoom/10december01.doc>>. Acesso em fevereiro de 2002
39. **Australian Prudential Regulatory Authority announces use of xbrl to collect financial statements from more than 12,000 super funds, insurers and banks in Australia.** Disponível em: <www.xbrl.org/PressRoom/29Oct011.doc>. Acesso em novembro de 2001
40. Bruey, Don. Arquivos **ElementsAndAttributes.html**. Disponível em <http://groups.yahoo.com/group/xbml-public>. Acesso em Janeiro 2002.
41. **Australian Prudential Regulatory Authority.** Disponível em: <<http://www.apra.gov.au>>

42. Coffin, Zachary & Matherne, Louis. **XBRL: A Technology Standard to Reduce Time, Cut Costs, and Enable Better Analysis for Tax Preparers.** Disponível em <<http://groups.yahoo.com/group/xbrl-public/files/>>. Acesso em abril de 2001
43. Watson, Liv A. McGuire, Brian L. & Cohen, Eric E. **Looking at Business Reports Through XBRL-Tinted Glasses.** Disponível em <<http://www.mamag.com/strategicfinance/2000/09g.htm>>. Acesso em outubro de 2001
44. XBRL Solutions. Apresenta textos sobre XBRL. Disponível em: <<http://www.xbrlsolutions.com>>.

Apêndice A – XBRL Instance Schema

```

<?xml version="1.0" encoding="UTF-8"?>
<schema targetNamespace="http://www.xbrl.org/2001/instance" xmlns:xbrli="http://www.xbrl.org/2001/instance"
xmlns:link="http://www.xbrl.org/2001/XLink/xbrlinkbase" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" xsi:schemaLocation="http://www.w3.org/2001/XMLSchema
http://www.w3.org/2001/XMLSchema.xsd">
  <annotation>
    <documentation>
      Taxonomy schema for XBRL.
      This schema extends the XSchema schema with several linking structures useful for XBRL taxonomies.
    </documentation>
  </annotation>
  <import namespace="http://www.xbrl.org/2001/XLink/xbrlinkbase"
    schemaLocation="xbrl-linkbase.xsd"/>
  <!-- ***** attributes ***** -->
  <attribute name="balance">
    <simpleType>
      <restriction base="string">
        <enumeration value="debit"/>
        <enumeration value="credit"/>
      </restriction>
    </simpleType>
  </attribute>
  <!-- ***** simple types ***** -->
  <!-- _____monetary -->
  <simpleType name="monetary">
    <annotation>
      <documentation>
        This datatype serves as the datatype for those financial concepts in a taxonomy which denote units in a currency.
        Instance items with this type should have a unit of measure from the ISO 4217 namespace of currencies.
      </documentation>
    </annotation>
    <restriction base="decimal"/>
  </simpleType>
  <!-- _____shares -->
  <simpleType name="shares">
    <annotation>
      <documentation>
        This datatype serves as the datatype for share based financial concepts.
      </documentation>
    </annotation>
    <restriction base="decimal">
      <minInclusive value="0"/>
    </restriction>
  </simpleType>
  <!-- ***** item types ***** -->
  <!-- _____itemType -->
  <!-- _____monetaryItemType -->
  <complexType name="monetaryItemType">
    <simpleContent>
      <extension base="xbrli:monetary">
        <attribute name="numericContext" type="IDREF" use="required"/>
        <anyAttribute namespace="##any" processContents="lax"/>
      </extension>
    </simpleContent>
  </complexType>
  <!-- _____sharesItemType -->
  <complexType name="sharesItemType">
    <simpleContent>
      <extension base="xbrli:shares">
        <attribute name="numericContext" type="IDREF" use="required"/>
        <anyAttribute namespace="##any" processContents="lax"/>
      </extension>
    </simpleContent>
  </complexType>

```

```

<!-- _____decimalItemType -->
<complexType name="decimalItemType">
  <simpleContent>
    <extension base="decimal">
      <attribute name="numericContext" type="IDREF" use="required"/>
      <anyAttribute namespace="##any" processContents="lax"/>
    </extension>
  </simpleContent>
</complexType>
<!-- _____stringItemType -->
<complexType name="stringItemType">
  <simpleContent>
    <extension base="string">
      <attribute name="nonNumericContext" type="IDREF" use="required"/>
      <anyAttribute namespace="##any" processContents="lax"/>
    </extension>
  </simpleContent>
</complexType>
<!-- _____uriItemType -->
<complexType name="uriItemType">
  <simpleContent>
    <extension base="anyURI">
      <attribute name="nonNumericContext" type="IDREF" use="required"/>
      <anyAttribute namespace="##any" processContents="lax"/>
    </extension>
  </simpleContent>
</complexType>
<!-- _____dateTimeItemType -->
<complexType name="dateTimeItemType">
  <simpleContent>
    <extension base="xbrli:dateUnion">
      <attribute name="nonNumericContext" type="IDREF" use="required"/>
      <anyAttribute namespace="##any" processContents="lax"/>
    </extension>
  </simpleContent>
</complexType>
<!-- ***** context: entityType ***** _->
<!-- _____context:entityType -->
<complexType name="entityType">
  <sequence>
    <element name="identifier">
      <complexType>
        <simpleContent>
          <extension base="string">
            <attribute name="scheme" type="anyURI" use="required"/>
          </extension>
        </simpleContent>
      </complexType>
    </element>
    <element ref="xbrli:segment" minOccurs="0"/>
  </sequence>
</complexType>
<!-- _____context:periodType -->
<complexType name="periodType">
  <choice>
    <sequence minOccurs="0">
      <element ref="xbrli:startDate"/>
      <element ref="xbrli:endDate"/>
    </sequence>
    <sequence minOccurs="0">
      <element ref="xbrli:startDate"/>
      <element ref="xbrli:duration"/>
    </sequence>
    <sequence minOccurs="0">
      <element ref="xbrli:duration"/>
      <element ref="xbrli:endDate"/>
    </sequence>
  </choice>

```



```

        <element ref="xbrli:instant" minOccurs="0"/>
        <element ref="xbrli:forever" minOccurs="0"/>
    </choice>
</complexType>
<!-- ***** context: period sub elements ***** -->
<simpleType name="dateUnion">
    <union memberTypes="date dateTime"/>
</simpleType>
<!-- _____startDate -->
<element name="startDate" type="xbrli:dateUnion"/>
<!-- _____endDate -->
<element name="endDate" type="xbrli:dateUnion"/>
<!-- _____duration -->
<element name="duration" type="duration"/>
<!-- _____instant -->
<element name="instant" type="xbrli:dateUnion"/>
<!-- _____forever -->
<element name="forever">
    <complexType/>
</element>
<!-- ***** context: segment element, scenario element ***** -->
<!-- _____context:segment element -->
<element name="segment">
    <complexType>
        <choice minOccurs="0" maxOccurs="unbounded">
            <element ref="xbrli:segment" minOccurs="0" maxOccurs="unbounded"/>
            <any namespace="##any" processContents="strict" minOccurs="0" maxOccurs="unbounded"/>
        </choice>
        <attribute name="name" type="string" use="optional"/>
    </complexType>
</element>
<!-- _____context:scenario element -->
<element name="scenario">
    <complexType>
        <choice minOccurs="0" maxOccurs="unbounded">
            <element ref="xbrli:scenario" minOccurs="0" maxOccurs="unbounded"/>
            <any namespace="##any" processContents="strict" minOccurs="0" maxOccurs="unbounded"/>
        </choice>
        <attribute name="name" type="string" use="optional"/>
    </complexType>
</element>
<!-- ***** context: unit element ***** -->
<!-- _____context:measure element -->
<element name="measure" type="QName"/>
<!-- _____context:operatorNameEnum -->
<simpleType name="operatorNameEnum">
    <restriction base="string">
        <enumeration value="multiply"/>
        <enumeration value="divide"/>
    </restriction>
</simpleType>
<!-- _____context:operator element -->
<element name="operator">
    <complexType>
        <choice minOccurs="2" maxOccurs="2">
            <element ref="xbrli:measure"/>
            <element ref="xbrli:operator"/>
        </choice>
        <attribute name="name" type="xbrli:operatorNameEnum" use="required"/>
    </complexType>
</element>
<!-- _____context:unitType -->
<complexType name="unitType">
    <choice>
        <element ref="xbrli:measure"/>
        <element ref="xbrli:operator"/>
    </choice>

```

```

</complexType>
<!-- ***** context: nonNumericContext, numericContext elements ***** -->
<!-- _____nonNumericContext -->
<element name="nonNumericContext">
  <complexType>
    <sequence>
      <element name="entity" type="xbrli:entityType"/>
      <element name="period" type="xbrli:periodType"/>
      <element name="unit" type="xbrli:unitType" minOccurs="0"/>
      <element ref="xbrli:scenario" minOccurs="0"/>
    </sequence>
    <attribute name="id" type="ID" use="required"/>
  </complexType>
</element>
<!-- _____numericContext -->
<element name="numericContext">
  <complexType>
    <sequence>
      <element name="entity" type="xbrli:entityType"/>
      <element name="period" type="xbrli:periodType"/>
      <element name="unit" type="xbrli:unitType"/>
      <element ref="xbrli:scenario" minOccurs="0"/>
    </sequence>
    <attribute name="id" type="ID" use="required"/>
    <attribute name="precision" type="string" use="required"/>
    <attribute name="cwa" type="boolean" use="required"/>
  </complexType>
</element>
<!-- ***** group elements ***** -->
<!-- _____item -->
<element name="item" type="anySimpleType" abstract="true"/>
<!-- _____tupleType -->
<complexType name="tupleType">
  <choice minOccurs="0" maxOccurs="unbounded">
    <element ref="xbrli:item" minOccurs="0" maxOccurs="unbounded"/>
    <element ref="xbrli:tuple" minOccurs="0" maxOccurs="unbounded"/>
  </choice>
  <anyAttribute namespace="##any" processContents="lax"/>
</complexType>
<!-- _____tuple -->
<element name="tuple" type="xbrli:tupleType" abstract="true"/>
<!-- _____group -->
  <element name="group">
    <complexType>
      <choice minOccurs="0" maxOccurs="unbounded">
        <element ref="xbrli:group" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="xbrli:item" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="xbrli:tuple" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="xbrli:nonNumericContext" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="xbrli:numericContext" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="link:linkbaseRef" minOccurs="0" maxOccurs="unbounded"/>
        <element ref="link:footnoteLink" minOccurs="0" maxOccurs="unbounded"/>
      </choice>
    </complexType>
  </element>
</schema>

```

Apêndice B – XBRL Linkbase Schema

```

<?xml version="1.0"?>
<schema targetNamespace="http://www.xbrl.org/2001/XLink/xbrlinkbase" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xl="http://www.xbrl.org/2001/XLink" xmlns:link="http://www.xbrl.org/2001/XLink/xbrlinkbase"
xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
<import namespace="http://www.xbrl.org/2001/XLink" schemaLocation="xl.xsd"/>
<import namespace="http://www.w3.org/XML/1998/namespace" schemaLocation="xml.xsd"/>
  <!-- useEnum -->
  <simpleType name="useEnum">
    <restriction base="string">
      <enumeration value="optional"/>
      <enumeration value="required"/>
      <enumeration value="prohibited"/>
    </restriction>
  </simpleType>
  <!-- xbrlArcType -->
  <complexType name="xbrlArcType">
    <complexContent>
      <extension base="xl:arcType">
        <attribute name="use" type="link:useEnum"/>
        <attribute name="priority" type="decimal"/>
      </extension>
    </complexContent>
  </complexType>
  <!-- calculationLink -->
  <element name="calculationLink" type="xl:extendedType" substitutionGroup="xl:extended">
  </element>
  <!-- presentationLink -->
  <element name="presentationLink" type="xl:extendedType" substitutionGroup="xl:extended">
  </element>
  <!-- definitionLink -->
  <element name="definitionLink" type="xl:extendedType" substitutionGroup="xl:extended"/>
  <!-- labelLink -->
  <element name="labelLink" type="xl:extendedType" substitutionGroup="xl:extended">
  </element>
  <!-- footnoteLink -->
  <element name="footnoteLink" type="xl:extendedType" substitutionGroup="xl:extended"/>
  <!-- referenceLink -->
  <element name="referenceLink" type="xl:extendedType" substitutionGroup="xl:extended"/>
  <!-- extendedLink -->
  <element name="extendedLink" type="xl:extendedType" substitutionGroup="xl:extended"/>
  <!-- calculationArc -->
  <element name="calculationArc" substitutionGroup="xl:arc">
    <complexType>
      <complexContent>
        <extension base="link:xbrlArcType">
          <attribute name="weight" type="decimal" use="required"/>
        </extension>
      </complexContent>
    </complexType>
  </element>
  <!-- presentationArc -->
  <element name="presentationArc" substitutionGroup="xl:arc">
    <complexType>
      <complexContent>
        <extension base="link:xbrlArcType">
          <attribute name="order" type="decimal" use="required"/>
        </extension>
      </complexContent>
    </complexType>
  </element>
  <!-- definitionArc -->
  <element name="definitionArc" type="link:xbrlArcType" substitutionGroup="xl:arc"/>
  <!-- labelArc -->
  <element name="labelArc" type="link:xbrlArcType" substitutionGroup="xl:arc">
  </element>
  <!-- footnoteArc -->

```

```

<element name="footnoteArc" type="link:xbrlArcType" substitutionGroup="xl:arc"/>
<!--      referenceArc      -->
<element name="referenceArc" type="link:xbrlArcType" substitutionGroup="xl:arc"/>
<!--      arc      -->
<element name="arc" type="xl:arcType" substitutionGroup="xl:arc"/>
<!--      loc      -->
<element name="loc" type="xl:locatorType" substitutionGroup="xl:locator"/>
<!--      linkbase      -->
<element name="linkbase">
  <complexType>
    <choice minOccurs="0" maxOccurs="unbounded">
      <element ref="link:linkbaseRef" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="xl:extended" minOccurs="0" maxOccurs="unbounded"/>
    </choice>
  </complexType>
</element>
<!--      linkbaseRef      -->
<element name="linkbaseRef">
  <complexType>
    <complexContent>
      <extension base="xl:simpleType">
        <sequence>
          <any namespace="http://www.w3.org/1999/xhtml" minOccurs="0" maxOccurs="unbounded"
processContents="skip"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
</element>
<!--      label      -->
<element name="label" substitutionGroup="xl:resource">
  <complexType mixed="true">
    <complexContent mixed="true">
      <extension base="xl:resourceType">
        <sequence>
          <any namespace="http://www.w3.org/1999/xhtml" processContents="skip" minOccurs="0"
maxOccurs="unbounded"/>
        </sequence>
        <attribute ref="xml:lang"/>
      </extension>
    </complexContent>
  </complexType>
</element>
<!--      footnote      -->
<element name="footnote" substitutionGroup="xl:resource">
  <complexType>
    <complexContent>
      <extension base="xl:resourceType">
        <attribute ref="xml:lang"/>
      </extension>
    </complexContent>
  </complexType>
</element>
<!--      part      -->
<element name="part" type="string" abstract="true"/>
<!--      reference      -->
<element name="reference" substitutionGroup="xl:resource">
  <annotation>
    <documentation>
      This datatype defines the reference to authoritative literature that may appear for a financial concept.
      These are references to published documents, not online resources.
    </documentation>
  </annotation>
  <complexType>
    <complexContent>
      <extension base="xl:resourceType">
        <choice minOccurs="0" maxOccurs="unbounded">

```

```
    <element ref="link:part" minOccurs="0" maxOccurs="unbounded"/>
  </choice>
  </extension>
</complexContent>
</complexType>
</element>
</schema>
```

Apêndice C – Linkbases

Os arquivos seguintes apresentam os documentos de linkbases de XBRL (label, reference, definition, presentation e calculation). Não serão apresentados todos os elementos criados para o estudo de casos, já que eles podem ser entendidos por analogia aos que estão presentes neste apêndice.

