

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE CIÊNCIAS EXATAS E DA NATUREZA
DEPARTAMENTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

**CoreKTV - Uma infraestrutura baseada em
conhecimento para TV Digital Interativa: um
estudo de caso para o *middleware* Ginga**

JÔNATAS PEREIRA CABRAL DE ARAUJO

JOÃO PESSOA
Setembro - 2011

UNIVERSIDADE FEDERAL DA PARAÍBA
CENTRO DE CIÊNCIAS EXATAS E DA NATUREZA
DEPARTAMENTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

JÔNATAS PEREIRA CABRAL DE ARAUJO

**CoreKTV - Uma infraestrutura baseada em
conhecimento para TV Digital Interativa: um
estudo de caso para o *middleware* Ginga**

Dissertação apresentada ao Programa de Pós-Graduação em Informática da Universidade Federal da Paraíba, como requisito parcial para a obtenção do título Mestre em Informática (Sistemas de Computação).

Linha de Pesquisa: Computação Distribuída

Orientadora: Prof. Dra. Natasha Correia Queiroz Lino.

JOÃO PESSOA
Setembro – 2011

A663c Araújo, Jônatas Pereira Cabral de.

CoreKTV – uma infraestrutura baseada em conhecimento para TV Digital Interativa: um estudo de caso para o Middleware Ginga / Jônatas Pereira Cabral de Araújo.-- João Pessoa, 2011.

139f. : il.

Orientadora: Natasha Correia Queiroz Lino

Dissertação (Mestrado) – UFPB/CCEN

1. Multimídia (Informática). 2. CoreKTV. 3. TV Digital Interativa. 4. Ginga. 5. TV Semântica. 6. Semântica Multimídia.

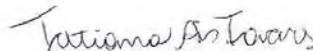
1

Ata da Sessão Pública de Defesa de Dissertação de Mestrado do JÔNATAS PEREIRA CABRAL DE ARAUJO, candidato ao Título de Mestre em Informática na Área de Sistemas de Computação, realizada em 23 de setembro de 2011.

2

3
4 Aos vinte e três dias do mês de setembro do ano dois mil e onze, às quatorze e trinta horas,
5 no Auditório do CCEN - da Universidade Federal da Paraíba, reuniram-se os membros da
6 Banca Examinadora constituída para examinar o candidato ao grau de Mestre em
7 Informática, na área de "Sistemas de Computação", na linha de pesquisa "Computação
8 Distribuída", o Sr. JÔNATAS PEREIRA CABRAL DE ARAUJO. A comissão
9 examinadora composta pelos professores doutores: Natasha Correia Queiroz Lino (DI -
10 UFPB), Orientador e Presidente da Banca Examinadora, Claurton de Albuquerque Siebra
11 (DI-UFPB) como examinador interno e Aquiles Burlamarqui (UFRN), como examinador
12 externo. Dando início aos trabalhos, a Prof^ª. Natasha Correia Queiroz Lino, cumprimentou
13 os presentes, comunicou aos mesmos a finalidade da reunião e passou a palavra ao
14 candidato para que o mesmo fizesse, oralmente, a exposição do trabalho de dissertação
15 intitulado "CoreKTV - Uma infraestrutura baseada em conhecimento para TV Digital
16 Interativa: um estudo de caso para o middleware Ginga". Concluída a exposição, o
17 candidato foi argüido pela Banca Examinadora que emitiu o seguinte parecer: "aprovado".
18 Assim sendo, deve a Universidade Federal da Paraíba expedir o respectivo diploma de
19 Mestre em Informática na forma da lei e, para constar, a professora Tatiana Aires Tavares,
20 Sra. Coordenadora do PPGI, lavrou a presente ata, que vai assinada por ela, e pelos
21 membros da Banca Examinadora. João Pessoa, 23 de setembro de 2011.
22


23


Tatiana Aires Tavares

24

25

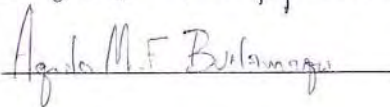
Prof.ª. Dra. Natasha Correia Queiroz Lino
Orientador (DI-UFPB)



