
Instituto de Computação
Universidade Estadual de Campinas

Especificação de Bibliotecas Digitais de Objetos Complexos

Ticiania Oniki Toffoli

Este exemplar corresponde à redação final da Dissertação devidamente corrigida e defendida por Ticiania Oniki Toffoli e aprovada pela Banca Examinadora.

Campinas, 28 de Fevereiro de 2012.

Prof. Dr. Ricardo da Silva Torres
(Orientador)

Dissertação apresentada ao Instituto de Computação, UNICAMP, como requisito parcial para a obtenção do título de Mestre em Ciência da Computação.

FICHA CATALOGRÁFICA ELABORADA POR
MARIA FABIANA BEZERRA MULLER - CRB8/6162
BIBLIOTECA DO INSTITUTO DE MATEMÁTICA, ESTATÍSTICA E
COMPUTAÇÃO CIENTÍFICA - UNICAMP

T571e Toffoli, Ticiania Oniki, 1982-
Especificação de bibliotecas digitais de objetos complexos /
Ticiania Oniki Toffoli. – Campinas, SP : [s.n.], 2012.

Orientador: Ricardo da Silva Torres.
Dissertação (mestrado) – Universidade Estadual de Campinas,
Instituto de Computação.

1. Bibliotecas digitais. 2. Engenharia de software auxiliada
por computador - Estudo de casos. I. Torres, Ricardo da Silva,
1977-. II. Universidade Estadual de Campinas. Instituto de
Computação. III. Título.

Informações para Biblioteca Digital

Título em inglês: Specification of digital libraries of digital complex objects

Palavras-chave em inglês:

Digital libraries

Computer-aided software engineering - Case studies

Área de concentração: Ciência da Computação

Titulação: Mestre em Ciência da Computação

Banca examinadora:

Ricardo da Silva Torres [Orientador]

Marcos André Gonçalves

Islene Calciolari Garcia

Data de defesa: 28-02-2012

Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Dissertação Defendida e Aprovada em 28 de Fevereiro de 2012, pela Banca examinadora composta pelos Professores Doutores:



Prof. Dr. Marcos André Gonçalves
DCC/UFMG



Prof^a. Dr^a. Islene Calciolari Garcia
IC / UNICAMP



Prof. Dr. Ricardo da Silva Torres
IC / UNICAMP

Especificação de Bibliotecas Digitais de Objetos Complexos

Ticiano Oniki Toffoli

Fevereiro de 2012

Banca Examinadora:

- Prof. Dr. Ricardo da Silva Torres (Orientador)
- Prof. Dr. Marcos André Gonçalves
Departamento de Ciência da Computação - UFMG
- Profa. Dra. Islene Calciolari Garcia
Instituto de Computação - UNICAMP
- Prof. Dr. André Santanchè (Suplente)
Instituto de Computação - UNICAMP
- Prof. Dr. Luciano Antonio Digiampietri (Suplente)
Escola de Artes, Ciências e Humanidades - USP

Resumo

Bibliotecas digitais são avançados e complexos sistemas de informação que armazenam, agregam e gerenciam informações correlatas para que comunidades específicas possam ter acesso a objetos digitais de interesse. Grandes volumes de dados de diferentes tipos e formatos vêm sendo gerados. Muitos desses dados são organizados em objetos digitais que podem ser objetos *complexos*, ou seja, objetos compostos de outros objetos digitais. Como a especificação e a implementação de uma biblioteca digital são tarefas cruciais para um gerenciamento eficaz de documentos, ferramentas vêm sendo criadas para auxiliar na especificação e na implementação de bibliotecas digitais. Entretanto, poucas ferramentas existentes permitem a especificação de objetos complexos em bibliotecas digitais. Além disso, dificuldades são encontradas na especificação, na modelagem e no reuso desse tipo complexo de dado. O objetivo desta dissertação foi propor um metamodelo de bibliotecas digitais de objetos complexos para especificar instâncias de bibliotecas digitais que gerenciem objetos complexos. O novo metamodelo foi incluído na ferramenta 5SGraph, ferramenta gráfica baseada no formalismo 5S (*Streams, Structures, Spaces, Scenarios e Societies*) que permite a especificação formal de bibliotecas digitais. A validação do novo metamodelo foi realizada por estudos de caso, por uma avaliação com usuários potenciais da ferramenta 5SGraph e pela implementação de um protótipo de uma biblioteca digital de documentos legais. Com os resultados obtidos, comprovou-se que a ferramenta 5SGraph estendida pode ser usada para especificar bibliotecas digitais de objetos complexos. As principais contribuições deste trabalho são: a caracterização de documentos legais como objetos complexos; a especificação de um metamodelo para especificação de bibliotecas digitais de objetos complexos; a implementação de um metamodelo para uso na ferramenta 5SGraph com o objetivo de especificar e instanciar bibliotecas digitais contendo objetos complexos; uma apresentação de estudos de casos para especificar objetos complexos de bibliotecas digitais na ferramenta 5SGraph; a validação do uso do novo metamodelo na ferramenta 5SGraph por usuários potenciais; e a especificação e a implementação de um protótipo de biblioteca digital de objetos complexos do tipo documento legal.

Abstract

Digital libraries are advanced and complex information systems that store, aggregate, and manage correlated information. These systems are used by specific communities to access digital objects of interest. Due to the creation of huge collections of heterogeneous data (in terms of type and format), some of these data are organized in digital *complex* objects, in the sense that they are composed by other digital objects. Since specification and implementation of digital libraries are decisive tasks to achieve an effective management of documents, specific software tools have been created to help and facilitate the specification and implementation of digital libraries. However, few existing tools used to model digital libraries allow the specification of complex objects. Besides, this kind of complex data make more difficult the tasks of specification, modeling, and reusing of complex objects. The objective of this dissertation was to propose a digital library metamodel in order to specify instances of digital libraries that manage complex objects. The new metamodel was included in the 5SGraph tool. 5SGraph is a software tool based on the 5S theory (*Streams, Structures, Spaces, Scenarios, and Societies*) and provides a visual environment for the formal specification of digital libraries. The new metamodel was validated through: case studies, an evaluation with potential users, and the implementation of a digital library prototype containing legal documents. According to the results, the extended 5SGraph tool can be used to specify digital libraries of complex objects. The main contributions of this work are: the characterization of legal documents as complex objects; the specification of a metamodel to be used in the specification of digital libraries of complex objects; the implementation of a metamodel in the 5SGraph tool, making the tool capable of specifying digital libraries of complex objects; the presentation of some case studies using the 5SGraph tool to specify complex objects of digital libraries; validation of the new metamodel in the 5SGraph with potential users; and the specification and implementation of a digital library prototype containing complex objects related to legal documents.

Agradecimentos

Meus sinceros agradecimentos ao meu orientador neste trabalho, o professor Ricardo da Silva Torres, por me aceitar como aluna, por todo o seu apoio durante o desenvolvimento deste mestrado (desde o apontamento de assuntos para o trabalho até às inúmeras indicações de correções e melhorias), por todo o seu entusiasmo como orientador e por todos os seus ensinamentos nesta fase.

Gostaria também de agradecer:

- aos meus pais, por toda a educação que me deram e por me tornarem esta pessoa que sou;
- a toda a minha família, por estarem sempre presentes na minha vida;
- ao meu eterno companheiro, Fernando Perin Munerato, por todo seu amor, alegria e paciência;
- aos meus ex-chefes, Paulo Kato e Yoneko Abe, por incentivarem os meus estudos;
- aos meus amigos queridos, que sempre me deram ajuda, força e coragem pra encarar este desafio; agradecimento especial ao Thiago Laubstein;
- aos pesquisadores e estudiosos de todos os trabalhos relacionados deste trabalho;
- aos colegas do IC e aos voluntários na validação do trabalho;
- aos membros das bancas;
- e a todas as entidades que apoiam os estudantes e a pesquisa nacional: UNICAMP, FAPESP, CNPq, CAPES, Microsoft, AMD, entre outras.

Sumário

Resumo	v
Abstract	vi
Agradecimentos	vii
1 Introdução	1
1.1 Contexto e Motivação	1
1.1.1 Aplicação: Gerenciamento de Documentos Legais	2
1.1.2 Aplicação: Gerenciamento de Objetos Digitais de Imagens	5
1.2 Contribuições da dissertação	7
1.3 Organização da dissertação	9
2 Revisão Bibliográfica	10
2.1 Bibliotecas Digitais	10
2.2 Formalismo 5S	10
2.2.1 Extensões	12
2.2.2 Ferramentas Baseadas no Formalismo 5S	13
2.3 Objetos Complexos	14
2.4 Formalismo 5S e Objetos Complexos	16
2.4.1 Documento Sobreposto	17
2.4.2 Descritor de Imagem	19
2.5 Documentos Legais	20
2.5.1 Definições	21
2.5.2 Estrutura	21
2.5.3 Criação e Publicação	23
2.5.4 Relacionamentos	24
2.6 Projetos/Aplicações de Documentos Legais	26
2.6.1 Governos	28
2.6.2 Universidades	29

2.7	Ontologias Formais	29
2.7.1	Definição	29
2.7.2	FRBR _{OO}	30
3	Elementos Legais como Objetos Complexos	33
3.1	Dispositivos	33
3.2	Texto Normativo	34
3.3	Documento Legal	36
3.4	Serviços	38
4	Especificação de Bibliotecas Digitais de Objetos Complexos	39
4.1	Especificação do Metamodelo	39
4.2	Extensão da ferramenta 5SGraph	41
4.3	Estudos de Caso	44
4.3.1	Documentos Legais	44
4.3.2	Documentos Sobrepostos	49
4.3.3	Descritores de Imagem	51
4.3.4	Imagens	52
4.3.5	Reuso de Objetos Complexos	55
5	Validação	59
5.1	Experimento com potenciais usuários	59
5.1.1	Protocolo Experimental	59
5.1.2	Perfis dos Usuários	60
5.1.3	Medidas para Análise	63
5.1.4	Resultados	63
5.2	Biblioteca Digital de Documentos Legais da DAC	68
6	Conclusões	74
6.1	Contribuições	74
6.2	Trabalhos Futuros	75
A	Questionário	77
	Bibliografia	84

Lista de Tabelas

2.1	Sistemas governamentais de alguns países que armazenam e pesquisam documentos legais.	28
2.2	Documentos diretivos de universidades públicas e serviços relacionados a esses documentos.	29
2.3	Artefatos jurídicos correlacionados às classes do modelo FRBR _{OO}	32
3.1	Objeto digital simples do Inciso I do Artigo 9° sendo alterado pela Deliberação CONSU-A-027/2002 (ver também Figura 3.1).	33
3.2	Objeto digital complexo referente à alteração do Artigo 9° do Manual do Aluno na Deliberação CONSU-A-027/2002.	34
3.3	Objeto digital complexo do Artigo 1° da Deliberação CONSU-A-027/2002.	34
4.1	Termos legais traduzidos para o inglês.	44
5.1	Dados coletados da Tarefa 1.	64
5.2	Dados coletados da Tarefa 2.	65
5.3	Dados coletados da Tarefa 3.	65
5.4	Resumo da média das medidas coletadas em cada tarefa.	65
5.5	Conceitos da Biblioteca Digital de Documentos Legais da UNICAMP de acordo com o formalismo 5S.	68

Lista de Figuras

1.1	Exemplo de ficha catalográfica em papel (contém os principais dados da Deliberação CONSU-A-001/1987).	5
1.2	Texto completo da Deliberação CONSU-A-001/1987 em papel.	6
1.3	Arquitetura de um sistema CBIR. Fonte [37].	7
2.1	Um objeto digital simples. Fonte [40].	12
2.2	Tipos de objetos digitais: (a) atômico; (b) composto e (c) complexo. (DOI é acrônimo de <i>Digital Object Identifier</i> e identifica um objeto digital). . . .	15
2.3	Objeto digital complexo.	15
2.4	Objetos digitais de uma biblioteca digital contendo <i>superimposed information</i> . Fonte [59].	18
2.5	Exemplo de um subdocumento. Fonte [59].	19
2.6	Exemplo da divisão da estrutura de uma lei em três partes básicas.	22
2.7	Composição dos elementos da parte normativa de uma lei. Adaptado de [16].	23
2.8	Conteúdo da publicação da Deliberação CONSU-A-027/2002. Fonte [3]. . . .	24
2.9	Conteúdo do artigo 9º do Manual do Aluno no período de 10/08/1998 (CONSU-A-011/1998) a 20/12/2002 (CONSU-A-027/2002).	25
2.10	Conteúdo do artigo 9º do Manual do Aluno no período de 21/12/2002 (CONSU-A-027/2002) a 16/12/2005 (CONSU-A-024/2005).	25
2.11	Evolução da norma no tempo. Fonte [16].	31
3.1	Objetos digitais da Deliberação CONSU-A-027/2002.	35
3.2	Versionamento de um objeto digital: (a) alteração no próprio objeto e (b) alteração ocasionada pela inclusão de novo objeto digital.	35
3.3	Textos legais como <i>superimposed information</i>	37
4.1	Relacionamentos entre componentes do metamodelo de bibliotecas digitais de objetos complexos.	40
4.2	Interface da ferramenta 5SGraph.	42
4.3	Novos elementos <i>ComplexObj</i> e <i>Object</i> no arquivo do metamodelo da ferramenta 5SGraph.	43

4.4	Novos elementos <i>ComplexObj</i> e <i>Object</i> do metamodelo da ferramenta 5SGraph visualizados pela interface.	43
4.5	<i>Streams</i> da biblioteca digital de documentos legais e detalhes da <i>stream Text</i> , na ferramenta 5SGraph.	45
4.6	Dispositivo Item (<i>Item</i>) como objeto digital simples na ferramenta 5SGraph.	45
4.7	<i>Caput</i> do dispositivo Item (<i>Item</i>) definido como <i>stream</i> do tipo <i>Text</i> , na ferramenta 5SGraph.	46
4.8	Dispositivos no arquivo gerado pela ferramenta 5SGraph.	46
4.9	Dispositivos como objetos complexos e detalhe da composição do objeto complexo Artigo (<i>Article</i>), na ferramenta 5SGraph.	47
4.10	<i>Caput</i> do Artigo (<i>ArticleCaput</i>) definido como <i>stream</i> do tipo <i>Text</i> , na ferramenta 5SGraph.	47
4.11	Elemento Parágrafo (<i>Paragraph</i>) de um Artigo como objeto digital do tipo <i>Paragraph</i> , na ferramenta 5SGraph.	47
4.12	Artigo (<i>Article</i>) no arquivo gerado pela ferramenta 5SGraph.	48
4.13	Composição do objeto complexo Parte Normativa (<i>NormativePart</i>), na ferramenta 5SGraph.	48
4.14	Composição do objeto complexo Documento Legal (<i>LegalDocument</i>), na ferramenta 5SGraph.	49
4.15	Parte Normativa (<i>NormativePart</i>) e Documento Legal (<i>LegalDocument</i>) no arquivo gerado pela ferramenta 5SGraph.	50
4.16	Documento base (<i>BaseDocument</i>) especificado na ferramenta 5SGraph. . .	50
4.17	Subdocumento (<i>Subdocument</i>) especificado na ferramenta 5SGraph. . . .	51
4.18	Especificação do subdocumento (<i>Subdocument</i>) como objeto digital do documento sobreposto (<i>SuperDocument</i>), na ferramenta 5SGraph.	51
4.19	Composição do documento sobreposto (<i>SuperDocument</i>), na ferramenta 5SGraph.	51
4.20	Estrutura do documentos base, do subdocumento e do documento sobreposto no arquivo gerado pela ferramenta 5SGraph.	52
4.21	Aplicação em Java especificada na ferramenta 5SGraph.	53
4.22	<i>Stream</i> de um programa Java no arquivo gerado pela ferramenta 5SGraph.	53
4.23	Exemplo de objetos digitais de software, na ferramenta 5SGraph.	53
4.24	Descritor de imagem especificado na ferramenta 5SGraph.	53
4.25	Descritor de imagem (<i>ImageDescriptor</i>) como objeto complexo no arquivo gerado pela ferramenta 5SGraph.	54
4.26	Objeto digital de imagem (IDO) especificado na ferramenta 5SGraph. . . .	54
4.27	Objeto complexo de imagem (ICO) especificado na ferramenta 5SGraph. .	54
4.28	<i>StructuredFeatureVector</i> especificado na ferramenta 5SGraph.	55

4.29	Vetor de características (<i>Feature Vector</i>) especificado na ferramenta 5SGraph.	55
4.30	Valores das distâncias (<i>Distances</i>) entre imagens especificados na ferramenta 5SGraph.	56
4.31	Arquivo contendo objeto complexo do tipo imagem.	56
4.32	Nó ICO sendo salvo para posterior reuso em outro modelo da ferramenta 5SGraph.	57
4.33	Arquivo gerado pela ferramenta 5SGraph contendo componente ICO.	57
4.34	Nó sendo carregado na ferramenta 5SGraph para reuso em novo modelo.	57
4.35	Inconsistência do tipo de objeto na ferramenta 5SGraph.	58
4.36	Inconsistência do tipo de <i>stream</i> na ferramenta 5SGraph.	58
5.1	Grau de formação dos participantes.	61
5.2	Nível de conhecimento dos participantes em inglês.	61
5.3	Nível de conhecimento dos participantes em bibliotecas digitais.	61
5.4	Nível de conhecimento dos participantes sobre objetos digitais complexos.	61
5.5	Nível de conhecimento dos participantes sobre 5S.	62
5.6	Nível de conhecimento dos participantes sobre 5SGraph.	62
5.7	Experiência dos participantes na criação de bibliotecas digitais.	62
5.8	Experiência dos participantes na criação de bibliotecas digitais de objetos complexos.	62
5.9	Uso de ferramentas na especificação de objetos digitais.	62
5.10	Nível de conhecimento dos participantes sobre objetos digitais complexos após uso da ferramenta 5SGraph.	64
5.11	Opinião dos participantes sobre as instruções das tarefas.	64
5.12	Opinião dos participantes sobre o nível de complexidade para especificar objetos digitais na ferramenta 5SGraph.	67
5.13	Opinião dos participantes sobre a funcionalidade da ferramenta 5SGraph de especificar objetos complexos.	67
5.14	Opinião dos participantes sobre a utilidade da ferramenta 5SGraph.	67
5.15	Parte do XMLSchema de documentos legais. Fonte [27].	69
5.16	Parte do arquivo XML de uma Deliberação da UNICAMP mostrando os metadados.	71
5.17	Parte do arquivo XML de uma Deliberação da UNICAMP mostrando a parte normativa.	72
5.18	Interface de busca do protótipo de biblioteca digital de documentos legais.	72
5.19	Resultados de uma busca no protótipo de biblioteca digital de documentos legais.	73

Capítulo 1

Introdução

1.1 Contexto e Motivação

Com o avanço e o uso em massa de novas tecnologias, grandes coleções de informações digitais vêm sendo produzidas. Este cenário faz surgir a demanda por aplicações com o objetivo de armazenar, processar e gerenciar esses dados. Estas aplicações precisam tratar informações complexas, com grandes volumes de dados heterogêneos, como textos, imagens, vídeos, softwares, entre outros. Uma das formas de gerenciar esses dados consiste no uso de bibliotecas digitais, que são avançados e complexos sistemas de informação que disseminam conhecimento e oferecem busca estruturada, navegação e preservação de documentos [39]. Além disso, são acessíveis em ambientes distribuídos (acesso remoto e compartilhado) visando atender às necessidades de comunidades de usuários específicos [42].

As informações ou documentos em uma biblioteca digital são manifestados em forma de objetos digitais, apresentados em diversos tipos de conteúdo. Quando o objeto digital contém um ou mais objetos digitais, ele é denominado de objeto complexo. Portanto, um objeto complexo é um objeto digital simples ou uma composição recursiva de outros objetos complexos, necessitando de uma estrutura para organizar seus componentes [59]. Para o gerenciamento desses objetos digitais complexos surgem as bibliotecas digitais de objetos digitais complexos. Entretanto, somente alguns poucos sistemas de bibliotecas digitais, como o Fedora [9, 54], permitem o gerenciamento desses tipos de objetos.

Por serem sistemas complexos englobando diversos tópicos, como armazenamento e recuperação de informação, interação humano-computador, interoperabilidade, computação distribuída, entre outros, formalismos vêm sendo utilizados para ajudar a modelar, definir e descrever bibliotecas digitais.

Um exemplo de formalismo muito utilizado para descrever bibliotecas digitais e seus elementos constituintes é o 5S [40, 42]. Esse formalismo apresenta os conceitos de *Streams*, *Structures*, *Spaces*, *Scenarios* e *Societies* para definir conceitos de bibliotecas digitais.

Conceitos relacionados aos objetos complexos e à busca de imagens baseada em conteúdo foram formalizados por Kozievitch *et al.* [52,61]. Além desse trabalho, outros trabalhos estenderam o formalismo 5S para definir: uma biblioteca digital integrada de serviços de navegação, de busca e de visualização [66,67]; um modelo de qualidade para bibliotecas digitais [43]; uma biblioteca digital com gerenciamento de informação pessoal [56]; um modelo de referência de biblioteca digital [60]; um gerador de biblioteca digital instanciada no DSpace [44]; o conceito de recomendação [64]; e componentes de uma biblioteca digital contendo *superimposed information* [59,61].

Para um gerenciamento eficaz de documentos, a especificação e a implementação de uma biblioteca digital são tarefas cruciais. Por isso, ferramentas são necessárias para auxiliar os desenvolvedores nessas tarefas e facilitar a construção de bibliotecas digitais. Várias ferramentas baseadas no formalismo 5S foram criadas: 5SL [40,41], uma linguagem descrita em XML, baseada nos conceitos 5S e considerada um metamodelo (uma biblioteca digital genérica); 5SGraph [40,71,72], uma ferramenta visual usada para modelagem de bibliotecas digitais que utiliza um metamodelo de bibliotecas digitais para descrever uma instância de uma biblioteca digital expressa em 5SL; 5SGen [47], um gerador de biblioteca digital genérica que implementa os serviços definidos para uma biblioteca digital; 5SQual [57], uma ferramenta que analisa a qualidade dos objetos digitais, dos metadados e dos serviços de bibliotecas digitais; e 5SSuite [66,68], um agrupamento de ferramentas que integra o processo de geração de bibliotecas digitais. Entretanto, poucas ferramentas são adequadas para definir, modelar, usar e reusar objetos complexos.

As subseções a seguir apresentam exemplos de aplicações que manipulam objetos digitais complexos.

1.1.1 Aplicação: Gerenciamento de Documentos Legais

Do ponto de vista de aplicação, bibliotecas digitais que gerenciem documentos legais vêm despertando interesse na comunidade em geral. Normas, leis ou documentos legais podem ser vistos como objetos complexos, pois são compostos por artigos, podem possuir texto, imagem, áudio, além de serem interligados a outros documentos legais.

A Diretoria Acadêmica (DAC) da Universidade Estadual de Campinas (UNICAMP) é um exemplo de comunidade com interesse em aplicações de documentos legais. A DAC possui, como um dos objetivos, subsidiar tecnicamente as decisões institucionais com plena observância dos preceitos legais. Por isso, os funcionários da DAC precisam diariamente consultar leis da UNICAMP e leis educacionais publicadas por outras instituições como o Ministério da Educação, o Conselho Nacional de Educação e o Diário Oficial do Estado de São Paulo. Algumas consultas são exemplificadas abaixo. Quando ocorre:

- a publicação de legislação pelo Ministério da Educação (MEC) alterando a carga

horária mínima de alguns cursos de graduação, uma consulta a esta lei deve atualizar os regimentos dos cursos de graduação da UNICAMP, reformulando as grades desses cursos. Esta alteração deve ser realizada dentro dos prazos estabelecidos pela norma;

- o processo de aproveitamento de estudos, os cursos reconhecidos no *site* da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) devem ser consultados com a finalidade de obter embasamento legal. Esse processo é realizado quando um determinado aluno ingressante, que anteriormente cursou disciplinas em outra Universidade, solicita a convalidação dos créditos já cursados com os créditos requisitados pela UNICAMP. Após a verificação do reconhecimento do curso, deve-se comparar a ementa e a carga horária das disciplinas.

Os exemplos acima referem-se a leis educacionais publicadas por outros órgãos. Para exemplificar uma consulta à legislação da UNICAMP, tem-se o caso de análise de recurso referente ao jubramento de um aluno por não cumprimento dos créditos no prazo máximo permitido para conclusão do curso. Nessa consulta, é necessário checar o prazo máximo de conclusão do curso definido no texto do regimento do curso de graduação, vigente no ano de ingresso do aluno.

Entretanto, ao realizar consultas diárias a leis educacionais, a DAC enfrenta muitos problemas:

- heterogeneidade de fontes de dados: as leis referentes ao ensino superior no Brasil (tais como, revalidação de diplomas; regulação, supervisão e avaliação de cursos) estão distribuídas nas esferas federal, estadual e municipal, e na própria Universidade, ocasionando buscas em diferentes fontes. Verifica-se ausência de um sistema de informação que trate documentos que se referem a um mesmo tema, no caso, que reúna os preceitos legais de interesse para uma instituição como uma Universidade;
- baixa eficácia ou inexistência dos sistemas de busca: dificuldade em encontrar o preceito legal desejado devido a serviços de busca inexistentes ou pouco eficazes encontrados nos portais que disponibilizam textos de normas. Muitas vezes os usuários precisam procurar os atos normativos manualmente (usando algum comando *Procurar*) diretamente no texto dos documentos. Além disso, a ordenação dos resultados encontrados pelo sistema nem sempre é satisfatória. Poucos *sites* fornecem o serviço de *feedback* do usuário, no qual é permitido que o usuário, a partir da interface, identifique quais foram os documentos/respostas relevantes ou quais documentos devem sempre ser retornados nas próximas consultas referentes ao mesmo assunto pesquisado;
- dificuldades em encontrar o texto legal válido para uma determinada data e em conhecer o período de vigência de um texto normativo devido à grande quantidade

de documentos que representam uma mesma lei: um documento legal é gerado para criar uma norma; posteriormente, outros documentos são gerados para alterar, revogar ou vetar partes desta mesma norma. Novamente, esses novos documentos podem ter seus textos alterados por novas publicações. Por causa desses vários eventos, podem existir várias entidades de uma norma: o conteúdo original promulgado após o processo legislativo; a versão consolidada desta norma em uma determinada data (resolução de consultas temporais); a versão traduzida desta norma para outra língua, *etc.*;

- inexistência de *links* hipertextuais entre documentos legislativos: uma norma pode fazer referências a outras normas, seja para realizar alterações ou revogar um dispositivo, porém, nem todos preceitos legais encontrados possuem este relacionamento explícito, pois esta funcionalidade é disponibilizada apenas por algumas autoridades emittentes ou portais de legislação;
- ausência ou quebra de página: existência de muitos *links* perdidos para os textos das normas, também dependendo da autoridade emittente e do portal disponibilizado;
- inconsistências entre textos publicados e o texto original: problemas estruturais dos textos legais; grafia incorreta ou ausência de partes do texto ao repassar o texto original para uma publicação. Estes problemas são ocasionados por erros humanos devido a digitações incorretas do texto;
- citação de documentos legais em nível de artigos e parágrafos na resolução de decisões: a DAC precisa definir quais foram as normas juntamente com seus respectivos artigos e/ou parágrafos utilizados para a tomada de decisão. Com o surgimento de um novo parecer semelhante, ocorre a procura pelo antigo parecer de modo que sejam identificadas as normas e artigo(s) e/ou parágrafo(s) utilizados;
- disponibilização não digital de documentos de interesse: muitos documentos sobre a legislação educacional da Universidade se encontram em papel, não disponibilizados em forma digital, impedindo o acesso rápido à legislação. As Figuras 1.1 e 1.2 exemplificam como a DAC procura pelas leis ainda não digitalizadas: a Figura 1.1 exibe uma ficha catalográfica em papel com os principais dados da Deliberação CONSU-A-001/1987 que são utilizados como metadados (data e página de publicação no Diário Oficial do Estado; palavras-chaves; ementa) e a Figura 1.2 exibe o texto completo da Deliberação em papel (cópia do texto da lei colado em uma folha para armazenamento em arquivos);
- leitura diária de publicações oficiais para conhecimento de novas leis e alterações normativas em leis educacionais.