Label Link – arquivo bcb_label.xml

```

<linkbase
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbase xbrl-linkbase.xsd">
  <labellink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/balancoPatrimonial">
    <!--labels definidos aqui.-->
    <label xlink:type="resource" xlink:label="bcb_bp_br" xlink:title="bcb_bp_br"
      xlink:role="http://www.xbrl.org/linkprops/label/total" xml:lang="br">BALANÇO PATRIMONIAL</label>
    <label xlink:type="resource" xlink:label="bcb_ativo_br"
      xlink:title="bcb_ativo_br"
      xlink:role="http://www.xbrl.org/linkprops/label/total"
      xml:lang="br">ATIVO TOTAL</label>
    <label xlink:type="resource" xlink:label="bcb_atv_circ_real_lprazo_br"
      xlink:title="bcb_atv_circ_real_lprazo_br"
      xlink:role="http://www.xbrl.org/linkprops/label/total"
      xml:lang="br">ATIVO CIRCULANTE E REALIZÁVEL A LONGO PRAZO</label>
    <label xlink:type="resource" xlink:label="bcb_atv_ext_br"
      xlink:title="bcb_atv_ext_br"
      xlink:role="http://www.xbrl.org/linkprops/label/total"
      xml:lang="br">ATIVO CIRCULANTE E REALIZÁVEL A LONGO PRAZO
      EXTERNO</label>
    ...
  <!--Definições das localizações dos elementos-->
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#bp"
    xlink:label="bcb_bp" xlink:title="bp" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#ativo"
    xlink:label="bcb_ativo" xlink:title="ativo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_circ_real_lprazo"
    xlink:label="bcb_atv_circ_real_lprazo" xlink:title="atv_circ_real_lprazo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_ext"
    xlink:label="bcb_atv_ext" xlink:title="atv_ext" />
  ...
  <!--Definições dos arcos de ligação entre elementos e seus labels-->
  <labelArc xlink:type="arc" xlink:from="bcb_bp" xlink:to="bcb_bp_br"
    xlink:show="embed" xlink:actuate="onRequest" xlink:title="Vá para o label
    bcb_bp_br" xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-label"/>
  <labelArc xlink:type="arc" xlink:from="bcb_bp_br" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest" xlink:title="Vá para o elemento
    bcb_bp" xlink:arcrole="http://www.xbrl.org/linkprops/arc/label-element"/>
  <labelArc xlink:type="arc" xlink:from="bcb_ativo" xlink:to="bcb_ativo_br"
    xlink:show="embed" xlink:actuate="onRequest" xlink:title="Vá para o label
    bcb_ativo_br" xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-label"/>
  <labelArc xlink:type="arc" xlink:from="bcb_ativo_br" xlink:to="bcb_ativo"
    xlink:show="replace" xlink:actuate="onRequest" xlink:title="Vá para o elemento
    bcb_ativo" xlink:arcrole="http://www.xbrl.org/linkprops/arc/label-element"/>
  <labelArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo"
    xlink:to="bcb_atv_circ_real_lprazo_br" xlink:show="embed"
    xlink:actuate="onRequest" xlink:title="Vá para o label bcb_atv_circ_real_lprazo_br"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-label"/>
  <labelArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo_br"
    xlink:to="bcb_atv_circ_real_lprazo" xlink:show="replace"
    xlink:actuate="onRequest" xlink:title="Vá para o elemento
    bcb_atv_circ_real_lprazo"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/label-element"/>

```

```
<labelArc xlink:type="arc" xlink:from="bcb_atv_ext" xlink:to="bcb_atv_ext_br"
xlink:show="embed" xlink:actuate="onRequest" xlink:title="Vá para o label
    bcb_atv_ext_br" xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-label"/>
<labelArc xlink:type="arc" xlink:from="bcb_atv_ext_br" xlink:to="bcb_atv_ext"
xlink:show="replace" xlink:actuate="onRequest" xlink:title="Vá para o elemento
    bcb_atv_ext" xlink:arcrole="http://www.xbrl.org/linkprops/arc/label-element"/>
    ...
</labellink>
</linkbase>
```

Reference Link – arquivo bcb_reference.xml

```

<linkbase
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:bcb="www.bcb.gov.br"
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbase xbrl-linkbase.xsd">
  <referencelink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/balancoPatrimonial">
<!-- Referências definidas aqui -->
    <reference xlink:type="resource" xlink:label="bcb_bp_ref" xlink:title="bcb_bp_ref">
      <bcb:referencia>As demonstrações contábeis são elaboradas de acordo com a legislação aplicável ao Banco Central,
com destaque para a Lei 4.320/64, Lei 4.595/64 (alterada pelo Decreto-Lei 2.376/87), Decreto-Lei 278/67, Lei 7.862/89,
Medida Provisória 2.179/2001 e a Lei Complementar 101/2000 (Lei de Responsabilidade Fiscal). Em conformidade com essa
legislação, o Banco Central apura resultado e elabora balanços semestrais em junho e dezembro de cada ano, bem como
balançetes nos demais meses.
      </bcb:referencia>
    </reference>
<!-- Definições das localizações do elemento-->
    <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#bp"
      xlink:label="bcb_bp" xlink:title="bp"/>
<!--Definições dos arcos de ligação entre o elemento e sua referência-->
    <referenceArc xlink:type="arc" xlink:from="bcb_bp" xlink:to="bcb_bp_ref"
      xlink:show="embed"
      xlink:actuate="onRequest" xlink:title="Vá para a referência bcb_bp_ref"
      xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-reference"/>
    <referenceArc xlink:type="arc" xlink:from="bcb_bp_ref" xlink:to="bcb_bp"
      xlink:show="replace"
      xlink:actuate="onRequest" xlink:title="Vá para o elemento bcb_bp"
      xlink:arcrole="http://www.xbrl.org/linkprops/arc/reference-element"/>
  </referencelink>
</linkbase>

```


Definition Link – arquivo bcb_definition.xml

```

<linkbase
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbase xbrl-linkbase.xsd">
  <definitionlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/balancoPatrimonial">
  <!--Definições das localizações dos elementos-->
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#bp"
    xlink:label="bcb_bp" xlink:title="bp"
    xlink:role="http://www.xbrl.org/linkprops/locator/root" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#ativo"
    xlink:label="bcb_ativo" xlink:title="ativo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_circ_real_lprazo"
    xlink:label="bcb_atv_circ_real_lprazo" xlink:title="atv_circ_real_lprazo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_ext"
    xlink:label="bcb_atv_ext" xlink:title="atv_ext" />
    ...
  <!--Definições dos arcos-->
  <!-- Relacionamento de filho para pai -->
  <definitionArc xlink:type="arc" xlink:from="bcb_ativo" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest" xlink:title="Definição: Vá para bcb_bp"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <definitionArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo"
    xlink:to="bcb_ativo" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Definição: Vá para bcb_ativo"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <definitionArc xlink:type="arc" xlink:from="bcb_atv_ext"
    xlink:to="bcb_atv_circ_real_lprazo" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Definição: Vá para bcb_atv_circ_real_lprazo"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
    ...
  </definitionlink>
  <definitionlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/incomeStatement"/>
  <definitionlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/statementOfComprehensiveIncome"/>
  <definitionlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/statementOfStockholdersEquity"/>
  <definitionlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/cashFlows"/>
</linkbase>

```

Presentation Link – arquivo bcb_presentation.xml

```

<linkbase
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbase xbrl-linkbase.xsd">
  <presentationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/balancoPatrimonial">
  <!--Definições das localizações dos elementos-->
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#bp"
    xlink:label="bcb_bp" xlink:title="bp"
    xlink:role="http://www.xbrl.org/linkprops/locator/root" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#ativo"
    xlink:label="bcb_ativo" xlink:title="ativo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_circ_real_lprazo"
    xlink:label="bcb_atv_circ_real_lprazo" xlink:title="atv_circ_real_lprazo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_ext"
    xlink:label="bcb_atv_ext" xlink:title="atv_ext" />
    ...
  <!--Definições dos arcos-->
  <!-- Relacionamento de filho para pai -->
  <presentationArc xlink:type="arc" xlink:from="bcb_ativo" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Apresentação: Vá para bcb_bp" order="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <presentationArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo"
    xlink:to="bcb_ativo" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Apresentação: Vá para bcb_ativo" order="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
    <presentationArc xlink:type="arc" xlink:from="bcb_atv_ext"
      xlink:to="bcb_atv_circ_real_lprazo" xlink:show="replace"
      xlink:actuate="onRequest"
      xlink:title="Apresentação: Vá para bcb_atv_circ_real_lprazo" order="1"
      xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <presentationArc xlink:type="arc" xlink:from="bcb_atv_op_ext" xlink:to="bcb_atv_ext"
    xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Apresentação: Vá para bcb_atv_ext" order="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <presentationArc xlink:type="arc" xlink:from="bcb_atv_prov_crd_liq_dv_ext"
    xlink:to="bcb_atv_ext" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Apresentação: Vá para bcb_atv_ext" order="2"
    ...
  </presentationlink>
  <presentationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/incomeStatement"/>
  <presentationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/statementOfComprehensiveIncome"/>
  <presentationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/statementOfStockholdersEquity"/>
  <presentationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/cashFlows"/>
</linkbase>

```

Calculation Link – arquivo bcb_calculation.xml

```

<linkbase
  xmlns="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.xbrl.org/2001/XLink/xbrllinkbase xbrl-linkbase.xsd">
  <calculationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/balancoPatrimonial">
  <!--Definições das localizações dos elementos-->
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#bp"
    xlink:label="bcb_bp" xlink:title="bp"
    xlink:role="http://www.xbrl.org/linkprops/locator/root" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#ativo"
    xlink:label="bcb_ativo" xlink:title="ativo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_circ_real_lprazo"
    xlink:label="bcb_atv_circ_real_lprazo" xlink:title="atv_circ_real_lprazo" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_ext"
    xlink:label="bcb_atv_ext" xlink:title="atv_ext" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_op_ext"
    xlink:label="bcb_atv_op_ext" xlink:title="atv_op_ext" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_prov_crd_liq_dv_ext"
    xlink:label="bcb_atv_prov_crd_liq_dv_ext" xlink:title="atv_prov_crd_liq_dv_ext" />
  <loc xlink:type="locator" xlink:href="bcb_taxonomia.xsd#atv_int"
    xlink:label="bcb_atv_int" xlink:title="atv_int" />
  ...
  <!--Definições dos arcos-->
  <!-- Relacionamento de filho para pai -->
  <calculationArc xlink:type="arc" xlink:from="bcb_ativo" xlink:to="bcb_bp"
    xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_bp" weight="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <calculationArc xlink:type="arc" xlink:from="bcb_atv_circ_real_lprazo"
    xlink:to="bcb_ativo" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_ativo" weight="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <calculationArc xlink:type="arc" xlink:from="bcb_atv_ext"
    xlink:to="bcb_atv_circ_real_lprazo" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_atv_circ_real_lprazo" weight="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <calculationArc xlink:type="arc" xlink:from="bcb_atv_op_ext" xlink:to="bcb_atv_ext"
    xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_atv_ext" weight="1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  <calculationArc xlink:type="arc" xlink:from="bcb_atv_prov_crd_liq_dv_ext"
    xlink:to="bcb_atv_ext" xlink:show="replace" xlink:actuate="onRequest"
    xlink:title="Cálculo: Vá para bcb_atv_ext" weight="-1"
    xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"/>
  ...
</calculationlink>
  <calculationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/incomeStatement"/>
  <calculationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/statementOfComprehensiveIncome"/>
  <calculationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/statementOfStockholdersEquity"/>
  <calculationlink xlink:type="extended"
    xlink:role="http://www.xbrl.org/linkprops/extended/cashFlows"/>
</linkbase>