Prof. Dr. Claurton de Albuquerque Siebra
Examinador Interno (DI-UFPB)



Prof. Dr. Aquiles Burlamarqui
Examinador Externo (UFRN)



26

*Dedico este trabalho aos meus pais,
José Pereira e Margareth, por todo o
amor, atenção e ensinamentos
passados em minha vida.*

Agradecimentos

Primeiramente à Deus, pelo dom da vida e por permitir ter forças, a cada dia, para buscar superar as dificuldades e obstáculos que surgem para atingir os objetivos em nossa vida.

Aos pais, José Pereira e Margareth, pelo apoio e suporte em todos os momentos desta jornada.

À minha orientadora, Natasha Queiroz, pela paciência, apoio e disposição para ajudar ao longo dos últimos três anos.

À minha grande amiga e namorada Jamilli, pela cumplicidade, companheirismo e incansável apoio e paciência nos momentos difíceis e estressantes dessa jornada e pelas alegrias e bons momentos compartilhados.

Aos meus irmãos e aos meus primos pelas constantes palavras de incentivo.

Aos meus grandes amigos da Quinta dos Amigos, Alysson (Boy), Arthur (Cabeção), Bruno (Galego), Eduardo (Careca), Diego Pizzol (Gago), Rafael Targino, Rodrigo Cartaxo e Renan Cartaxo, pela amizade.

Aos companheiros do projeto *Knowledge TV*, em especial a Junior e Ramon, pelas sugestões e críticas que, certamente, tornaram mais ricas as contribuições obtidas neste trabalho.

Aos integrantes do Lavid, pela ótima convivência de trabalho ao longo dos últimos quatro anos, em especial aos professores Guido Lemos e Raoni Kulesza, pelas oportunidades oferecidas, e aos amigos André Felipe, Caio Caroca e Alan Lívio, pelas discussões, sugestões e críticas no trabalho diário.

À todos que, direta ou indiretamente, contribuíram neste trabalho.

RESUMO

O surgimento da TV Digital promoveu um cenário de aumento da quantidade de canais, serviços e conteúdo disponíveis ao usuário. Entretanto, tais mudanças não foram refletidas na forma como o conteúdo multimídia é descrito. O modelo tradicional apresenta uma estrutura sintática, no qual as informações não possuem semântica, além do seu formato de representação para transmissão que não permite interoperabilidade com outros sistemas. Com a convergência entre as plataformas de TV e Web esse cenário passa a ser ainda mais problemático, uma vez que na Web já há um movimento em curso com o objetivo de tornar os sistemas interoperáveis e o formato de representação homogêneo. Neste cenário, este trabalho propõe uma infraestrutura para modelagem de conteúdo multimídia em TVDI baseada em conhecimento alinhado aos conceitos e padrões da Web Semântica, visando à integração entre as plataformas de TVDI e Web, aplicada ao middleware do padrão brasileiro, o Ginga.

Palavras-chave: CoreKTV; Knowledge TV; KTV; TV Digital Interativa; Ginga; TV Semântica; Semântica Multimídia; Ontologia.

ABSTRACT

The advent of Digital TV has promoted a scenario of increasing number of channels, content and services available to the user. However, these changes were not reflected in the way multimedia content is described. The traditional model presents a syntactic structure, which have no semantic information. In addition, its representation format for transmission does not allow interoperability with other systems. With the convergence between TV and web platforms that scenario becomes even more problematic, since there is a movement in the Web, leaded by the World Wide Web Consortium, in order to make systems interoperable and based on homogeneous representation format. In this scenario, this work proposes a knowledge based infrastructure for modeling multimedia content in TVDI, aligned with the concepts and standards of the Semantic Web aiming to integrate the platform and Web TVDI, applied to the Brazilian middleware standard – the Ginga middleware.

Keywords: CoreKTV; KnowledgeTV; KTV; Interactive Digital TV; Ginga ; Semantic TV; Multimedia Semantic; Ontology.