DOCUMENTAÇÃO ACADÊMICA U
Por: <i>Univamp - Unigão</i>
PUBL. <i>DOE</i>
DATA <i>09/04/87</i>
PAG. <i>13</i>
PALAVRAS CHAVES: <i>dir.: Consu</i>

Deliberação Consu 1/87, de 09/04/87.

Dispõe sobre o exame e deliberação de processos destinados ao extinto Conselho Diretor.

Figura 1.1: Exemplo de ficha catalográfica em papel (contém os principais dados da Deliberação CONSU-A-001/1987).

A principal referência desse tipo de aplicação é o projeto nacional LexML [27] que disponibilizou o Portal LexML [22] para os cidadãos realizarem consultas de normas nas esferas federal, estadual e municipal, publicados por órgãos ou por entidades nacionais.

1.1.2 Aplicação: Gerenciamento de Objetos Digitais de Imagens

Além de documentos legais, outros tipos de objetos complexos amplamente utilizados são imagens e descritores de imagens. Aplicações voltadas à recuperação de imagem baseada em conteúdo (*content-based image retrieval* ou CBIR) gerenciam esses tipos de objetos. Basicamente, essas aplicações tentam recuperar imagens similares a uma especificação e/ou a um padrão de imagem definidos pelo usuário, automatizando o processo de busca. Para que essa recuperação seja possível, algoritmos de processamento de imagens são utilizados para codificar propriedades visuais das imagens, como cor, textura e forma, em vetores de características, indexando-os e possibilitando comparações de similaridade utilizando essas características [36, 65].

A Figura 1.3 mostra a arquitetura típica de um sistema de recuperação de imagens por conteúdo. Os módulos e setas tracejadas são responsáveis por extrair os vetores de características das imagens e armazená-los na base de imagens. Geralmente, esse processo é realizado apenas uma vez para cada imagem e para cada descritor. O descritor de imagens é responsável por caracterizar as propriedades da imagem e realizar a comparação das imagens baseada nas propriedades extraídas.

A interface então permite ao usuário especificar uma consulta e visualizar as imagens recuperadas. O módulo de processamento de consultas extrai o vetor de características da consulta e aplica uma métrica de distância para avaliar a similaridade entre a imagem da

DOCUMENTAÇÃO ACADEMICA U
UNICAMP
PASTA 019002
PUBL. DOE
DATA 09/04/87
PAG. 13
PALAVRAS CHAVES: dir.: consu

Deliberação CONSU 1/87 <i>Dispõe sobre o exame e deliberação de processos destinados ao extinto Conselho Diretor</i> O Reitor da Universidade Estadual de Campinas, na qualidade de Presidente do Conselho Universitário, tendo em vista o decidido pelo Conselho na 1.ª Sessão Ordinária, realizada em 31 de março de 1987, faz expedir a seguinte deliberação: Artigo 1.º — Os Processos que se encontram na Secretaria Geral e se destinam ao exame e deliberação do extinto Conselho Diretor serão apreciados, diretamente, pelo Conselho Universitário Pleno de 14-4-87. Artigo 2.º — Não ocorrendo deliberação final, o processo retornará à Secretaria Geral para instrução nos termos do Regimento Interno do CONSU. Artigo 3.º — Esta Deliberação entra em vigor nesta data.

Figura 1.2: Texto completo da Deliberação CONSU-A-001/1987 em papel.

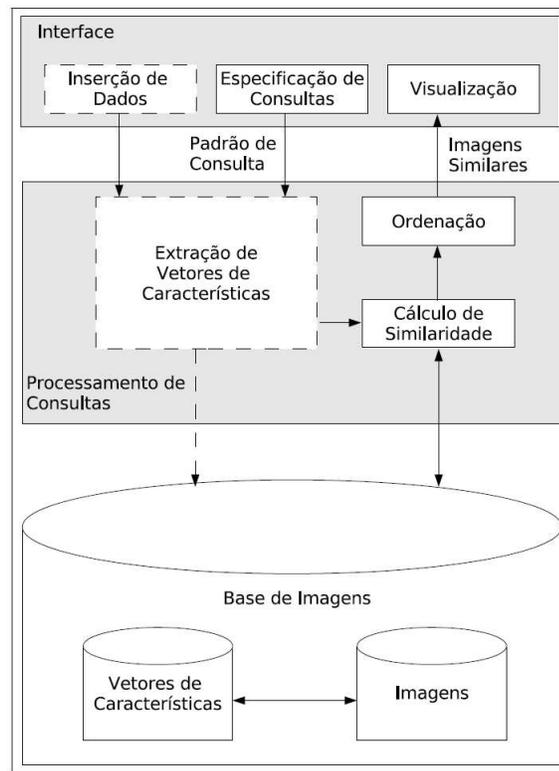


Figura 1.3: Arquitetura de um sistema CBIR. Fonte [37].

consulta e as imagens da base. Quanto maior o valor da distância, menor a semelhança das imagens. Em seguida, esse módulo ordena as imagens, retornando as mais similares para a interface [37].

Kozievitch [50] formaliza, utilizando o formalismo 5S, objetos complexos e o processo de recuperação de imagem baseada em conteúdo. O objeto digital complexo do tipo imagem é composto de uma imagem, de um conjunto de vetores de características e das distâncias entre esta imagem e outras imagens de acordo com os descritores definidos. A ferramenta SuperIDR [59] é um exemplo de aplicação que gerencia imagens, subimagens, metadados e anotações.

1.2 Contribuições da dissertação

Uma biblioteca digital de objetos complexos e conceitos relacionados foram definidos formalmente usando o formalismo 5S. Para respaldar essas definições, aplicações [52, 61] que gerenciam objetos complexos foram implementadas sobre esses conceitos. Porém, nada ainda tem sido realizado em relação ao apoio à especificação de bibliotecas digitais

de objetos complexos por usuários. Além disso, devido à complexidade de alguns objetos digitais, surge a necessidade de aprimorar a especificação de bibliotecas digitais. Assim, o foco do trabalho foi voltado a responder as seguintes questões:

- *Como deve ser um metamodelo de uma biblioteca digital de objetos complexos?*
- *Quais modificações precisam ser realizadas no metamodelo de uma biblioteca digital de objetos digitais para que o metamodelo seja utilizado na instanciação de bibliotecas digitais de objetos complexos?*

Assim, o objetivo desta dissertação é especificar bibliotecas digitais de objetos complexos, criando um metamodelo para bibliotecas digitais de objetos complexos.

Além disso, uma análise de extensão na ferramenta de especificação visual de biblioteca digital 5SGraph foi realizada, para que esta possibilite aos usuários especificar uma biblioteca digital de objetos complexos na linguagem 5SL. Com essa mudança na ferramenta, usuários podem definir os elementos necessários de uma biblioteca digital de objetos complexos, modelando-os visualmente na ferramenta. Essa extensão da ferramenta 5SGraph propicia facilidade aos usuários no processo de análise de dados complexos e viabiliza a construção de bibliotecas digitais de objetos complexos. Uma outra funcionalidade da ferramenta é a possibilidade de reuso de objetos complexos em novas instâncias de bibliotecas digitais.

A ferramenta 5SGraph estendida foi validada por usuários potenciais e utilizada em estudos de casos para especificar objetos digitais complexos dos tipos: tese de doutorado, documento legal, imagem e descritor de imagens. Também foi implementado um protótipo de biblioteca digital de documentos legais, baseado no modelo gerado pela ferramenta 5SGraph.

Após pesquisa e estudo, as principais contribuições deste trabalho são:

- a caracterização de documentos legais como objetos complexos;
- a especificação de um metamodelo para especificação de bibliotecas digitais de objetos complexos;
- a implementação de um metamodelo para uso na ferramenta 5SGraph com o objetivo de especificar e instanciar bibliotecas digitais contendo objetos complexos;
- estudos de casos para especificar objetos complexos de bibliotecas digitais na ferramenta 5SGraph;
- a validação do uso do novo metamodelo na ferramenta 5SGraph por usuários potenciais;

- a especificação e a implementação de um protótipo de biblioteca digital de objetos complexos do tipo documento legal, sendo esta biblioteca digital baseada no modelo gerado a partir do metamodelo de bibliotecas digitais de objetos complexos.

1.3 Organização da dissertação

O restante da dissertação está organizada como segue: o próximo capítulo apresenta a revisão bibliográfica, com conceitos básicos e trabalhos correlatos; o Capítulo 3 caracteriza documentos legais como objetos digitais complexos; o Capítulo 4 demonstra a aplicação do formalismo 5S para a criação do metamodelo de uma biblioteca digital de objetos complexos, além de exibir as alterações no metamodelo da ferramenta 5SGraph para especificar bibliotecas digitais de objetos complexos; e o Capítulo 5 descreve a validação e a análise dos resultados referentes às alterações do metamodelo da ferramenta 5SGraph. Por fim, o Capítulo 6 apresenta as conclusões e discute possíveis extensões para este trabalho.

Capítulo 2

Revisão Bibliográfica

Este capítulo apresenta uma visão geral sobre bibliotecas digitais, objetos complexos, formalismo 5S e documentos legais.

2.1 Bibliotecas Digitais

Uma biblioteca digital pode ser definida, de maneira bastante ampla, como uma coleção organizada de objetos digitais associada a um conjunto de serviços acessíveis em ambientes distribuídos, visando atender às necessidades de comunidades de usuários [42]. É um dos tipos mais avançados e complexos de sistemas de informação, por envolver, dentre outras coisas, busca estruturada, navegação, recomendação, preservação de documento, serviços de informação multimídia e disseminação seletiva da informação [39].

Os gerenciadores de bibliotecas digitais são softwares desenvolvidos para facilitar a construção de bibliotecas digitais. Eles fornecem armazenamento e acesso às coleções de qualquer tipo de documentos. Os principais e mais utilizados gerenciadores, disponibilizados como software livre, são: DSpace [6, 69], Greenstone [10, 70] e Fedora [9, 54].

Por serem sistemas complexos englobando diversas áreas e/ou serviços (armazenamento e recuperação de informação, interação humano-computador, interoperabilidade, computação distribuída, entre outros), formalismos vêm sendo utilizados para ajudar a modelar, definir e descrever bibliotecas digitais. A próxima seção apresenta o formalismo 5S (*Streams, Structures, Spaces, Scenarios, Societies*).

2.2 Formalismo 5S

O formalismo 5S apresenta uma série de definições relacionadas a bibliotecas digitais [40, 42]. O objetivo deste formalismo é definir, relacionar e unificar conceitos de objetos

digitais, metadados, coleções e serviços. Essa definição tem como base os conceitos de:

- *Streams*: definem os tipos de dados; são sequências de itens arbitrários usados para descrever conteúdo estático e dinâmico (como *bits*, caracteres, imagens, *etc.*);
- *Structures*: definem como a informação é estruturada e organizada; podem ser vistos como grafos rotulados direcionados que impõem organização;
- *Spaces*: definem propriedades e operações; são conjuntos de operações que obedecem certas restrições;
- *Scenarios*: definem o comportamento da biblioteca digital; consistem de sequências de eventos ou ações que modificam estados de uma computação para atender a um requisito funcional;
- *Societies*: são conjuntos de atores, usuários, entidades e atividades e os relacionamentos entre eles.

A definição de um **objeto digital** (Def. 16 em [40]) é uma tupla $do = (h, SM, ST, StructuredStreams)$, em que:

1. $h \in H$, sendo H um conjunto de identificadores universais únicos (*labels*);
2. $SM = \{sm_1, sm_2, \dots, sm_n\}$ é um conjunto de *streams*;
3. $ST = \{st_1, st_2, \dots, st_m\}$ é um conjunto de *Structural Metadata Specifications*;
4. $StructuredStreams = \{stsm_1, stsm_2, \dots, stsm_p\}$ é um conjunto de funções de *StructuredStream* definidas a partir de *streams* no conjunto SM do objeto digital e de estruturas no conjunto ST .

Structural Metadata Specifications definem relações entre o objeto e suas partes, ou seja, definem a estrutura interna do objeto digital. *StructuredStreams* definem o mapeamento de uma estrutura para *streams*. Por exemplo, definem o texto de um capítulo de um livro.

A Figura 2.1 exemplifica a composição de um objeto digital contendo diferentes *streams*. Pode-se observar: *Handle* como identificador do objeto digital; as *streams* do tipo *text*, *Audio* e *Image* definidas para o objeto digital; *Structure* apresentando a estrutura dos dados do objeto digital; e as caixas rotuladas com as *streams* indicando a composição e a organização das *streams* no objeto digital.

A definição de uma **biblioteca digital** (Def. 24 em [40]) é uma tupla com quatro elementos $(R, Cat, Serv, Soc)$, em que:

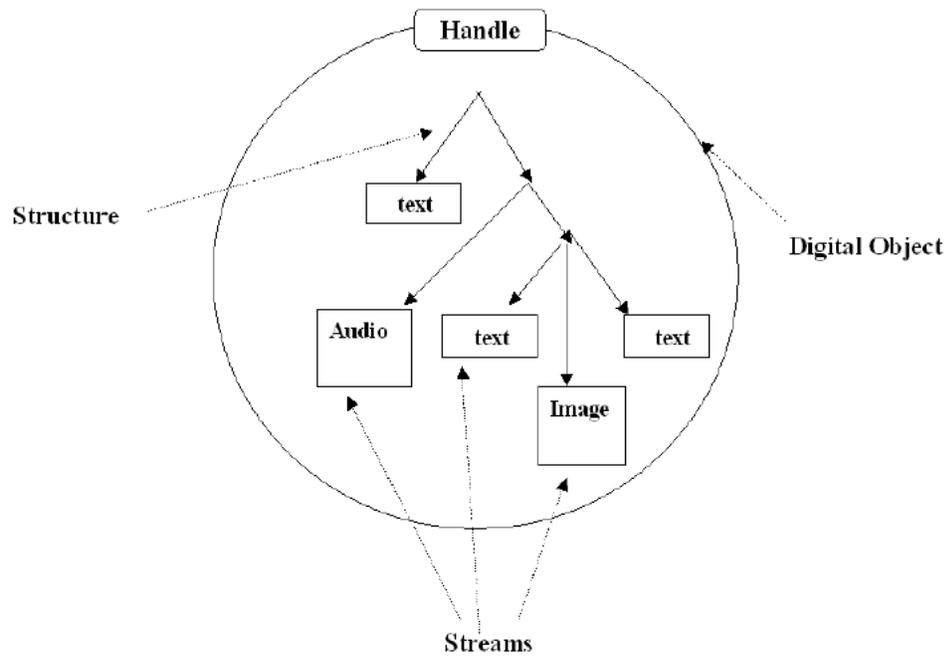


Figura 2.1: Um objeto digital simples. Fonte [40].

1. R é um repositório;
2. $Cat = \{DM_{C_1}, DM_{C_2}, \dots, DM_{C_K}\}$ é um conjunto de catálogos de metadados para todas as coleções $\{C_1, C_2, \dots, C_K\}$ no repositório;
3. $Serv$ é um conjunto de serviços contendo, ao menos, os serviços de indexação, de busca e de navegação;
4. Soc é uma sociedade.

2.2.1 Extensões

As definições do formalismo 5S foram estendidas para definir: uma biblioteca digital integrada de serviços (navegação, busca e visualização) [66,67]; um modelo de qualidade para bibliotecas digitais [43]; uma biblioteca digital com gerenciamento de informação pessoal [56]; um modelo de referência de biblioteca digital [60]; um gerador de biblioteca digital instanciada no DSpace [44]; conceitos do serviço de recomendação e conceitos de conteúdo de software [64]; componentes de uma biblioteca digital contendo *superimposed information* [59,61], que representam novas informações (conteúdo e/ou estrutura) criadas

para referenciar informações já existentes; e objetos complexos e busca de imagens baseada em conteúdo [48, 50, 61].

2.2.2 Ferramentas Baseadas no Formalismo 5S

Na construção e no desenvolvimento de bibliotecas digitais, dificuldades são encontradas devido aos vários aspectos complexos que devem ser tratados [40]:

- a especificação do conteúdo a ser armazenado na biblioteca digital;
- a organização, a estruturação, a descrição e o acesso ao conteúdo;
- os serviços oferecidos;
- os agentes automatizados que usam esses serviços, interagindo uns com os outros.

Por isso, ferramentas são necessárias para facilitar a construção de bibliotecas digitais por desenvolvedores, ajudando no levantamento de requisitos, na modelagem conceitual e na metodologia do desenvolvimento.

Várias ferramentas baseadas no formalismo 5S foram criadas: 5SL, 5SGraph, 5SGen, 5SQual e 5SSuite. Entretanto, elas ainda não lidam com objetos complexos.

5SL [40, 41] é uma linguagem de alto nível e de domínio específico (na qual abstrações e notações do domínio são usadas) e que permite especificação declarativa de propriedades de bibliotecas digitais em XML. Sua base formal é composta por submodelos baseados nos conceitos básicos do 5S e fornece uma ferramenta de especificação precisa que facilita a prototipagem e ajuda na validação da implementação. Assim, é útil na geração e na instalação de uma implementação da biblioteca digital. Pode ser considerada um meta-modelo (uma biblioteca digital genérica) cuja instanciação produz modelos de sistemas específicos de bibliotecas digitais.

5SGraph [40, 71, 72] é uma ferramenta visual de modelagem de bibliotecas digitais que utiliza um metamodelo de bibliotecas digitais para descrever uma instância de uma biblioteca digital, expressa em 5SL. Sendo visual, é útil para usuários leigos que não possuem o conhecimento teórico da linguagem descritiva e facilita os processos de análise dos dados e de modelagem. É configurada para acionar outras ferramentas visuais já existentes e amplamente utilizadas como, por exemplo, ferramentas de modelagem UML. Além disso, a ferramenta garante a consistência e as restrições do metamodelo 5S sobre a instância do modelo e permite o reuso de elementos e modelos, reduzindo o tempo e o esforço dos usuários. Uma de suas vantagens é a visualização síncrona dos elementos do metamodelo com os elementos do modelo sendo descrito.

5SGen [47] é um gerador de biblioteca digital genérica para semiautomaticamente produzir serviços de bibliotecas digitais; transforma conceitos definidos na linguagem 5SL

em classes orientadas a objetos. A origem da transformação é a captura dos aspectos estruturais e comportamentais dos serviços da biblioteca digital a partir dos modelos 5SL de *Societies* e de *Scenarios*. O primeiro modelo é transformado em um modelo de representação XMI¹ com o uso de JDOM² e XPath³ e então a implementação *open source* XMI2Java gera as classes Java que implementam os gerenciadores de serviços da biblioteca digital.

5SQual [57, 58] verifica automaticamente a qualidade dos objetos digitais (acessibilidade, significância, similaridade, atualizações), dos metadados (completude e integridade) e dos serviços (eficiência e confiabilidade) de uma biblioteca digital.

5SSuite [66, 68] é uma ferramenta utilizada para integrar o processo de geração de uma biblioteca digital, incluindo a definição dos requisitos, o modelo conceitual, prototipação e a geração de código. Essa ferramenta engloba 5SGraph, 5SGen e SchemaMapper (uma ferramenta que mapeia o esquema local em um esquema global) e foi validada na área de arqueologia.

2.3 Objetos Complexos

As informações ou documentos em uma biblioteca digital são manifestados em forma de objetos digitais, apresentados em diversos tipos de conteúdo, podendo ser classificados em [34]:

- atômico (Figura 2.2a): formado por um ou mais dados apresentados em um mesmo formato. Exemplo: uma tese ou uma dissertação em formato único, na qual textos e imagens estão presentes no mesmo formato textual.
- composto (Figura 2.2b): formado por vários elementos que podem estar em diferentes formatos; é uma agregação de unidades distintas de informação que quando combinadas formam uma unidade lógica única [55]. Exemplo: uma tese ou dissertação que possui o texto em um formato e figuras ou outros dados em um formato não textual. Juntamente, esses dados formam o objeto digital que representa a tese ou a dissertação. A Figura 2.1 também exemplifica um objeto composto.
- complexo (Figura 2.2c): formado por vários dados, tendo pelo menos um deles como objeto digital (de qualquer tipo, podendo ser também um objeto complexo, e permitindo a recursividade da composição de objetos complexos). Exemplo: uma tese ou dissertação que é formada por objetos digitais que representam os textos e as imagens.

¹<http://www.omg.org/spec/XMI/> (último acesso em 20/01/2012).

²<http://www.jdom.org> (último acesso em 20/01/2012).

³<http://www.w3schools.com/xpath/> (último acesso em 20/01/2012).

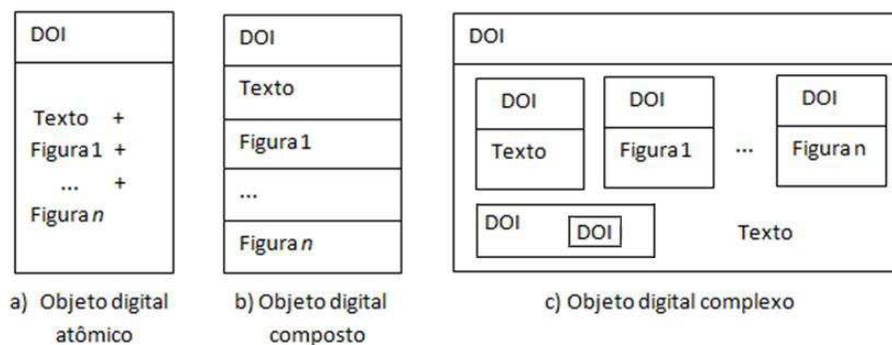


Figura 2.2: Tipos de objetos digitais: (a) atômico; (b) composto e (c) complexo. (DOI é acrônimo de *Digital Object Identifier* e identifica um objeto digital).

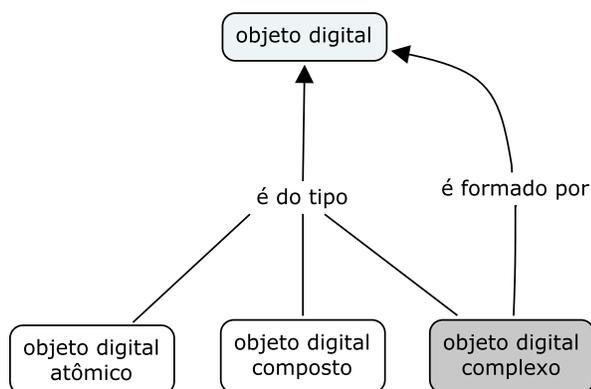


Figura 2.3: Objeto digital complexo.

O objeto digital simples é caracterizado pelo fato de não ser complexo. Portanto, tanto o objeto atômico quanto o objeto composto representam um objeto digital simples.

O objeto complexo, por conter outros objetos digitais (ver Figura 2.3), deve ser armazenado com essa composição explícita de seus componentes, identificando cada documento (todo e partes) unicamente. Além disso, o gerenciamento dos serviços de busca e de navegação de um objeto complexo deve ser provido pela biblioteca digital, que ora deve retornar o objeto complexo, ora deve retornar apenas os objetos digitais, independentes do objeto complexo ao qual pertencem.

Várias tecnologias e formatos vêm sendo propostos para tratar objetos digitais complexos [49]: MPEG-21 (*Moving Picture Experts Group 21*), OAI-ORE (*Open Archive Initiative Object Reuse and Exchange*) [52, 55], METS (*Metadata Encoding and Trans-*

mission Standard) [23], DCC (*Digital Content Components*) [52], Buckets [52], IMS CP (*IMS Content Package*), RAS (*Reusable Asset Specification*) [38] e HTML5 [12, 63]. Um estudo de comparação entre algumas dessas tecnologias, sob a perspectiva do formalismo 5S, foi realizado por Kozievitch *et al.* [52]. Conclusões desse estudo apontam que há incompatibilidades entre algumas dessas tecnologias e formatos, já que cada um foi criado para um domínio específico.

2.4 Formalismo 5S e Objetos Complexos

O conceito formal de objeto complexo não consta no formalismo inicial do 5S definido por Gonçalves *et al.* [42] para descrever uma biblioteca digital mínima, mas foi incluído posteriormente por Kozievitch [50] juntamente com conceitos relacionados à busca de imagens baseada em conteúdo. Os conceitos propostos foram validados em estudos de caso com a ferramenta SuperIDR [59] que gerencia imagens, metadados, marcas e anotações relacionados ao domínio de parasitologia e de ictiologia [62]. Seus principais serviços incluem a navegação, a busca textual, a busca visual e a busca multimodal. Apesar desse trabalho formalizar, especificar e implementar objetos complexos de bibliotecas digitais, a definição e a modelagem dos objetos complexos utilizados não foram validados com o uso e a ajuda de uma ferramenta de especificação e de modelagem de bibliotecas digitais.

Kozievitch [50] também define o formalismo 5S relacionado aos conceitos de agregação de dados, analisando e comparando as tecnologias de objetos complexos: DCCs (*Digital Content Components*), Buckets, e OAI-ORE (*Open Archives Initiative - Object Reuse and Exchange*).

Além da formalização desses conceitos, um protótipo [53] foi criado para gerenciar objetos complexos de imagens com a disponibilização dos serviço de busca de imagens baseada em conteúdo a partir de medidas de similaridade e vetores de características. O acesso aos componentes individuais e ao objeto complexo como entidade única foi implementado. Os dados também são publicados como itens de um repositório de dados, usando o protocolo OAI-PMH¹ (*Open Archives Initiative Protocol for Metadata Harvesting*).

A definição de um **objeto complexo** [51] é uma tupla $cdo = (h, SCDO, S)$, na qual:

1. $h \in H$, sendo H um conjunto universal de identificadores únicos (*labels*);
2. $SCDO = \{DO \cup SM\}$, sendo $DO = \{do_1, do_2, \dots, do_n\}$, em que do_i é um objeto digital ou um outro objeto complexo; e $SM = \{sm_a, sm_b, \dots, sm_z\}$ é um conjunto de *streams*;

¹<http://www.openarchives.org/pmh/> (último acesso em 20/01/2012).