```

Apêndice D – Taxonomia - arquivo bcb_taxonomia.xsd

```

<schema
  targetNamespace="www.bcb.gov.br"
  xmlns="http://www.w3.org/2001/XMLSchema"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:link="http://www.xbrl.org/2001/XLink/xbrlinkbase"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  xmlns:xbrli="http://www.xbrl.org/2001/instance"
  xmlns:bcb="http://www.bcb.gov.br"
  elementFormDefault="qualified">
  <annotation>
    <appinfo>
<link:linkbaseRef xlink:type="simple" xlink:href="bcb_presentation.xml"
  xlink:actuate="onRequest" xlink:role="http://www.xbrl.org/linkprops/linkRef/presentation"
  xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbase">
  <xhtml:p>Links para relacionamentos de apresentação</xhtml:p>
</link:linkbaseRef>
<link:linkbaseRef xlink:type="simple" xlink:href="bcb_calculation.xml"
  xlink:actuate="onRequest" xlink:role="http://www.xbrl.org/linkprops/linkRef/calculation"
  xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbase">
  <xhtml:p>Links para relacionamentos de cálculos</xhtml:p>
</link:linkbaseRef>
<link:linkbaseRef xlink:type="simple" xlink:href="bcb_definition.xml"
  xlink:actuate="onRequest" xlink:role="http://www.xbrl.org/linkprops/linkRef/definition"
  xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbase">
  <xhtml:p>Links para relacionamentos de definição</xhtml:p>
</link:linkbaseRef>
<link:linkbaseRef xlink:type="simple" xlink:href="bcb_label.xml"
  xlink:actuate="onRequest" xlink:role="http://www.xbrl.org/linkprops/linkRef/label"
  xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbase">
  <xhtml:p>Links para labels</xhtml:p>
</link:linkbaseRef>
<link:linkbaseRef xlink:type="simple" xlink:href="bcb_reference.xml"
  xlink:actuate="onRequest" xlink:role="http://www.xbrl.org/linkprops/linkRef/reference"
  xlink:arcrole="http://www.w3.org/1999/xlink/properties/linkbase">
  <xhtml:p>Links para referências</xhtml:p>
</link:linkbaseRef>
    </appinfo>
  </annotation>
<!-- Import de XBRL instance schema (exigido em toda taxonomia XBRL) -->
<import namespace="http://www.xbrl.org/2001/instance" schemaLocation="xbrl-instance.xsd"/>
<!-- Definição de elementos -->
  <element name="bp" id="bp" type="xbrli:tupleType" substitutionGroup="xbrli:tuple"/>
  <element name="ativo" id="ativo" type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item">
    <annotation>
      <documentation>Seção que contém os ativos do balanço
        patrimonial.</documentation>
    </annotation>
  </element>
  <element name="atv_circ_real_lprazo" id="atv_circ_real_lprazo"
    type="xbrli:monetaryItemType" substitutionGroup="xbrli:item"/>
  <element name="atv_ext" id="atv_ext" type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item"/>
  <element name="atv_op_ext" id="atv_op_ext" type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item"/>
  <element name="atv_prov_crd_liq_dv_ext" id="atv_prov_crd_liq_dv_ext"
    type="xbrli:monetaryItemType" substitutionGroup="xbrli:item"/>
  <element name="atv_int" id="atv_int" type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item"/>
  <element name="atv_opr_int" id="atv_opr_int" type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item"/>
    . . .
  <element name="passivo" id="passivo" type="xbrli:monetaryItemType"
    substitutionGroup="xbrli:item">
    <annotation>
      <documentation>Seção que contém os passivos do balancete

```

```

                                patrimonial.</documentation>
        </annotation>
    </element>
    <element name="pssv_circ_exg_lprazo" id="pssv_circ_exg_lprazo"
        type="xbrli:monetaryItemType" substitutionGroup="xbrli:item"/>
    <element name="pssv_ext" id="pssv_ext" type="xbrli:monetaryItemType"
        substitutionGroup="xbrli:item"/>
        ...
    <!-- Os elementos abaixo foram adicionados para declarar a diretoria da instituição -->
    <element name="diretores" type="string" substitutionGroup="link:part">
    <element name="presidente" type="string" substitutionGroup="link:part"/>
    <element name="diretorDipec" type="string" substitutionGroup="link:part"/>
    <element name="diretorDinor" type="string" substitutionGroup="link:part"/>
    <element name="diretorDirad" type="string" substitutionGroup="link:part"/>
    <element name="diretorDipom" type="string" substitutionGroup="link:part"/>
    <element name="diretorDifis" type="string" substitutionGroup="link:part"/>
    <element name="diretorDirex" type="string" substitutionGroup="link:part"/>
    <element name="diretorDifip" type="string" substitutionGroup="link:part"/>
    <element name="chefedeafi" type="string" substitutionGroup="link:part"/>
    </element>
    <!-- O elemento abaixo foi adicionado para declarar o elememnto cenário usado no documento XBRL -->
    <element name="cenarioType">
        <simpleType>
            <restriction base="string">
                <enumeration value="final"/>
                <enumeration value="preliminar"/>
            </restriction>
        </simpleType>
    </element>
    <!-- O elemento descrição foi adicionado para ser usado no documento XBRL -->
    <element name="descrição" type="string" substitutionGroup="link:part"/>
    <!-- O elemento abaixo foi adicionado para ser usado como elemento de referência -->
    <element name="referencia" type="string" substitutionGroup="link:part"/>
</schema>

```

Apêndice E – Documento XBRL - arquivo bcb_bp_28_02_2001.xml

```

<?xml version="1.0"?>
<?xml-stylesheet type="text/xsl" href="bcb_InstView.xsl"?>
<group
  xmlns="http://www.xbrl.org/2001/instance"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:link="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:iso4217="http://www.iso.org/4217"
  xmlns:bcb="www.bcb.gov.br"
  xsi:schemaLocation="www.bcb.gov.br bcb_taxonomia.xsd">
  <bcb:bp>
<bcb:ativo numericContext="c1">270190618</bcb:ativo>
<bcb:atv_circ_real_lprazo numericContext="c1">
  261376808</bcb:atv_circ_real_lprazo>
<bcb:atv_ext numericContext="c1">81072891</bcb:atv_ext>
<bcb:atv_op_ext numericContext="c1">81162933</bcb:atv_op_ext>
<bcb:atv_prov_crd_liq_dv_ext numericContext="c1">
  90042</bcb:atv_prov_crd_liq_dv_ext>
<bcb:atv_int numericContext="c1">180303917</bcb:atv_int>
<bcb:atv_opr_int numericContext="c1">148906640</bcb:atv_opr_int>
<bcb:atv_emp_if numericContext="c1">8469741</bcb:atv_emp_if>
<bcb:atv_tpf_ma numericContext="c1">138444311</bcb:atv_tpf_ma>
<bcb:atv_ntn_a10 numericContext="c1">1903229</bcb:atv_ntn_a10>
<bcb:atv_ntn_p numericContext="c1">2788</bcb:atv_ntn_p>
<bcb:atv_otr_opr numericContext="c1">86599</bcb:atv_otr_opr>
  <bcb:atv_prov_pdr_crd_liq_dv_otr numericContext="c1">
    28</bcb:atv_prov_pdr_crd_liq_dv_otr>
<bcb:atv_otr_crd numericContext="c1">30561931</bcb:atv_otr_crd>
<bcb:atv_crd_rcb numericContext="c1">36328489</bcb:atv_crd_rcb>
<bcb:atv_crd_ctr numericContext="c1">1745142</bcb:atv_crd_ctr>
<bcb:atv_tit_rcb numericContext="c1">1717865</bcb:atv_tit_rcb>
<bcb:atv_crd_da numericContext="c1">885102</bcb:atv_crd_da>
<bcb:atv_dvd_ci numericContext="c1">8918</bcb:atv_dvd_ci>
<bcb:atv_prov_crd_liq_dv_otr_crd numericContext="c1">
  10123585</bcb:atv_prov_crd_liq_dv_otr_crd>
<bcb:atv_otr_cnt numericContext="c1">111896</bcb:atv_otr_cnt>
<bcb:atv_dep_vir numericContext="c1">417968</bcb:atv_dep_vir>
<bcb:atv_crd_agr numericContext="c1">59432</bcb:atv_crd_agr>
<bcb:atv_dm_ctn numericContext="c1">52376</bcb:atv_dm_ctn>
<bcb:atv_prov_crd_liq_dv_pdr_jud numericContext="c1">
  417880</bcb:atv_prov_crd_liq_dv_pdr_jud>
<bcb:atv_vlr_bens numericContext="c1">14221</bcb:atv_vlr_bens>
<bcb:atv_amx numericContext="c1">1937</bcb:atv_amx>
<bcb:atv_imv_nuso numericContext="c1">12284</bcb:atv_imv_nuso>
<bcb:atv_dps_apr numericContext="c1">709229</bcb:atv_dps_apr>
<bcb:atv_dps_apr_ctr numericContext="c1">577218</bcb:atv_dps_apr_ctr>
<bcb:atv_dps_apr_nbc numericContext="c1">131980</bcb:atv_dps_apr_nbc>
<bcb:atv_otr_dps_apr numericContext="c1">31</bcb:atv_otr_dps_apr>
<bcb:atv_prmt numericContext="c1">8813810</bcb:atv_prmt>
<bcb:atv_invt numericContext="c1">8079757</bcb:atv_invt>
<bcb:atv_qt_cap_org_intl numericContext="c1">
  8079757</bcb:atv_qt_cap_org_intl>
<bcb:atv_fmi numericContext="c1">8006408</bcb:atv_fmi>
<bcb:atv_bis numericContext="c1">73349</bcb:atv_bis>
<bcb:atv_imbz numericContext="c1">734053</bcb:atv_imbz>
<bcb:atv_mov numericContext="c1">103122</bcb:atv_mov>
<bcb:atv_imov numericContext="c1">764800</bcb:atv_imov>
<bcb:atv_dprc_acm numericContext="c1">133998</bcb:atv_dprc_acm>
<bcb:atv_intg numericContext="c1">129</bcb:atv_intg>
<bcb:passivo numericContext="c1">270190618</bcb:passivo>
<bcb:pssv_circ_exg_lprazo numericContext="c1">
  238023319</bcb:pssv_circ_exg_lprazo>
<bcb:pssv_ext numericContext="c1">18388337</bcb:pssv_ext>
<bcb:pssv_obr_moed_est numericContext="c1">
  9902422</bcb:pssv_obr_moed_est>

```

<bcb:pssv_dpt_org_intl numericContext="c1">8431559</bcb:pssv_dpt_org_intl>
 <bcb:pssv_dm_ctn numericContext="c1">54356</bcb:pssv_dm_ctn>
 <bcb:pssv_int numericContext="c1">219634982</bcb:pssv_int>
 <bcb:pssv_dpt_if numericContext="c1">35033784</bcb:pssv_dpt_if>
 <bcb:pssv_rb numericContext="c1">19739312</bcb:pssv_rb>
 <bcb:pssv_dpt_sbpe numericContext="c1">16370254</bcb:pssv_dpt_sbpe>
 <bcb:pssv_rcdj numericContext="c1">7048271</bcb:pssv_rcdj>
 <bcb:pssv_dm_ctn_if numericContext="c1">268165</bcb:pssv_dm_ctn_if>
 <bcb:pssv_rec_tpf_if numericContext="c1">8392218</bcb:pssv_rec_tpf_if>
 <bcb:pssv_dpt_gf numericContext="c1">72446542</bcb:pssv_dpt_gf>
 <bcb:pssv_otr_dpt numericContext="c1">27261</bcb:pssv_otr_dpt>
 <bcb:pssv_otr_dpt_vgo numericContext="c1">2736973</bcb:pssv_otr_dpt_vgo>
 <bcb:pssv_rec_tpf_otr numericContext="c1">2736972</bcb:pssv_rec_tpf_otr>
 <bcb:pssv_dm_ctn_dpt numericContext="c1">27260</bcb:pssv_dm_ctn_dpt>
 <bcb:pssv_otr_exg numericContext="c1">11031472</bcb:pssv_otr_exg>
 <bcb:pssv_otr_exg_nbc numericContext="c1">94611389</bcb:pssv_otr_exg_nbc>
 <bcb:pssv_otr_exg_opr_tit numericContext="c1">
 15254398</bcb:pssv_otr_exg_opr_tit>
 <bcb:pssv_otr_exg_rst_tn numericContext="c1">
 1113019</bcb:pssv_otr_exg_rst_tn>
 <bcb:pssv_otr_exg_dm_ctn numericContext="c1">
 52666</bcb:pssv_otr_exg_dm_ctn>
 <bcb:pssv_otr_ctn numericContext="c1">1095923</bcb:pssv_otr_ctn>
 <bcb:pssv_otr_obr_moed_est numericContext="c1">
 19781</bcb:pssv_otr_obr_moed_est>
 <bcb:pssv_rem_disp_gf numericContext="c1">227755</bcb:pssv_rem_disp_gf>
 <bcb:pssv_vlr_rec_ctr numericContext="c1">577218</bcb:pssv_vlr_rec_ctr>
 <bcb:pssv_rct_apr_nbc numericContext="c1">185842</bcb:pssv_rct_apr_nbc>
 <bcb:pssv_otr_dm_ctn numericContext="c1">85327</bcb:pssv_otr_dm_ctn>
 <bcb:pssv_mc numericContext="c1">28152323</bcb:pssv_mc>
 <bcb:pssv_pl numericContext="c1">3472415</bcb:pssv_pl>
 <bcb:pssv_ptr numericContext="c1">2576356</bcb:pssv_ptr>
 <bcb:pssv_rsv_ctg numericContext="c1">404406</bcb:pssv_rsv_ctg>
 <bcb:pssv_rsv_avl numericContext="c1">491653</bcb:pssv_rsv_avl>
 <bcb:pssv_cr numericContext="c1">542561</bcb:pssv_cr>
 <bcb:pssv_cr_crd numericContext="c1">11730759</bcb:pssv_cr_crd>
 <bcb:pssv_cr_dvd numericContext="c1">11188198</bcb:pssv_cr_dvd>
 <bcb:diretores nonNumericContext="c2">
 <bcb:presidente nonNumericContext="c2">Armínio Fraga
 Neto</bcb:presidente>
 <bcb:diretorDifip nonNumericContext="c2">Carlos Eduardo de
 Freitas</bcb:diretorDifip>
 <bcb:diretorDirex nonNumericContext="c2">Daniel Luiz
 Gleizer</bcb:diretorDirex>
 <bcb:diretorDirad nonNumericContext="c2">Edison Bernades dos
 Santos</bcb:diretorDirad>
 <bcb:diretorDipec nonNumericContext="c2">Ilan
 Goldfajn</bcb:diretorDipec>
 <bcb:diretorDipom nonNumericContext="c2">Luiz Fernando
 Figueiredo</bcb:diretorDipom>
 <bcb:diretorDinor nonNumericContext="c2">Sergio Darcy da Silva
 Alves</bcb:diretorDinor>
 <bcb:diretorDifis nonNumericContext="c2">Tereza Cristina Grossi
 Togni</bcb:diretorDifis>
 <bcb:chefedeafi nonNumericContext="c2">Jefferson Moreira - CRC: DF
 7333</bcb:chefedeafi>
 </bcb:diretores>
 </bcb:bp>
 <numericContext id="c1" precision="10" cwa="true">
 <entity>
 <identifier scheme="http://www.bcb.com.br/">Banco Central do
 Brasil</identifier>
 </entity>
 <segment/>
 </segment>
 <period>
 <instant>28-02-2001</instant>
 </period>
 </numericContext>

```
<unit>
<measure>iso4217:BRL</measure>
</unit>
<scenario name="Balço Patrimonial">
  <bcbscenarioType>final</bcbscenarioType>
</scenario>
</numericContext>
<nonNumericContext id="c2">
  <entity>
    <identifier scheme="http://www.bcb.com.br/">Banco Central do
      Brasil</identifier>
  </entity>
  <period>
    <instant>28-02-2001</instant>
  </period>
</nonNumericContext>
</group>
```