Lista de Siglas

Sigla	Significado
ACAP	<i>Advanced Common Application Platform</i>
API	<i>Aplication Program Interface</i>
ARIB	<i>Association of Radio Industries and Businesses</i>
ATSC	<i>Advanced Television Systems Committee</i>
BIT	<i>Broadcaster Information Table</i>
CAT	<i>Conditional Acces Table</i>
CTIC	Centro de Pesquisa e Desenvolvimento em Tecnologias Digitais para a Informação e Comunicação
DASE	<i>DTV Application Software Environment</i>
DDL	<i>Description Definition Language</i>
DSM-CC	<i>Digital Storage Media Command and Control Protocol</i>
DTMB	<i>Digital Terrestrial Multimedia Broadcast</i>
DVB	<i>Digital Video Broadcast</i>
EIT	<i>Event Information Table</i>
HDTV	<i>High Definition TV</i>
HTML	<i>Hypertext Markup Language</i>
ISDB	<i>Integrated Services Digital Broadcasting</i>
ITU	<i>International Telecommunication Union</i>
JVM	<i>Java Virtual Machine</i>
KDD	<i>Knowledge Discovery Database</i>
KTV	<i>Knowledge TV</i>
LDTV	<i>Low Definition TV</i>
LGPL	<i>Lesser General Public License</i>
MCT	Ministério da Ciência e Tecnologia
MHP	<i>Multi Home Platform</i>

MPEG	Moving Picture Experts Group
NCL	<i>Nested Context Language</i>
NIT	<i>Network Information Table</i>
OCAP	<i>OpenCable Application Platform</i>
OLAP	<i>Online Analytics Processing</i>
OWL	<i>Ontology Web Language</i>
PAT	<i>Program Association Table</i>
PDA	<i>Personal Digital Assistant</i>
PES	<i>Packetized Elementary Stream</i>
PMT	<i>Program Map Table</i>
PSI	<i>Program Specific Information</i>
PUC-RJ	Pontifícia Universidade Católica – Rio de Janeiro
RDF	<i>Resource Description Framework</i>
RDFS	<i>Resource Description Framework Schema</i>
RNP	Rede Nacional de Pesquisa
SBTVD	Sistema Brasileiro de TV Digital
SDT	<i>Service Description Table</i>
SDTV	<i>Standard Definition Table</i>
SI	<i>Service Information</i>
STB	<i>Set Top Box</i>
TOT	<i>Time Offset Table</i>
TS	<i>Transport Stream</i>
TVDI	TV Digital Interativa
UFPB	Universidade Federal da Paraíba
URI	<i>Uniform Resource Identifier</i>
W3C	<i>World Wide Web Consortium</i>