3. S é uma estrutura que compõe o objeto complexo cdo em suas partes em $SCDO$.

Os metadados dos objetos complexos são armazenados em um catálogo separado, assim como definido para os metadados dos objetos simples em [40]. Por isso, eles não constam da definição de objeto complexo.

A estrutura S no objeto complexo pode ser qualquer estrutura que representa as partes de um todo, como uma lista, uma árvore, ou um grafo. Como exemplo há o gerenciador Fedora [9] que utiliza grafos para representar os relacionamentos entre objetos digitais.

A definição de um **objeto complexo mínimo** [50] é uma tupla $cdo = (h, SCDO, S)$, na qual:

1. $h \in H$, sendo H um conjunto universal de identificadores únicos (*labels*);
2. $SCDO = \{DO \cup SM\}$, sendo $DO = \{do_1\}$, em que do_1 é um objeto digital; e $SM = \{sm_1, sm_2, \dots, sm_n\}$ um conjunto de *streams*;
3. S é uma estrutura que indica que $\{do_1\}$ é um componente de cdo .

As subseções a seguir apresentam exemplos de objetos complexos.

2.4.1 Documento Sobreposto

Esta subseção descreve o documento sobreposto (*superimposed document*), um tipo de objeto complexo. Em [59], para definir formalmente uma biblioteca digital contendo *superimposed information*, os seguintes tipos de objetos digitais são definidos (ver Figura 2.4):

- documento base (*base document*): informação completa de um documento; um objeto digital torna-se um documento base quando um subdocumento é criado;
- subdocumento (*subdocument*): parte de um documento base, definido por um endereço de início e fim do documento base; herda as especificações dos metadados descritivos e estruturais definidas para o endereço do documento base;
- documento sobreposto (*superimposed document*): um documento que consiste de um ou mais subdocumentos e de outras informações associadas.

Os eventos temporais são expressos por: $t_{BD} < t_{sd} \leq t_{sidoc}$, em que:

- t_{BD} representa o tempo de criação de um documento base BD;
- t_{sd} representa o tempo de criação de um subdocumento sb no documento base BD;

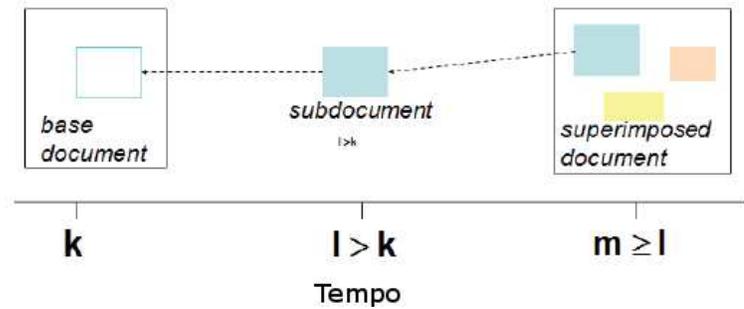


Figura 2.4: Objetos digitais de uma biblioteca digital contendo *superimposed information*. Fonte [59].

- t_{sidoc} representa o tempo de criação de um documento sobreposto que contém o subdocumento sb.

A Figura 2.5 exemplifica um subdocumento. À esquerda, o documento base é exibido com identificador hBD , título *enhance_cmaps.pdf* e um subdocumento destacado que é associado a uma especificação de apresentação PS. Essa especificação contém o tipo de conteúdo, o formato e o *software* a ser utilizado para exibir o subdocumento. À direita, a figura apresenta os detalhes dos componentes do subdocumento sd , incluindo um identificador hsd , uma *substream* $sm_{sd}[i, j]$, uma subestrutura st_{sd} e um endereço $addr$. $sm_{sd}[i, j]$ é a sequência de caracteres do subdocumento. Os índices i e j indicam os números do primeiro e do último caracteres da *substream* dentro do documento base. st_{sd} mostra o mapeamento entre a especificação dos metadados estruturais do documento base e as *streams* dentro do subdocumento. $addr$ indica ponteiros para o início e o fim do subdocumento, considerando a especificação da apresentação PS do documento base.

A definição de um subdocumento como objeto digital, utilizando o formalismo 5S, é apresentada em [59]. Assim, um documento sobreposto, por conter pelo menos um subdocumento, que é um objeto digital, é representado como um objeto complexo.

Uma biblioteca digital com informação sobreposta (SI-DL) gerencia um repositório de coleções de: documentos base, subdocumentos e documentos sobrepostos; e fornece os serviços de indexação, busca, navegação e de visão no contexto (*view in context*). Esse último serviço foi definido por Murthy em [59] e é representado pela navegação intra documentos. Por exemplo, dado um subdocumento, a apresentação/visão do documento base pode ser acionada e o conteúdo do documento base é recuperado e apresentado com o subdocumento destacado dentro do documento base.

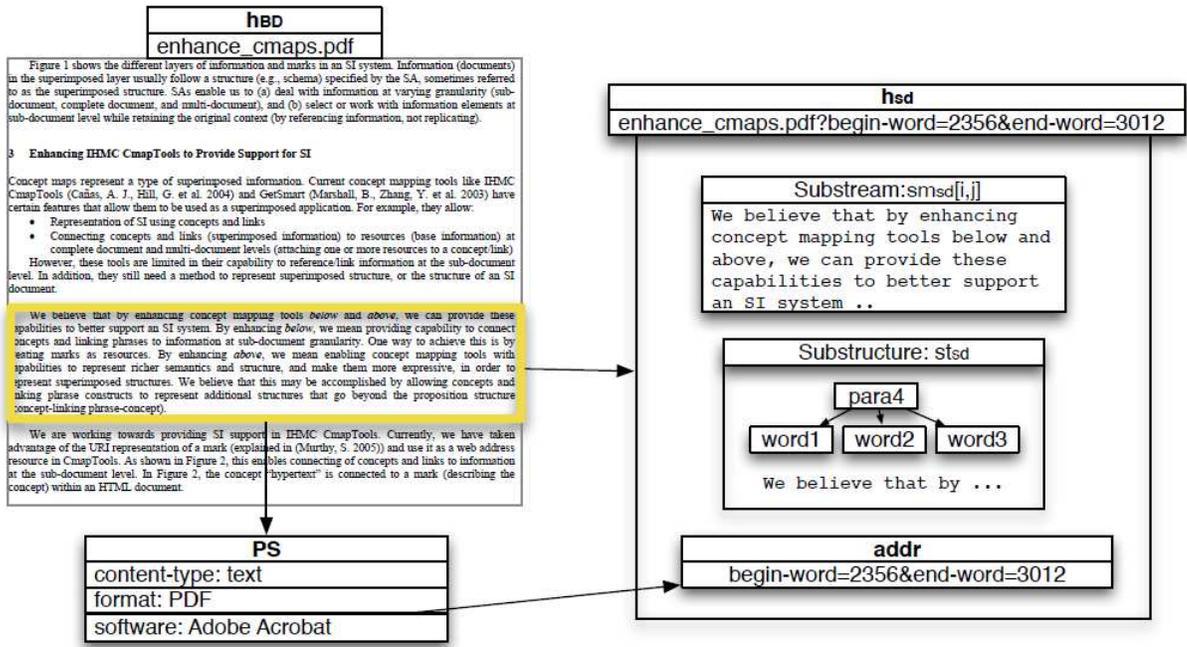


Figura 2.5: Exemplo de um subdocumento. Fonte [59].

2.4.2 Descritor de Imagem

Um dos principais componentes de um sistema de recuperação de imagens por conteúdo é o descritor de imagens, que é caracterizado por duas funções [50]:

- extração das características da imagem em vetores de características; e
- computação das distâncias entre imagens (análise dos vetores de características) com o uso de uma medida de similaridade.

Segue abaixo a definição formal usando o formalismo 5S [51].

Um **vetor de características** $\vec{f}v_i$ de uma imagem \hat{I} é um ponto no espaço R^n : $\vec{f}v_i = (fv_1, fv_2, \dots, fv_n)$, em que n é a dimensão do vetor.

Um **descritor de imagem** D é definido como uma tupla $(h_{desc}, \epsilon_D, \delta_D)$, em que:

- $h_{desc} \in H$, sendo H um conjunto universal de identificadores únicos (*labels*);
- $\epsilon_D : \{\hat{I}\} \rightarrow R^n$ é uma função que extrai um *vetor de características* $\vec{f}v_i$ de uma *imagem* \hat{I} ;

- $\delta_D : R^n \times R^n \rightarrow R$ é uma *função de similaridade* (por exemplo, baseada na distância métrica) que computa a similaridade entre duas imagens como uma função da distância entre os correspondentes *vetores de características*.

Nessa definição formal, o tipo de objeto digital que representa as funções citadas não é apresentado. Portanto, com o intuito de caracterizar o descritor de imagem como objeto complexo, pode-se utilizar o objeto digital de software para representar as funções de um descritor de imagem. O conceito de objeto digital de software foi definido por Pedronette [64]:

Um **objeto digital de software** $ObjDigSoft = (h_{desc}, C_{Mt}, C_{Md}, C_{Rep}, C_{Cn})$ é um objeto digital que atende às seguintes extensões e restrições:

1. $h_{desc} \in H$, sendo H um conjunto universal de identificadores únicos (*labels*);
2. $C_{Mt} = \{mt_1, mt_2, \dots, mt_n\}$ é um conjunto de métodos de software (ver Def. 2 e 3 em [64]);
3. $C_{Md} = \{md_1, md_2, \dots, md_n\}$ é um conjunto de especificações de metadados descritivos (ver Def. 12 em [40]), responsáveis por armazenar informações sobre o *software*, interfaces e parâmetros de configuração;
4. $C_{Rep} = \{R_1, R_2, \dots, R_n\}$ é um conjunto de repositórios (ver Def. 19 em [40]) acessados pelos métodos de software do conjunto C_{Mt} , para armazenamento ou recuperação de informações;
5. $C_{Cn} = \{cn_1, cn_2, \dots, cn_n\}$ é um conjunto de cenários implementados pelos métodos de software do elemento C_{Mt} , e $C_{Cn} \subset Serv$, em que $Serv$ é um serviço definido total ou parcialmente pelo conjunto C_{Cn} .

Um exemplo de gerenciamento de descritores de imagem é observado na ferramenta Eva [65], que avalia esses objetos complexos para a recuperação de imagem baseada em conteúdo.

2.5 Documentos Legais

Esta seção apresenta tópicos relacionados a um outro tipo de objeto complexo, o documento legal. A subseção 2.5.1 define os termos jurídicos utilizados nesta dissertação; a subseção 2.5.2 relata a estrutura de um texto normativo; a subseção 2.5.3 cita o ciclo de vida de uma lei e a subseção 2.5.4 apresenta os relacionamentos entre documentos legais.

2.5.1 Definições

Algumas definições [33] são listadas abaixo para facilitar o entendimento dos termos jurídicos no decorrer do texto:

Documento Legal é o documento emanado pelo poder público e que produz efeitos de ordem jurídica. Exemplos: Constituição Federal, Leis, Decretos, Deliberações e Portarias.

Lei é definida como norma, preceito ou regra, estabelecida com a finalidade de nortear as relações sociais de um país ou de uma instituição.

Nesta dissertação, os termos *texto normativo* e *texto legal* são utilizados para referenciar o texto de um documento legal.

2.5.2 Estrutura

A elaboração, a redação, a alteração e a consolidação das leis e demais atos normativos devem seguir a estrutura definida pela Lei Complementar n° 95/1998 [14] que foi alterada pela Lei Complementar n° 107/2001 [13]. Essas Leis Complementares definem os elementos de uma lei e os possíveis relacionamentos entre eles. De acordo com o 3° artigo da Lei Complementar n° 95/1998:

“Art. 3° - A lei será estruturada em três partes básicas:

I - parte preliminar, compreendendo a epígrafe, a ementa, o preâmbulo, o enunciado do objeto e a indicação do âmbito de aplicação das disposições normativas;

II - parte normativa, compreendendo o texto das normas de conteúdo substantivo relacionadas com a matéria regulada;

III - parte final, compreendendo as disposições pertinentes às medidas necessárias à implementação das normas de conteúdo substantivo, às disposições transitórias, se for o caso, a cláusula de vigência e a cláusula de revogação, quando couber.”

A Figura 2.6 indica esse particionamento na Lei n° 12.089.

Como forma de melhorar a articulação do texto de normas, é possível utilizar dispositivos para agrupamentos de artigos (Parte, Livro, Título, Capítulo, Seção e Subseção). O agrupamento de artigos poderá constituir Subseções; o de Subseções, a Seção; o de Seções, o Capítulo; o de Capítulos, o Título; o de Títulos, o Livro e o de Livros, a Parte. Essa sistematização é criada segundo critérios que visam à organização lógica e hierárquica da matéria normativa.

O artigo é a unidade básica de articulação e contém no seu *caput* a regra normativa propriamente dita. Os parágrafos servem para expressar os aspectos complementares bem

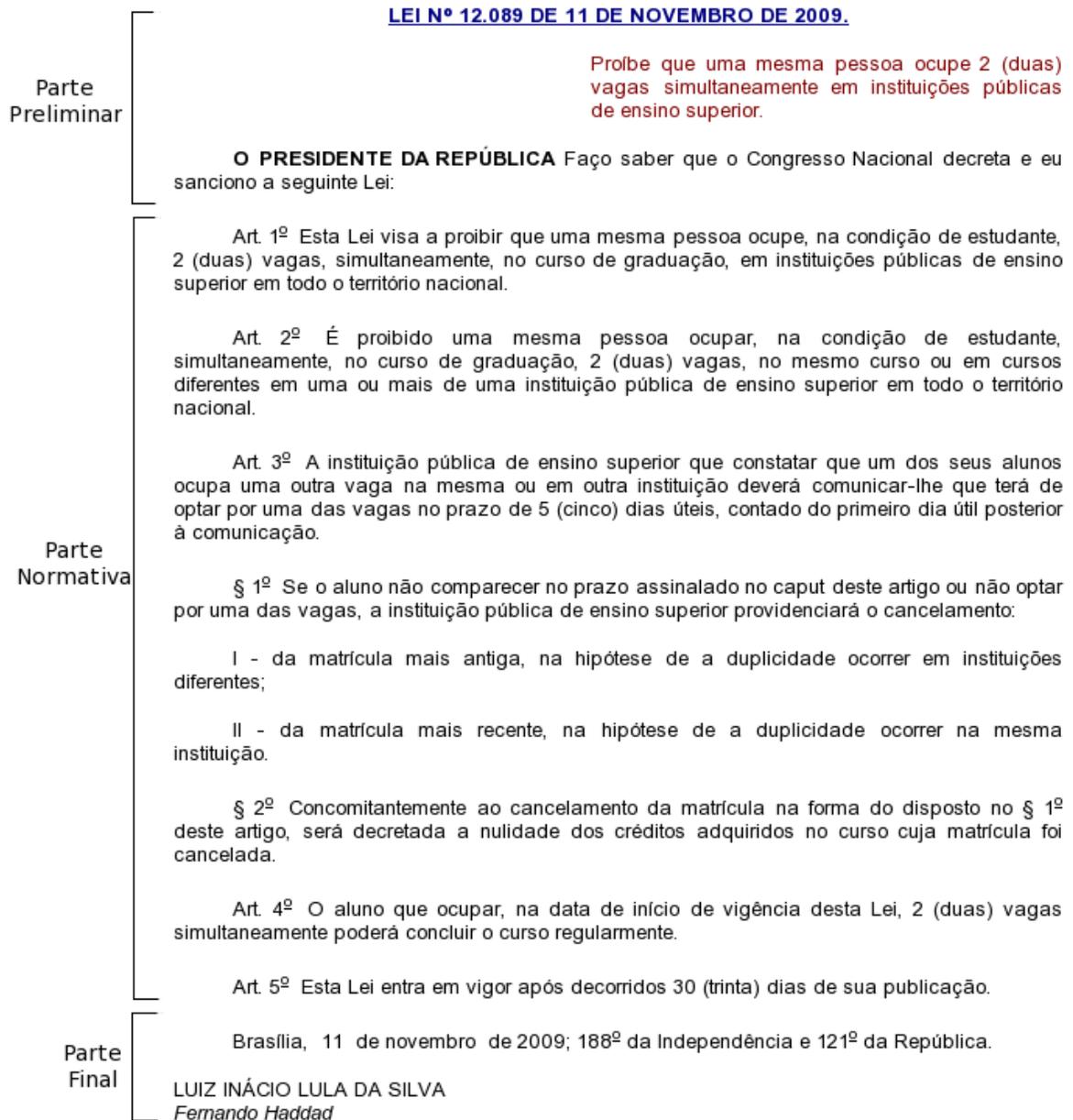


Figura 2.6: Exemplo da divisão da estrutura de uma lei em três partes básicas.

como as exceções à regra. As discriminações e enumerações necessárias ao *caput* e aos parágrafos são feitas de forma hierárquica, utilizando incisos, alíneas e itens [16]. Um artigo pode ser composto por Parágrafos ou Incisos; um Parágrafo por Incisos; um Inciso por Alíneas; e uma Alínea por Itens.

Para melhor entendimento e visualização, a Figura 2.7 especifica a composição dos elementos da parte normativa.

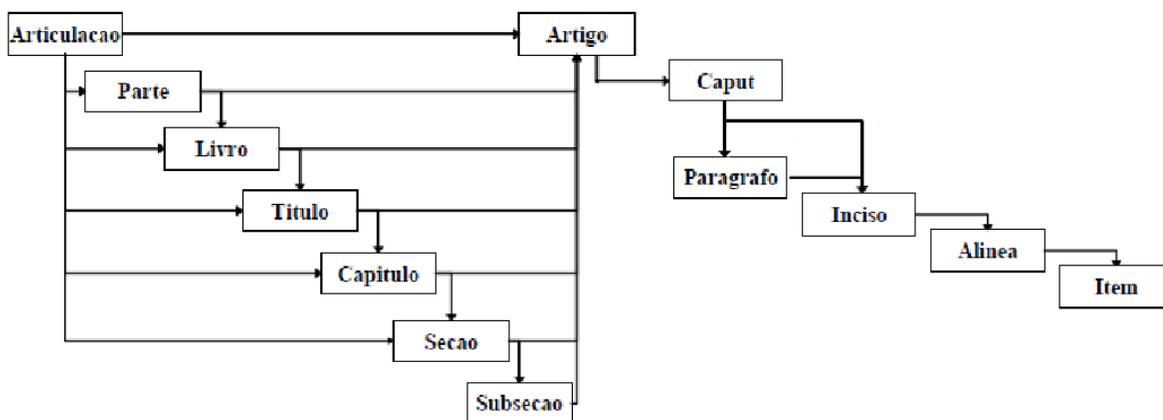


Figura 2.7: Composição dos elementos da parte normativa de uma lei. Adaptado de [16].

2.5.3 Criação e Publicação

A seguir, são descritos os processos de criação e publicação de uma norma jurídica juntamente com a evolução da norma no tempo [16].

Uma norma (obra) é criada no momento da sua assinatura pela autoridade competente, gerando o documento original assinado. No entanto, é só com a promulgação e publicação que a norma passa a ser executória. Cada nova publicação é considerada um exemplar. Uma nova publicação pode afetar o conteúdo de uma norma, tanto em relação a sua forma (expressão textual) quanto em relação a sua matéria (teor normativo). A expressão textual é originada a partir do original de documentos oficiais; a expressão normativa de uma norma pode ser afetada por ações de inclusão, alteração ou revogação de dispositivos. Além disso, não há limite para a quantidade de alterações que determinada lei pode sofrer por meio de leis alteradoras.

Por tudo isso, uma edição específica de uma publicação oficial agrega várias entidades, como, por exemplo: o conteúdo original promulgado após o processo legislativo; o conteúdo da publicação (ver Figura 2.8); a versão consolidada desta norma em uma determinada data (ver Figuras 2.9 e 2.10 que mostram textos diferentes para o artigo 9º do

Regimento Geral dos Cursos de Graduação para períodos de datas distintos); a versão traduzida desta norma para outra língua, *etc.*

Deliberação CONSU-A-027/2002, de 17/12/2002

Reitor: Carlos Henrique de Brito Cruz
Secretária Geral: Patrícia Maria Morato Lopes Romano

Dispõe sobre alteração da [Deliberação CONSU-A-011/1998](#), que baixou o Manual do Aluno.

O Reitor da Universidade Estadual de Campinas, na qualidade de Presidente do Conselho Universitário, tendo em vista o decidido pelo Conselho em sua 5ª Sessão Extraordinária de 2002, realizada em 17-12-02, baixa a seguinte deliberação:

Artigo 1º - Fica suprimido o § 2º do Artigo 55 e alterada a redação do caput do Artigo 9º e seus incisos I e II que passam a vigorar com a seguinte redação.

"Artigo 9º - O graduado em curso da Unicamp poderá requerer, nos períodos de matrícula, o retorno para complementar, mediante aprovação das Coordenadorias dos Cursos:

I - Outra habilitação ou modalidade do curso em que se graduou ainda que envolva períodos distintos desse mesmo curso.

II - Outro curso, desde que tenha ingresso vinculado ao mesmo curso no qual se graduou."

Artigo 2º - Essa Deliberação entrará em vigor na data de sua publicação, revogadas as disposições em contrário.

Publicada no DOE em 21/12/2002

Figura 2.8: Conteúdo da publicação da Deliberação CONSU-A-027/2002. Fonte [3].

2.5.4 Relacionamentos

Os seguintes relacionamentos (a maioria é identificada explicitamente no texto legal) podem ocorrer entre documentos legais [27]:

- Citações: um texto normativo pode citar outras normas. Exemplo: a Lei Federal nº 11.705 cita a Lei Federal nº 9.294. Essas citações são facilmente convertidas em *hiperlinks*;
- Dependências Diretas: citação para revogar o texto publicado, dando origem ao chamado "texto compilado" da norma original;
- Dependências Hierárquicas: usualmente normas mais específicas e detalhadas regulamentam as normas mais gerais. Essa é uma relação típica entre, por exemplo, as Leis Federais e a Constituição da República;

Artigo 9º - O graduado em curso da UNICAMP poderá, a qualquer momento, nos períodos de matrícula, retornar para complementar outra habilitação ou modalidade do curso em que se graduou, mediante aprovação da Coordenadoria do Curso.

§ 1º - Ao aceitar o reingresso de um aluno para cumprir habilitação ou modalidade adicional, a Coordenadoria do Curso indicará o ano do catálogo a ser seguido pelo aluno.

§ 2º - O aluno reingressante terá direito ao limite máximo de trancamentos previsto no Artigo 46, descontados os já obtidos anteriormente ao ingresso para complementação da nova habilitação ou modalidade.

§ 3º - Ao aluno reingressante aplicam-se todas as disposições do Manual do Aluno.

§ 4º - O aluno reingressante que vier a cancelar sua matrícula não mais poderá reingressar no mesmo curso.

§ 5º - O aluno reingressante que vier a cancelar sua matrícula ou tenha esta cancelada pela UNICAMP, terá transformadas em seu Histórico Escolar as ocorrências referentes ao reingresso para a condição de Estudante Especial.

§ 6º - Concluída a habilitação ou modalidade referente ao retorno, esta será apostilada no diploma do aluno ou lhe será expedido um novo diploma, desde que a nova situação o exija.

Figura 2.9: Conteúdo do artigo 9º do Manual do Aluno no período de 10/08/1998 (CONSU-A-011/1998) a 20/12/2002 (CONSU-A-027/2002).

Artigo 9º - O graduado em curso da Unicamp poderá requerer, nos períodos de matrícula, o retorno para complementar, mediante aprovação das Coordenadorias dos Cursos:

I - Outra habilitação ou modalidade do curso em que se graduou ainda que envolva períodos distintos desse mesmo curso.

II - Outro curso, desde que tenha ingresso vinculado ao mesmo curso no qual se graduou.

§ 1º - Ao aceitar o reingresso de um aluno para cumprir habilitação ou modalidade adicional, a Coordenadoria do Curso indicará o ano do catálogo a ser seguido pelo aluno.

§ 2º - O aluno reingressante terá direito ao limite máximo de trancamentos previsto no Artigo 46, descontados os já obtidos anteriormente ao ingresso para complementação da nova habilitação ou modalidade.

§ 3º - Ao aluno reingressante aplicam-se todas as disposições do Manual do Aluno.

§ 4º - O aluno reingressante que vier a cancelar sua matrícula não mais poderá reingressar no mesmo curso.

§ 5º - O aluno reingressante que vier a cancelar sua matrícula ou tenha esta cancelada pela UNICAMP, terá transformadas em seu Histórico Escolar as ocorrências referentes ao reingresso para a condição de Estudante Especial.

§ 6º - Concluída a habilitação ou modalidade referente ao retorno, esta será apostilada no diploma do aluno ou lhe será expedido um novo diploma, desde que a nova situação o exija.

Figura 2.10: Conteúdo do artigo 9º do Manual do Aluno no período de 21/12/2002 (CONSU-A-027/2002) a 16/12/2005 (CONSU-A-024/2005).

- Vinculação por Assunto: vários documentos legais podem decorrer sobre um mesmo assunto. Sobre o assunto “Ciência e Tecnologia” há como exemplos, a Lei de Biossegurança e a Lei Arouca (sobre procedimentos para o uso científico de animais);
- Relações Espaciais e Temporais: respectivamente, definem em quais locais vigoram as normas e pontuam quais são as normas que estão vigentes em uma determinada data ou em um período de datas.

2.6 Projetos/Aplicações de Documentos Legais

O principal projeto de referência sobre documentos legais deste trabalho é um projeto do governo brasileiro denominado LexML [27]. Esse projeto é uma iniciativa de diversos órgãos nacionais, incluindo os Poderes Executivo, Legislativo e Judiciário, a Advocacia Geral da União e o Ministério Público, em busca do estabelecimento de padrões abertos, da integração de processos de trabalho e do compartilhamento de dados de interesse comum, com foco em informações legislativas e jurídicas.

O projeto possui os seguintes objetivos: identificar, de forma unívoca e persistente, os recursos de informação legislativa e jurídica; e estruturar o conteúdo dos documentos utilizando o formato XML. As seguintes categorias de documentos compõem o escopo inicial: proposições legislativas; normas jurídicas; jurisprudência (acórdãos, súmulas e decisões monocráticas).

Esse projeto é baseado nas experiências bem sucedidas dos projetos:

- *Norme in Rete* [25], da Itália;
- *Akoma Ntoso* [1], da África, e
- *MetaLex CEN* [2], da Europa.

O projeto *Norme in Rete* consiste nas definições de: um identificador uniforme, com o uso de URN (*Uniform Resource Name*); um formato para estruturar o inteiro teor de normas, utilizando DTD¹ e XML Schema²; e um portal para a resolução de identificadores, traduzindo URN em URL. Além disso, as seguintes ferramentas foram implementadas pelo subprojeto XMLeges [32]: um editor de legislações, uma ferramenta para localizar referências em textos normativos e um extrator de semântica de documentos.