Apêndice F – Definição de atributos Xlink - arquivo Xlink.xsd

```

<schema targetNamespace="http://www.w3.org/1999/xlink" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <!--      typeEnum      -->
  <simpleType name="typeEnum">
    <restriction base="string">
      <enumeration value="simple"/>
      <enumeration value="extended"/>
      <enumeration value="locator"/>
      <enumeration value="arc"/>
      <enumeration value="resource"/>
      <enumeration value="title"/>
    </restriction>
  </simpleType>
  <!--      showEnum      -->
  <simpleType name="showEnum">
    <restriction base="string">
      <enumeration value="new"/>
      <enumeration value="replace"/>
      <enumeration value="embed"/>
      <enumeration value="other"/>
      <enumeration value="none"/>
    </restriction>
  </simpleType>
  <!--      actuateEnum    -->
  <simpleType name="actuateEnum">
    <restriction base="string">
      <enumeration value="onLoad"/>
      <enumeration value="onRequest"/>
      <enumeration value="other"/>
      <enumeration value="none"/>
    </restriction>
  </simpleType>
  <attribute name="type" type="xlink:typeEnum"/>
  <attribute name="role" type="string"/>
  <attribute name="arcrole" type="string"/>
  <attribute name="title" type="string"/>
  <attribute name="show" type="xlink:showEnum"/>
  <attribute name="actuate" type="xlink:actuateEnum"/>
  <attribute name="label" type="NCName"/>
  <attribute name="from" type="NCName"/>
  <attribute name="to" type="NCName"/>
  <attribute name="href" type="anyURI"/>
</schema>

```

Apêndice G – Definição de elementos Xlink - arquivo Xl.xsd

```

<schema targetNamespace="http://www.xbrl.org/2001/XLink" xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xl="http://www.xbrl.org/2001/XLink" xmlns:xlink="http://www.w3.org/1999/xlink" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <import namespace="http://www.w3.org/1999/xlink" schemaLocation="xlink.xsd"/>
  <!--          titleType          -->
  <complexType name="titleType">
    <complexContent>
      <restriction base="anyType">
        <attribute ref="xlink:type"/>
      </restriction>
    </complexContent>
  </complexType>
  <!--          simpleType          -->
  <complexType name="simpleType">
    <complexContent>
      <restriction base="anyType">
        <attribute ref="xlink:type"/>
        <attribute ref="xlink:href"/>
        <attribute ref="xlink:role"/>
        <attribute ref="xlink:arcrole"/>
        <attribute ref="xlink:title"/>
        <attribute ref="xlink:show"/>
        <attribute ref="xlink:actuate"/>
      </restriction>
    </complexContent>
  </complexType>
  <!--          extendedType          -->
  <complexType name="extendedType">
    <choice minOccurs="0" maxOccurs="unbounded">
      <element ref="xl:locator" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="xl:arc" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="xl:resource" minOccurs="0" maxOccurs="unbounded"/>
      <element ref="xl:title"/>
    </choice>
    <attribute ref="xlink:type"/>
    <attribute ref="xlink:role"/>
    <attribute ref="xlink:title"/>
  </complexType>
  <!--          locatorType          -->
  <complexType name="locatorType">
    <choice minOccurs="0" maxOccurs="unbounded">
      <element ref="xl:title"/>
    </choice>
    <attribute ref="xlink:type"/>
    <attribute ref="xlink:href"/>
    <attribute ref="xlink:role"/>
    <attribute ref="xlink:title"/>
    <attribute ref="xlink:label"/>
  </complexType>
  <!--          arcType          -->
  <complexType name="arcType">
    <choice minOccurs="0" maxOccurs="unbounded">
      <element ref="xl:title"/>
    </choice>
    <attribute ref="xlink:type"/>
    <attribute ref="xlink:arcrole"/>
    <attribute ref="xlink:title"/>
    <attribute ref="xlink:show"/>
    <attribute ref="xlink:actuate"/>
    <attribute ref="xlink:from"/>
    <attribute ref="xlink:to"/>
  </complexType>
  <!--          arcLinkbaseDefType          -->
  <complexType name="arcLinkbaseDefType">
    <complexContent>

```

```

    <restriction base="xl:arcType">
      <choice minOccurs="0" maxOccurs="unbounded">
        <element ref="xl:title"/>
      </choice>
      <attribute ref="xlink:type"/>
      <attribute ref="xlink:arcrole"/>
      <attribute ref="xlink:title"/>
      <attribute ref="xlink:show"/>
      <attribute ref="xlink:actuate"/>
      <attribute ref="xlink:from"/>
      <attribute ref="xlink:to"/>
    </restriction>
  </complexContent>
</complexType>
<!-- resourceType -->
<complexType name="resourceType">
  <complexContent>
    <restriction base="anyType">
      <attribute ref="xlink:type"/>
      <attribute ref="xlink:role"/>
      <attribute ref="xlink:title"/>
      <attribute ref="xlink:label"/>
    </restriction>
  </complexContent>
</complexType>
<!-- elements -->
<element name="title" type="xl:titleType" abstract="true"/>
<element name="simple" type="xl:simpleType" abstract="true"/>
<element name="extended" type="xl:extendedType" abstract="true"/>
<element name="locator" type="xl:locatorType" abstract="true"/>
<element name="arc" type="xl:arcType" abstract="true"/>
<element name="resource" type="xl:resourceType" abstract="true">
</element>
<!-- element arcLinkbaseDef -->
<element name="arcLinkbaseDef" type="xl:arcLinkbaseDefType" abstract="true"
  substitutionGroup="xl:arc"/>
</schema>

```

Apêndice H – Folha de estilo - arquivo bcb_InstView.xsl

<!--

Essa folha de estilo foi escrita baseada no código XSLT desenvolvido por Don Bruey, Creative Solutions (January 2002, dbruey@creativesolutions.com), para a transformação de um documento XBRL Versão 2 em outro HTML. Modificações foram feitas para se utilizar os linkbases calculation, reference e label. Os outros tipos de linkbase (presentation e definition) não foram aplicados neste exemplo, porém o código original que permite usá-los foi mantido.

Este código XSLT, quando aplicado a um documento codificado com a versão 2.0 da especificação XBRL, fornecerá uma tabela HTML contendo informações de seus itens. Essas informações são obtidas pela carga da taxonomia, a qual aponta para os linkbases label, calculation e reference, que auxiliam na formatação da tabela.

Essa transformação faz uso de uma outra transformação, contida no arquivo TxCompress.xsl, cuja saída é uma árvore contendo as informações da taxonomia usada por este código.

Na criação deste código não foi considerado seu desempenho quando utilizado com grandes taxonomias e arquivos de documentos.

O processador XSLT utilizado foi o MSXML 4, se este código for testado em outro processador, a função msxml node-set() deverá ser substituída por outra apropriada. Esta é a única função XSLT usada que não é padronizada. -->

```
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xbrl="http://www.xbrl.org/2001/instance"
  xmlns:msxsl="urn:schemas-microsoft-com:xsl"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:xbrlLink="http://www.xbrl.org/2001/XLink/xbrlLinkbase"
  xmlns:bcb="www.bcb.gov.br">
<xsl:output method="html" indent="yes"/>

<!-- Inclui outro arquivo XSL que manuseia as informações contidas na taxonomia usadas por esta transformação. -->
<xsl:include href="bcb_TxCompress.xsl"/>
<!-- Elemento para formatar um valor numérico -->
<xsl:decimal-format decimal-separator="," grouping-separator="."/>
<xsl:template match="/">
<!-- Variável que chama o template para carregar a taxonomia -->
<xsl:variable name="taxonomyInfo">
  <xsl:call-template name="loadTaxonomies" />
</xsl:variable>
<!-- Variável que copia os elementos context para a árvore resultante -->
<xsl:variable name="contexts">
  <xsl:copy-of select="//xbrl:nonNumericContext | //xbrl:numericContext" />
</xsl:variable>
<!-- Chamada do template que irá fazer a transformação do documento em uma tabela HTML, passando os parâmetros que
contêm as variáveis que carregam os elementos da taxonomia e os contextos -->
<xsl:call-template name="outputInstElts">
  <xsl:with-param name="txElts" select="$taxonomyInfo"/>
  <xsl:with-param name="contexts" select="$contexts"/>
</xsl:call-template>
</xsl:template>
<!-- Template que contém um loop que informa para outro template (itemOutput) o item de saída -->
<xsl:template name="outputInstElts">
<xsl:param name="txElts"/>
<xsl:param name="contexts"/>
<HTML>
  <HEAD>
    <table border="0">
      <tr>
        <td><IMG SRC="bcb.gif" width="35"/></td>
        <td>
          <font face="impact" size="5" color="blue"><b>Demonstrações Contábeis do Banco Central do Brasil<br/>
          </b></font>
        </td>
      </tr>
    </table>
    <hr/>
    <BlockQuote/>
  </HEAD>
  <body text="#003366" BGCOLOR="#FFFFDD">
<!-- Seleciona cada elemento filho do elemento raiz que contenha um dos atributos contexto -->
```

```

    <xsl:for-each select=".*[@numericContext | @nonNumericContext]">
<!-- Chama o template que envia o item para a saída, com os parâmetros que passam o elemento, as informações contida na
taxonomia e o contexto do elemento -->
    <xsl:call-template name="itemOutput">
        <xsl:with-param name="item" select="." />
        <xsl:with-param name="txElts" select="$txElts" />
        <xsl:with-param name="contexts" select="$contexts" />
    </xsl:call-template>
</xsl:for-each>
<CENTER>
<TABLE cellPadding="10" border="0">
<TBODY>
<TR>
<TD align="right" width="50%"><A href="http://www.bcb.gov.br/"
target="_top"><IMG
src="a-home.gif"
align="middle" border="0"/></A>&#160;<A href="http://www.bcb.gov.br/"
target="_top"><FONT face="arial" size="1">Volta à homepage</FONT></A>&#160;&#160;</TD>
<TD align="middle" width="50%"><A
href="http://www.bcb.gov.br/htms/bc_atende/Default.htm" target="_top"><IMG
src="a-atend.gif"
align="middle" border="0"/></A>&#160;<A
href="http://www.bcb.gov.br/htms/bc_atende/Default.htm" target="_top"><FONT
face="arial" size="1">BC
Atende</FONT></A>&#160;&#160;</TD></TR></TBODY></TABLE></CENTER>
</body>
</HTML>
</xsl:template>
<!-- Fim do template <outputInstElts> -->
<!-- Este template fornece as informações de saída de cada elemento, com os seguintes parâmetros:
$item - O item do documento XBRL
$txElts - Uma árvore das informações da taxonomia criada na transformação TxCompress.xml. Este passo acontece no
template loadTaxonomies.
$contexts - Uma árvore contendo todos os elementos numericContext e nonNumericContext. Usada para encontrar as
informações de cada elemento neste template. -->
<xsl:template name="itemOutput">
    <xsl:param name="item" />
    <xsl:param name="txElts" />
    <xsl:param name="contexts" />
<!-- Variável com as informações de contexto associada ao item. Associa o item ao contexto cujo atributo ID é igual ao
atributo de contexto do item. A função node-set converte a árvore em um nó-->
    <xsl:variable name="curContext">
        <xsl:choose>
            <xsl:when test="$item/@nonNumericContext">
                <xsl:copy-of select="msxsl:node-set($contexts)//xbrl:nonNumericContext[@id= $item/@nonNumericContext]" />
            </xsl:when>
            <xsl:when test="$item/@numericContext">
                <xsl:copy-of select="msxsl:node-set($contexts)//xbrl:numericContext[@id= $item/@numericContext]" />
            </xsl:when>
        </xsl:choose>
    </xsl:variable>
<!-- Variável que contém o período, nas várias possibilidades em que pode ser especificado (instant, forever, duration,
startDate/duration,
duration/endDate, startDate/endDate). -->
    <xsl:variable name="periodValue">
        <xsl:choose>
            <xsl:when test="msxsl:node-set($curContext)//xbrl:period/xbrl:instant">
                <xsl:value-of select="msxsl:node-set($curContext)//xbrl:period/xbrl:instant"/>
            </xsl:when>
            <xsl:when test="msxsl:node-set($curContext)//xbrl:period/xbrl:duration">
                Duração: <xsl:value-of select="msxsl:node-set($curContext)//xbrl:period/xbrl:duration"/>
            </xsl:choose>
            <xsl:when test="msxsl:node-set($curContext)//xbrl:period/xbrl:startDate">
                Início: <xsl:value-of select="msxsl:node-set($curContext)//xbrl:period/xbrl:startDate"/>
            </xsl:when>
            <xsl:when test="msxsl:node-set($curContext)//xbrl:period/xbrl:endDate">