XML *eXtensible Markup Language*

Lista de Figuras

Figura 1	- Sistema de TV Digital Interativa	25
Figura 2	- Processo de geração do sinal da TVDI	26
Figura 3	- Arquitetura de receptor de TVDI	28
Figura 4	- Arquitetura de referência para <i>middleware</i> de TVDI	29
Figura 5	- Arquitetura conceitual do <i>middleware</i> Ginga	31
Figura 6	- Camadas lógicas da Web Semântica	38
Figura 7	- Classificação de ontologias quanto à generalidade	40
Figura 8	- Exemplo de sentença RDF	43
Figura 9	- Modelo de dados RDF	43
Figura 10	- Exemplo de uso de primitivas RDF-Schema	45
Figura 11	- Declaração de classe OWL	46
Figura 12	- Declaração de indivíduo usando primitivas RDF	47
Figura 13	- Declaração de instância em uma ontologia	47
Figura 14	- Visão do <i>Knowledge TV</i> em um ambiente de TVDI	50
Figura 15	- Arquitetura conceitual do <i>Knowledge TV</i>	52
Figura 16	- Arquitetura <i>Knowledge TV</i> em um MW de TVDI	53
Figura 17	- Arquitetura dos módulos do <i>Knowledge TV</i>	54
Figura 18	- Visão geral do processo de construção da Ontologia Núcleo do KTV	56
Figura 19	- Visão geral da Ontologia Núcleo do KTV	64
Figura 20	- Hierarquia de classes para <i>ContentDescription</i>	65
Figura 21	- Hierarquia <i>Genre</i>	66
Figura 22	- Hierarquia <i>EntertainmentGenre</i>	68
Figura 23	- Hierarquia <i>FictionGenre</i>	68
Figura 24	- Hierarquia <i>InformativeGenre</i>	69
Figura 25	- Hierarquia <i>LifestyleGenre</i>	70
Figura 26	- Hierarquia <i>MusicGenre</i>	71
Figura 27	- Hierarquia <i>SportsGenre</i>	72
Figura 28	- Hierarquia <i>Language</i>	74
Figura 29	- Hierarquia <i>Classification</i>	74
Figura 30	- Visão completa da hierarquia <i>Classification</i>	75
Figura 31	- Hierarquia <i>IntendedAudience</i>	76
Figura 32	- Hierarquia <i>AgeGroupAudience</i>	77
Figura 33	- Hierarquia <i>EducationStandardAudience</i>	77
Figura 34	- Hierarquia <i>FamiliarStageAudience</i>	78
Figura 35	- Hierarquia <i>GenderAudience</i>	78
Figura 36	- Hierarquia <i>SocialGroupAudience</i>	78
Figura 37	- Hierarquia <i>SpecificGroupAudience</i>	79
Figura 38	- Visão completa de hierarquia <i>ContentDescription</i>	80
Figura 39	- Hierarquia <i>Credits</i>	81
Figura 40	- Hierarquia <i>MediaFormat</i>	82
Figura 41	- Hierarquia <i>Group</i>	83

Figura 42	- Hierarquia <i>Origination</i>	83
Figura 43	- Hierarquia <i>GeographicArea</i>	84
Figura 44	- Componente <i>Semantic Integration</i> na arquitetura do <i>middleware</i> Ginga	90
Figura 45	- Estrutura do Agente Provedor	92
Figura 46	- Arquitetura do componente Capturador de Dados	93
Figura 47	- Arquitetura do componente <i>OWL Parser</i>	94
Figura 48	- Documento de entrada XML descrevendo um conteúdo multimídia	95
Figura 49	- Arquivo XML traduzido para o formato de ontologia	96
Figura 50	- Diagrama de caso de uso da Gerência de Informação do Usuário (GIU)	99
Figura 51	- Diagrama de caso de uso da Unidade de Filtragem (UF)	102
Figura 52	- Diagrama de caso de uso da Unidade de Recomendação (UR)	104

Lista de Tabelas

Tabela 1	- Classes essenciais do RDF-Schema	44
Tabela 2	- Metadados MPEG-2 PSI/SI	58
Tabela 3	- Metadados TV-Anytime	58
Tabela 4	- Classificação dos dados operacionais	59
Tabela 5	- Modelo dados do usuário	60
Tabela 6	- Quadro comparativo de trabalhos relacionados	112

Sumário

Capítulo 1	Introdução	18
1.1	Motivação	18
1.2	Objetivos	21
1.3	Estrutura	21
Capítulo 2	Fundamentação teórica	23
2.1	Motivação	X
2.2	TV Digital Interativa	23
2.2.1	Componentes da TVDI	24
2.2.1.1	<i>Middleware</i> para TVDI	28
2.2.2	Metadados para TVDI	31
2.2.2.1	Tabelas de informação de serviços MPEG-2 PSI/SI	33
2.2.2.2	TV-Anytime	34
2.3	A visão da Web Semântica	36
2.3.1	Ontologias	39
2.3.2	Linguagens da Web Semântica	41
2.3.2.1	RDF	42
2.3.2.2	RDF- <i>Schema</i>	43
2.3.2.3	OWL	45
2.4	Conclusão	47
Capítulo 3	<i>Knowledge TV</i>	49
3.1	Contextualização	49
3.2	A camada semântica	50
3.3	Conclusão	54
Capítulo 4	<i>Uma Ontologia Núcleo para o Knowledge TV</i>	55
4.1	Visão geral	55
4.2	Dados operacionais	57
4.3	Modelagem semântica da Ontologia Núcleo	60
4.3.1	Classes	61
4.3.2	Hierarquias de classes	63
4.3.2.1	Descrição de conteúdo	65
4.3.2.2	Créditos	80