O *Akoma Ntoso* é um projeto promovido pelas Nações Unidas e realizado com o apoio de pesquisadores da Universidade de Bolonha com o objetivo de capacitar os parlamentos da África com novas tecnologias de informação e comunicação.

¹<http://www.w3schools.com/dtd/> (último acesso em 20/01/2012).

²<http://www.w3schools.com/schema/> (último acesso em 20/01/2012).

O Centro Europeu de Normatização (CEN) define um metamodelo que servirá de denominador comum em relação aos diversos esquemas nacionais europeus. Outro projeto europeu relacionado é o Projeto ESTRELLA [8, 35] que tem como objetivo desenvolver: a linguagem *Legal Knowledge Interchange Format* (LKIF) construída em padrões de Web Semântica baseados em XML, incluindo RDF e OWL; e aplicações para interação com outros sistemas baseados em conhecimento legal.

Como resultado do projeto LexML Brasil, é apresentado o Portal LexML [22] com acesso gratuito dos cidadãos a consultas de normas e de julgados nas esferas federal, estadual e municipal, publicados por órgãos ou por entidades da Administração Direta ou Indireta. É composto de uma infraestrutura que permite manipular eficazmente a gigantesca quantidade de informações existentes no país.

A seguir, algumas características do projeto LexML são detalhadas:

- modelo de referência [16]: o modelo conceitual de classes utilizado pelo sistema é baseado na ontologia FRBR_{OO} (ver Subseção 2.7.2), podendo identificar as várias entidades de uma norma (documento original assinado, documento publicado, texto consolidado, *etc.*);
- metadados: autoridade emitente, tipo do documento, identificador, data representativa (referente à data da assinatura), data da versão (referente à data de início de vigência de uma nova norma ou de sua alteração por uma outra norma), data da visão (referente à data de um evento que gere uma nova versão ou uma variante de uma versão), contexto, ciclo de vida, eventos gerados, notas e recursos.
- identificador LeXML unívoco de nomes para normas [17]: a gramática para formação de nomes uniformes de Normas, Julgados e Projetos de Normas utiliza a notação URN; definido para não gerar problemas de *links* perdidos; é composto por informações do documento;
- estrutura dos textos normativos [18]: é definida através de XML Schema e segue a técnica legislativa nacional;
- coleta de metadados [19]: realizada para reunir os metadados de documentos legislativos e jurídicos disponíveis nos sítios dos diversos órgãos governamentais; utiliza o protocolo OAI-PMH;
- serviço de resolução de URN [20]: *parser* para identificação de documentos a partir de URNs;
- vocabulários controlados [21]: identificam acrônimos que indicam os órgãos do governo, localidades, autoridades emitentes e tipos de documentos.

A Subseção 2.6.1 lista as tecnologias utilizadas por alguns países na implementação desse tipo de sistema e a Subseção 2.6.2 mostra como algumas universidades estão disponibilizando sua legislação para a comunidade universitária.

2.6.1 Governos

Vários outros governos possuem um sistema jurídico informatizado, ocasionando uma rápida solução para os processos legislativos e jurídicos.

Tabela 2.1: Sistemas governamentais de alguns países que armazenam e pesquisam documentos legais.

Sistemas	Documentos	Metadados/ Ontologias Formais	Tecnologias	Serviços
Portal LexML	Legislação, Jurisprudência, Proposições Legislativas	IFLA FRBR ¹	HTML, URN, XML Schema, RDF ² , OAI-PMH	Busca por termos, por filtros.
				Navegação para outros sites do governo que publicam leis.
				Interoperabilidade permitindo inclusão de novos documentos indexados na busca.
<i>Norme in Rete</i> (Itália)	Legislação, Jurisprudência	IFLA FRBR	URN, URL, DTD, XML Schema	Busca, Navegação.
<i>Akoma Ntoso</i> (África)	Legislação, Gravação de debates, Minutas	Dublin Core, IFLA FRBR, FOAF ³	HTML, DTD, XML Schema	Busca, Navegação.
MetaLex CEN (Europa)	Legislação	IFLA FRBR	IRI, URN, URL, RDF, XML Schema, OWL Schema ⁴	Busca por termos, por filtros, Navegação, Interoperabilidade entre países europeus.

¹<http://www.ifa.org/publications/functional-requirements-for-bibliographic-records> (último acesso em 20/01/2012)

²<http://www.w3.org/RDF/> (último acesso em 20/01/2012).

³<http://xmlns.com/foaf/spec/> (último acesso em 20/01/2012).

⁴<http://www.w3.org/TR/owl-features/> (último acesso em 20/01/2012).

A Tabela 2.1 lista sistemas de alguns países indicando também as tecnologias e os serviços utilizados por cada um. Além desses sistemas, outros projetos internacionais são listados em [15] e podem ser consultados.

2.6.2 Universidades

A Tabela 2.2 lista os documentos diretivos e os serviços relacionados a esses documentos que são disponibilizados por algumas universidades públicas nacionais.

Tabela 2.2: Documentos diretivos de universidades públicas e serviços relacionados a esses documentos.

Universidade	Documentos Diretivos	Tipos de Arquivos	Serviços	Observações
UNICAMP	Deliberações [5], Portarias [26], Resoluções [30]	HTML	Busca, Navegação	Busca por termos.
	Estatutos [7], Regimento [29]	PDF	-	-
UNESP	Artigos de Periódicos, Estatutos, Portarias, Regimento, Resoluções [11]	HTML	Busca, Navegação	Busca por: tipo de documento, termos, local do documento (cabeçalho ou corpo); exibição de número máximo de registros.
USP	Decretos, Estatutos, Portarias, Regimentos, Resoluções [24]	HTML	Busca, Navegação	Busca por termos.
UFSCar	Estatuto, Regimento [28], Pareceres, Atos Normativos e Resoluções [4]	PDF	-	-

2.7 Ontologias Formais

O tratamento de documentos legais requisita o uso de ontologias para otimizar os serviços da biblioteca digital.

2.7.1 Definição

Uma ontologia é uma especificação explícita de uma conceitualização; descreve um modelo abstrato de termos, relacionados entre si [45], visando definir um entendimento comum sobre um domínio. Dessa forma, são utilizadas para que pessoas de um mesmo domínio

compartilhem e reuam o mesmo vocabulário, garantindo a consistência de consultas e asserções. Ontologias são representadas por uma linguagem de representação processada por computador, como a *Web Ontology Language* (OWL) [31], publicada pelo *World Wide Web Consortium* (W3C), que é uma comunidade internacional que desenvolve padrões com o objetivo de garantir o crescimento da *Web*.

2.7.2 FRBR_{OO}

Vários sistemas legislativos adotaram a ontologia formal *Functional Requirements for Bibliographic Records Object Oriented* (FRBR_{OO}) [46], que captura e representa semânticas de registros bibliográficos e de dados de museus, facilitando a integração e a troca dessas informações. Por ser orientada a objetos, é aplicável em diferentes especificações funcionais e em diferentes ambientes. Essa ontologia foi criada a partir do modelo *Functional Requirements for Bibliographic Records* (FRBR) e do padrão *Comité international de documentation Conceptual Reference Model* (CIDOC CRM) (ISO 21.127:2006), por um grupo de estudo internacional formado pela *International Federation of Library Associations and Institutions* (IFLA) e pelo *International Council of Museums* (ICOM).

As classes/entidades e propriedades do modelo FRBR_{OO} são identificadas pelas letras “F” e “R”, respectivamente, seguidas de um número e de um nome. Propriedades identificam os relacionamentos entre as classes.

Nesse modelo, para representar uma obra (instância de ‘*F1 Work*’), a forma de expressão dessa obra (‘*F2 Expression*’) é manifestada em um original (‘*F4 Manifestation Singleton*’). A partir desse original, o processo de produção produz diversos exemplares (‘*F5 Item*’), onde um exemplar de uma publicação oficial é uma instância da classe ‘*F5 Item*’ e possui uma expressão de publicação (‘*F24 Publication Expression*’) [16].

Normalmente, o texto publicado reflete o texto do original. No entanto, por problemas no processo de publicação, podem ocorrer inconsistências entre o texto do original e o texto publicado. Nestes casos, a publicação de uma comunicação oficial é realizada com as devidas retificações.

Cada entidade da norma pode ser modelada como uma instância da classe ‘*F14 Individual Work*’ com a respectiva instância da classe ‘*F22 Self Contained Expression*’.

Na publicação de uma norma, é criada uma instância da classe ‘*F15 Complex Work*’ com o objetivo de referenciar as duas instâncias da classe ‘*F14 Individual Work*’ (uma referente ao documento original assinado e a outra referente ao documento oficial publicado).

Quando é publicada uma norma retificadora, no momento da entrada em vigor da norma que altera outra já existente, cria-se uma obra derivada que é representada por uma nova instância da classe ‘*F14 Individual Work*’ com a respectiva instância da classe

'F22 Self Contained Expression'. Essa dinâmica é ilustrada pela Figura 2.11 que mostra a modificação da "Lei 1" após a publicação da "Lei 8".

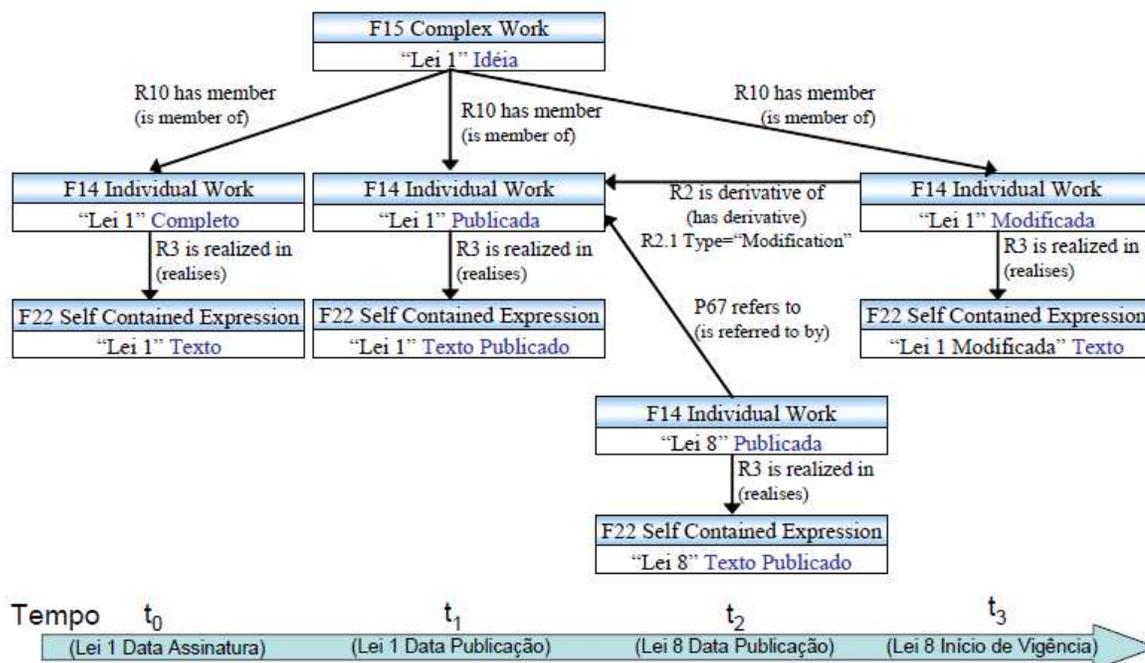


Figura 2.11: Evolução da norma no tempo. Fonte [16].

A Tabela 2.3 especifica as classes do modelo FRBR_{OO} e as relaciona com instâncias de normas jurídicas.

Além de normas jurídicas, resultados do processo legislativo, outros documentos de interesse jurídico, como julgados e projetos de normas, também podem ser modelados com o uso desse modelo.

O capítulo a seguir detalha o documento legal como um objeto complexo.

Tabela 2.3: Artefatos jurídicos correlacionados às classes do modelo FRBR_{OO}.

Classe FRBR _{OO}	Descrição da classe FRBR _{OO} [16, 46]	Instância (norma) [16]
<i>F1 Work</i>	Representa o conteúdo conceitual de uma obra; criação artística ou intelectual distinta.	Norma jurídica; deliberação; portaria.
<i>F2 Expression</i>	É a representação de uma obra ou a forma de expressão de um objeto físico: texto; fala; poema; tradução para outra língua.	Texto.
<i>F3 Manifestation Product Type</i>	Define as características dos objetos que serão produtos de um específico processo de produção ou publicação (<i>'F5 Item'</i>).	Tipo de publicação do produto “Diário Oficial da União - Seção 1”, publicado em 4 de Outubro de 2010 pela Imprensa Nacional.
<i>F4 Manifestation Singleton</i>	Objeto físico único de uma expressão (<i>'F2 Expression'</i>).	O documento original assinado (artefato).
<i>F5 Item</i>	Exemplar específico de uma manifestação.	Exemplar da Publicação Oficial.
<i>F14 Individual Work</i>	Especialização de <i>'F1 Work'</i> . Representa os conceitos associados com um conjunto específico de símbolos realizados em uma <i>'F22 Self Contained Expression'</i> .	Conceitos associados com o documento original (criação) ou com a norma publicada (publicação).
<i>F15 Complex Work</i>	Especialização de <i>'F1 Work'</i> . Representa um conjunto que possui outras instâncias de <i>'F1 Work'</i> como membros. Serve para representar a evolução de uma obra no tempo e/ou para representar obras que são compostas por outras obras.	Relaciona o documento original e a norma publicada.
<i>F17 Aggregation Work</i>	Agregação e/ou rearranjo de expressões de outros trabalhos.	Diário Oficial “Completo”.
<i>F18 Serial Work</i>	Comprime trabalhos que resultam em sequências de manifestações com propriedades comuns.	Diário Oficial “Idéia”.
<i>F19 Publication Work</i>	Especialização de <i>'F1 Work'</i> . Representa obras que foram planejadas para resultar em um <i>'Manifestation Product Type'</i> (tipo de produto) e que veicula expressões de outras obras.	Diário Oficial “Completo”.
<i>F22 Self Contained Expression</i>	Especialização de <i>'F2 Expression'</i> . Representa a realização de <i>'F14 Individual Work'</i> em um determinado tempo.	Texto normativo resultante do processo legislativo (criação) ou texto da norma publicada (publicação).
<i>F24 Publication Expression</i>	Especialização de <i>'F22 Self Contained Expression'</i> . Representa o <i>layout</i> e conteúdo completo fornecido pelo editor para publicação; pode conter elementos adicionais (além das expressões das obras veiculadas) tais como tabelas de conteúdo (sumário), cabeçalhos e notas do editor.	O conteúdo da página publicada na internet com o conteúdo referente ao texto de uma norma.

Capítulo 3

Elementos Legais como Objetos Complexos

Este capítulo demonstra como os dispositivos legais, um texto normativo e um documento legal podem ser caracterizados como objetos digitais complexos e lista alguns serviços essenciais para esses tipos de objetos digitais.

3.1 Dispositivos

Os dispositivos de um texto legal podem ser definidos como objeto digital atômico, pois possuem: um identificador único (*label*); o texto (*streams*); a estrutura textual definida pelas Leis Complementares (*Structural Metadata Specifications*); e o texto associado à estrutura textual (*StructuredStreams*).

Entretanto, quando existem dispositivos filhos no texto (por exemplo, um parágrafo possui incisos), os dispositivos (no exemplo, o parágrafo) podem ser considerados objetos complexos, pois compõem-se de objetos digitais, que são os dispositivos filhos (os incisos).

Tabela 3.1: Objeto digital simples do Inciso I do Artigo 9° sendo alterado pela Deliberação CONSU-A-027/2002 (ver também Figura 3.1).

Objeto Digital	Descrição
Identificador único	art1_cpt_asp1_art9_incl
<i>Stream</i>	<i>Stream</i> referente ao <i>Caput</i> do Inciso I: “Outra habilitação ou modalidade do curso em que se graduou ainda que envolva períodos distintos desse mesmo curso.”
<i>Structural Metadata Specifications</i>	Lei Complementar n° 95: “os incisos serão representados por algarismos romanos”

Tabela 3.2: Objeto digital complexo referente à alteração do Artigo 9º do Manual do Aluno na Deliberação CONSU-A-027/2002.

Objeto Digital	Descrição
Identificador único	art1_cpt_asp1_art9
Stream e Objetos Digitais	<i>Stream</i> referente ao <i>Caput</i> do Artigo 9º: “O graduado em curso da Unicamp ... mediante aprovação das Coordenadorias dos Cursos:” Objeto digital simples referente à alteração do Inciso I (Tabela 3.1) Objeto digital simples referente à alteração do Inciso II
<i>Structural Metadata Specifications</i>	Lei Complementar nº 95: “os artigos desdobrar-se-ão em parágrafos ou em incisos”

Tabela 3.3: Objeto digital complexo do Artigo 1º da Deliberação CONSU-A-027/2002.

Objeto Digital	Descrição
Identificador único	art1
Streams e Objetos Digitais	<i>Stream</i> referente ao <i>Caput</i> do Artigo 1º: “Fica suprimido o ... Artigo 9º e seus incisos I e II que passam a vigorar com a seguinte redação. Objeto digital complexo referente à alteração do Artigo 9º (Tabela 3.2)
<i>Structural Metadata Specifications</i>	Lei Complementar nº 95: “os artigos desdobrar-se-ão em parágrafos ou em incisos; os parágrafos em incisos, os incisos em alíneas e as alíneas em itens”

As Tabelas 3.1, 3.2 e 3.3 apresentam os objetos digitais referentes aos dispositivos da Deliberação CONSU-A-027/2002, apresentada na Figura 2.8: o objeto digital simples referente ao Inciso I do Artigo 9º, o objeto digital complexo referente ao Artigo 9º sendo alterado e o objeto complexo referente ao Artigo 1º.

Já a Figura 3.1 apresenta visualmente no texto legal a delimitação dos objetos digitais. Os quadrados apresentados em preto indicam os objetos digitais referentes aos incisos do Artigo 9º (comparar com Tabela 3.1); o quadrado em vermelho representa o objeto digital do Artigo 9º sendo alterado (analisar juntamente com a Tabela 3.2); e os quadrados azuis destacam os objetos digitais dos artigos do documento legal (verificar Tabela 3.3 para melhor entendimento). O quadrado verde representa a parte normativa.

3.2 Texto Normativo

Exemplos de textos normativos são listados a seguir: o Manual do Aluno da Unicamp, a Constituição da República Federativa do Brasil, a Lei Áurea, entre outros. Os textos normativos são definidos por publicações de documentos legais. Por exemplo, o Manual

Deliberação CONSU-A-027/2002, de 17/12/2002

Reitor: Carlos Henrique de Brito Cruz
Secretária Geral: Patricia Maria Morato Lopes Romano

Dispõe sobre alteração da [Deliberação CONSU-A-011/1998](#), que baixou o Manual do Aluno.

O Reitor da Universidade Estadual de Campinas, na qualidade de Presidente do Conselho Universitário, tendo em vista o decidido pelo Conselho em sua 5ª Sessão Extraordinária de 2002, realizada em 17-12-02, baixa a seguinte deliberação:

Artigo 1º - Fica suprimido o § 2º do Artigo 55 e alterada a redação do caput do Artigo 9º e seus incisos I e II que passam a vigorar com a seguinte redação:

"Artigo 9º - O graduado em curso da Unicamp poderá requerer, nos períodos de matrícula, o retorno para complementar, mediante aprovação das Coordenadorias dos Cursos:

I - Outra habilitação ou modalidade do curso em que se graduou ainda que envolva períodos distintos desse mesmo curso.

II - Outro curso, desde que tenha ingresso vinculado ao mesmo curso no qual se graduou."

Artigo 2º - Essa Deliberação entrará em vigor na data de sua publicação, revogadas as disposições em contrário.

Publicada no DOE em 21/12/2002

Figura 3.1: Objetos digitais da Deliberação CONSU-A-027/2002.

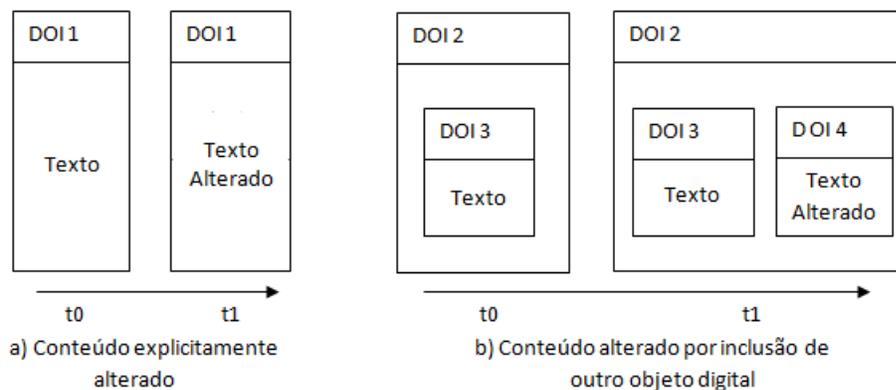


Figura 3.2: Versionamento de um objeto digital: (a) alteração no próprio objeto e (b) alteração ocasionada pela inclusão de novo objeto digital.

do Aluno foi baixado pela Deliberação CONSU-A-011/1998 (documento legal A) e posteriormente alterado por outras Deliberações (documentos legais B, C, *etc.*). Por causa dessas várias alterações, seguem abaixo exemplos de versionamentos de objetos digitais.

Um objeto digital pode ser alterado explicitamente, quando se cria uma nova versão de seu conteúdo. A Figura 3.2a exemplifica o caso de uma alteração explícita no conteúdo do próprio objeto digital, quando, no tempo t_0 , existe o objeto digital original e, no tempo t_1 , existe o mesmo objeto com uma nova versão de seu conteúdo, sendo que $t_0 < t_1$. Entretanto, em objetos digitais complexos, a alteração pode ser ocasionada pela inclusão de outros objetos digitais (ver Figura 3.2b). Esse tipo de alteração ocorre, por exemplo, nos textos normativos, ocasionado pela publicação de um novo documento legal. Por isso, um texto normativo pode possuir conteúdos distintos em determinados períodos de tempo. Assim, quando ocorre a consolidação do texto normativo em uma determinada data, forma-se o texto consolidado. Por exemplo, no cenário de análise de jubramento de um aluno de graduação, checka-se o prazo máximo permitido para a conclusão do curso no Regimento Geral dos Cursos de Graduação. O objeto digital DOI 2 da Figura 3.2b representa o Regimento Geral dos Cursos de Graduação. Hoje, seu texto consolidado, representado pelo DOI 3 da Figura 3.2b diz que o prazo máximo de conclusão do curso é igual ao número de períodos proposto pela Unidade de Ensino para o cumprimento do currículo pleno acrescido de 50%. No futuro, uma nova deliberação, representada pelo DOI 4 da Figura 3.2b, pode alterar o texto do Regimento, alterando o prazo máximo de conclusão dos cursos. Portanto, a data de ingresso do aluno afeta a análise do prazo máximo de conclusão de curso. Com este exemplo, conclui-se que DOI 2 (um texto normativo) é um objeto digital complexo por ser definido por diversos outros objetos digitais, no exemplo, DOI 3 e DOI 4 (documentos legais); além disso, seu conteúdo é definido temporalmente.

3.3 Documento Legal

Um documento legal, ou seja, a publicação de um texto normativo, como por exemplo, a Deliberação CONSU-A-027/2002 (que altera o Manual do Aluno), pode ser definido como objeto complexo, pois possui: um identificador único (exemplo: CONSU-A-027/2002); as partes da lei (que são objetos digitais); o texto legal (*streams*); e uma estrutura definida pelas Leis Complementares n° 95/1998 e n° 107/2001. A Figura 3.1 exibe o objeto complexo referente à Deliberação CONSU-A-027/2002 representado pelo quadrado rosa.

Complementando, uma aplicação que gere documentos legais pode ser comparada a uma aplicação contendo *superimposed information*, conforme descrito na Subseção 2.4.1. Isto porque um documento legal é composto por dispositivos que podem ser definidos como partes (subdocumentos) do documento principal (documento base) e o texto consolidado

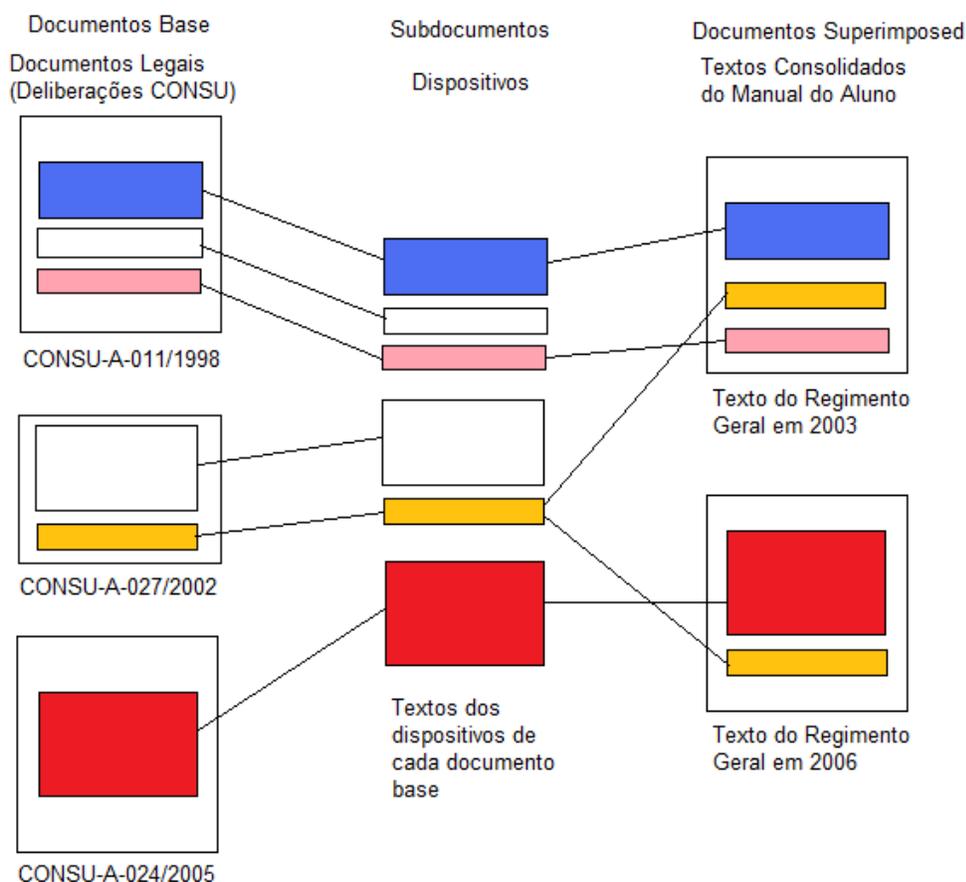


Figura 3.3: Textos legais como *superimposed information*.

de uma lei em uma determinada data é formado por dispositivos dispersos em um ou mais documentos legais publicados, já que uma nova publicação pode alterar apenas alguns artigos, mantendo vigentes os outros artigos não afetados. Por isso, um texto consolidado pode ser comparado a um documento sobreposto (*superimposed*) que é composto por vários subdocumentos. A Figura 3.3 mostra a consolidação de um texto normativo (o Regimento Geral dos Cursos de Graduação ou, como antigamente definido, o Manual do Aluno) em períodos de datas distintas. Assim, o texto do Manual, em 2003, é formado por artigos definidos pela Deliberação CONSU-A-011/1998 e por outros artigos que foram sendo alterados ou suprimidos pelas Deliberações que foram sendo publicadas depois, como a Deliberação CONSU-A-027/2002. Da mesma maneira, o texto do Manual, em 2006, já é outro texto, diferente do texto de 2003, pois foram publicadas novas Deliberações após 2003, como a Deliberação CONSU-A-024/2005, que novamente altera textos e inclui novos artigos no Manual do Aluno.