```

```

    Fim: <xsl:value-of select="msxsl:node-set($curContext)//xbrl:period/xbrl:endDate"/>
  </xsl:when>
</xsl:choose>
</xsl:when>
  <xsl:when test="msxsl:node-set($curContext)//xbrl:period/xbrl:startDate and msxsl:node-
set($curContext)//xbrl:period/xbrl:endDate">
    Datas de <xsl:value-of select="msxsl:node-set($curContext)//xbrl:period/xbrl:startDate"/> até <xsl:value-of
select="msxsl:node-set($curContext)//xbrl:period/xbrl:endDate"/>
  </xsl:when>
  <xsl:when test="msxsl:node-set($curContext)//xbrl:period/xbrl:forever">
    Para sempre
  </xsl:when>
</xsl:choose>
</xsl:variable>
<!-- Fim da variável $periodValue -->
<!-- Variável que mostra o valor da unidade do contexto. Seu valor deve ser semelhante a "ISO4217:EUR/xbrli:shares" -->
<xsl:variable name="unitValue">
  <xsl:if test="msxsl:node-set($curContext)//xbrl:unit" >
    <xsl:choose>
      <xsl:when test="msxsl:node-set($curContext)//xbrl:unit/xbrl:operator">
        <xsl:choose>
          <xsl:when test="msxsl:node-set($curContext)//xbrl:unit/xbrl:operator/@name = 'divide'">
            <xsl:value-of select="msxsl:node-set($curContext)//xbrl:unit/xbrl:operator/xbrl:measure[1]"/><xsl:value-of
select="msxsl:node-set($curContext)//xbrl:unit/xbrl:operator/xbrl:measure[2]"/>
          </xsl:when>
          <xsl:otherwise>
            <xsl:value-of select="msxsl:node-set($curContext)//xbrl:unit/xbrl:operator/xbrl:measure[1]"/>*<xsl:value-of
select="msxsl:node-set($curContext)//xbrl:unit/xbrl:operator/xbrl:measure[2]"/>
          </xsl:otherwise>
        </xsl:choose>
      </xsl:when>
      <xsl:otherwise>
        <xsl:value-of select="msxsl:node-set($curContext)//xbrl:unit/xbrl:measure"/>
      </xsl:otherwise>
    </xsl:choose>
  </xsl:if>
</xsl:variable>
<!-- fim da variável $unitValue -->
<!-- Variável que obtém o label com o atributo xml:lang="br" para o item em questão. Cujo predicado:
[local-name(.) = local-name($item) and namespace-uri(.) = namespace-uri($item)]
poderia ser escrito da seguinte forma:
[name(.) = name($item)], desde que os elementos compartilhem o mesmo local e URI namespace -->
<xsl:variable name="labelValue">
  <xsl:value-of select="msxsl:node-set($xElts)//*[ local-name(.) = local-name($item) and namespace-uri(.) = namespace-
uri($item)]//bcb:labelSet/bcb:label[@xml:lang='br']" />
</xsl:variable>
<!-- Variável que recupera o valor do atributo Xlink:role do elemento label, usado para diferenciar as cores da célula da
tabela, em função de ser um elemento totalizador dos valores de seus filhos -->
<xsl:variable name="roleValue">
  <xsl:value-of select="msxsl:node-set($xElts)//*[ local-name(.) = local-name($item) and namespace-uri(.) = namespace-
uri($item)]//bcb:labelSet/bcb:label[@xml:lang='br']/@xlink:role" />
</xsl:variable>
<!-- Variável que recupera o valor do atributo xbrlink:weight do elemento Calculation, para formatar a saída de valor
negativo -->
<xsl:variable name="weightValue">
  <xsl:value-of select="msxsl:node-set($xElts)//*[ local-name(.) = local-name($item) and namespace-uri(.) = namespace-
uri($item)]//bcb:calcSet/bcb:weight" />
</xsl:variable>
<!-- Variável que recupera os elementos de Referência -->
<xsl:variable name="referenceValue">
  <xsl:value-of select="msxsl:node-set($xElts)//*[ local-name(.) = local-name($item) and namespace-uri(.) = namespace-
uri($item)]//bcb:reportSet/bcb:reference" />
</xsl:variable>
<!-- Saída da informação em uma tabela HTML, com labels, valores, referências, etc.
Este trecho do código usa as variáveis acima para formatar a saída -->
<!-- Testa se a posição do item é a primeira para inserir dados do início do relatório -->
<xsl:if test="position()=1">

```

```

<TITLE>Balanco Patrimonial - Banco Central do Brasil em <xsl:value-of select="$periodValue"/></TITLE>
<table Width="95%" Border="1" cellspacing="0">
  <tr><td Align="left"><H5><xsl:value-of select="$referenceValue"/></H5></td></tr>
</table>
<table Width="500" Border="0" cellspacing="0" Align="center">&#160;
  <tr><td Align="Center"><H3><xsl:value-of select="$labelValue"/> - EM <xsl:value-of select="$periodValue"
/></H3></td></tr>
  <tr><td Align="Right">Em R$ mil</td></tr>
</table>
</xsl:if>
<!-- Testa se a posição do item é diferente da primeira para inserir a tabela -->
<xsl:if test="position()!<=1">
<xsl:choose>
<!-- executa o código se o item é numérico -->
<xsl:when test="format-number($item, '#') != 'NaN'">
  <table Border="1" cellspacing="0" Align="center">
    <tr>
      <xsl:choose>
        <xsl:when test="$roleValue='http://www.xbrl.org/linkprops/label/total'">
          <xsl:choose>
            <!-- define a cor da célula do elemento Ativo Total e Passivo Total-->
            <xsl:when test="$labelValue='ATIVO TOTAL'">
              <Td Width="400" bgcolor="#E7A50F"><font face="Arial" Size="2"><B><div Align="Left"><xsl:value-of
select="$labelValue"/></div></B></font></Td>
              <Td Width="20%" bgcolor="#E7A50F"><font face="Arial" Size="2"><B><div Align="Right">
<xsl:value-of select="format-number($item, '#.###')"/></div></B></font></Td>
            </xsl:when>
            <xsl:when test="$labelValue='PASSIVO TOTAL'">
              <Td Width="400" bgcolor="#E7A50F"><font face="Arial" Size="2"><B><div Align="Left"><xsl:value-of
select="$labelValue"/></div></B></font></Td>
              <Td Width="20%" bgcolor="#E7A50F"><font face="Arial" Size="2"><B><div Align="Right">
<xsl:value-of select="format-number($item, '#.###')"/></div></B></font></Td>
            </xsl:when>
            <!-- Define a cor das células dos outros elementos, cujo valor do atributo role seja 'total' -->
            <xsl:otherwise>
              <Td Width="400" bgcolor="#E7A27F"><font face="Arial" Size="2"><B><div Align="Left"><xsl:value-of
select="$labelValue"/></div></B></font></Td>
              <xsl:choose>
                <xsl:when test="$item<0">
                  <Td Width="20%" bgcolor="#E7A27F"><font color="RED" face="Arial" Size="2"><B><div Align="Right">
<xsl:value-of select="format-number(substring-after($item, '-'), '#.###')"/>
</div></B></font></Td>
                </xsl:when>
                <xsl:otherwise>
                  <Td Width="20%" bgcolor="#E7A27F"><font face="Arial" Size="2"><B><div Align="Right">
<xsl:value-of select="format-number($item, '#.###')"/> </div></B></font></Td>
                </xsl:otherwise>
              </xsl:choose>
            </xsl:otherwise>
          </xsl:choose>
        </xsl:when>
        <!-- Define a cor das células dos outros elementos, cujo valor do atributo role seja 'standard' -->
        <xsl:otherwise>
          <Td Width="400" bgcolor="#F6D8C4"><font face="Arial" Size="2"><B><div Align="Left"><xsl:value-
of
select="$labelValue"/></div></B></font></Td>
        <!-- Define a cor do texto dos elementos cujo valor seja usado para subtrair o valor do elemento pai -->
        <xsl:choose>
          <xsl:when test="$weightValue='-1'">
            <Td Width="20%" bgcolor="#F6D8C4"><font color="RED" face="Arial" Size="2"><B><div
Align="Right">
<xsl:value-of select="format-number($item, '#.###')"/>
</div></B></font></Td>
          </xsl:when>
          <!-- Define a cor do texto dos outros elementos -->
          <xsl:otherwise>
            <Td Width="20%" bgcolor="#F6D8C4"><font face="Arial" Size="2"><B><div Align="Right">
<xsl:value-of select="format-number($item, '#.###')"/>

```

```

        </div></B></font></Td>
    </xsl:otherwise>
    </xsl:choose>
</xsl:otherwise>
</xsl:choose>
</tr>
</table>
</xsl:when>
<!-- executa o código se o item for não numérico -->
<xsl:otherwise>
    <xsl:choose>
    <xsl:when test="$labelValue='Diretoria Colegiada'">
        <p/>
        <font face="Arial" Size="3"><B><div Align="left">
        <xsl:value-of select="$labelValue"/></div></B></font>
        <hr/>
    </xsl:when>
    <xsl:when test="$labelValue='Presidente'">
        <font face="Arial" Size="3"><B><div Align="left"><xsl:value-of select="$labelValue"/> :
        <xsl:value-of select="$item"/></div></B></font>
        <hr/>Diretores :
    </xsl:when>
    <xsl:when test="$labelValue='Chefe do Departamento de Administração Financeira'">
        <hr/>
        <font face="Arial" Size="1"><B><div Align="left"><xsl:value-of select="$labelValue"/> :
        <xsl:value-of select="$item"/></div></B></font>
    </xsl:when>
    <xsl:when test="$labelValue='Contador'">
        <font face="Arial" Size="1"><B><div Align="left"><xsl:value-of select="$labelValue"/> -
        <xsl:value-of select="$item"/></div></B></font>
    </xsl:when>
    <xsl:otherwise>
        <font face="Arial" Size="2"><B><div Align="Left"><xsl:value-of select="$item"/></div></B></font>
    </xsl:otherwise>
    </xsl:choose>
</xsl:otherwise>
</xsl:choose>
</xsl:if>
</xsl:template>
<!-- Este template chama outro template, loadSingleTaxonomy, e diz para iniciar com cada porção do atributo
@xsi:schemaLocation, dividindo seu caminho para encontrar cada taxonomia que necessita ser carregada. -->
<xsl:template name="loadTaxonomies">
    <xsl:variable name="schemaLoc" select="/xbrl:group/@xsi:schemaLocation"/>
    <xsl:call-template name="loadSingleTaxonomy">
        <xsl:with-param name="schemaLocPart" select="$schemaLoc" />
        <xsl:with-param name="stringIndex" select="1" />
    </xsl:call-template>
</xsl:template>
<!-- Analisa o valor do atributo schemaLocation e determina a taxonomia a ser carregada, o arquivo correspondente à
segunda string. -->
<xsl:template name="loadSingleTaxonomy">
    <xsl:param name="schemaLocPart" />
    <xsl:param name="stringIndex" />
    <xsl:choose>
    <!-- Se o resto da divisão do índice, cujo valor atribuído inicialmente como 1, for 1, seleciona a segunda parte da string (após
o espaço em branco) -->
    <xsl:when test="$stringIndex mod 2 = 1">
        <xsl:variable name="fileNameStart" select="substring-after($schemaLocPart, ' ')" />
        <xsl:variable name="fileName">
            <xsl:choose>
            <xsl:when test="contains($fileNameStart, ' ')">
                <xsl:value-of select="substring-before($fileNameStart, ' ')" />
            </xsl:when>
            <xsl:otherwise>
                <xsl:value-of select="$fileNameStart" />
            </xsl:otherwise>
            </xsl:choose>
        </xsl:variable>
    </xsl:when>
    <xsl:otherwise>
        <xsl:variable name="fileNameStart" select="substring-after($schemaLocPart, ' ')" />
        <xsl:variable name="fileName">
            <xsl:choose>
            <xsl:when test="contains($fileNameStart, ' ')">
                <xsl:value-of select="substring-before($fileNameStart, ' ')" />
            </xsl:when>
            <xsl:otherwise>
                <xsl:value-of select="$fileNameStart" />
            </xsl:otherwise>
            </xsl:choose>
        </xsl:variable>
    </xsl:otherwise>
    </xsl:choose>

```



```
</xsl:variable>
<!-- Carrega o documento de taxonomia e aplica os templates. O resultado é uma árvore com as informações da taxonomia
-->
<xsl:variable name="taxonomyDocument" select="document($fileName)"/>
<xsl:apply-templates select="$taxonomyDocument" mode="taxonomy"/>
</xsl:when>
</xsl:choose>
<!-- Passa o resto da string, se existir, e retorna para a função com novo índice para encontrar outros arquivos de taxonomia
-->
<xsl:if test="string($schemaLocPart)" >
<xsl:call-template name="loadSingleTaxonomy" >
<xsl:with-param name="schemaLocPart" select="substring-after($schemaLocPart, ' ')" />
<xsl:with-param name="stringIndex" select="$stringIndex + 1" />
</xsl:call-template>
</xsl:if>
</xsl:template>
</xsl:stylesheet>
```

Apêndice I – Folha de estilo - arquivo bcb_TxCompress.xsl

<!--

Essa folha de estilo foi escrita baseada no código XSLT desenvolvido por Don Bruey, Creative Solutions (January 2002, dbruey@creativesolutions.com), que quando aplicado a um documento esquema de taxonomia XBRL, versão 2.0, produz uma árvore resultante com várias informações sobre a taxonomia e seus linkbases. Assim como no arquivo Instview.xsl, modificações foram feitas para se utilizar os linkbases calculation, reference e label. Os outros tipos de linkbase (presentation e definition) não foram aplicados neste exemplo, porém o código original que permite usá-los foi mantido.

As informações contidas nos linkbases são encontradas por meio dos elementos linkbaseRef, que informam onde os linkbases estão localizados.

Este arquivo deve ser usado em conjunto com o InstView.xsl, já que quando utilizado sozinho não produz nenhuma saída com informações úteis. Algumas características de XBRL, como o atributo priority, não são abordadas neste código, bem como não foi considerado seu desempenho quando utilizado com grandes taxonomias e arquivos de documentos.

O processador XSLT utilizado foi o MSXML 4, se este código for testado em outro processador, a função msxml node-set() deverá ser substituída por outra apropriada. Esta é a única função XSLT usada que não é padronizada.

Os templates abaixo são divididos em diferentes tipos de categorias, em função dos links estendidos usados por XBRL.