4.3.2.3	Formato de mídia	81
4.3.2.4	Grupo	82
4.3.2.5	Distribuição original	83
4.3.2.6	Área geográfica	84
4.3.3	Propriedades	84
4.3.4	Usando a ontologia <i>OWL Time</i>	87
4.4	Componente <i>Semantic Integration</i>	89
4.4.1	Agente Provedor	91
4.4.2	Agente Monitor	92
4.5	Capturador de dados	92
4.6	<i>OWL Parser</i>	94
4.7	Conclusão	96
Capítulo 5	Cenário de uso: recomendação de conteúdo em TVDI	97
5.1	Motivação	97
5.2	Recomendação de conteúdo	98
5.2.1	Gerência de informações do usuário	99
5.2.2	Unidade de filtragem de informações do conteúdo	101
5.2.3	Nível de personalização	103
5.2.4	Estratégia de recomendação	104
5.3	Conclusão	105
Capítulo 6	Trabalhos relacionados	106
6.1	Infraestrutura semântica para TVDI	106
6.1.1	Araújo e Ricarte (2010)	106
6.1.2	Aroyo <i>et. al.</i> (2007)	107
6.1.3	Butkus e Petersen (2007)	108
6.2	Cenário de recomendação de conteúdo	108
6.2.1	Blanco-Fernandez <i>et. al.</i> (2006)	109
6.2.2	Ávila (2010)	109
6.2.3	Lucas e Zorzo (2009)	110
6.2.4	Zhang <i>et. al</i> (2005)	111
6.3	Conclusão	112
Capítulo 7	Conclusões	113
7.1	Contribuições	113

7.2	Trabalhos futuros	114
7.3	Considerações finais	115
Capítulo 8	Referências bibliográficas	118
Apêndice A	Tabelas de dados operacionais	123
Apêndice B	API do componente de núcleo <i>Semantic Integration</i>	131

Capítulo

1

Introdução

“Se avexe não, toda caminhada começa no primeiro passo ”

Accioly Neto

1.1 Motivação

A televisão tem sido considerada o principal meio para transmitir informação e entretenimento à população durante as últimas décadas. Isto aconteceu devido a sua ampla área de cobertura e facilidade na aquisição de aparelhos de TV. Pesquisas recentes (IBGE; 2008) mostram que, particularmente no Brasil, o percentual de lares que possuem ao menos um aparelho de TV é superior a 95%. Este índice torna-se ainda mais significativo quando comparado ao de lares que possuem computadores com acesso à internet, que embora apresente uma expansão contínua, ainda continua com penetração inferior aos 35% dos lares no país. Diante de tais números, é possível observar que além de veículo difusor de informação e diversão, a TV também se apresenta como um meio de grande potencial a permitir inclusão digital e disseminação cultural junto à população de uma região ou país. Neste cenário, o advento da TV Digital Interativa (TVDI) no

Brasil (SBTVD; 2010) pode ser compreendido como um elemento catalisador no processo de massificação e inserção do ambiente digital junto à população.

Resultante da convergência de diversos domínios de conhecimento e tecnológicos, como informática, multimídia e telecomunicações, por exemplo, a TVDI tem permitido diversificar os serviços e conteúdos oferecidos ao usuário. A partir dessa integração, essa convergência entre os serviços televisivos e as novas tendências tecnológicas tem expandido o desenvolvimento de plataformas fornecedoras de serviços personalizados e buscas baseadas em objetos, entrega e recepção de informações, resultando em número cada vez maior de opções para escolha dos usuários (LUGMAYR, NIIRANEN, KALLI; 2004). Com o crescente número de canais e serviços interativos disponibilizados na TVDI, o público passa a ter acesso a uma gama cada vez maior de informações armazenadas local e remotamente. Além disso, observa-se também a mudança no comportamento do usuário em relação ao conteúdo consumido, deixando de ser passivo, que apenas absorve informação e passando a adotar uma postura ativa, que interage com a programação transmitida.