3.4 Serviços

A seguir, alguns serviços essenciais para uma biblioteca digital de documentos legais são apresentados:

- inserção de documentos legais: necessário para que os documentos legais e os textos normativos sejam armazenados em um modelo de dados eficaz para a realização dos outros serviços. A publicação e a inserção de um novo documento legal ocasiona a criação de um texto normativo ou alterações em um texto normativo, fazendo com que este seja sempre atualizado por novas publicações. Além disso, um documento legal possui vários relacionamentos com outros documentos legais, gerando *hiperlinks* entre os documentos;
- busca: a requisição de um usuário pode ser por um documento legal ou por um texto normativo em uma determinada data; o usuário também pode especificar busca por similaridade (digitando parte do texto normativo), busca estruturada (por indicação explícita de data, apelido da lei, autoridade emitente, *etc.*) ou busca multimodal, que engloba conjuntamente as buscas textual e estrutural;
- navegação: o usuário pode navegar de um documento legal para outro documento legal; de um documento legal para o texto consolidado em uma determinada data; do texto consolidado para um documento legal; de um texto consolidado para outro texto consolidado em outra data (navegação na linha do tempo). Além dessas navegações entre documentos, também pode-se ter o serviço *view in context* [59]. Nesse tipo de serviço, é disponibilizada a navegação intra documentos, ou seja, *links* são criados entre dispositivos que se relacionam na hierarquia de um documento legal. Dessa maneira, se o retorno de uma consulta é um parágrafo, este é apresentado destacado no documento legal ao qual pertence.

Capítulo 4

Especificação de Bibliotecas Digitais de Objetos Complexos

Este capítulo descreve a proposta relacionada à especificação de bibliotecas digitais de objetos complexos.

Para modelar conceitualmente uma biblioteca digital, a linguagem 5SL, uma linguagem declarativa e formal, que utiliza a sintaxe XML, pode ser utilizada. 5SL é a representação da teoria 5S e serve como metamodelo para instanciar modelos de sistemas específicos de bibliotecas digitais [40]. Simplificando, um metamodelo descreve uma biblioteca digital genérica [71], estruturando os conceitos de uma biblioteca digital (definindo os componentes, suas propriedades, estrutura hierárquica, restrições de cardinalidade e de semântica) e exibindo os relacionamentos entre esses conceitos.

Neste trabalho, a linguagem 5SL foi utilizada como base para especificar bibliotecas digitais de objetos complexos. A proposta é analisar mudanças e alterações necessárias no metamodelo atual para que seja possível a especificação de bibliotecas digitais de objetos complexos.

A especificação de uma biblioteca digital envolve a definição do conteúdo a ser armazenado, como esse conteúdo é organizado, estruturado, descrito e acessado; quais serviços são oferecidos pela biblioteca; e como os usuários utilizam e interagem com esses serviços [40]. Portanto, para iniciar a especificação de bibliotecas digitais de objetos complexos, a seguinte questão precisa ser respondida: *Quais modelos definidos pelo formalismo 5S são afetados para especificar uma biblioteca digital de objetos complexos?*

4.1 Especificação do Metamodelo

Primeiramente, é necessário atacar os modelos que definem as *streams* e as estruturas dos objetos utilizados na biblioteca. O metamodelo atual especifica apenas objetos digitais

simples. Como alterá-lo para que objetos complexos possam ser definidos?

As *streams* definidas são muito comuns a qualquer tipo de biblioteca digital, pois tratam da formação de um dado digital. Quanto à estrutura dos objetos, por serem objetos com composição distintas, um objeto atômico, um objeto composto e um objeto complexo devem ser diferenciados. Um objeto composto possui mais de um componente *Stream* em relação a um objeto atômico e esses objetos digitais são referenciados pelo componente *Document* no metamodelo atual. Esse componente foi renomeado para *SimpleObject* no novo metamodelo para que a mesma nomenclatura dos conceitos até então apresentada seja utilizada. A diferença dos objetos simples para os objetos complexos é indicada pelo novo conceito/componente *ComplexObject*, que representa um objeto complexo. Dessa maneira, a coleção da biblioteca digital é alterada para conter objetos complexos, além de objetos atômicos e compostos.

Sendo imprescindível que um objeto complexo contenha ao menos um outro objeto digital, podendo este ser também um objeto complexo, inclusive do mesmo tipo, o componente *Object* foi criado para representar um objeto digital de um objeto complexo. Assim, um componente do tipo *ComplexObject* pode possuir um ou mais componentes *Object* e, opcionalmente, componentes do tipo *Stream*, identificando então a estrutura completa do objeto complexo. A Figura 4.1 exhibe os relacionamentos entre os componentes *Object*, *SimpleObject*, *ComplexObject* e *Stream* do metamodelo.

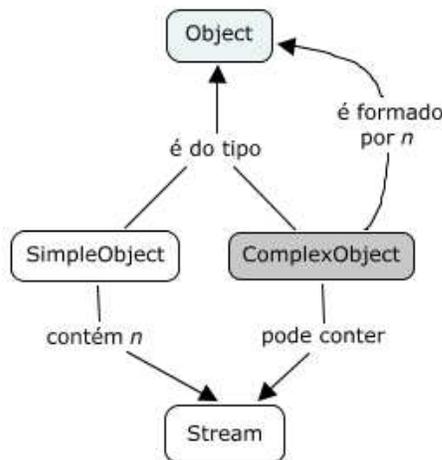


Figura 4.1: Relacionamentos entre componentes do metamodelo de bibliotecas digitais de objetos complexos.

Resumidamente, o metamodelo genérico até então utilizado para especificar uma biblioteca digital foi alterado para incluir os seguintes elementos:

ComplexObject: representa um objeto digital complexo. É formado por pelo menos um objeto digital e pode possuir *Streams* que não são objetos digitais;

Object: representa um objeto da biblioteca digital, podendo ser um objeto atômico, composto ou complexo; é parte de um objeto complexo.

A nomenclatura dos elementos foi definida na língua inglesa, para manter a coerência com os outros elementos do metamodelo atual.

Posteriormente, os outros modelos do formalismo 5S foram analisados. Os modelos *Spaces* e *Societies* não são afetados pelas *streams* ou pela ocorrência de objetos complexos; são afetados diretamente pelos requisitos da aplicação, e por isso, não sofrem alterações significativas em bibliotecas digitais de objetos complexos.

Entretanto, o modelo *Scenarios* pode ser alterado com a inclusão do serviço de visão no contexto (*view in context*) [59], que permite a navegação intra-documentos, e do serviço de navegação temporal (quando houver).

4.2 Extensão da ferramenta 5SGraph

A ferramenta 5SGraph [71] foi criada com o objetivo de fazer com que os desenvolvedores de bibliotecas digitais compreendam e produzam modelos complexos de bibliotecas digitais. Para alcançar esse objetivo, a ferramenta (ver interface pela Figura 4.2) carrega e mostra um metamodelo de bibliotecas digitais (parte inferior da interface) para que usuários criem modelos de bibliotecas digitais (parte superior da interface) seguindo a estrutura desse metamodelo carregado. Os modelos então são gerados na linguagem 5SL em arquivos XML [40, 41].

Entretanto, a ferramenta, até então, não continha um metamodelo que permitisse especificar uma biblioteca digital de objetos complexos. Com a grande quantidade de informação em diferentes formatos e com muitos dados agregados, é essencial que uma ferramenta possibilite a especificação desse tipo de objetos para documentar as estruturas dos objetos e para facilitar o entendimento dessas estruturas pelos analistas de requisitos, desenvolvedores e usuários. A seguir, as alterações que foram realizadas no metamodelo da ferramenta 5SGraph são detalhadas.

O componente já existente *Document* foi renomeado para *SimpleObject*. Os novos componentes *Object* e *ComplexObject* descritos na seção anterior foram criados no metamodelo da ferramenta 5SGraph com a nomenclatura *Object* e *ComplexObj*. O conceito *ComplexObject* foi encurtado para *ComplexObj* no metamodelo da ferramenta 5SGraph para que o conceito seja completamente visível após abertura na interface da ferramenta 5SGraph. Com essas alterações, é possível modelar bibliotecas digitais de objetos complexos na ferramenta.

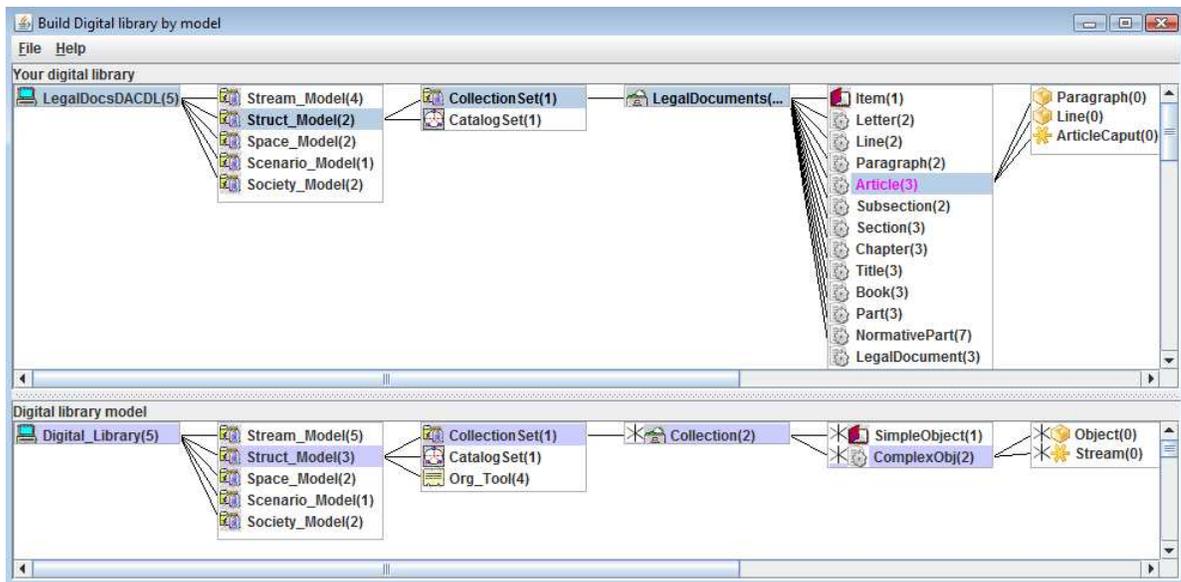


Figura 4.2: Interface da ferramenta 5SGraph.

A Figura 4.3 mostra as alterações do modelo *Structural*, indicadas por quadrados em negrito, no arquivo XML referente ao metamodelo genérico da ferramenta 5SGraph. O elemento *Collection* foi alterado para que possa conter, além de objetos atômicos e compostos, também objetos complexos. Então, a tag *ComplexObj* foi incluída como nó filho da tag *Collection*, com valor do atributo *constraint* sendo igual a “*” (asterisco), indicando que uma coleção pode ter vários objetos complexos, da mesma forma que especificado para objetos digitais simples (tag *Document* renomeada para *SimpleObject*). No quadrado maior, estão exibidos os dois novos elementos criados. O primeiro elemento, *ComplexObj*, é composto de pelo menos um objeto digital, representado pela tag *Object*, e pode ser composto de várias outras *Streams* (tag *Stream*). O ícone *co.gif*, de “complex object”, foi escolhido para representar o objeto complexo. Da mesma forma que se pode informar a estrutura de um objeto simples na ferramenta, um arquivo com a estrutura do objeto complexo, normalmente um DTD ou um XML Schema, pode ser carregado. Essa funcionalidade é criada, conjuntamente, pela tag *TextField* e pelo seu atributo *load* com valor igual a *true*. O segundo elemento representa um objeto digital, simples ou complexo, que será especificado como parte de um objeto complexo. O ícone *object.gif* foi escolhido para representá-lo na ferramenta 5SGraph. A tag *ComboBox* permite ao usuário indicar qual é o objeto que está sendo referenciado para compor o objeto complexo, pois uma lista dos objetos já definidos na biblioteca é apresentada.

A Figura 4.4 exhibe as alterações realizadas no metamodelo genérico, visualizadas pela ferramenta 5SGraph. Os novos elementos criados (*ComplexObj* e *Object*) estão destacados

```

<Collection type="DataType">
  <SubNodes>
    <SimpleObject constraint="*" />
    <ComplexObj constraint="*" />
  </SubNodes>
  <Icon name="coll.gif" />
  <Property>
    <TextField name="Description" />
    <TextField name="Creator" />
    <TextField name="Maintainer" />
  </Property>
</Collection>
<SimpleObject type="DataType">
  <SubNodes >
    <Stream constraint="*" />
  </SubNodes>
  <Icon name="doc.gif" />
  <Property>
    <TextField name="Structed_Stream" load="true"/>
  </Property>
</SimpleObject>
<ComplexObj type="DataType">
  <SubNodes>
    <Object constraint="*" />
    <Stream constraint="*" />
  </SubNodes>
  <Icon name="co.gif" />
  <Property>
    <TextField name="Structed_Stream" load="true"/>
  </Property>
</ComplexObj>
<Object type="DataType">
  <SubNodes />
  <Icon name="object.gif" />
  <Property>
    <ComboBox name="Object" src="Collection" />
  </Property>
</Object>

```

Figura 4.3: Novos elementos *ComplexObj* e *Object* no arquivo do metamodelo da ferramenta 5SGraph.

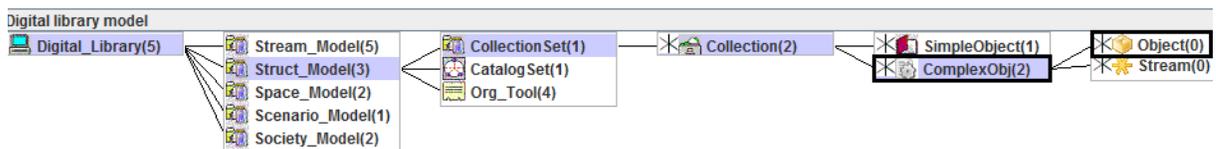


Figura 4.4: Novos elementos *ComplexObj* e *Object* do metamodelo da ferramenta 5SGraph visualizados pela interface.

em negrito. Exemplos e maiores detalhes podem ser visualizados na próxima Seção.

4.3 Estudos de Caso

Esta seção apresenta a especificação de objetos digitais complexos de algumas bibliotecas digitais, usando a ferramenta 5SGraph. Para a especificação dos objetos complexos, só é necessário o preenchimento dos dados dos modelos de *Stream* e de *Struct*. Por isso, os dados dos outros modelos (*Space*, *Scenario* e *Society*) não são discutidos e exibidos nas próximas subseções. A funcionalidade de reuso de componentes/nós de um modelo anteriormente especificado também foi analisado para objetos complexos.

4.3.1 Documentos Legais

O primeiro estudo de caso demonstra a especificação de dispositivos e de documentos legais, descritos na Seção 2.5 e no Capítulo 3. A especificação de documento legal é genérica para diversos tipos de documentos: Deliberação, Portaria, Regimento, Estatuto, entre outros. Os dispositivos e os termos foram definidos em inglês na ferramenta, conforme a Tabela 4.1, para manter a coerência textual com o metamodelo.

Tabela 4.1: Termos legais traduzidos para o inglês.

Em português	Traduzidos para o inglês
Parte	<i>Part</i>
Livro	<i>Book</i>
Título	<i>Title</i>
Capítulo	<i>Chapter</i>
Seção	<i>Section</i>
Subseção	<i>Subsection</i>
Artigo	<i>Article</i>
Parágrafo	<i>Paragraph</i>
Inciso	<i>Line</i>
Alínea	<i>Letter</i>
Item	<i>Item</i>
Parte Normativa	<i>Normative Part</i>
Documento Legal	<i>Legal Document</i>

Inicialmente, as *streams* da biblioteca digital de documentos legais foram definidas (ver Figura 4.5): texto (*Text*), referente aos caracteres que formam os textos normativos; imagem (*Image*) relacionada a possíveis imagens dentro de um texto normativo; áudio (*Audio*), com a possibilidade de publicar áudio dos documentos, fornecendo acesso

para cegos; e páginas *Web* (*WebPage*), onde são exibidos os documentos legais e textos normativos. Detalhes da configuração da *stream Text* é também exibida na mesma figura.

Na Figura 4.6, o dispositivo *Item* (*Item*) é especificado como um objeto digital simples, já que não possui dispositivos filhos, apenas o texto legal, chamado de *caput*, que é definido como *stream* do tipo *Text* (Texto) (ver Figura 4.7).

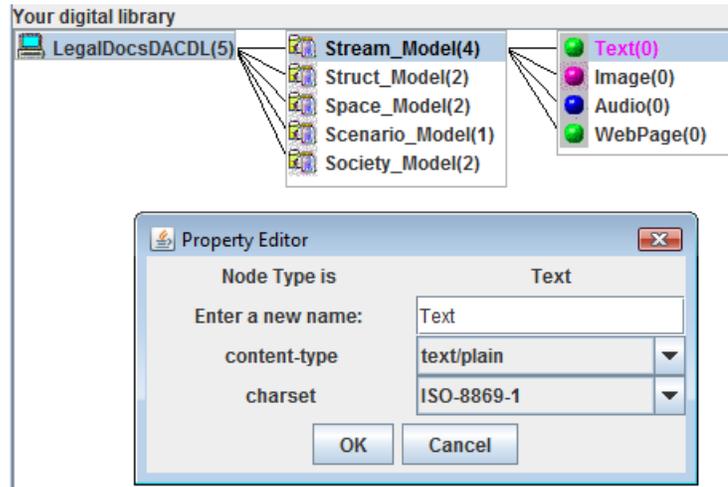


Figura 4.5: *Streams* da biblioteca digital de documentos legais e detalhes da *stream Text*, na ferramenta 5SGraph.

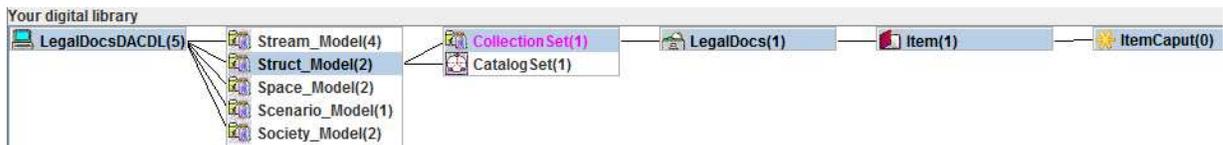


Figura 4.6: Dispositivo *Item* (*Item*) como objeto digital simples na ferramenta 5SGraph.

A Figura 4.8 mostra os dispositivos *Item*, *Alínea* (*Letter*), *Inciso* (*Line*) e *Parágrafo* (*Paragraph*) especificados no arquivo gerado pela ferramenta 5SGraph. A estrutura normativa definida pelas Leis Complementares é respeitada.

Na Figura 4.9, o dispositivo *Artigo* (*Article*) é detalhado e podem-se ver os outros dispositivos criados como objetos complexos, pela possibilidade de conterem outros tipos de dispositivos. As Figuras 4.10 e 4.11 detalham dois dos componentes de um artigo: o texto do *caput* (*ArticleCaput*) e o dispositivo filho *Parágrafo* (*Paragraph*). O dispositivo *Inciso* (*Line*) pode ser também um dos objetos digitais de um *Artigo*. Na última figura



Figura 4.7: *Caput* do dispositivo Item (*Item*) definido como *stream* do tipo *Text*, na ferramenta 5SGraph.

```

<SimpleObject name="Item">
  <Structured_Stream></Structured_Stream>
  <Stream name="ItemCaput">
    <Stream>"Text"</Stream>
  </Stream>
</SimpleObject>
<ComplexObj name="Letter">
  <Structured_Stream></Structured_Stream>
  <Object name="Item">
    <Object>"Item"</Object>
  </Object>
  <Stream name="LetterCaput">
    <Stream>"Text"</Stream>
  </Stream>
</ComplexObj>
<ComplexObj name="Line">
  <Structured_Stream></Structured_Stream>
  <Object name="Letter">
    <Object>"Letter"</Object>
  </Object>
  <Stream name="LineCaput">
    <Stream>"Text"</Stream>
  </Stream>
</ComplexObj>
<ComplexObj name="Paragraph">
  <Structured_Stream></Structured_Stream>
  <Object name="Line">
    <Object>"Line"</Object>
  </Object>
  <Stream name="ParagraphCaput">
    <Stream>"Text"</Stream>
  </Stream>
</ComplexObj>

```

Figura 4.8: Dispositivos no arquivo gerado pela ferramenta 5SGraph.



Figura 4.9: Dispositivos como objetos complexos e detalhe da composição do objeto complexo Artigo (*Article*), na ferramenta 5SGraph.

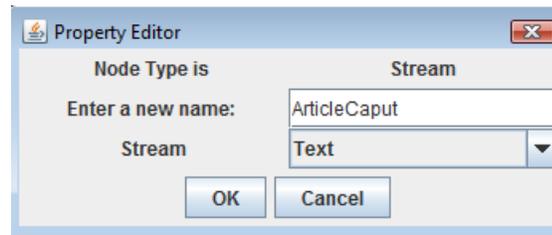


Figura 4.10: *Caput* do Artigo (*ArticleCaput*) definido como *stream* do tipo *Text*, na ferramenta 5SGraph.

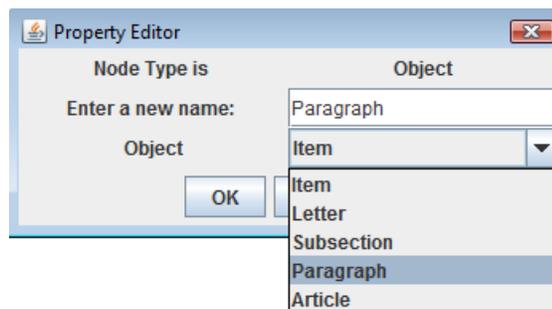


Figura 4.11: Elemento Parágrafo (*Paragraph*) de um Artigo como objeto digital do tipo *Paragraph*, na ferramenta 5SGraph.

citada, pode-se ver que o usuário pode escolher os objetos digitais (simples ou complexos) até então definidos no modelo para definir o tipo do objeto que está sendo criado.

A Figura 4.12 mostra a especificação do Artigo no arquivo gerado pela ferramenta 5SGraph.

```
<ComplexObj name="Article">
  <Structured_Stream></Structured_Stream>
  <Object name="Paragraph">
    <Object>"Paragraph"</Object>
  </Object>
  <Object name="Line">
    <Object>"Line"</Object>
  </Object>
  <Stream name="ArticleCaput">
    <Stream>"Text"</Stream>
  </Stream>
</ComplexObj>
```

Figura 4.12: Artigo (*Article*) no arquivo gerado pela ferramenta 5SGraph.

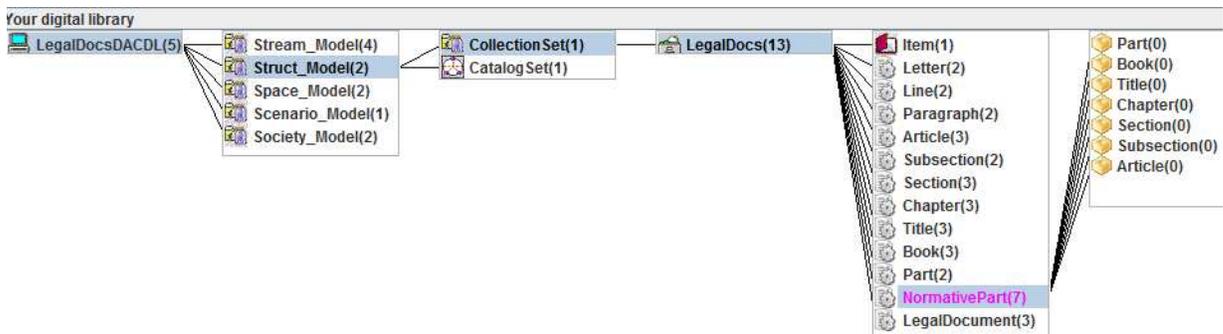


Figura 4.13: Composição do objeto complexo Parte Normativa (*NormativePart*), na ferramenta 5SGraph.

Na Figura 4.13, a parte normativa de um documento legal é detalhada com todos os dispositivos de um texto normativo, de acordo com a estrutura dos documentos legais definida pelas Leis Complementares. O último objeto incluído é o Artigo (*Article*), pois ele é a unidade básica de um documento legal. Os dispositivos Parágrafo, Inciso, Alínea e Item estão sempre associados a um Artigo específico. Portanto, se existirem, são objetos do artigo.

Finalizando, um documento legal é composto de parte preliminar (*PreliminaryPart*), parte normativa (*NormativePart*) e parte final (*FinalPart*). Essa composição também foi especificada na ferramenta 5SGraph, conforme Figura 4.14. As partes preliminar e final são compostas apenas de textos e a parte normativa é composta por artigos, que podem ser agrupados ou detalhados em outros dispositivos.

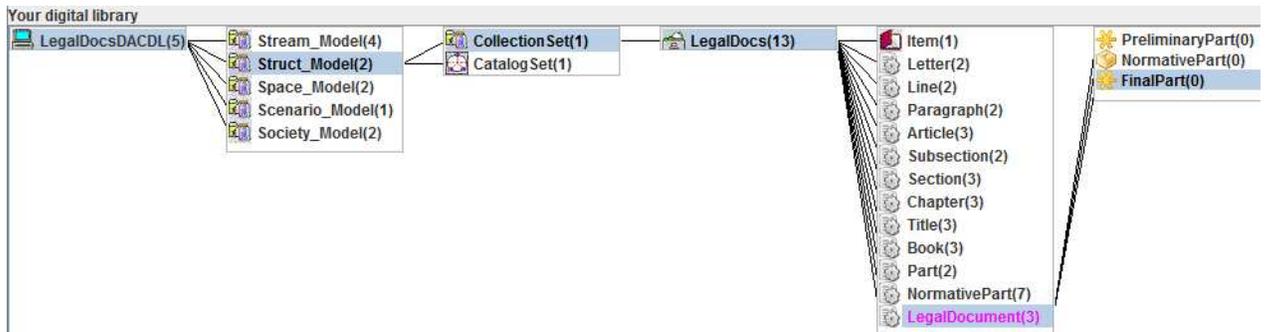


Figura 4.14: Composição do objeto complexo Documento Legal (*LegalDocument*), na ferramenta 5SGraph.