O prefixo de namespace "tempLink" é usado onde o código define seus próprios elementos e atributos.

-->

```
<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:xbrllink="http://www.xbrl.org/2001/XLink/xbrllinkbase"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xmlns:tempLink="http://www.creativesolutions.com/XBRL/linkbase"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:msxsl="urn:schemas-microsoft-com:xslt"
  xmlns:bcb="www.bcb.gov.br">
  <xsl:output method="xml" indent="yes"/>
  <!-- Esse template é chamado pelo InstView.xsl e constrói a árvore resultante da taxonomia -->
  <xsl:template match="/" mode="taxonomy">
    <xsl:variable name="extLinks">
      <xsl:apply-templates select="*" />
    </xsl:variable>
    <xsl:element name="bcb:TaxonomyElements">
      <xsl:for-each select="//xsd:element">
        <xsl:call-template name="schemaOutput_XML">
          <xsl:with-param name="schemaElt" select="." />
          <xsl:with-param name="extendedLinks" select="$extLinks" />
        </xsl:call-template>
      </xsl:for-each>
    </xsl:element>
  </xsl:template>
  <xsl:template match="/">
    <xsl:apply-templates select="/" mode="taxonomy" />
  </xsl:template>
  <!-- Este template recebe dois parâmetros, um com o elemento schema que está sendo processado e outro com uma lista de nós das informações encontradas nos linkbases. A saída são elementos XML com a informação desejada -->
  <xsl:template name="schemaOutput_XML">
    <xsl:param name="schemaElt" />
    <xsl:param name="extendedLinks" />
    <xsl:variable name="namespaceURI" select="*/@targetNamespace[1]" />
    <xsl:element name="{ $schemaElt/@name }" namespace="{ $namespaceURI }">
      <xsl:attribute name="xsd:type"><xsl:value-of select="$schemaElt/@type"/></xsl:attribute>
      <xsl:attribute name="xsd:substitutionGroup"><xsl:value-of select="$schemaElt/@substitutionGroup"/></xsl:attribute>
    <!-- carrega as informações dos elementos contidas no labelLink -->
    <xsl:element name="bcb:labelSet">
      <xsl:for-each select="msxsl:node-set($extendedLinks)//tempLink:label[substring-after(@tempLink:from, '#') = $schemaElt/@name]">
        <!-- para cada elemento carrega as informações do seu label, usadas em InstView.xsl -->
        <xsl:element name="bcb:label">
          <xsl:attribute name="xml:lang"><xsl:value-of select="@xml:lang" /></xsl:attribute>
          <xsl:attribute name="xlink:role"><xsl:value-of select="@xlink:role" /></xsl:attribute>
        <!-- atribui o valor do elemento tempLink:label para o elemento bcb:label -->
        <xsl:value-of select="." />
      </xsl:element>
    </xsl:for-each>
  </xsl:element>
  <!-- Para cada elemento carrega as informações do calculationLink, usadas em InstView.xsl -->
```

```

<xsl:element name="bcb:calcSet">
  <xsl:element name="bcb:weight">
    <xsl:for-each select="msxsl:node-set($extendedLinks)//tempLink:calculationArc[substring-after(@tempLink:from, '#')
= ($schemaElt/@name)]">
      <xsl:value-of select="@xbrllink:weight"/>
    </xsl:for-each>
  </xsl:element>
</xsl:element>
<!-- carrega as informações do referenceLink, usadas em InstView.xml -->
<xsl:element name="bcb:reportSet">
  <xsl:element name="bcb:reference">
    <xsl:for-each select="msxsl:node-set($extendedLinks)//tempLink:reference[substring-after(@tempLink:from, '#') =
      $schemaElt/@name and @xlink:arcrole = 'http://www.xbrl.org/linkprops/arc/element-reference']">
      <!-- atribui o valor do elemento tempLink:reference para o elemento bcb:reference -->
      <xsl:copy-of select="."/>
    </xsl:for-each>
  </xsl:element>
</xsl:element>
</xsl:template>
<!-- Este template encontra os elementos linkbaseRef, cria uma variável de saída cujo conteúdo são os linkbases e então
aplica os templates às variáveis. Os elementos de saída, que são descritos nos templates abaixo, serão usados pelo template
raiz deste código para uso na saída da tabela HTML.-->
<xsl:template match="xbrllink:linkbaseRef">
  <!-- Carrega o linkbase usando a função document() -->
  <xsl:variable name="lbase" select="document(@xlink:href)" />
  <!-- aplica os templates ao conteúdo do linkbase -->
  <xsl:apply-templates select="$lbase" mode="linkbase"/>
</xsl:template>
<!-- Este é o template raiz usando o atributo mode com o valor linkbase. Este template do elemento raiz é chamado pelo
template do elemento xbrllink:linkbaseRef, nunca por default, já que tem um atributo mode indicando que apenas o template
linkbaseRef pode invocá-lo -->
<xsl:template match="/" mode="linkbase">
  <xsl:variable name="linkbaseResults">
    <!-- Aqui os templates baseados no atributo mode com o valor resource são chamados, ou seja, elementos cdo tipo
resource são processados e seus arcos os utilizam são encontrados. Labels e references são os dois tipos de linkbases que têm
templates usando este modo -->
    <xsl:apply-templates select="*" mode="resource-based"/>
    <!-- Aqui os templates baseados no atributo mode com o valor locator são chamados, ou seja, elementos cdo tipo resource
são processados e seus arcos os utilizam são encontrados. Nesta transformação apenas o link estendido Calculation faz este
processamento -->
    <xsl:apply-templates select="*" mode="arc-based"/>
  </xsl:variable>
  <!-- copia o resultado da transformação para a saída -->
  <xsl:copy-of select="$linkbaseResults" />
</xsl:template>
<!-- os elementos calculationArc, definitionArc, and presentationArc são muito similares, a diferença entre eles é a existência
de um atributo próprio para presentationArc and calculationArc.
A transformação troca os atributos to/from do elemento arco pelo elemento especificado em xlink:href do elemento locator,
cujo atributo xlink:label seja igual aos originais to/from. Os outros atributos são retirados diretamente do elemento arco.
A saída deste template deve ser semelhante a esta:
calculationArc:
<tempLink:calculationArc
  tempLink:from="bcb_taxonomia.xsd#bcb_ativo"
  tempLink:to="bcb_taxonomia.xsd#bcb_bp"
  xlink:show="replace"
  xlink:actuate="onRequest"
  xbrllink:weight="1"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"
  tempLink:extRole="http://www.xbrl.org/linkprops/extended/BalançoPatrimonial"/>
presentationArc:
<tempLink:presentationArc
  tempLink:from="bcb_taxonomia.xsd#bcb_ativo"
  tempLink:to="bcb_taxonomia.xsd#bcb_bp"
  xlink:show="replace"
  xlink:actuate="onRequest"
  xbrllink:order="1"

```

```

  xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"
  tempLink:extRole="http://www.xbrl.org/linkprops/extended/BalançoPatrimonial"/>
definitionArc:
<tempLink:definitionArc
  tempLink:from="bcb_taxonomia.xsd#bcb_ativo"
  tempLink:to="bcb_taxonomia.xsd#bcb_bp"
  xlink:show="replace"
  xlink:actuate="onRequest"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/child-parent"
  tempLink:extRole="http://www.xbrl.org/linkprops/extended/BalançoPatrimonial"/>

```

onde, para todos os três tipos de arcos, tempLink:from - o valor de href no locator referenciado pelo atributo xlink:from de seu elemento arco tempLink:to - o valor de href no locator referenciado pelo atributo xlink:to de seu elemento arco

```

  xlink:show - copiado diretamente do elemento tipo arco
  xlink:actuate - copiado diretamente do elemento tipo arco
  xbrllink:order - apenas para presentationArc - copiado diretamente do elemento tipo arco
  xbrllink:weight - apenas para calculationArc - copiado diretamente do elemento tipo arco
  xlink:arcrole - copiado diretamente do elemento tipo arco
  tempLink:extRole - copiado do atributo xlink:role do nó de contexto do elemento pai de link estendido.

```

-->

```

<xsl:template
  match="xbrllink:calculationArc | xbrllink:presentationArc | xbrllink:definitionArc"
  mode="arc-based">
  <!-- Variáveis que armazenam informações para uso posterior -->
  <xsl:variable name="fromElt" select="@xlink:from" />
  <xsl:variable name="toElt" select="@xlink:to" />
  <xsl:variable name="resultFrom" select="..xbrllink:loc[ @xlink:label=$fromElt]/@xlink:href" />
  <xsl:variable name="resultTo" select="..xbrllink:loc[ @xlink:label=$toElt]/@xlink:href" />
  <xsl:variable name="resultElementName" select="concat('tempLink:', local-name(.))" />
  <!-- Escreve um elemento para saída com o nome apropriado -->
  <xsl:element name="{ $resultElementName }">
    <xsl:attribute name="tempLink:from"><xsl:value-of select="$resultFrom" /></xsl:attribute>
    <xsl:attribute name="tempLink:to"><xsl:value-of select="$resultTo" /></xsl:attribute>
    <xsl:attribute name="xlink:show"><xsl:value-of select="@xlink:show" /></xsl:attribute>
    <xsl:attribute name="xlink:actuate"><xsl:value-of select="@xlink:actuate" /></xsl:attribute>
    <!--
    elementos calculationArc copiam o atributo weight
    elementos presentationArc copiam o atributo order
    elementos definitionArc não têm qualquer atributo especial
    -->
    <xsl:choose>
      <xsl:when test="local-name(.) = 'calculationArc'">
        <xsl:attribute name="xbrllink:weight"><xsl:value-of select="@weight" /></xsl:attribute>
      </xsl:when>
      <xsl:when test="local-name(.) = 'presentationArc'">
        <xsl:attribute name="xbrllink:order"><xsl:value-of select="@order" /></xsl:attribute>
      </xsl:when>
    </xsl:choose>
    <xsl:attribute name="xlink:arcrole"><xsl:value-of select="@xlink:arcrole" /></xsl:attribute>
    <xsl:attribute name="tempLink:extRole"><xsl:value-of select="..@xlink:role[1]" />
  </xsl:attribute>
  </xsl:element>
</xsl:template>

```

<!-- elementos xbrllink:reference são diferentes dos outros elementos linkbase, pois podem conter textos e outros elementos (exceto para os elementos labels que, também, podem conter textos)

A saída deste template deve ser semelhante a esta:

```

<tempLink:reference xlink:href="bcb_taxonomia.xsd#bp"
  xlink:show="embed" xlink:actuate="onRequest"
  xlink:arcrole="http://www.xbrl.org/linkprops/arc/element-reference">
  <bcbr:referencia> Publicação em acordo com a Lei ....</bcbr:referencia>
</tempLink:reference>

```

-->

```

<xsl:template match="xbrllink:reference" mode="resource-based">
  <xsl:variable name="xlinkLabel" select="@xlink:label" />
  <xsl:variable name="partElement" select="text()" />
  <xsl:for-each select="..xbrllink:referenceArc[ @xlink:to = $xlinkLabel]">
    <xsl:variable name="fromElt" select="@xlink:from" />

```

```

<xsl:variable name="elementURI" select="../xbrl:loc[@xlink:label=$fromElt]/@xlink:href" />
<xsl:if test="$elementURI != """>
  <xsl:element name="tempLink:reference">
    <xsl:attribute name="tempLink:from"><xsl:value-of select="$elementURI" />
  </xsl:attribute>
  <xsl:attribute name="xlink:show"><xsl:value-of select="@xlink:show" /></xsl:attribute>
  <xsl:attribute name="xlink:actuate"><xsl:value-of select="@xlink:actuate" />
</xsl:attribute>
  <xsl:attribute name="xlink:arcrole"><xsl:value-of select="@xlink:arcrole" />
</xsl:attribute>
  <xsl:attribute name="tempLink:extRole"><xsl:value-of select="../@xlink:role[1]" />
</xsl:attribute>
<xsl:copy-of select="../xbrl:reference[@xlink:label = $xlinkLabel]/*" />
  </xsl:element>
</xsl:if>
</xsl:for-each>
</xsl:template>
<!-- templat default para prevenir que nós textos tornem-se saída quando os template acima não forem executados -->
<xsl:template match="text()" />
<!-- elemento xbrl:label elements com processamento baseado no tipo resource
  O melhor método para lidar com labels é iniciar com um elemento label resource e encontrar cada um de seus elementos
  irmãos do tipo arc, que tenham o atributo "xlink:to" com o mesmo valor do atributo "xlink:label" do elemento label. Depois
  de encontrado cada arco, é alcançado o elemento locator, o qual informa o ponto inicial do arco que aponta para este label
  (ex:taxonomia.xsd#elemento) e criado um elemento label de saída com todas as informações desejadas.
  Este código não funciona quando um labelArc aponta de um elemento locator para outro. Este template trabalha apenas com
  links de sentido único, de um elemento para um label.
  A saída deste template deve ser semelhante a esta:
  <tempLink:label tempLink:from="bcb_taxonomia.xsd#ativo"
    tempLink:role="http://www.xbrl.org/linkprops/label/standard"
    xml:lang="br"
    tempLink:extRole="http://www.xbrl.org/linkprops/extended/BalançoPatrimonial">
  Ativo Total
  </tempLink:label>
  onde
    tempLink:from é o atributo "from" de cada arco que está apontando para o label
    tempLink:role é o mesmo atributo role do elemento de tipo arc
    xml:lang é o atributo padrão xml, copiado do elemento resource
    tempLink:extRole é tirado do elemento pai labelLink
  -->
  <xsl:template match="xbrl:label" mode="resource-based">
    <xsl:variable name="xlinkLabel" select="@xlink:label" />
    <xsl:variable name="xlinkRole" select="@xlink:role" />
    <xsl:variable name="xmlLang" select="@xml:lang" />
    <xsl:variable name="labelText" select="text()" />
    <xsl:for-each select="../xbrl:labelArc[@xlink:to = $xlinkLabel]">
      <xsl:variable name="fromElt" select="@xlink:from" />
      <xsl:variable name="elementURI" select="../xbrl:loc[@xlink:label=$fromElt]/@xlink:href" />
      <xsl:if test="$elementURI != """>
        <xsl:element name="tempLink:label">
          <xsl:attribute name="tempLink:from"><xsl:value-of select="$elementURI" />
        </xsl:attribute>
          <xsl:attribute name="xlink:role"><xsl:value-of select="$xlinkRole" /></xsl:attribute>
          <xsl:attribute name="xml:lang" ><xsl:value-of select="$xmlLang" /></xsl:attribute>
          <xsl:attribute name="tempLink:extRole"><xsl:value-of select="../@xlink:role[1]" />
        </xsl:attribute>
          <xsl:value-of select="$labelText" />
        </xsl:element>
      </xsl:if>
    </xsl:for-each>
  </xsl:template>
</xsl:stylesheet>