É interessante notar que este tipo de interatividade não muda nenhum conteúdo ou sua linearidade inerente, mas somente como os usuários controlam a visualização desse conteúdo. A tecnologia da TVDI pretende aumentar esse conceito de interatividade em pelo menos dois níveis. Primeiro, a interatividade com o conteúdo de programas de TV, onde os programas podem mudar baseado na entrada dos usuários. Formas avançadas desse conceito podem incluir novelas onde os usuários podem escolher ou influenciar detalhes ou mesmo o final do enredo. Segundo, interatividade com o conteúdo relacionado ao da TV, cujos exemplos incluem adquirir mais informação sobre o programa visualizado, previsões do tempo, esportes, filmes, notícias, etc. Esse novo aspecto permite o desenvolvimento de um novo conjunto de recursos e serviços para seus usuários. Tais recursos e serviços são geralmente aplicações interativas que são executadas de maneira síncrona ou assíncrona, junto com o conteúdo gerado pelos provedores (LINO *et. al;* 2011).

Essas mudanças tornaram-se ainda mais fortes a partir do momento em que os aparelhos de TV passaram a oferecer conexão com a Internet, permitindo ao usuário buscar o que lhe interessa ao invés de apenas escolher dentre o que é recebido em seu terminal. Isso faz com que o universo da grande rede também migre para a TV, levando consigo os problemas de interoperabilidade entre sistemas e informações, agravado pela carência nas harmonizações de padrões entre as duas

plataformas. Um dos objetivos originais da Web era dar suporte à troca de informações entre usuários, também dando suporte à participação de processos computacionais que pudessem auxiliar usuários humanos em vários aspectos durante essa comunicação. Entretanto, os processos computacionais atuais que são executados na Web apenas lidam com o transporte de informações, de tal modo que eles não têm acesso ao significado do conteúdo das páginas. A principal razão desse formato está na maneira que a informação é estruturada, que é mais apropriada para a manipulação de um usuário humano do que um processo computacional. Logo, hoje nós temos uma Web de documentos, ao invés de uma Web de informação, onde computadores podem prover apenas uma ajuda limitada durante o acesso e o processamento das informações. Nesse cenário, os usuários têm que executar várias funções, dentre elas a interpretação de informações (LINO *et. al.*; 2010).

Para tentar harmonizar a *web*, há o consócio W3C, que através de recomendações e especificações, define protocolos e padrões para homogeneizar o formato das informações transmitidas, tornando-as compartilhadas, de forma que elas se tornem operáveis em quaisquer pontos da internet. A partir desta convergência, este mesmo desafio passa a ser enfrentado na plataforma de TV, que passa a busca um meio de tornar o conteúdo transmitido capaz de ter suas informações compartilhadas e utilizadas para diferentes propósitos, através de um modelo único de representação.

Dessa forma, este trabalho apresenta a CoreKTV, uma infraestrutura baseada em conhecimento para TV Digital Interativa. Inserido no contexto do projeto *Knowledge TV: Uma Camada Semântica para prover serviços na plataforma da TV Digital Interativa*, os benefícios trazidos a partir da abordagem baseada em conhecimento abrangem harmonização com padrões internacionalmente estabelecidos pela indústria e academia, tornando possível a interoperabilidade entre sistemas e agentes computacionais, além do compartilhamento de informações e ampla possibilidade de automatização de operações por meio de raciocínio automático. De uma perspectiva prática, essa infraestrutura irá auxiliar a extração, classificação e organização da informação na plataforma da TVDI, estruturando seu conteúdo (vídeo, áudio e dados) de maneira que tal conteúdo possa ser apropriado tanto para usuários humanos como processos computacionais.

1.2 Objetivos

Este trabalho tem como principal objetivo a construção de uma infraestrutura para modelagem de conteúdo multimídia em TVDI, baseada em conhecimento alinhada aos conceitos e padrões da Web Semântica, de forma a possibilitar a interoperabilidade junto a outros sistemas integrados à plataforma da TV Digital Interativa, além de extensível para o desenvolvimento de infraestruturas de domínio específicas na plataforma do *Knowledge TV*.