A Figura 4.15 mostra a especificação da parte normativa e do documento legal no arquivo gerado pela ferramenta 5SGraph.

4.3.2 Documentos Sobrepostos

Outro estudo de caso é uma demonstração da teoria sobre documentos sobrepostos (*superimposed documents*), com a especificação dos documentos na ferramenta 5SGraph.

Primeiramente, especifica-se o documento base (*base document*) com *streams* de texto e de imagem (ver Figura 4.16). Depois, especifica-se o subdocumento (*subdocument*) com as *substreams* do documento base, a especificação da apresentação (PS) e o endereço (*Address*) do começo e do final da *substream* (ver Figura 4.17). Por último, o documento sobreposto (*superimposed document*) é especificado sendo composto por um ou mais objetos digitais do tipo subdocumento, já definido anteriormente, e por *streams*. Por conter objetos digitais, é especificado como objeto complexo. A Figura 4.18 mostra o subdocumento sendo especificado para o documento sobreposto e a Figura 4.19 exhibe a composição do documento sobreposto. Para checagem do arquivo gerado pela ferramenta 5SGraph, a Figura 4.20 exhibe trecho do arquivo referente aos objetos digitais.

O documento base também poderia ser definido como objeto complexo, contendo um ou mais objetos digitais e *streams* (como, por exemplo, um documento legal). Nesse caso, o subdocumento poderia ser definido ou como objeto simples, caso possua *substreams* do documento base ou como objeto complexo, quando contém objetos digitais do documento base.

```

<ComplexObj name="NormativePart">
  <Structured_Stream></Structured_Stream>
  <Object name="Part">
    <Object>"Part"</Object>
  </Object>
  <Object name="Book">
    <Object>"Book"</Object>
  </Object>
  <Object name="Title">
    <Object>"Title"</Object>
  </Object>
  <Object name="Chapter">
    <Object>"Chapter"</Object>
  </Object>
  <Object name="Section">
    <Object>"Section"</Object>
  </Object>
  <Object name="Subsection">
    <Object>"Subsection"</Object>
  </Object>
  <Object name="Article">
    <Object>"Article"</Object>
  </Object>
</ComplexObj>
<ComplexObj name="LegalDocument">
  <Structured_Stream></Structured_Stream>
  <Object name="NormativePart">
    <Object>"NormativePart"</Object>
  </Object>
  <Stream name="PreliminaryPart">
    <Stream>"Text"</Stream>
  </Stream>
  <Stream name="FinalPart">
    <Stream>"Text"</Stream>
  </Stream>
</ComplexObj>

```

Figura 4.15: Parte Normativa (*NormativePart*) e Documento Legal (*LegalDocument*) no arquivo gerado pela ferramenta 5SGraph.



Figura 4.16: Documento base (*BaseDocument*) especificado na ferramenta 5SGraph.

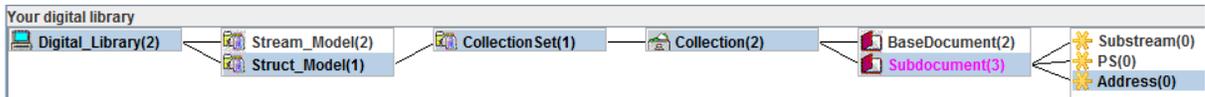


Figura 4.17: Subdocumento (*Subdocument*) especificado na ferramenta 5SGraph.

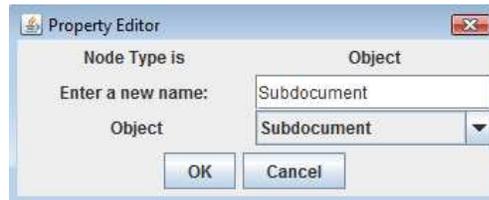


Figura 4.18: Especificação do subdocumento (*Subdocument*) como objeto digital do documento sobreposto (*SuperDocument*), na ferramenta 5SGraph.

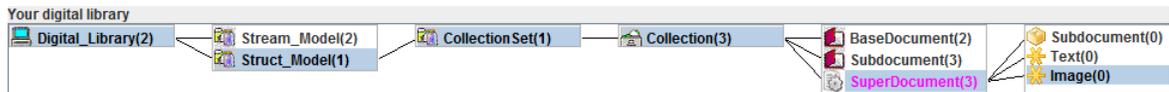


Figura 4.19: Composição do documento sobreposto (*SuperDocument*), na ferramenta 5SGraph.

4.3.3 Descritores de Imagem

Este estudo de caso especifica um descritor de imagem como objeto complexo. Para essa composição o objeto digital de *software* é utilizado. Esse tipo de objeto foi definido com o tipo de *stream Application*. Hipoteticamente, definiu-se que a linguagem de implementação Java seria utilizada na geração dos *softwares* e, por isso, a *stream JavaApplication* foi criada (ver Figura 4.21). A Figura 4.22 mostra a especificação dessa *stream* no arquivo gerado pela ferramenta 5SGraph.

Com esse tipo de *stream*, as duas funções realizadas por um descritor de imagens podem ser especificadas: a extração do vetor de características de imagens e a computação/comparação de distâncias entre imagens. A Figura 4.23 mostra a especificação dessas duas funções, como *FVExtraction* e *DistanceComp*, respectivamente, na ferramenta 5SGraph.

Ao final (Figura 4.24), o descritor de imagem (*ImageDescriptor*) é especificado com os dois objetos digitais de *software* criados anteriormente. A Figura 4.25 mostra a especificação do descritor de imagem como objeto complexo no arquivo gerado pela ferramenta 5SGraph.

```

<SimpleObject name="BaseDocument">
  <Structured_Stream></Structured_Stream>
  <Stream name="Text">
    <Stream>"Text"</Stream>
  </Stream>
  <Stream name="Image">
    <Stream>"Image"</Stream>
  </Stream>
</SimpleObject>
<SimpleObject name="Subdocument">
  <Structured_Stream></Structured_Stream>
  <Stream name="Substream">
    <Stream>"Text"</Stream>
  </Stream>
  <Stream name="PS">
    <Stream>"Text"</Stream>
  </Stream>
  <Stream name="Address">
    <Stream>"Text"</Stream>
  </Stream>
</SimpleObject>
<ComplexObj name="SuperDocument">
  <Structured_Stream></Structured_Stream>
  <Object name="Subdocument">
    <Object>"Subdocument"</Object>
  </Object>
  <Stream name="Text">
    <Stream>"Text"</Stream>
  </Stream>
  <Stream name="Image">
    <Stream>"Image"</Stream>
  </Stream>
</ComplexObj>

```

Figura 4.20: Estrutura do documentos base, do subdocumento e do documento sobreposto no arquivo gerado pela ferramenta 5SGraph.

4.3.4 Imagens

O último estudo de caso especifica objeto de imagem e objeto complexo de imagem de acordo com a formalização realizada por Kozievitch [50].

O objeto de imagem, definido como IDO (*Image Digital Object*) na Figura 4.26, possui:

- *substreams* (regiões) de uma imagem (*stream*), indicadas pela *stream Image* do tipo *Image*;
- uma descrição do conteúdo da imagem, com o detalhamento do vetor de característica para cada descritor utilizado, declarada como *stream FeatureVectors* do tipo *Text*.

Já o objeto complexo de imagem, definido como ICO (*Image Complex Object*) na Figura 4.27, possui:

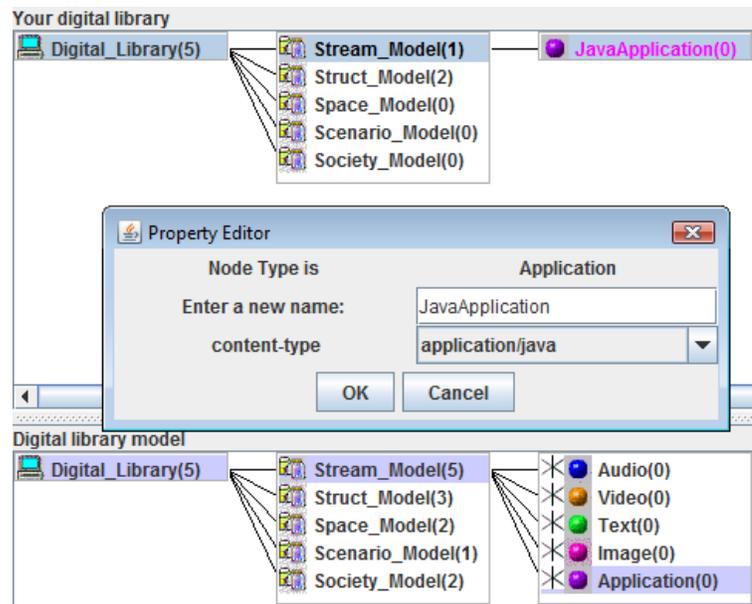


Figura 4.21: Aplicação em Java especificada na ferramenta 5SGraph.

```

<Stream_Model name="Stream_Model">
  <Application name="JavaApplication">
    <content-type>"application/java"</content-type>
  </Application>
</Stream_Model>

```

Figura 4.22: *Stream* de um programa Java no arquivo gerado pela ferramenta 5SGraph.

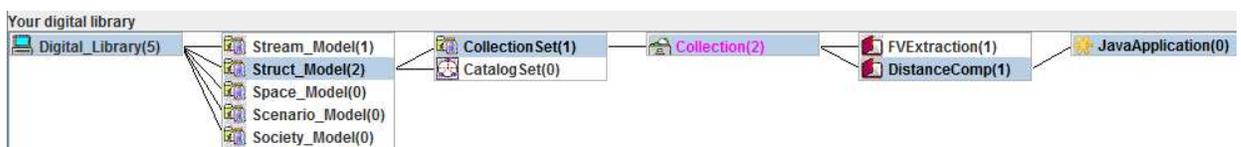


Figura 4.23: Exemplo de objetos digitais de software, na ferramenta 5SGraph.

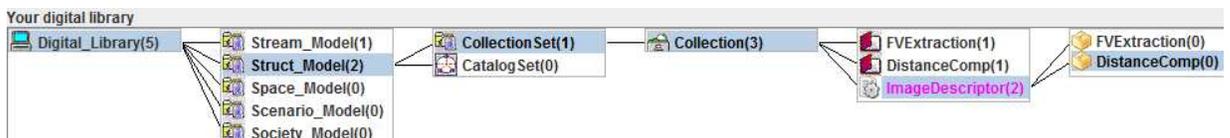


Figura 4.24: Descritores de imagem especificados na ferramenta 5SGraph.

```

<SimpleObject name="FVExtraction">
  <Structured_Stream></Structured_Stream>
  <Stream name="JavaApplication">
    <Stream>"JavaApplication"</Stream>
  </Stream>
</SimpleObject>
<SimpleObject name="DistanceComp">
  <Structured_Stream></Structured_Stream>
  <Stream name="JavaApplication">
    <Stream>"JavaApplication"</Stream>
  </Stream>
</SimpleObject>
<ComplexObj name="ImageDescriptor">
  <Structured_Stream></Structured_Stream>
  <Object name="FVExtraction">
    <Object>"FVExtraction"</Object>
  </Object>
  <Object name="DistanceComp">
    <Object>"DistanceComp"</Object>
  </Object>
</ComplexObj>

```

Figura 4.25: Descritor de imagem (*ImageDescriptor*) como objeto complexo no arquivo gerado pela ferramenta 5SGraph.

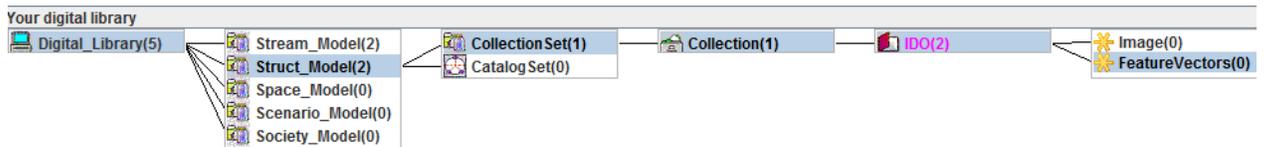


Figura 4.26: Objeto digital de imagem (IDO) especificado na ferramenta 5SGraph.

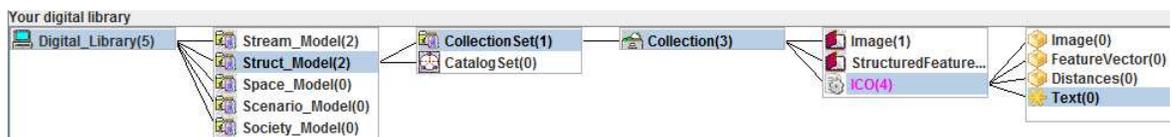


Figura 4.27: Objeto complexo de imagem (ICO) especificado na ferramenta 5SGraph.

- um objeto digital de imagem (indicando o arquivo da imagem e os pixels do arquivo que representam a imagem); é representado pelo objeto digital *Image*, definido como *stream* do tipo *image/jpeg*;
- o detalhamento do vetor de característica para cada descritor utilizado (ver Figura 4.28, onde *StructuredFeatureVector* foi definido na ferramenta 5SGraph como *stream* do tipo *Text* e, posteriormente, ver Figura 4.29, na qual o vetor de características *FeatureVector* foi declarado como objeto do tipo *StructuredFeatureVector*);
- valores das distâncias entre outras imagens; definidos como *Distances*, objeto digital do tipo *StructuredFeatureVector*, na Figura 4.30;
- outras possíveis *streams*, como a descrição textual da imagem definida como *stream* do tipo *Text*.

A Figura 4.31 exibe a especificação dos objetos gerados no arquivo pela ferramenta 5SGraph.

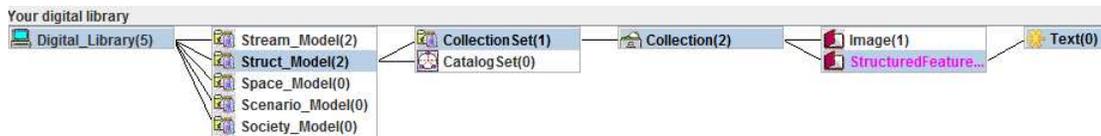


Figura 4.28: *StructuredFeatureVector* especificado na ferramenta 5SGraph.

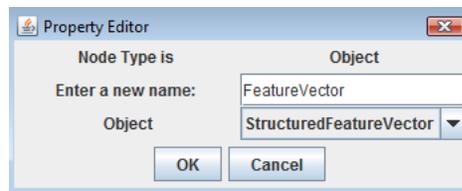


Figura 4.29: Vetor de características (*Feature Vector*) especificado na ferramenta 5SGraph.

4.3.5 Reuso de Objetos Complexos

A ferramenta 5SGraph oferece a funcionalidade de reusar componentes de um modelo de bibliotecas digitais. Essa funcionalidade é acessada clicando em cima de um componente especificado com o botão direito e escolhendo a opção “Save node as...” e salvando o componente em um arquivo XML. Para reusar o componente em um novo modelo de

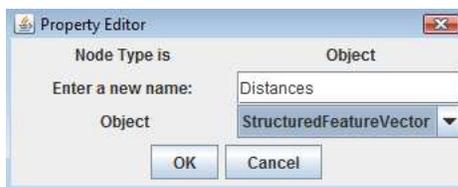


Figura 4.30: Valores das distâncias (*Distances*) entre imagens especificados na ferramenta 5SGraph.

```

<SimpleObject name="Image">
  <Structed_Stream></Structed_Stream>
  <Stream name="Image">
    <Stream>"Image"</Stream>
  </Stream>
</SimpleObject>
<SimpleObject name="StructuredFeatureVector">
  <Structed_Stream></Structed_Stream>
  <Stream name="Text">
    <Stream>"Text"</Stream>
  </Stream>
</SimpleObject>
<ComplexObj name="ICO">
  <Structed_Stream></Structed_Stream>
  <Object name="Image">
    <Object>"Image"</Object>
  </Object>
  <Object name="FeatureVector">
    <Object>"StructuredFeatureVector"</Object>
  </Object>
  <Object name="Distances">
    <Object>"StructuredFeatureVector"</Object>
  </Object>
  <Stream name="Text">
    <Stream>"Text"</Stream>
  </Stream>
</ComplexObj>

```

Figura 4.31: Arquivo contendo objeto complexo do tipo imagem.

bibliotecas digitais, deve-se clicar com o botão direito no componente que deve ser o componente pai do componente a ser reusado e escolha a opção “Load sub node...”. Em seguida, indica-se o arquivo XML referente ao componente que deve ser reusado neste novo modelo.

Para exemplificar o reuso de objetos complexos, salvamos o componente *Image Complex Object* ou ICO (ver Figura 4.32) em um arquivo XML para reutilizá-lo em outro modelo de biblioteca digital de imagens. O arquivo gerado pode ser visto na Figura 4.33. Apenas o objeto complexo indicado e seus nós filhos foram gravados no arquivo. Assim, em um novo modelo, pode-se clicar em “Load sub node...” (ver Figura 4.34) para reusar

o componente salvo anteriormente.