```

Apêndice J – Códigos Java e JSP das aplicações

Classe LendoXML

```

// package DomJava;
import javax.xml.parsers.*;
import org.w3c.dom.*;
import java.io.File;
public class LendoXML {
    private static AtualizaTabela atualiza;
    // -----
    public static void main (String args[])
        throws Exception
    {
        java.io.BufferedReader reader = new java.io.BufferedReader(new java.io.InputStreamReader(System.in));
        System.out.print(" Digite o nome do arquivo XML: ");
        String nome = reader.readLine();
        File docFile = new File( nome );
        Document doc = null;
        // Parse the file
        try {
            DocumentBuilderFactory dbf = DocumentBuilderFactory.newInstance();
            DocumentBuilder db = dbf.newDocumentBuilder();
            doc = db.parse(docFile);
        }
        catch (Exception e) {
            System.out.print("Problem parsing the file [" + nome + "]\n" );
        }
        Element root = doc.getDocumentElement();
        LendoXML.atualiza = new AtualizaTabela("BP");
        stepThrough(root);
        LendoXML.atualiza.atualizar();
    }
    private static void lerNumericContext(Element numericContext)
    {
        for (Node child = numericContext.getFirstChild();
            child != null;
            child = child.getNextSibling())
        {
            if (child.getNodeType() == child.ELEMENT_NODE)
            {
                for (Node filho = child.getFirstChild();
                    filho != null;
                    filho = filho.getNextSibling())
                {
                    if (filho.getNodeName().equals("identifier") || filho.getNodeName().equals("instant") ||
                        filho.getNodeName().equals("measure") )
                    {
                        String nomeDaTag = filho.getNodeName();
                        String __str = new String(getTextContents(filho));
                        LendoXML.atualiza.adicionarCampo(nomeDaTag, __str);
                    }
                }
                if (child.getNodeName().equals("scenario"))
                {
                    String nomeDaTag = child.getNodeName();
                    // pegar o valor do atributo 'name' e armazena no campo do banco de dados de nome 'scenario'
                    NamedNodeMap atributos = child.getAttributes();
                    for (int i = 0; i < atributos.getLength(); i++)
                    {
                        Node atr = atributos.item(i);
                        if (atr.getNodeName().equals("name"))
                        {
                            String __str = new String(atr.getNodeValue());
                            LendoXML.atualiza.adicionarCampo(nomeDaTag, __str);
                        }
                    }
                }
            }
        }
    }
    // pegar o valor do filho 'scenarioType' e armazena

```



```

        lerNumericContext(start);
    }
    else
    if (nomeDaTag.equals("bcb:bp"))
    {
        lerBP(start);
    }
    else
    for (Node child = start.getFirstChild();
        child != null;
        child = child.getNextSibling())
    {
        if (child.getNodeType() == child.ELEMENT_NODE)
            stepThrough((Element)child);
    }
}
/**
 * Returns the contents of all immediate child text nodes, can strip whitespace
 * <p>
 * Takes a node as input and merges all its immediate text nodes into a
 * string. If the strip whitespace flag is set, whitespace at the beginning
 * and end of each merged text node will be removed
 *
 * @param node          node to extract text contents of
 * @param b_strip_whitespace  flag to set whitespace removal
 * @return              string containing text contents of the node
 */
public static String getTextContents( Node node )
{
    NodeList childNodes;
    StringBuffer contents = new StringBuffer();
    childNodes = node.getChildNodes();
    for(int i=0; i < childNodes.getLength(); i++ )
    {
        if( childNodes.item(i).getNodeType() == Node.TEXT_NODE )
        {
            contents.append(childNodes.item(i).getNodeValue());
        }
    }
    return contents.toString();
}
}

```

Classe AtualizaTabela

```

// package DomJava;
import java.sql.*;
public class AtualizaTabela
{
    String campos;
    String valores;
    String nomeDaTabela;
    public AtualizaTabela(String nomeDaTabela)
    {
        this.campos = this.valores = null;
        this.nomeDaTabela = nomeDaTabela;
    }
    public void adicionarCampo(String campo, Object valorDoCampo)
    {
        if (this.campos == null)
        {
            this.campos = ("insert into " + this.nomeDaTabela + " ( " + campo);
            this.valores = (" values ( " + paraSQL(valorDoCampo) );
        }
        else
        {
            this.campos += (" , " + campo);
            this.valores += (" , " + paraSQL(valorDoCampo));
        }
    }
}

```



```

    }
}
private String paraSQL(Object obj)
{
    if (obj instanceof String)
        return "" + obj.toString() + "";
    else
        return obj.toString();
}
public int atualizar()
throws SQLException, ClassNotFoundException
{
    String url = "jdbc:odbc:DemonstrativosContabeis";
    String username = "";
    String password = "";
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    Connection conn = DriverManager.getConnection(url, username, password);
    Statement st = conn.createStatement();
    String sql = this.campos + ")" + this.valores + ")";
    int linhas = st.executeUpdate(sql);
    conn.commit();
    st.close();
    if(linhas==1)
        System.out.println("Tabela Atualizada");
    else
        System.out.println("Problemas na Atualização da Tabela");
    return linhas;
}
}

```

Classe LendoXML

```

import java.sql.*;
public class AcessaBancoDados
{
    String nomeDaTabela;
    String campo;
    String url = "jdbc:odbc:DemonstrativosContabeis";
    String username = "";
    String password = "";
    private ResultSet rs;
    public AcessaBancoDados(String nomeDaTabela)
    throws ClassNotFoundException, SQLException
    {
        this.campo = null;
        this.nomeDaTabela = nomeDaTabela;
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection conn = DriverManager.getConnection(url, username, password);
        Statement st = conn.createStatement();
        String sql = ("select * from " + this.nomeDaTabela + " order by instant" );
        this.rs = st.executeQuery(sql);
    }
    public String recuperarCampo(String campo)
    throws SQLException, java.io.IOException
    {
        return rs.getString(campo);
    }
    public boolean proximaLinha()
    throws SQLException
    {
        return this.rs.next();
    }
}

```

Arquivo mostraGrafico.JSP

```

<% @page import="java.util.Date"% >
<html>
<body BACKGROUND='fundo_moeda.gif' BGPROPERTIES=FIXED TEXT=#003366>

```

```

<table border='0' align='left'>
<tr>
<h3>Evolução do Ativo </h3>
<%
  AcessaBancoDados acessaBancoDados = new AcessaBancoDados("BP");
  String ativo, valorAtivo, data;
  int resultAtivo;
  while (acessaBancoDados.proximaLinha())
  {
    ativo = acessaBancoDados.recuperarCampo("ativo");
    valorAtivo = "R$ " + ativo.substring(0,3) + "." + ativo.substring(3,6) + "." + ativo.substring(6,9) ;
    resultAtivo = Integer.parseInt(ativo) / 1000000;
    data = acessaBancoDados.recuperarCampo("instant");
  }
  %>
  <td valign='bottom'><%= valorAtivo %><br>'><br><%= data %></td>
<%
}
%>
</tr>
</table>
</body>
</html>

```

Arquivo mostraGraficoativo.JSP

```

<% @page import="java.util.Date"%>
<html>
<body BACKGROUND='fundo_moeda.gif' BGPROPERTIES=FIXED TEXT=#003366>
<table border='0' align='left'>
<tr>
<h3>Evolução do Ativo</h3>
<h5> (R$ milhões) </h1>
<%
  AcessaBancoDados acessaBancoDados = new AcessaBancoDados("BP");
  String ativo, data;
  int resultAtivo, valorAtivo;
  while (acessaBancoDados.proximaLinha())
  {
    ativo = acessaBancoDados.recuperarCampo("ativo");
    valorAtivo = Integer.parseInt(ativo) / 1000000;
    resultAtivo = Integer.parseInt(ativo) / 1000000;
    data = acessaBancoDados.recuperarCampo("instant");
  }
  %>
  <td valign='bottom'><%= valorAtivo %><br>'><br><h6><%= data %></h6></td>
<%
}
%>
</tr>
</table>
</body>
</html>

```

Arquivo mostraGraficoImobilizacaoPL.JSP

```

<% @page import="java.util.*"%>
<% @page import="java.text.DecimalFormat"%>
<html>
<body BACKGROUND='fundo_moeda.gif' BGPROPERTIES=FIXED TEXT=#003366>
<table border='0' align='left'>
<tr>
<h4>Imobilização do Patrimônio Líquido</h4>
<%
  AcessaBancoDados acessaBancoDados = new AcessaBancoDados("BP");
  DecimalFormat doisDigitos = new DecimalFormat("0.00");
  String data;
  double ativo_permanente, patrimonio_liquido, imobilizacao_pl;

```

```

while (acessaBancoDados.proximaLinha())
{
    ativo_permanente = (double)Integer.parseInt(acessaBancoDados.recuperarCampo("atv_prmt"));
    patrimonio_liquido = (double)Integer.parseInt(acessaBancoDados.recuperarCampo("pssv_pl"));
    imobilizacao_pl = ativo_permanente / patrimonio_liquido * 100.;
    data = acessarBancoDados.recuperarCampo("instant");
%>
    <td valign='bottom'><%= doisDigitos.format(imobilizacao_pl) %> %> <br><img src='pixel_azul.gif' width='70'
height='<%= imobilizacao_pl/2 %>'><br><h6><%= data %></h6></td>
<%
}
%>
</tr>
</table>
</body>
</html>

```

Arquivo mostraGraficoLiquidez.JSP

```

<% @page import="java.util.*"%>
<% @page import="java.text.DecimalFormat"%>
<html>
<body BACKGROUND='fundo_moeda.gif' BGGPROPERTIES=FIXED TEXT=#003366>
<table border='0' align='left'>
<tr>
<h4>Liquidez Geral</h4>
<%
    AcessoBancoDados acessoBancoDados = new AcessoBancoDados("BP");
    DecimalFormat doisDigitos = new DecimalFormat("0.00");
    String data;
    double ativo_circulante_lp, passivo_circulante_lp, liquidez_geral, valorParaGrafico;
    while (acessaBancoDados.proximaLinha())
    {
        ativo_circulante_lp = (double)Integer.parseInt(acessaBancoDados.recuperarCampo("atv_circ_real_lprazo"));
        passivo_circulante_lp = (double)Integer.parseInt(acessaBancoDados.recuperarCampo("pssv_circ_exg_lprazo"));
        liquidez_geral = (ativo_circulante_lp / passivo_circulante_lp) * 100.;
        data = acessarBancoDados.recuperarCampo("instant");
%>
        <td valign='bottom'><%= doisDigitos.format(liquidez_geral) %> %> <br><img src='pixel_verde.gif' width='70'
height='<%= liquidez_geral/2 %>'><br><h6><%= data %></h6></td>
<%
}
%>
</tr>
</table>
</body>
</html>

```

Apêndice L – Resumo dos elementos e atributos XBRL [40]

XBRL V2.0 - Elementos do documento XBRL

Elementos	Atributos	Valores	Notas/Links
item - definido pelo usuário			Cada item é definido em um documento de taxonomia, suportando zero ou mais linkbases. Um item terá um nome que descreve o valor de um fato financeiro que este irá ter em um documento XBRL. Cada item em um documento pode ter um atributo chamado numericContext or nonNumericContext, os quais referem-se a um outro elemento contendo informações como um período, unidades de medida e outras informações que esclareça o significado do item.
	NumericContext	IDREF	O valor para o atributo deve ser igual ao elemento numericContext no documento XBRL - a informação do contexto completa a informação do item. Usado para itens que têm um tipo numérico.
	nonNumericContext	IDREF	O valor para o atributo deve ser igual ao elemento numericContext no documento XBRL - a informação do contexto completa a informação do item. Usado para itens que têm um tipo não numérico.
Context (numericContext & nonNumericContext)			Partes do contexto dos elementos numericContext e nonNumericContext são divididas em subelementos e atributos: *atributo de identificação - id *elemento entity *elemento period *elemento scenario *elemento unit - requerido para numericContext, opcional para nonNumericContext. O elemento numericContext contém atributos adicionais que ajudam no processamento dos valores numéricos: *precision *cwa
	id	ID	O valor desse atributo corresponde ao do elemento numericContext ou nonNumericContext.
entity de numericContext e nonNumericContext			Elemento que contém informação sobre a entidade para a qual os valores usados nesse contexto são relatados.
identifier de Entity de numericContext e nonNumericContext		string	Uma string define o uso que contém informação sobre essa entidade, veja a descrição do atributo Scheme abaixo.
	Scheme	URI	URI para fornecer uma sugestão para o usuário do tipo de informação encontrada no documento. <identifier scheme=www.nasdaq.com>SAMP</identifier> refere-se a companhia com o ticker "SAMP" em NASDAQ. O valor de Scheme refere-se à URI do namespace onde o elemento é definido.
segment de numericContext e nonNumericContext		Opcional conteúdo misto	Elemento de recipiente genérico que contém informação adicional gerada pelo produtor do documento. Não há uma estrutura definida para esse elemento, podendo conter tantos marcadores quantos forem necessários para o entendimento do documento.
period de numericContext e nonNumericContext		vários conteúdos do período	Ambos os contextos, numeric e non-numeric, usam o subelemento period. Veja a especificação para possíveis valores.
scenario de numericContext e nonNumericContext		Opcional conteúdo misto	Ambos os contextos, numeric e non-numeric, usam o subelemento Scenario. Pode conter alguma string ou marcador adicional.
unit de numericContext e nonNumericContext		Unidade de medida de valores de itens usados	Exemplos de especificações: <unit> <measure>ISO4217:GBO</measure> </unit>

		no contexto	<pre><unit> <operator name="divide"> <measure>ISO4217:EUR</measure> <measure>xbri:shares</measure> </operator> </unit></pre>
numericContext			
	precision	inteiro	<p>Uma aplicação pode assumir se algarismos significativos são válidos. Por exemplo, com a precisão de "4" e um item no valor de 98,765.08, a aplicação pode somente assumir 98,760.00 durante o processamento.</p> <p>O valor da precisão não refere-se aos dígitos depois do separador decimal.</p>
	cwa	Dois valores: true e false	<p>Se true, a informação de toda quantidade especificada usada nesse contexto está completa, e outros valores, como total, podem ser calculados usando esta informação. Se falso, o leitor não pode fazer esses tipos de suposições sobre a integralidade da dado.</p>
Tuple			<p>Nenhum elemento do tipo "tuple" aparecerá em um documento XBRL. "Tuple" é um elemento abstrato no qual outros elementos são baseados. Caso o elemento seja "tuple" ele irá atuar como um recipiente para outros elementos ou tuples, com o qual este tem um relacionamento familiar.</p>
Group			<p>O elemento grup é um recipiente de elementos dentro do documento XBRL, é o elemento raiz de todo documento XBRL. Irá geralmente conter elementos, namespaces e informações sobre SchemaLocation para ajudar o leitor a localizar taxonomias que são requeridas por esse documento, para ajuda-lo a distinguir nomes de elementos queertencam a certos namespaces, como o namespace XBRL, namespace da taxonomia ou namespace Xlink.</p>
Footnotes			Veja sobre esse elemento na tabela de elementos XLink.
linkbaseRef			Veja sobre esse elemento na tabela elementos XLink.