Na construção deste objetivo geral, foram estabelecidos os seguintes objetivos específicos:

- Estudo do ambiente de TV Digital Interativa e seus componentes;
- Estudo do middleware brasileiro de TVDI, o Ginga;
- Estudo dos conceitos e das técnicas para modelagem e construção de bases de conhecimento;
- Levantamento dos metadados que descrevem o conteúdo transmitido como dados operacionais a partir dos padrões MPEG-2 PSI/SI e TV-Anytime;
- Definição de um modelo semântico para representação de informações de conteúdo multimídia para TV;
- Definição de uma forma de representação de conhecimento núcleo para programas de TVDI;
- Definição de um mecanismo de tradução de metadados de TVDI (MPEG-2 PSI/SI e TV-Anytime) para forma de representação em ontologias OWL;
- Definição de uma arquitetura de integração como *middleware* do SBTVD, o Ginga;
- Levantamento de cenário de uso para infraestrutura proposta.

1.3 Estrutura da dissertação

Esta dissertação está organizada em nove capítulos. Seguindo a esta introdução, no capítulo dois, é apresentada a fundamentação teórica acerca das principais áreas envolvidas no trabalho: TV Digital Interativa como um ambiente de convergência digital e Web Semântica.

No capítulo três é apresentado o projeto *Knowledge TV*, do qual este trabalho faz parte e contribui no seu desenvolvimento.

O capítulo quatro apresenta a abordagem e o desenvolvimento da ontologia núcleo para o *Knowledge TV*, a qual será responsável por modelar semanticamente as informações acerca da programação da TV.

No capítulo cinco é apresentado o cenário de uso proposto para validação e discussão da abordagem desenvolvida.

No capítulo seis é feito o levantamento do estado da arte em plataformas semânticas integradas ao ambiente de TVDI, com a apresentação ao final de um quadro-resumo que retrata as diferenças entre as abordagens encontradas e a desenvolvida neste trabalho.

O capítulo sete trata das conclusões da pesquisa do trabalho, além de discutir as dificuldades encontradas ao longo do desenvolvimento do projeto. No capítulo oito, encontram-se as referências bibliográficas utilizadas no trabalho.

Nos apêndices A e B são encontradas as tabelas de dados operacionais utilizadas neste trabalho e a API do componente *Semantic Integration*, respectivamente.

Capítulo

2

Fundamentação Teórica

“O propósito da computação é insight, não números”

Richard Hamming

Neste capítulo é apresentado o contexto teórico das principais áreas temáticas que envolvem este trabalho: (i) TV Digital Interativa, seus componentes, o Ginga, *middleware* do SBTVD, no qual o trabalho foi desenvolvido e utilização de metadados como fonte de dados acerca do conteúdo transmitido; (ii) a visão da Web Semântica, ontologias e linguagens da Web Semântica para construção de bases de conhecimento.

2.1 TV Digital Interativa

Os sistemas de TV Digital surgiram como possibilidade à substituição dos tradicionais sistemas analógicos a partir da década 1990. Este processo iniciou-se nos Estados Unidos, espalhando-se em seguida na Europa, no Japão e mais recentemente no Brasil e na China. Embora para cada uma dessas regiões tenham surgidos diferentes propostas, obedecendo a especificidades e objetivos regionais, a

maioria de sistemas de TV Digital, como o ATSC¹ americano, o DVB² europeu, o ISDB³ japonês, o SBTVD⁴ brasileiro e o DTMB chinês, mantém um núcleo de convergência, definido pela União Internacional de Telecomunicação – ITU, através de um modelo de referência para transmissão de sinais de TV Digital.