Figura 4.32: Nó ICO sendo salvo para posterior reuso em outro modelo da ferramenta 5SGraph.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<DLModel>
  <ComplexObj name="ICO">
    <Structured_Stream></Structured_Stream>
    <Object name="Image">
      <Object>"Image"</Object>
    </Object>
    <Object name="FeatureVector">
      <Object>"StructuredFeatureVector"</Object>
    </Object>
    <Object name="Distances">
      <Object>"StructuredFeatureVector"</Object>
    </Object>
    <Stream name="Text">
      <Stream>"Text"</Stream>
    </Stream>
  </ComplexObj>
</DLModel>
```

Figura 4.33: Arquivo gerado pela ferramenta 5SGraph contendo componente ICO.

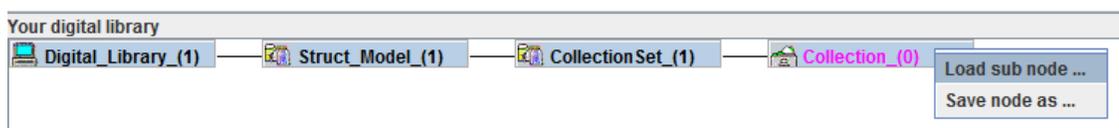


Figura 4.34: Nó sendo carregado na ferramenta 5SGraph para reuso em novo modelo.

Após o carregamento do componente ICO no novo modelo, pode-se notar que o objeto complexo e seus filhos foram carregados. Entretanto, a especificação do tipo de *streams* e de objetos fica incoerente devido a não especificação desses dados antes de carregar o componente. Para exemplificar essa inconsistência, a Figura 4.35 exibe o objeto *Image* especificado como objeto do tipo ICO, pois apenas esse tipo de objeto foi especificado na coleção. Da mesma forma, como nenhuma *stream* foi especificada, a Figura 4.36 mostra que a *stream Text* não possui a especificação do tipo de *stream*. Por isso, após

o carregamento de um componente reusado, devem-se checar os dados carregados e, se necessário, gerar os componentes que não foram carregados ou especificados e que são utilizados no objeto carregado.



Figura 4.35: Inconsistência do tipo de objeto na ferramenta 5SGraph.

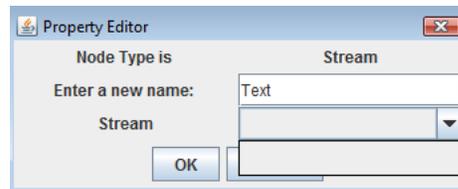


Figura 4.36: Inconsistência do tipo de *stream* na ferramenta 5SGraph.

Capítulo 5

Validação

Este capítulo descreve as avaliações realizadas relacionadas à especificação de bibliotecas digitais de objetos complexos com o uso da ferramenta 5SGraph estendida. Além dos estudos de caso apresentados no capítulo anterior, uma validação com usuários do novo metamodelo inserido na ferramenta 5SGraph e uma implementação de um protótipo de uma biblioteca digital de objetos complexos, de acordo com a especificação em 5SL, foram realizadas. O objetivo foi analisar e avaliar a concreta utilização desse tipo de especificação por usuários potenciais na implementação de bibliotecas digitais de objetos complexos.

5.1 Experimento com potenciais usuários

A validação do novo metamodelo criado foi realizada com alunos do Instituto de Computação da UNICAMP com conhecimentos em bibliotecas digitais para verificar se a ferramenta 5SGraph ajuda os usuários na construção de modelos, baseados no formalismo 5S, visando à especificação de bibliotecas digitais de objetos complexos. As seguintes medidas foram analisadas: eficácia, eficiência e satisfação do usuário. Além disso, esse experimento permitiu analisar se o que foi proposto por este trabalho pode ser estendido e aperfeiçoado de acordo com críticas, sugestões e comentários relatados pelos participantes.

5.1.1 Protocolo Experimental

O processo de avaliação adotado seguiu como exemplo o protocolo utilizado em avaliação anterior realizada por Zhu [71] sobre a especificação de bibliotecas digitais usando a ferramenta 5SGraph em que objetos complexos não foram considerados.

As seguintes etapas foram adotadas para a realização da análise do novo metamodelo:

- escolha de participantes da área de computação e de pessoas com conhecimento na área de bibliotecas digitais;

- entrega de um questionário para coleta dos níveis de conhecimento do participante;
- indicação de três tarefas em ordem crescente de dificuldade a serem realizadas na ferramenta 5SGraph relacionadas à definição de objetos de uma biblioteca digital;
- entrega de um questionário final sobre aspectos de usabilidade, de aprendizagem e de eficácia da ferramenta;
- análise dos arquivos gerados por cada participante após cada tarefa realizada na ferramenta 5SGraph;
- obtenção dos resultados através das medidas de eficácia, de eficiência e de satisfação do usuário.

As tarefas do experimento consistem em especificar objetos digitais usando a ferramenta 5SGraph. Para o cumprimento das tarefas, o usuário precisava especificar:

- um objeto digital simples (uma tese de doutorado com *streams* do tipo texto e *streams* do tipo imagem);
- um objeto complexo contendo um objeto digital do tipo *software* e *streams* do tipo texto (uma tese de doutorado que possui, além do texto, um *software* que foi desenvolvido durante a pesquisa realizada);
- vários objetos complexos, alguns contendo mais de um objeto digital (um documento legal desmembrado em artigos, parágrafos, incisos, alíneas e itens);

Os documentos elaborados para o experimento se encontram no Apêndice A.

Já que para especificar objetos digitais, somente os modelos *Stream* e *Struct* precisam ser preenchidos, as tarefas não contemplavam o preenchimento dos outros modelos das bibliotecas pelos usuários. Os arquivos, em linguagem 5SL, gerados por cada tarefa foram encaminhados pelos usuários para apuração dos resultados.

5.1.2 Perfis dos Usuários

Um questionário foi disponibilizado a seis usuários com o objetivo de traçar o perfil dos participantes quanto aos conhecimentos nas áreas e nas tecnologias envolvidas no experimento.

A Figura 5.1 apresenta o grau de instrução dos participantes. Pode-se verificar que todos possuem nível superior (são ou foram alunos do Instituto de Computação da UNICAMP) e que a metade dos participantes está cursando doutorado na área de tecnologia. A Figura 5.2 mostra bons conhecimentos dos participantes na língua inglesa.

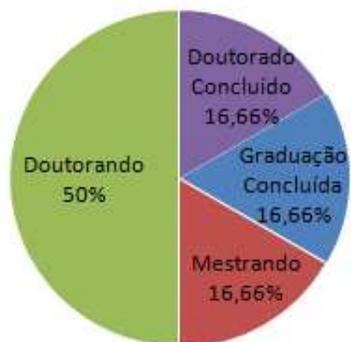


Figura 5.1: Grau de formação dos participantes.



Figura 5.2: Nível de conhecimento dos participantes em inglês.



Figura 5.3: Nível de conhecimento dos participantes em bibliotecas digitais.

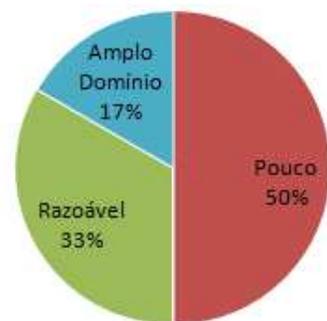


Figura 5.4: Nível de conhecimento dos participantes sobre objetos digitais complexos.

A Figura 5.3 exibe o nível de conhecimento em bibliotecas digitais dos participantes. O gráfico mostra que todos possuem alguma noção sobre o tópico, sendo que a maioria (quase 84%) possui de razoável a amplo domínio sobre o assunto. Já sobre objetos digitais complexos (ver Figura 5.4), a metade (50%) possui pouco conhecimento, 33% razoável conhecimento e 17% amplo conhecimento.

Quanto ao conhecimento sobre o formalismo 5S (ver Figura 5.5), um dos participantes não conhecia o 5S, a maioria teve pouco ou razoável contato e um outro participante possui amplo domínio. Consequentemente, o conhecimento e a prática da ferramenta 5SGraph pelos participantes é baixa, como se pode observar na Figura 5.6. Todos os usuários possuem de razoável a nenhum conhecimento sobre a ferramenta.

Também foi questionado aos participantes se possuíam alguma experiência na criação de bibliotecas digitais (ver Figura 5.7). Apenas um dos participantes nunca criou uma biblioteca digital. O restante já criou bibliotecas digitais de relatórios técnicos na disci-

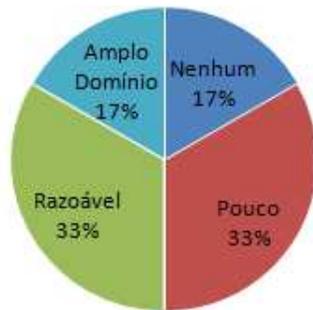


Figura 5.5: Nível de conhecimento dos participantes sobre 5S.

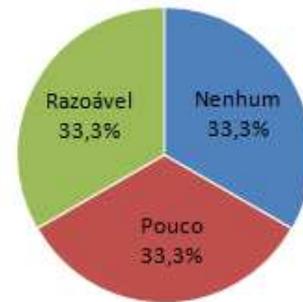


Figura 5.6: Nível de conhecimento dos participantes sobre 5SGraph.

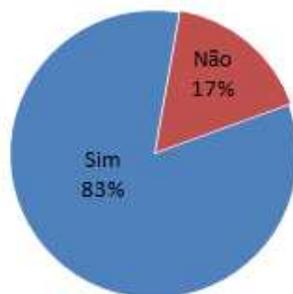


Figura 5.7: Experiência dos participantes na criação de bibliotecas digitais.

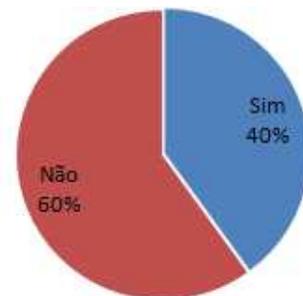


Figura 5.8: Experiência dos participantes na criação de bibliotecas digitais de objetos complexos.



Figura 5.9: Uso de ferramentas na especificação de objetos digitais.

plina de Bibliotecas Digitais; de teses e dissertações; de espécies de peixes; de documentos históricos; e de documentos digitais da UNICAMP. Dos participantes que já criaram uma biblioteca digital, pode-se ver, pela Figura 5.8, que 40% deles já criou uma biblioteca digital com objetos complexos (anotações sobre partes de imagens), e, pela Figura 5.9, que ninguém usou uma ferramenta de especificação na construção dessas bibliotecas digitais.

Com todos esses dados, considera-se que o perfil dos participantes é adequado para esse experimento.

5.1.3 Medidas para Análise

As medidas que foram utilizadas para análise dos resultados estão citadas abaixo.

Eficácia

Taxa de completude: percentual de participantes que completaram corretamente cada tarefa. Valor de 0 a 100.

Alcance do objetivo: percentual que indica quanto cada tarefa foi completada corretamente. Valor de 0 a 100.

Eficiência

Tempo: tempo em minutos utilizado para completar cada tarefa.

Praticidade: tempo mínimo dividido pelo tempo da tarefa. Esse tempo é o menor período de tempo que foi necessário para finalizar a tarefa, e é medido pelo tempo que um habilidoso usuário na ferramenta gasta realizando a tarefa. Valor de 0 a 1. Neste experimento, o habilidoso usuário ou o especialista utilizado em cada tarefa foi o participante que finalizou a tarefa no menor tempo.

Satisfação

A satisfação do usuário foi medida a partir das respostas dadas no questionário final. Esse questionário é composto por perguntas relacionadas aos níveis de aprendizado e de satisfação do aluno ao utilizar a ferramenta para especificar objetos complexos.

5.1.4 Resultados

Nesta subseção, os dados coletados durante e após a realização das tarefas são analisados para verificar a efetiva utilidade da ferramenta 5SGraph na especificação de objetos complexos de bibliotecas digitais.

Questionados sobre outras indicações para a denominação dos novos conceitos criados, todos os participantes responderam que não usariam outras denominações. Além

disso, também responderam sentir-se capazes de especificar outros objetos complexos na ferramenta 5SGraph.

De maneira geral, conforme Figura 5.10, notou-se um maior entendimento sobre objetos digitais complexos após a especificação desses objetos na ferramenta 5SGraph do que antes do experimento (comparar com a Figura 5.4). Os 50% dos participantes que declararam-se com pouco conhecimento sobre objetos complexos no questionário inicial declararam possuir razoável conhecimento ao final do experimento.

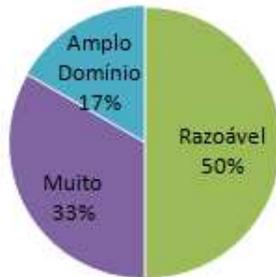


Figura 5.10: Nível de conhecimento dos participantes sobre objetos digitais complexos após uso da ferramenta 5SGraph.

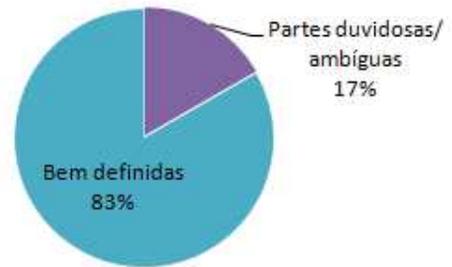


Figura 5.11: Opinião dos participantes sobre as instruções das tarefas.

Tabela 5.1: Dados coletados da Tarefa 1.

Número do Participante	Alcance do Objetivo (%)	Tempo (min)	Praticidade
1	100	6	0,5
2	100	4	0,75
3	100	5	0,6
4	100	4	0,75
5	100	3	1
6	100	4	0,75

Eficácia

De acordo com os dados coletados (listados nas Tabelas 5.1, 5.2, 5.3 e resumidos com as médias das medidas na Tabela 5.4), a alta taxa de completude e a alta taxa de alcance dos objetivos das tarefas mostram que a ferramenta é eficaz na especificação de objetos complexos. A Tarefa 3 resultou em valores menores por ser mais complexa e possuir mais detalhes (como a especificação também de *streams*, além do objeto digital, nos objetos complexos) que não foram percebidos por todos os usuários ou também pelo

Tabela 5.2: Dados coletados da Tarefa 2.

Número do Participante	Alcance do Objetivo (%)	Tempo (min)	Praticidade
1	100	8	0,5
2	100	8	0,5
3	90	10	0,4
4	100	4	1
5	100	12	0,34
6	100	7	0,57

Tabela 5.3: Dados coletados da Tarefa 3.

Número do Participante	Alcance do Objetivo (%)	Tempo (min)	Praticidade
1	100	12	0,5
2	80	12	0,5
3	75	12	0,5
4	100	6	1
5	100	12	0,5
6	80	9	0,66

Tabela 5.4: Resumo da média das medidas coletadas em cada tarefa.

	Tarefa 1	Tarefa 2	Tarefa 3
Taxa de Completude (%)	100	80	50
Alcance do Objetivo (%)	100	98,33	89,17
Tempo médio (min)	4,33	8,17	10,5
Praticidade	0,72	0,55	0,61

não entendimento das instruções da tarefa por alguns usuários (ver Figura 5.11 na qual um usuário declarou que as instruções possuíam partes ambíguas/duvidosas).

Um problema relatado por dois participantes foi a dificuldade em entender a especificação da composição dos objetos. Da maneira que foi proposta a especificação de objetos digitais na ferramenta 5SGraph, o usuário deve primeiro especificar os objetos simples para depois especificar os objetos complexos. Por exemplo, para especificar um descritor de imagem (objeto complexo), o usuário primeiro especifica as funções de extração de vetores de características e de cálculo das distâncias das imagens para depois especificar que o descritor é composto por essas funções. Esses participantes sugeriram analisar a ordem inversa do processo de construção de um objeto digital, iniciando a especificação pelo objeto complexo e depois especificando os objetos que o compõe.

Eficiência

Quanto à eficiência, o tempo médio gasto em cada tarefa pelos participantes (ver Tabela 5.4) foi, para a Tarefa 1, de mais de quatro minutos; para a Tarefa 2, um pouco mais de oito minutos; e para a Tarefa 3, dez minutos e meio.

No questionário final, a maioria dos participantes (83%) indicou que as instruções das tarefas estavam bem definidas (ver Figura 5.11). Por isso, essa não seria a causa do alto tempo gasto nas tarefas. A demora foi devida a não ter tido explicações iniciais sobre a ferramenta e pela ausência da leitura de um manual sobre 5SGraph. Alguns participantes comentaram que tiveram dificuldades iniciais no entendimento das funcionalidades da ferramenta.

Outro fator que influenciou o tempo foi a presença de poucos detalhes dos tipos das *streams* a serem utilizadas em cada tarefa (por exemplo, o texto da tese deve ser especificado com o tipo de conteúdo *text/plain* ou com *text/latex*, com codificação *ISO-8869-1* ou então *UTF-8*). Essa ausência de informações precisas foi intencionalmente provocada com o objetivo de que os participantes pudessem realmente refletir, escolher e definir sobre os objetos da biblioteca.

Os resultados para a medida de praticidade (ver Tabela 5.4) mostram que a Tarefa 1, por ser bem simples, foi facilmente entendida e realizada por todos os participantes (valor 0,72). Porém, a realização da Tarefa 2, com a novidade da especificação de objeto complexo, mostrou maior dificuldade (o valor da praticidade decaiu em relação à Tarefa 1). Nota-se aqui novamente a dificuldade em especificar objetos complexos iniciando a especificação por objetos com menor granularidade. Os resultados da Tarefa 3 mostram que os participantes compreenderam como especificar objetos complexos após a execução da Tarefa 2, pois obtiveram uma melhora na medida, mesmo a Tarefa 3 sendo uma especificação mais complexa, com o envolvimento de vários objetos. Essa melhora na Tarefa 3 é similar aos resultados do experimento realizado por Zhu [71], no qual os usuários também melhoraram a medida de praticidade das tarefas seguintes após a execução da Tarefa 1, mostrando que, após entendimento das funcionalidades básicas da ferramenta, novas especificações podem ser executadas rapidamente.

Satisfação

Pela Figura 5.12, nota-se que a metade dos participantes declararam facilidade para especificar objetos complexos na ferramenta 5SGraph. A outra metade dos participantes considera que a especificação de objetos complexos na ferramenta foi uma tarefa realizada com certa complexidade. Entre esses últimos participantes, comentários foram deixados sobre a interface não ser intuitiva e sugestões foram dadas sobre a especificação mudar sua abordagem para *top-down*, ou seja, que o início da especificação seja pela definição do objeto complexo ao invés de iniciá-la pelos objetos simples. Algumas melhorias também foram sugeridas na interface: inclusão automática de nó no modelo quando o usuário



Figura 5.12: Opinião dos participantes sobre o nível de complexidade para especificar objetos digitais na ferramenta 5SGraph.



Figura 5.13: Opinião dos participantes sobre a funcionalidade da ferramenta 5SGraph de especificar objetos complexos.

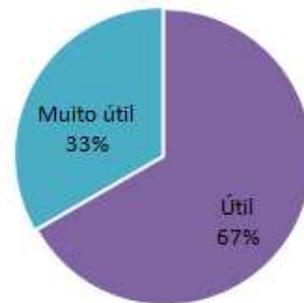


Figura 5.14: Opinião dos participantes sobre a utilidade da ferramenta 5SGraph.

clica em um conceito no metamodelo e este ainda não está definido no modelo sendo criado; criação automática do nome da *stream* ou do objeto simples após escolha no *combobox*, podendo o nome indicado ser alterado; realizar a validação da estrutura dos objetos definidos com o arquivo de estrutura indicado e carregado na interface.

Por fim, foi questionado se a ferramenta atende ao propósito de especificar objetos complexos, se ela pode ser útil aos desenvolvedores nesta tarefa e se os participantes a reusariam para especificar uma biblioteca digital. A partir das respostas (ver Figuras 5.13 e 5.14), as opiniões dos participantes indicam que a ferramenta é satisfatória, útil e que seria reusada para especificar bibliotecas digitais.

Analisando todos os resultados, pode-se concluir que a ferramenta 5SGraph estendida é capaz de especificar bibliotecas digitais de objetos complexos e que melhorias podem ser realizadas na interface para aumentar a satisfação dos usuários.

5.2 Biblioteca Digital de Documentos Legais da DAC

Esta seção apresenta a construção do protótipo de uma biblioteca digital de documentos legais, baseada no modelo criado a partir do uso da ferramenta 5SGraph estendida. Esse modelo foi apresentado nos estudos de caso (ver subseção 4.3.1).

O objetivo desse protótipo é armazenar e gerenciar documentos legais da UNICAMP referentes à educação para que sejam acessíveis a usuários a partir da realização de buscas usando palavras-chave. Além disso, disponibilizar *links* para navegações entre documentos legais relacionados.

A definição formal dos conceitos desse protótipo, de acordo com o formalismo 5S, é descrita na Tabela 5.5.

Tabela 5.5: Conceitos da Biblioteca Digital de Documentos Legais da UNICAMP de acordo com o formalismo 5S.

Conceito 5S	Biblioteca Digital de Documentos Diretivos Educacionais da UNICAMP
<i>Societies</i>	Funcionários da DAC, alunos, professores
<i>Streams</i>	Texto das Deliberações/Portarias
<i>Structures</i>	Padrão de Metadados, XML, RDF
<i>Spaces</i>	Espaço Vetorial para indexação e busca de documentos legais
	Base de dados para armazenamento e classificação dos documentos
<i>Scenarios</i>	Busca (por palavras-chave)
	Visualização (documento inteiro; artigos)
	Navegação (<i>links</i> entre documentos relacionados)

O desenvolvimento desta biblioteca digital foi baseado no projeto LexML [27]. Este projeto utiliza XML para modelar os documentos legais e URN para identificar univocamente um documento. Além disso, ele utiliza a ontologia FRBR_{OO} para identificar um documento legal ou uma versão consolidada do texto normativo.

Nesse protótipo, apenas documentos da UNICAMP referentes ao Regimento Geral dos Cursos de Graduação são gerenciados.

Desenvolvimento

As seguintes etapas foram realizadas para o gerenciamento desses documentos legais:

- *parser* dos textos das Deliberações/Portarias;
- geração de documentos XML;
- inserção dos documentos XML em um banco de dados;
- criação de interface para buscar, consultar e navegar pelos documentos.

Como os documentos legais foram idealizados em XML, o XMLSchema do projeto LexML foi utilizado como modelo de dados, pois ele já define a estrutura de textos normativos do país segundo as regras de elaboração, redação e consolidação das leis nacionais [13,14]. Portanto, esse XMLSchema engloba todos os possíveis elementos pertencentes a um texto normativo brasileiro, englobando também a modelagem dos textos legais publicados pela UNICAMP (Deliberações e Portarias), não precisando ser alterado para contemplá-los. Além disso, o XMLSchema segue a especificação dos objetos digitais no modelo definido com o uso da ferramenta 5SGraph, já que obedece as regras de hierarquia e de estrutura dos dispositivos definidas pelas Leis Complementares. A Figura 5.15 exibe a especificação do dispositivo Artigo (*Article*) no XMLSchema, mostrando que o Artigo pode possuir um título, possui um rótulo (numeração) e pode ser composto por dispositivos filhos (generalizados no grupo *LXcontainers*).

```

<xsd:complexType name="ArticleType" >
  <xsd:sequence >
    <xsd:element ref="TituloArtigo" minOccurs="0" maxOccurs="1" />
    <xsd:element ref="Rotulo" minOccurs="1" maxOccurs="1" />
    <xsd:choice>
      <xsd:sequence >
        <xsd:group ref="LXcontainers" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:element ref="DispositivoGenerico" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:choice>
  </xsd:sequence>
  <xsd:attributeGroup ref="corereqArt"/>
  <xsd:attributeGroup ref="linkopt"/>
</xsd:complexType>
<xsd:group name="LXcontainers">
  <xsd:choice>
    <xsd:element ref="_Caput"/>
    <xsd:element ref="_Paragrafo"/>
    <xsd:element ref="_Inciso"/>
    <xsd:element ref="_Alinea"/>
    <xsd:element ref="_Item"/>
  </xsd:choice>
</xsd:group>

```

Figura 5.15: Parte do XMLSchema de documentos legais. Fonte [27].

O *parser* dos textos das Deliberações e das Portarias foi realizado a partir dos textos publicados no *site* da Procuradoria Geral da UNICAMP¹. Da mesma forma que o modelo de dados, o *parser* do projeto LexML também foi aproveitado como base para a criação do *parser* dos documentos legais da UNICAMP. Entretanto, o *parser* teve que ser alterado para contemplar diferenças na apresentação dos textos legais da UNICAMP. Algumas diferenças incluem:

¹http://www.pg.unicamp.br/deliberacoes_consult.php (último acesso em 20/01/2012).

- as assinaturas (nome e cargo) de um documento legal, de acordo com as Leis Complementares, são exibidas no final do documento. Porém, no texto apresentado para as Deliberações e para as Portarias da UNICAMP, as assinaturas são exibidas no início do documento;
- a ordem dos dados de uma assinatura, em um documento legal, é nome e cargo da autoridade; no texto da UNICAMP, a ordem é inversa, cargo e nome da autoridade;
- a parte final de um documento legal possui local, data e assinaturas; para o texto dos documentos da UNICAMP, a parte final inclui apenas o texto “Publicado no DOE em <data>”;
- o artigo em um documento oficial é apresentado com a nomenclatura “Art. <número> - *caput*”; no documento da UNICAMP, o artigo é apresentado com a nomenclatura “Artigo <número> *caput*”;
- a omissão de texto é definida no documento legal com uma linha contendo vários pontos (.....); para o documento da UNICAMP, o texto da omissão textual também pode conter ponto e vírgula no final da linha;
- alterações em um documento legal são indicadas por aspas duplas sem pontuação no final; o texto normativo da UNICAMP pode conter pontuação após as aspas duplas.

Além dessas diferenças, notou-se que os textos legais da UNICAMP não seguem rigorosamente as Leis Complementares, pois há ocorrências de: alíneas como nós filhos de parágrafos; letras das alíneas sem o uso de parênteses; incisos com ponto ao invés de traço após o número romano; entre outras. Por causa desses erros estruturais, dificuldades foram encontradas na geração automática desse tipo de objetos digitais, comprometendo a leitura automatizada do texto. Nesses casos, correções manuais são necessárias para que os arquivos XML sejam gerados.

Tanto o *parser* quanto à geração dos arquivos XML foram implementados utilizando o *software* ANTLRWorks¹. Além disso, os arquivos fonte do projeto LexML foram utilizados como base para o desenvolvimento, sofrendo alterações específicas para contemplar os documentos legais da UNICAMP. As Figuras 5.16 e 5.17 exibem partes (respectivamente, metadados e parte normativa) do arquivo XML gerado referente à Deliberação CONSU-A-017/2010.

Após a transformação dos textos normativos para arquivos XML, validações dos arquivos XML sobre o arquivo XMLSchema do projeto LexML foram realizadas para analisar

¹<http://www.antlr.org/works/index.html> (último acesso em 20/01/2012).

```

<?xml version="1.0" encoding="UTF-8"?>
<LexML xmlns="http://www.lexml.gov.br/1.0" xmlns:xsd="http://www.w3.org/2001/XMLSchema-instance"
  <Metadado>
    <Identificacao xml:base="http://www.lexml.gov.br/" FRBRitemRef="self">
      <JurisdicaoAutoridade xlink:href="/resources/localidade.rdf#br;sao.paulo;campinas" />
      <Autoridade xlink:href="/resources/autoridade.rdf#procuradoria.geral.unicamp" />
      <TipoDocumento xlink:href="/resources/tipoDocumento.rdf#br:estadual.unicamp:deliberacao" />
      <IdentificadorDocumento>
        <DataRepresentativa>2010-11-30</DataRepresentativa>
        <Identificador>Deliberação CONSU-A-017/2010</Identificador>
      </IdentificadorDocumento>
      <DataVersao>2010-11-30</DataVersao>
      <DataVisao>2010-11-30</DataVisao>
      <FormaConteudo>
        <TipoConteudo xlink:href="/resources/tipoConteudo.rdf#texto" />
        <Lingua xlink:href="/resources/lingua.rdf#pt-br" />
      </FormaConteudo>
    </Identificacao>
    <Contexto>
      <FRBRObraComplexa URN="br:estadual.unicamp:deliberacao;consu;CONSU-A-017-2010" showAs="1" />
      <Apelido URN="br:estadual.unicamp:deliberacao;consu;regimento.geral.cursos.graduacao" />
      <FRBRObraIndividual URN="br:estadual.unicamp:deliberacao;consu;CONSU-A-017-2010" showAs="1" />
      <FRBRItem id="self" mimeType="text/xml" xlink:href="/resources/tipoConteudo.rdf#texto" />
    </FRBRObraIndividual>
    </FRBRObraComplexa>
  </Contexto>
</Metadado>

```

Figura 5.16: Parte do arquivo XML de uma Deliberação da UNICAMP mostrando os metadados.

a estrutura textual normativa dos arquivos gerados. Assim, foi possível a inserção dos mesmos em um banco de dados XML (eXist-db¹).

Para a implementação da interface, a linguagem de consulta a bancos de dados XML, *XML Query Language*², foi utilizada para acessar os arquivos no banco XML e, juntamente com JavaScript³, para exibir os resultados na interface. Lucene⁴ também foi utilizado para indexar os textos legais permitindo consultas mais rápidas.

A Figura 5.18 exibe a interface de busca do protótipo da biblioteca digital de documentos legais, exemplificando a procura de documentos legais que possuam o texto “modalidade”. A Figura 5.19 mostra os resultados encontrados, destacando onde a palavra-chave da busca se encontra no texto retornado.

¹<http://exist-db.org/> (último acesso em 20/01/2012).

²<http://www.ibiblio.org/xql/xql-proposal.html> (último acesso em 20/01/2012).

³<https://developer.mozilla.org/en/JavaScript/> (último acesso em 20/01/2012).

⁴<http://lucene.apache.org/> (último acesso em 20/01/2012).