XBRL V2.0 - Elementos Xlink

Elementos	Atributos	Valores	Notas/Links
linkbaseRef			Usado em taxonomias para informar ao usuário onde encontrar um linkbase XBRL. Pode também ser usado em um arquivo linkbase para direcionar para um outro arquivo linkbase, ou em um documento XBRL para direcionar o usuário para um linkbase no qual o conteúdo seja necessário para completar os fatos do documento XBRL.
	xlink:type	Simple	Sempre tem valor simple - links simple são links com um só caminho, neste caso, para o documento linkbase.
	xlink:href	URI que aponta para um arquivo linkbase	Exemplos: label_linkbase_em.xml http://www.my.com/linkbases/labels_em.xml
	xlink:role	.../label, .../reference, .../presentation, .../calculation ou .../definition.	Identifica o tipo de linkbase que está sendo especificado - o leitor da taxonomia pode escolher quais linkbases, se houver mais de um. Usando este valor de atributo, pode-se saber qual o tipo do linkbase está sendo referido sem a abertura e leitura desse.
	xlink:arcrole		Valor fixo definido pela especificação Xlink: www.w3.org/1999/xlink/properties/linkbase em http://www.w3.org/TR/2001/REC-xlink-20010627/#xlg
	xlink:actuate	OnRequest	Sempre tem valor onRequest. O usuário de uma taxonomia não precisa ter conhecimento de todos os linkbases, apenas os que são apropriados para o andamento do seu processo. Pelo o atributo xlink:role deste elemento, o usuário decide se é necessário carregar o linkbase.
calculationLink definitionLink presentationLink labelLink referenceLink			Estes são os cinco tipos de links estendidos definidos pelo XBRL. Seus nomes tem descrição própria assim como o tipo de informação que cada um possui.
	xlink:type	Extended	Sempre tem valor extended
	xlink:role		Pela criação de vários valores o autor da taxonomia tem a possibilidade de dividir um linkbase em seções para que o leitor possa limitar pedidos de informações somente daquela seção da taxonomia que é do seu interesse. Por exemplo, se o leitor está somente interessado na informação do balanço patrimonial, o programa pode limitar a leitura do linkbase para aqueles elementos cujo atributo role é ".../extended/BalancoPatrimonial" Note que não há um valor oficial da definição dessas seções na taxonomia ou linkbase, eles são proprietários de uma taxonomia e o autor da taxonomia pode criar o valor que lhe seja conveniente.

Locator			<p>Esses elementos são encontrados em linkbases reference e label. Eles existem para dar nomes aos elementos do arquivo de taxonomia, para uso interno dentro do linkbase. Estes nomes são então usados para definir relacionamentos entre elementos através de elementos arc. Nos exemplos de linkbases para elementos locator, observe os valores atribuídos à xlink:label, então veja como os elementos arc usam este valor para definir arcos.</p> <p>Em geral, elementos locator-type referem-se a elementos de um arquivo de taxonomia, mas eles poderiam também referir-se a um label externo.</p>
	xlink:type	Locator	Sempre tem valor locator
	xlink:href	Uma expressão Xpointer que aponta para um elemento no arquivo de esquema da taxonomia.	<p>Este atributo informa qual o elemento da taxonomia está sendo referenciado.</p> <p>Exemplo: <i>ias.xsd#assets.currentAssets</i> <i>ias.xsd#xpointer("//*[local-name(.)= 'element' and contains(@name, 'asset')]")</i></p> <p>Argumentos para funções xpointer em expressões xpointer são restritas em XBRL para conter apenas expressões Xpath válidas. Funções de extensão Xpointer não são permitidas. A URI pode apontar para um arquivo de taxonomia ou para um documento externo que contém um label ou informação literária.</p>
	xlink:label	NCName	Os valores do atributo xlink:label são usados em elementos arc para definir arcos entre elementos da taxonomia e elementos xlink:resource em um linkbase.
	xlink:role	Opcional: o valor definido XBRL é http://www.xbrl.org/linkprops/locator/root	O valor definido em XBRL identifica o elemento atual como o elemento "raiz", representando o conceito de nó em uma árvore, como "balanceSheet".
Elementos do tipo resource (label e reference)			Esse elemento contém os dados de um linkbase. Por exemplo, em um linkbase label o texto pode ser encontrado neste elemento. Outros elementos Xlink que são encontrados em um linkbase label existem para o suporte a Xlink, para criar arcos de um elemento a outro ou para fornecer suporte de nome para benefício de Xlink..
	xlink:type	resource	Sempre tem valor resource
	xlink:label	NCName	O valor desse atributo é usado pelos elementos arc para ligar os elementos locator. Por exemplo, observe os elementos resource. Siga os labels através de elementos arcs, que usam esses labels de elementos locator, que definem nomes locais para elementos da taxonomia.
	xlink:role	Opcional - dois valores definidos por XBRL são: http://www.xbrl.org/linkprops/label/standard http://www.xbrl.org/linkprops/label/total	Informação mais detalhada da descrição ou categorização de um conceito.
Elementos tipo arc			Esse elemento define o relacionamento "de-para" entre recursos. Use o atributo xlink:arcrole para determinar o significado dos atributos xlink:to e xlink:from para cada elemento arc.
	xlink:type	arc	Sempre tem valor arc.
	xlink:show	Um dos dois valores: embed e replace	Use embed quando ligar a um elemento de arquivo diferente (ex: definições label e elemento). Use replace ligando elementos do mesmo arquivo.
	xlink:actuate	onRequest	Sempre tem valor onRequest

	xlink:title	Opcional-string	Um valor opcional usado pelo Xlink, ou usado como uma descrição legível do significado do arco. Ele é simplesmente uma sugestão para o leitor acerca da razão do arc.
	xlink:to	NCName	Este é o destino do link.
	xlink:from	NCName	É o ponto inicial do link.
	xlink:arcrole	string que define a relação entre os atributos do elementos xlink:to e xlink:from	Para cálculoArc,definitionArc e presentationArc os seguintes valores são definidos por XBRL: http://www.xbrl.org/linkprops/arc/child-parent http://www.xbrl.org/linkprops/arc/parent-child http://www.xbrl.org/linkprops/arc/dimension-element http://www.xbrl.org/linkprops/arc/element-dimension Para elementos labelArc,os seguintes valores são definidos por XBRL: http://www.xbrl.org/linkprops/arc/label-element http://www.xbrl.org/linkprops/arc/element-label Para elementos referenceArc: http://www.xbrl.org/linkprops/arc/reference-element http://www.xbrl.org/linkprops/arc/element-reference http://www.xbrl.org/linkprops/arc/actual-element http://www.xbrl.org/linkprops/arc/element-actual
	use	Required, optional ou prohibited	Para um arco onde os elementos to/from já são acoplados em um link definido por uma outra parte, este atributo descreve uma modificação deste link. *optional é o valor default, isso significa que o arco pode ser atravessado por uma aplicação. * prohibited, o arco não pode ser atravessado. *required, ambos os elementos(to/from) devem existir para se ter um significado completo, ou seja, há uma dependência entre os elementos.
	priority	Opcional -integer	Define a prioridade da travessia de dois arcos equivalentes com conflitos no atributo xbrl:use (ex: um é proibido,o outro não).O arco com maior valor de prioridade toma a procedência. No caso dos atributos de prioridade serem iguais a aplicação decide qual arco é escolhido. O default é 0, então o arc que tiver valor de priority acima de 0 terá prioridade sobre o default.
calculationArc			São elementos arc com um atributo adicional.
	weigth	número decimal	Multiplicador que será aplicado no valor do elemento. Ex:"1.0" significa adicionar 100% do valor do elemento no elemento irmão para se calcular o valor do pai..
presentationArc			São elementos arc com um atributo adicional.
	order	número inteiro positivo	Ordem em que esse elemento é aplicado durante a apresentação (ex:"4" significa que esse é o 4º elemento a ser apresentado no documento).Se o valor da ordem de dois arcos são o mesmo, a aplicação decide qual usar primeiro
definitionArc			Veja elementos Arc acima.
labelArc			Veja elementos Arc acima.
referenceArc			Veja elementos Arc acima.
link:footnoteLink			
	xlink:type	extended	Sempre tem valor extended
link:loc			subelemento do footnoteLink. Explicado acima com nos elementos locator.
	xlink:type	locator	Sempre tem valor locator

	xlink:href	URI	Já explicado acima
	xlink:label	string	Já explicado acima
	xlink:role	Opcional - tem um valor XBRL definido: http://www.xbrl.org/linkprops/locator/root	Já explicado acima
link:footnote			subelemento de footnoteLink. Já explicado acima em elementos resource.
	xlink:type	Resource	Sempre tem valor resource
	xlink:label	string	Já explicado acima
	xlink:role	Opcional: http://www.xbrl.org/linkprops/label/standard http://www.xbrl.org/linkprops/label/total	Já explicado acima
link:footnoteArc			subelemento de footnoteLink- Já explicado acima em elementos arc.
	xlink:type	arc	Já explicado acima
	xlink:show	embed/replace	Já explicado acima
	xlink:actuate	onRequest	Sempre tem valor onRequest
	xlink:to	uso definido	Já explicado acima
	xlink:from	uso definido	Já explicado acima
	xlink:title	uso definido	Já explicado acima
	xlink:arcrole	http://www.xbrl.org/linkprops/arc/footnote http://www.xbrl.org/linkprops/arc/footnote-fact	Ve Já explicado acima

XBRL V2.0 – Elementos da Taxonomia

Elementos Atributos Valores Notas/Links

Elementos	Atributos	Valores	Notas/Links
element			"element" é o nome do principal elemento usado pelo XML Schema. Definir um tipo de elemento no XML Schema envolve declarar um elemento chamado "element" e adicionar de atributos apropriados e subelementos para descrever adequadamente o tipo de dados e alguma restrição que o elemento terá quando usado em um documento.
	name	NCName	O atributo name deve ser o nome de um elemento num documento XBRL. Ex:se o nome do elemento é 'assets.currentAssets', então o documento XBRL que usa essa taxonomia conter um elemento que parece este: <ci:assets.currentAssets...>4000</ ci:assets.currentAssets> Na taxonomia, os nomes de cada possíveis fatos financeiros são definidos e eles estão disponíveis para uso no documento XBRL. Um "NCName" é um formato de string XML definida, na prática é geralmente uma string que inicia-se com uma letra seguida pelo zero ou mais letras, números ou pontuação selecionada incluindo '.' (ponto), '-' (traço) e '_' (sublinhado).
	id	Opcional - ID	O valor deste atributo se usado deve ser único dentro do arquivo de taxonomia. Se usado ele deve ser igual ao valor do atributo name. A razão preliminar desse atributo existir é auxiliar as ferramentas que fazem uso de Xlink, rapidamente e eficientemente encontrando os elementos da taxonomia, quando processando linkbases ou outros arquivos XBRL.
	type	Tipos definido em XML Schema ou um dos seguintes tipos XBRL: monetaryItemType	Tipos de dados para o elemento. Elementos que são declarados monetaryItemType, sharesItemType ou decimalItemType todos devem se referir ao elemento numericContext no documento XBRL e elementos declarados como stringItemType, uriItemType e dateTimeItemType devem referir-se a elementos nonNumericContext no documento XBRL.

		sharesItemType decimalItemType stringItemType uriItemType dateTimeItemType	
	substitutionGroup	Tipo definidos em XML Schema ou um dos seguintes tipos XBRL: itemType tupleType	Este atributo XML Schema informa qual o tipo de dado abstrato é a base para o elemento. Por exemplo, item é um tipo abstrato. Elementos Artigos XBRL são baseados nesse tipo e herde suas propriedade, como atributos, restrições. Elementos XBRL usam esses tiposa como grupo de substituição. Uma outra forma de dizer isso é que um elemento que é baseado no elemento ' item', pode substituí-lo num documento XBRL
	balance	Opcional - um dos dois valores: Debit ou credit	Este atributo reserva ao autor da taxonomia a capacidade de esclarecer se uma quantidade é débito ou crédito.
anotation		Opcional - pode conter um outro elemento	Pode conter elementos adicionais não definidos. Algumas vezes a descrição de um subelemento está presente para fornecer informação adicional ao usuário da taxonomia, para esclarecer o papel ou o significado do elemento. Um outro subelemento do XML Schema é o appInfo, que fornece informações adicionais à aplicações. XBRL faz uso desse elemento appInfo para hospedar os elementos linkbaseRef, indicando ao usuário da taxonomia onde linkbases podem ser encontrados.