```

<Norma>
  <ParteInicial>
    <Epigrafe id="epigrafe">
      <p>Deliberação CONSU-A-017/2010, de 30/11/2010</p>
    </Epigrafe>
    <Ementa id="ementa">
      <p>Altera o Regimento Geral dos Cursos de Graduação da UNICAMP</p>
    </Ementa>
    <Preambulo id="preambulo">
      <p>&lt;p&gt;O Reitor da Universidade Estadual de Campinas, na qualidade de Presid
    </Preambulo>
  </ParteInicial>
  <Articulacao>
    <Artigo id="art1" seqOrdem="1">
      <Rotulo>Art. 1º</Rotulo>
      <Caput id="art1_cpt">
        <p>Ficam alterados: artigo 8º, do Capítulo I - Disposições Preliminares - Seção
      <Alteracao>
        <AbreAspas>&amp;#8220;</AbreAspas>
        <Artigo id="art1_cpt_asp1_art8" seqOrdem="1">
          <Rotulo>Art. 8º</Rotulo>
          <Caput id="art1_cpt_asp1_art8_cpt">
            <p>As formas de ingresso nos cursos de graduação e Programas Especiais da
            <Inciso id="art1_cpt_asp1_art8_cpt_inc1" seqOrdem="1">
              <Rotulo>I -</Rotulo>
              <p>vestibular nacional realizado pela Comissão Permanente para os Vesti
            </Inciso>
            <Inciso id="art1_cpt_asp1_art8_cpt_inc2" seqOrdem="2">
              <Rotulo>II -</Rotulo>
              <p>preenchimento de vagas remanescentes em cursos de graduação por:</p>
            </Inciso>
          </Caput>
        </Artigo>
        <Omissis>
          <Rotulo>a)</Rotulo>
          <p>.....
        </Omissis>
        <Omissis>
          <Rotulo>b)</Rotulo>

```

Figura 5.17: Parte do arquivo XML de uma Deliberação da UNICAMP mostrando a parte normativa.



Figura 5.18: Interface de busca do protótipo de biblioteca digital de documentos legais.

RESULTADOS ENCONTRADOS: 3

1 - [Deliberação CONSU-A-005/2001](#)

VI - quando o aluno, tendo cursado número maior de semestre do que o previsto para o curso, não atingir o prazo máximo permitido (para se avaliar as chances de conclusão do curso, deve-se fazer por períodos de semestres de que ele ainda dispõe);

2 - [Deliberação CONSU-A-027/2002](#)

I - Outra habilitação ou modalidade do curso em que se graduou ainda que envolva período

3 - [Deliberação CONSU-A-011/1998](#)

Art. 9º O graduado em curso da UNICAMP poderá, a qualquer momento, nos períodos de matrícula, solicitar a reavaliação mediante aprovação da Coordenadoria do Curso.

§ 1º Ao aceitar o reingresso de um aluno para cumprir habilitação ou modalidade adicional,

§ 2º O aluno reingressante terá direito ao limite máximo de trancamentos previsto no Artigo 1º ou modalidade.

§ 6º Concluída a habilitação ou modalidade referente ao retorno, esta será apostilada no diploma.

Art. 22 Cada curso de graduação pode ter mais de uma modalidade/habilitação.

Figura 5.19: Resultados de uma busca no protótipo de biblioteca digital de documentos legais.

Capítulo 6

Conclusões

Este trabalho propôs a introdução de novos conceitos, *ComplexObject* e *Object*, no metamodelo de bibliotecas digitais para permitir a especificação de bibliotecas digitais de objetos complexos. O metamodelo anterior possibilitava a especificação apenas de objetos atômicos e compostos, ou seja, objetos simples. O novo conceito *ComplexObject* identifica um objeto complexo e o conceito *Object* representa o objeto que compõe o objeto complexo. Além disso, o conceito *Document* foi renomeado para *SimpleObject* para clarear a nomenclatura utilizada no metamodelo. Com essas alterações na criação do novo metamodelo, foi possível alterar também o metamodelo da ferramenta 5SGraph, criando as *tags* *ComplexObj* e *Object* e renomeando a *tag* *Document* para *SimpleObject*, para permitir que usuários da ferramenta possam especificar objetos complexos. Essa ferramenta é baseada no formalismo 5S, permitindo que o usuário especifique, formalmente, os conceitos principais de uma biblioteca digital. Além disso, arquivos na linguagem 5SL são gerados para posteriormente serem utilizados na geração automática de bibliotecas digitais. Neste trabalho, somente a especificação de objetos complexos foi analisada e resolvida. A geração automática não foi contemplada, mas uma biblioteca digital de documentos legais, que são objetos complexos típicos, foi implementada respeitando-se a estrutura dos documentos legais e dos seus dispositivos definida com o uso da ferramenta 5SGraph.

Além dessa implementação, estudos de caso e um experimento com usuários potenciais foram realizados sobre a especificação de objetos complexos na ferramenta 5SGraph a fim de validar e analisar os novos conceitos.

Conclui-se que o novo metamodelo pode ser utilizado, na ferramenta 5SGraph, para especificar instâncias de bibliotecas digitais que contenham objetos complexos.

6.1 Contribuições

As contribuições deste trabalho são:

- a especificação de um metamodelo para especificar bibliotecas digitais de objetos complexos;
- a implementação de um metamodelo para uso na ferramenta 5SGraph com o objetivo de especificar e instanciar bibliotecas digitais contendo objetos complexos;
- estudos de casos sobre o uso do metamodelo criado para especificar objetos complexos (documento sobreposto, documento legal, imagem e descritor de imagens) na ferramenta 5SGraph;
- a validação do uso do novo metamodelo na ferramenta 5SGraph por usuários potenciais;
- a demonstração de documentos legais como objetos complexos;
- a especificação e a implementação de um protótipo de biblioteca digital de objetos complexos do tipo documento legal, sendo esta biblioteca digital baseada no modelo gerado a partir do metamodelo criado.

6.2 Trabalhos Futuros

Além das contribuições, várias oportunidades de extensões e trabalhos futuros foram identificadas. Algumas sugestões de trabalhos futuros incluem:

- melhorias na ferramenta 5SGraph: validação da estrutura (*Structured_Stream*) indicada para *SimpleObject* e *ComplexObj* e, se objeto complexo, validação da estrutura também sobre os objetos digitais informados; especificação *top-down* dos objetos digitais para comparação com a especificação *bottom-up* atual;
- indicação na ferramenta 5SGraph da opcionalidade de componentes de um objeto digital. Por exemplo, um documento legal pode ser iniciado por um Capítulo e depois esse dispositivo é decomposto por seus dispositivos; ou um documento legal pode conter apenas artigos. Como indicar na ferramenta 5SGraph que um documento legal pode ser composto de uma maneira ou de outra?
- indicação, se existentes, de objetos e *streams* faltantes na especificação do modelo após o carregamento de um componente (reuso) na ferramenta 5SGraph;
- geração automática de bibliotecas digitais de objetos complexos pela ferramenta 5SGen [47] a partir do arquivo na linguagem 5SL gerado pela ferramenta 5SGraph: analisar as mudanças necessárias para que a ferramenta 5SGen gere também objetos complexos para uma biblioteca digital;

- finalização da biblioteca digital de documentos legais: conclusão de uma versão completa e funcional para que a DAC e outros usuários potenciais possam usufruir desse sistema em atividades que envolvam busca de documentos e de dispositivos legais; implementação dos serviços de navegação tanto entre documentos relacionados quanto navegação por versões temporais do texto consolidado;
- incremento de funcionalidades na biblioteca digital de documentos legais: aperfeiçoar a busca de documentos, incluindo busca estruturada e *relevance feedback*;
- validação da biblioteca digital de documentos legais por funcionários da DAC;
- formalização do versionamento de objetos digitais sob o aspecto temporal usando 5S. Por exemplo, um texto legal possui uma nova versão após a publicação de alterações por um documento legal. A versão anterior é válida até a data da publicação e a versão nova é válida a partir da data da publicação até a publicação de um novo documento legal. Como formalizar esse aspecto temporal associado a objetos digitais?

Apêndice A

Questionário

Documento para os participantes da pesquisa sobre especificação de objetos digitais complexos usando a ferramenta 5SGraph

Projeto de Mestrado

Especificação de Bibliotecas Digitais de Objetos Complexos

Aluna

Ticiane Oniki Toffoli

Orientador

Ricardo da Silva Torres

Proposta

Esta pesquisa está sendo realizada para analisar os impactos e validar as alterações realizadas no metamodelo genérico de bibliotecas digitais da ferramenta 5SGraph para permitir a especificação de objetos digitais complexos. Algumas tarefas na ferramenta 5SGraph serão requisitadas aos usuários para avaliação das alterações realizadas. As respostas do questionário final serão úteis para melhorar a especificação de objetos complexos e validar os conceitos adotados para a especificação desses objetos.

Procedimentos

Questionário para conhecimento do perfil dos participantes.

Introdução sobre bibliotecas digitais, formalismo 5S, ferramenta 5SGraph e objetos complexos.

Realização de tarefas na ferramenta 5SGraph.

Questionário final sobre as tarefas e sobre os novos conceitos introduzidos no meta-modelo.

Confidencialidade

Todos os dados serão utilizados apenas como dados para a pesquisa.

Responsabilidades dos participantes

Os participantes assumem que:

- participam voluntariamente dessa pesquisa;
- todas as tarefas serão finalizadas;
- os resultados serão encaminhados para os responsáveis pela pesquisa.

Perfil dos Participantes

Qual seu grau de instrução?

- a. Aluno de graduação
- b. Graduação Concluída
- c. Mestrando
- d. Mestrado Concluído
- e. Doutorando
- f. Doutorado Concluído
- g. Cursando outro tipo de Pós-Graduação
- h. Outro tipo de Pós-Graduação Concluída

Qual seu nível de conhecimento na língua inglesa?

- (1) Nenhum (2) Pouco (3) Razoável (4) Muito (5) Amplo domínio

Qual seu nível de conhecimento em Bibliotecas Digitais?

- (1) Nenhum (2) Pouco (3) Razoável (4) Muito (5) Amplo domínio

Qual seu nível de conhecimento no formalismo 5S (*Streams, Structures, Spaces, Scenarios, Societies*)?

- (1) Nenhum (2) Pouco (3) Razoável (4) Muito (5) Amplo domínio

Qual seu nível de conhecimento sobre o conceito de objetos digitais complexos?

- (1) Nenhum (2) Pouco (3) Razoável (4) Muito (5) Amplo domínio

Qual seu nível de conhecimento e prática na ferramenta 5SGraph?

(1) Nenhum (2) Pouco (3) Razoável (4) Muito (5) Amplo domínio

Você já criou/configurou uma biblioteca digital?

- a. Sim (escreva detalhes sobre a biblioteca)
- b. Não

Se resposta anterior foi “Sim”, responda as questões abaixo.

Alguma biblioteca continha objetos digitais complexos?

- a. Sim (escreva detalhes sobre os objetos complexos)
- b. Não

Como a especificação dos objetos digitais foi realizada? (possível a marcação de ambas respostas a e b)

- a. Pela ferramenta 5SGraph
- b. Por outra ferramenta de especificação: _____
- c. Sem uso de ferramenta de especificação

Introdução

Com o avanço e o uso em massa de novas tecnologias, grandes coleções de informações digitais vêm sendo produzidas e necessitam de aplicações para armazenar, processar e gerenciar esses dados. Conseqüentemente, diversas aplicações precisam tratar informações complexas, com grandes volumes de dados heterogêneos, como textos, imagens, vídeos, softwares, entre outros. Uma das formas de gerenciar esses dados é utilizar bibliotecas digitais, que são avançados e complexos sistemas de informação que disseminam informação [1] e oferecem busca estruturada, navegação e preservação de documentos. Além disso, são acessíveis em ambientes distribuídos (acesso remoto e compartilhado) para atender às necessidades de comunidades de usuários específicos [2].

As informações ou documentos em uma biblioteca digital são manifestados em forma de objetos digitais, apresentados em diversos tipos de conteúdo. Quando o objeto digital é composto de um ou mais objetos digitais, ele é denominado de objeto complexo. Portanto, um objeto complexo é um simples objeto digital ou uma composição recursiva de outros objetos complexos, necessitando de uma estrutura para organizar seus componentes [3].

Para o gerenciamento desses objetos digitais complexos surgem as bibliotecas digitais

de objetos digitais complexos. Por serem sistemas complexos englobando diversos tópicos, como armazenamento e recuperação de informação, interação humano-computador, interoperabilidade, computação distribuída, entre outros, formalismos vêm sendo utilizados para ajudar a modelar, definir e descrever bibliotecas digitais. O formalismo 5S [2,4] define os seguintes conceitos de bibliotecas digitais:

Streams: definem os tipos de dados que serão gerenciados; são sequências de itens arbitrários usados para descrever conteúdo estático e dinâmico (como *bits*, caracteres, imagens, *etc.*).

Structures: definem como a informação (*streams*) é estruturada e organizada.

Spaces: definem propriedades e operações da biblioteca.

Scenarios: definem o comportamento da biblioteca digital; consistem de sequências de eventos ou ações que modificam estados de uma computação para atender a um requisito funcional.

Societies: são conjuntos de atores, usuários, entidades e atividades e os relacionamentos entre eles.

A especificação e a implementação de uma biblioteca digital são tarefas cruciais para um gerenciamento de documentos com qualidade. Por isso, ferramentas são necessárias para auxiliar os desenvolvedores nessas tarefas e facilitar a construção de bibliotecas digitais. Várias ferramentas baseadas no formalismo 5S foram criadas, e entre elas, está a ferramenta 5SGraph [4,5,6], que permite a modelagem visual de bibliotecas digitais com as seguintes especificações: do conteúdo a ser armazenado; de como esse conteúdo deve ser organizado, estruturado, descrito e acessado; de quais serviços são oferecidos pela biblioteca; e de como os usuários utilizam e interagem com esses serviços [4]. Entretanto, essa ferramenta e nenhuma outra estão adequadas para a especificação de objetos complexos.

Para sanar a falta de ferramentas que especifiquem bibliotecas digitais de objetos complexos, foi realizado um aprimoramento na ferramenta 5SGraph, para que esta especifique uma biblioteca digital de objetos complexos, na linguagem 5SL [7]. Para isto, dois novos conceitos foram criados no metamodelo:

ComplexObj: representa um objeto digital complexo. É formado por pelo menos um objeto digital e pode possuir outras *streams*.

Object: representa um objeto da biblioteca digital, podendo ser um objeto simples ou um objeto complexo.

Desta maneira, usuários podem modelar visualmente uma biblioteca digital de objetos complexos, facilitando os processos de análise dos dados e viabilizando sua construção.

Portanto, para analisar a efetiva utilização desses novos conceitos na ferramenta 5SGraph, estamos realizando esta pesquisa.

Referências

[1] E. A. Fox, G. Marchionini. Toward a Worldwide Digital Library. *Communications of the ACM*, 41(4):29-32, April 1998.

[2] M. A. Gonçalves, E. A. Fox, L. T. Watson, N. A. Kipp. Streams, Structures, Spaces, Scenarios, Societies (5S): a Formal Model for Digital Libraries. *ACM Transactions on Information Systems*, 22(2):270-312, April 2004.

[3] U. Murthy. Digital Libraries with Superimposed Information: Supporting Scholarly Tasks that Involve Fine Grain Information. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2011.

[4] M. A. Gonçalves. Streams, Structures, Spaces, Scenarios, and Societies (5S): A Formal Digital Library Framework and its Applications. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2004.

[5] Q. Zhu. 5SGraph: a Modeling Tool for Digital Libraries. Master's thesis, Virginia Polytechnic Institute and State University, 2002.

[6] Q. Zhu, M. A. Gonçalves, E. A. Fox. 5SGraph Demo: a Graphical Modeling Tool for Digital Libraries. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries, JCDL '03*, pages 385-385, Washington, DC, USA, 2003. IEEE Computer Society.

[7] M. A. Gonçalves, E. A. Fox. 5SL: a Language for Declarative Specification and Generation of Digital Libraries. In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, JCDL '02*, pages 263-272, New York, NY, USA, 2002. ACM.

Tarefas

1) Especificar objetos digitais de uma biblioteca digital (sem o uso de objetos complexos).

Nesta tarefa, o participante deve especificar uma tese de doutorado com *streams* do tipo texto (referentes ao texto da tese) e *streams* do tipo imagem (referente às Figuras que compõem uma tese) usando a ferramenta 5SGraph.

2) Especificar objetos complexos de uma biblioteca digital.

Um *software*, na linguagem Java, foi criado por um aluno em sua pós-graduação e deve ser armazenado como um objeto digital para acesso direto. Para especificar uma tese com essa característica, ela deve ser criada como um objeto complexo, contendo o texto da tese e o objeto digital referente ao *software*. Especifique esta situação usando a ferramenta 5SGraph.

3) Especificar objetos complexos compostos por outros objetos complexos de uma biblioteca digital.

Para esta tarefa, o participante deve especificar um documento legal usando a ferramenta 5SGraph. Um documento legal possui *stream* do tipo texto e o Artigo é sua unidade básica. Entretanto, um artigo pode ser composto por Parágrafos ou Incisos; um Parágrafo por Incisos; um Inciso por Alíneas; uma Alínea por Itens. Todos esses dispositivos possuem *stream* do tipo texto e podem ser acessados independentes do artigo. Por causa dessa estrutura, o participante deve criar também os objetos digitais referentes aos dispositivos para comporem o documento legal.

Questionário Final

As perguntas seguintes devem ser respondidas após o uso da ferramenta 5SGraph com o metamodelo genérico modificado para a especificação de objetos complexos.

As instruções das tarefas estavam claras?

(1) Não (2) Confusas (3) Razoáveis (4) Partes duvidosas/ambíguas (5) Sim

Após o uso da ferramenta, qual seu nível de conhecimento sobre o conceito de objetos digitais complexos?

(1) Nenhum (2) Pouco (3) Razoável (4) Muito (5) Amplo domínio

A especificação de objetos complexos na ferramenta 5SGraph com o uso dos conceitos foi uma tarefa

(1) Muito simples (2) Simples (3) Intermediária (4) Complexa (5) Muito complexa

Você usaria outros nomes para os conceitos de objeto digital complexo na ferramenta 5SGraph?

- a. Sim (Quais?)
- b. Não

Após as tarefas, sente-se capaz de especificar outros objetos digitais complexos usando a ferramenta 5SGraph?

- a. Sim
- b. Não (Por que?)

Você acha que a especificação dos objetos digitais de uma biblioteca digital pode ser realizada na ferramenta 5SGraph de maneira

(1) Não atende o propósito (2) Incompleta (3) Razoável (4) Satisfatória (5) Completa

Sob o ponto de vista de utilidade, pode-se dizer que a ferramenta é

(1) Inútil (2) Pouco útil (3) Útil em alguns aspectos (4) Útil (5) Muito útil

Se você precisasse especificar novamente uma biblioteca digital, você usaria a ferramenta 5SGraph?

- a. Sim
- b. Não (Por que?)

Você teve algum problema ao usar a ferramenta 5SGraph?

Escreva, se desejar, sugestões e comentários sobre os novos conceitos ou sobre a ferramenta 5SGraph.

Agradecemos sua participação!

Referências Bibliográficas

- [1] Akoma Ntoso. Disponível em <http://www.akomantoso.org/>. Acessado em 20/02/2012.
- [2] CEN Metalex. Disponível em <http://www.metalex.eu/>. Acessado em 20/02/2012.
- [3] Deliberação CONSU-A-027/2002. Disponível em http://www.pg.unicamp.br/mostra_norma.php?consolidada=undefined&id_norma=2736/. Acessado em 20/02/2012.
- [4] Deliberações da UFSCar. Disponível em <http://www.ufscar.br/~soc/>. Acessado em 20/02/2012.
- [5] Deliberações da UNICAMP. Disponível em http://www.pg.unicamp.br/deliberacoes_consus.php. Acessado em 20/02/2012.
- [6] DSpace. Disponível em <http://www.dspace.org/>. Acessado em 20/02/2012.
- [7] Estatutos da UNICAMP. Disponível em <http://www.sg.unicamp.br/download/documentos-e-legislacoes/d2cfb229abdfa931b53833d9467ea5ec>. Acessado em 20/02/2012.
- [8] Estrella Project. Disponível em <http://www.estrellaproject.org/>. Acessado em 20/02/2012.
- [9] Fedora Commons. Disponível em <http://www.fedora-commons.org/>. Acessado em 20/02/2012.
- [10] Greenstone Digital Library Software. Disponível em <http://www.greenstone.org/>. Acessado em 20/02/2012.
- [11] Grupo de Informações Documentárias da UNESP. Disponível em <http://www.unesp.br/gid/>. Acessado em 20/02/2012.

- [12] HTML5. Disponível em <http://www.w3schools.com/html5/>. Acessado em 20/02/2012.
- [13] Lei Complementar nº 107/2001. Disponível em http://www.planalto.gov.br/ccivil_03/Leis/LCP/Lcp107.htm. Acessado em 20/02/2012.
- [14] Lei Complementar nº 95/1998. Disponível em http://www.planalto.gov.br/ccivil_03/Leis/LCP/Lcp95.htm. Acessado em 20/02/2012.
- [15] LexML Brasil - Parte 1 - Apresentação. Disponível em <http://projeto.lexml.gov.br/documentacao/Apresentacao.pdf>. Acessado em 20/02/2012.
- [16] LexML Brasil - Parte 1 - Modelo de Referência. Disponível em <http://projeto.lexml.gov.br/documentacao/Parte-1-Modelo-de-Referencia.pdf>. Acessado em 20/02/2012.
- [17] LexML Brasil - Parte 2 - lexML URN. Disponível em <http://projeto.lexml.gov.br/documentacao/Parte-2-LexML-URN.pdf>. Acessado em 20/02/2012.
- [18] LexML Brasil - Parte 3 - lexML XML Schema. Disponível em <http://projeto.lexml.gov.br/documentacao/Parte-3-XML-Schema.pdf>. Acessado em 20/02/2012.
- [19] LexML Brasil - Parte 4 - Coleta de Metadados. Disponível em <http://projeto.lexml.gov.br/documentacao/Parte-4-Coleta-de-Metadados.pdf>. Acessado em 20/02/2012.
- [20] LexML Brasil - Parte 5 - Serviço de Resolução de URN. Disponível em <http://projeto.lexml.gov.br/documentacao/Parte-5-Servico-de-Resolucao-de-URN.pdf>. Acessado em 20/02/2012.
- [21] LexML Brasil - Parte 6 - Vocabulários Controlados. Disponível em <http://projeto.lexml.gov.br/documentacao/Parte-6-Vocabularios-Controlados.pdf>. Acessado em 20/02/2012.
- [22] LexML Brasil - Rede de Informação Legislativa e Jurídica. Disponível em <http://www.lexml.gov.br/>. Acessado em 20/02/2012.
- [23] Metadata Encoding and Transmission Standard. Disponível em <http://www.loc.gov/standards/mets/>. Acessado em 20/02/2012.
- [24] Normas USP. Disponível em <http://leginf.uspnet.usp.br/>. Acessado em 20/02/2012.

- [25] Norme in Rete. Disponível em <http://www.normeinrete.it/>. Acessado em 20/02/2012.
- [26] Portarias da UNICAMP. Disponível em <http://www.pg.unicamp.br/portarias.php>. Acessado em 20/02/2012.
- [27] Rede de Informação Legislativa e Jurídica - Projeto LexML Brasil. Disponível em <http://projeto.lexml.gov.br/>. Acessado em 20/02/2012.
- [28] Regimento geral da UFSCar. Disponível em http://www.ufscar.br/~soc/documentos/regimento_ufscar.pdf. Acessado em 20/02/2012.
- [29] Regimento geral da UNICAMP. Disponível em <http://www.sg.unicamp.br/download/documentos-e-legislacoes/2db7456fd9ef8d40bc1891d21ffffa1ee>. Acessado em 20/02/2012.
- [30] Resoluções da UNICAMP. Disponível em <http://www.pg.unicamp.br/resolucoes.php>. Acessado em 20/02/2012.
- [31] Web Ontology Language (OWL). Disponível em <http://www.w3.org/TR/owl-features/>. Acessado em 20/02/2012.
- [32] XMLeges. Disponível em <http://www.xmleges.org/eng/>. Acessado em 20/02/2012.
- [33] Manual de Redação da Presidência da República, 2002. Disponível em http://www.planalto.gov.br/ccivil_03/manual/manual.htm. Acessado em 20/02/2012.
- [34] C. Awre. Managing Compound Objects Within Fedora. 2009. Enhanced E-theses Project Deliverable 9, Universiteit Utrecht. Disponível em <http://igiturarchive.library.uu.nl/DARLIN/2010-0526-200241/UUindex.html>. Acessado em 20/02/2012.
- [35] A. Boer, R. Winkels, and F. Vitali. Metalex XML and the Legal Knowledge Interchange Format. In *Computable Models of the Law*, volume 4884 of *Lecture Notes in Computer Science*, pages 21–41. Springer Berlin Heidelberg, 2008.
- [36] R. da S. Torres and A. X. Falcão. Content-Based Image Retrieval: Theory and Applications. *Revista de Informática Teórica e Aplicada*, 13:161–185.
- [37] A. T. da Silva, L. P. Magalhães, and A. X. Falcão. Recuperação de Imagens por Realimentação de Relevância. In *Encontro dos Alunos e Docentes do Departamento de Engenharia de Computação e Automação Industrial*. Segundo Encontro dos Alunos e Docentes do Departamento de Engenharia de Computação e Automação Industrial, 2009.

- [38] P. Dourado, P. Ferreira, and A. Santanchè. Representação Unificada de Objetos Digitais Complexos: Confrontando o RAS com o IMS CP. In *III Workshop de Bibliotecas Digitais, Webmedia 2007*, 2006.
- [39] E. A. Fox and G. Marchionini. Toward a Worldwide Digital Library. *Communications of the ACM*, 41(4):29–32, April 1998.
- [40] M. A. Gonçalves. *Streams, Structures, Spaces, Scenarios, and Societies (5S): A Formal Digital Library Framework and its Applications*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2004.
- [41] M. A. Gonçalves and E. A. Fox. 5SL: a Language for Declarative Specification and Generation of Digital Libraries. In *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, JCDL '02*, pages 263–272, New York, NY, USA, 2002. ACM.
- [42] M. A. Gonçalves, E. A. Fox, L. T. Watson, and N. A. Kipp. Streams, Structures, Spaces, Scenarios, Societies (5S): a Formal Model for Digital Libraries. *ACM Transactions on Information Systems*, 22(2):270–312, April 2004.
- [43] M. A. Gonçalves, B. L. Moreira, E. A. Fox, and L. T. Watson. “What is a Good Digital Library?” - A Quality Model for Digital Libraries. *Information Processing and Management*, 43(5), 2007.
- [44] D. Gorton. Practical Digital Library Generation into DSpace with the 5S Framework. Master’s thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2010.
- [45] T. R. Gruber. Toward Principles for the Design of Ontologies Used for Knowledge Sharing. *International Journal of Human-Computer Studies - Special issue: the role of formal ontology in the information technology*, 43(5-6):907–928, December 1995.
- [46] CIDOC CRM Harmonisation International Working Group on FRBR. FRBR Object-Oriented Definition and Mapping to FRBRER (version 0.9 draft), January 2008. Disponível em http://www.cidoc-crm.org/docs/frbr_oo/frbr_docs/FRBR_oo_v0.9.pdf. Acessado em 13/02/2011.
- [47] R. Kelapure, M. A. Gonçalves, and E. A. Fox. Scenario-Based Generation of Digital Library Services. Technical report, Presented at European Conference on Digital Libraries (ECDL 2003), 2003.

- [48] N. Kozievitch and R. da S. Torres. Describing OAI-ORE from the 5S Framework Perspective. In *Proceedings of the role of digital libraries in a time of global change, and 12th international conference on Asia-Pacific digital libraries, ICADL '10*, pages 260–261, Berlin, Heidelberg, 2010. Springer-Verlag.
- [49] N. P. Kozievitch. Complex Objects in Digital Libraries. In *Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries, Doctoral Consortium, JCDL '09*, 2009. Disponível em <http://www.ieee-tcdl.org/Bulletin/v5n3/Kozievitch/kozievitch.html>. Acessado em 20/02/2012.
- [50] N. P. Kozievitch. *Objetos Complexos em Bibliotecas Digitais: Analisando o Gerenciamento de Componentes de Imagens*. PhD thesis, Universidade Estadual de Campinas, Campinas, SP, 2011.
- [51] N. P. Kozievitch, J. Almeida, R. da S. Torres, N. J. Leite, M. A. Gonçalves, U. Murthy, and E. A. Fox. Towards a Formal Theory for Complex Objects and Content-Based Image Retrieval. *Journal of Information and Data Management*, 2(3):321–336, 2011.
- [52] N. P. Kozievitch, E. A. Fox, and R. da S. Torres. Analyzing Compound Object Technologies from the 5S Perspective. Technical Report IC-11-01, Universidade Estadual de Campinas, January 2011.
- [53] N. P. Kozievitch, D. C. G. Pedronette, R. da S. Torres, A. Santanchè, E. A. Fox, and R. T. Calumby. A Complex-Based Infrastructure for Searching and Harvesting Complex Image Objects. *Information-Interaction-Intelligence (I3 Journal)*, 11:39–68, 2011.
- [54] C. Lagoze, S. Payette, E. Shin, and C. Wilper. Fedora: an Architecture for Complex Objects and their Relationships. *International Journal on Digital Libraries*, 6(2):124–138, April 2006.
- [55] C. Lagoze and H. V. Sompel. Compound Information Objects: the OAI-ORE Perspective. 2007. Open Archives Initiative Object Reuse and Exchange, White Paper. Disponível em <http://www.openarchives.org/ore/documents>. Acessado em 20/02/2012.
- [56] Y. Ma, E. A. Fox, and M. A. Gonçalves. Personal Digital Library: PIM through a 5S Perspective. In *Proceedings of the ACM first Ph.D. workshop in CIKM, PIKM '07*, pages 117–124, New York, NY, USA, 2007. ACM.

- [57] B. L. Moreira, M. A. Gonçalves, A. H. F. Laender, and E. A. Fox. 5SQual: a Quality Assessment Tool for Digital Libraries. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '07, pages 513–513, New York, NY, USA, 2007. ACM.
- [58] B. L. Moreira, M. A. Gonçalves, A. H. F. Laender, and E. A. Fox. Evaluating Digital Libraries with 5SQual. In *ECDL*, pages 466–470, 2007.
- [59] U. Murthy. *Digital Libraries with Superimposed Information: Supporting Scholarly Tasks that Involve Fine Grain Information*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2011.
- [60] U. Murthy, D. Gorton, R. da S. Torres, M. A. Gonçalves, E. A. Fox, and L. Delcambre. Extending the 5S Digital Library (DL) Framework: From a Minimal DL towards a DL Reference Model. In *1st Workshop on Digital Library Foundations, ACM IEEE Joint Conference on Digital Libraries*, Vancouver, Canada, June 2007.
- [61] U. Murthy, N. P. Kozievitch, J. Leidig, R. da S. Torres, S. Yang, M. A. Gonçalves, L. Delcambre, D. Archer, and E. A. Fox. Extending the 5S Framework of Digital Libraries to Support Complex Objects, Superimposed Information, and Content-Based Image Retrieval Services. Technical Report TR-10-05, Virginia Tech, Department of Computer Science, April 2010.
- [62] U. Murthy, L. T. Li, E. Hallerman, E. A. Fox, M. A. Pérez-Quñones, L. M. Delcambre, and R. da S. Torres. Use of Subimages in Fish Species Identification: a Qualitative Study. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, JCDL '11, pages 185–194, New York, NY, USA, 2011. ACM.
- [63] S. H. Park, N. Lynberg, J. Racer, P. McElmurray, and E. A. Fox. HTML5 ETDs. In *Proceedings of International Symposium on Electronic Thesis and Dissertations*, Austin, TX, USA, 2010.
- [64] D. C. G. Pedronette. Uma Plataforma de Serviços de Recomendação para Bibliotecas Digitais. Master's thesis, Universidade Estadual de Campinas, Campinas, SP, 2008.
- [65] O. Penatti and R. da S. Torres. Eva: an Evaluation Tool for Comparing Descriptors in Content-Based Image Retrieval Tasks. In *Proceedings of the international conference on Multimedia information retrieval*, MIR '10, pages 413–416, New York, NY, USA, 2010. ACM.

- [66] R. Shen. *Applying the 5S Framework to Integrating Digital Libraries*. PhD thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2006.
- [67] R. Shen, N. S. Vemuri, W. Fan, R. da S. Torres, and E. A. Fox. Exploring Digital Libraries: Integrating Browsing, Searching, and Visualization. In *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital libraries*, pages 1–10. ACM Press, 2006.
- [68] R. Shen, N. S. Vemuri, W. Fan, and E. A. Fox. Integration of Complex Archeology Digital Libraries: an ETANA-DL Experience. *Information Systems*, 33(7-8):699–723, 2008.
- [69] M. Smith, M. Barton, M. Bass, M. Branschofsky, G. McClellan, D. Stuve, R. Tansley, and J. H. Walker. DSpace An Open Source Dynamic Digital Repository. *D-Lib Magazine*, 9(1), January 2003.
- [70] I. H. Witten and D. Bainbridge. A Retrospective Look at Greenstone: Lessons from the First Decade. In *Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '07, pages 147–156, New York, NY, USA, 2007. ACM.
- [71] Q. Zhu. 5SGraph: a Modeling Tool for Digital Libraries. Master's thesis, Virginia Polytechnic Institute and State University, 2002.
- [72] Q. Zhu, M. A. Gonçalves, and E. A. Fox. 5SGraph Demo: a Graphical Modeling Tool for Digital Libraries. In *Proceedings of the 3rd ACM/IEEE-CS joint conference on Digital libraries*, JCDL '03, pages 385–385, Washington, DC, USA, 2003. IEEE Computer Society.