O termo TV Digital é aplicado aos sistemas que distribuem programação televisiva em formato digital para o usuário doméstico. Esses sistemas são capazes de distribuir um conteúdo de melhor qualidade audiovisual, eliminando problemas típicos em sistemas de TV analógica, como ruídos, distorções nas imagens e fantasmas (JONES *et. al.*; 2006). Algumas das melhorias possíveis com a TVDI são:

- **Imagens em** diferentes graus de resolução, que dependendo do dispositivo utilizado pelo usuário, podem ser de **alta resolução** (HDTV), **resolução padrão** (SDTV) ou de **baixa resolução** (LDTV).
- **Áudio de alta qualidade**, permitindo utilização de sistemas de cinco canais e tecnologias *surround*.
- **Multiprogramação**, com a possibilidade de enviar mais de um único serviço no mesmo canal de transmissão.
- **Portabilidade e Mobilidade**, que permite ao usuário receber o sinal da TV Digital em dispositivos portáteis como celulares, *palms*, além da recepção em ambientes em movimento.
- **Interatividade**, possibilitando ao telespectador uma nova experiência ao assistir TV, com execução de aplicações interativas, como enquetes, *quiz*, aplicações de *e-gov*, *e-bank*, etc.

2.1.1 Componentes da TV Digital Interativa

Do ponto de vista básico, um sistema de TV Digital Interativa possui ao menos três componentes, mostrados na Figura 1:

- Uma estação difusora, também chamada de *head-end*, responsável pela geração do sinal digital do conteúdo a ser transmitido;
- Um meio de difusão, no qual ocorre a transmissão do sinal entre a estação transmissora e os receptores e;

¹ <http://www.atsc.org>

² <http://www.dvb.org>

³ <http://www.dibeg.org>

⁴ <http://sbtvd.cpqd.com.br>

- Um receptor digital, capaz de receber, decodificar e exibir o sinal transmitido.

Além destas componentes, para que a interatividade seja plena é necessária a presença de um canal de interação, podendo ser unidirecional, chamado canal de retorno, responsável por transmitir dados do usuário para o provedor de conteúdo; ou bidirecional, através de conexão com a internet.

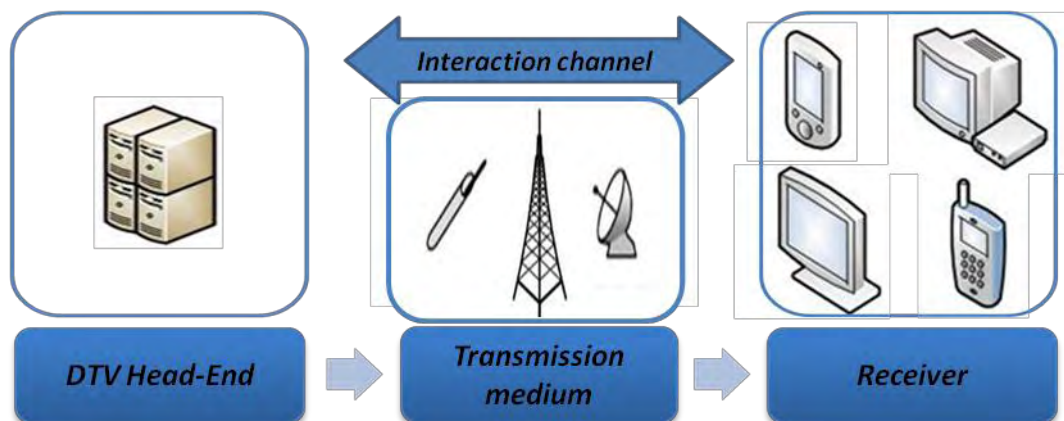


Figura 1 - Sistema de TV Digital Interativa

O processo de geração do sinal digital, de acordo com o modelo de convergência da ITU, adotado na maioria dos padrões de TV Digital, consiste das seguintes etapas, ilustradas na Figura 2:

- Codificação, onde é feita a digitalização e compressão do conteúdo (componentes de áudio e vídeo);
- Multiplexação, onde as diferentes componentes (áudio, vídeo, dados) são sincronizadas e combinadas para formar um único fluxo de dados;
- Modulação, onde são definidos os esquemas para transmissão do sinal específicos a cada um dos padrões de TVDI.

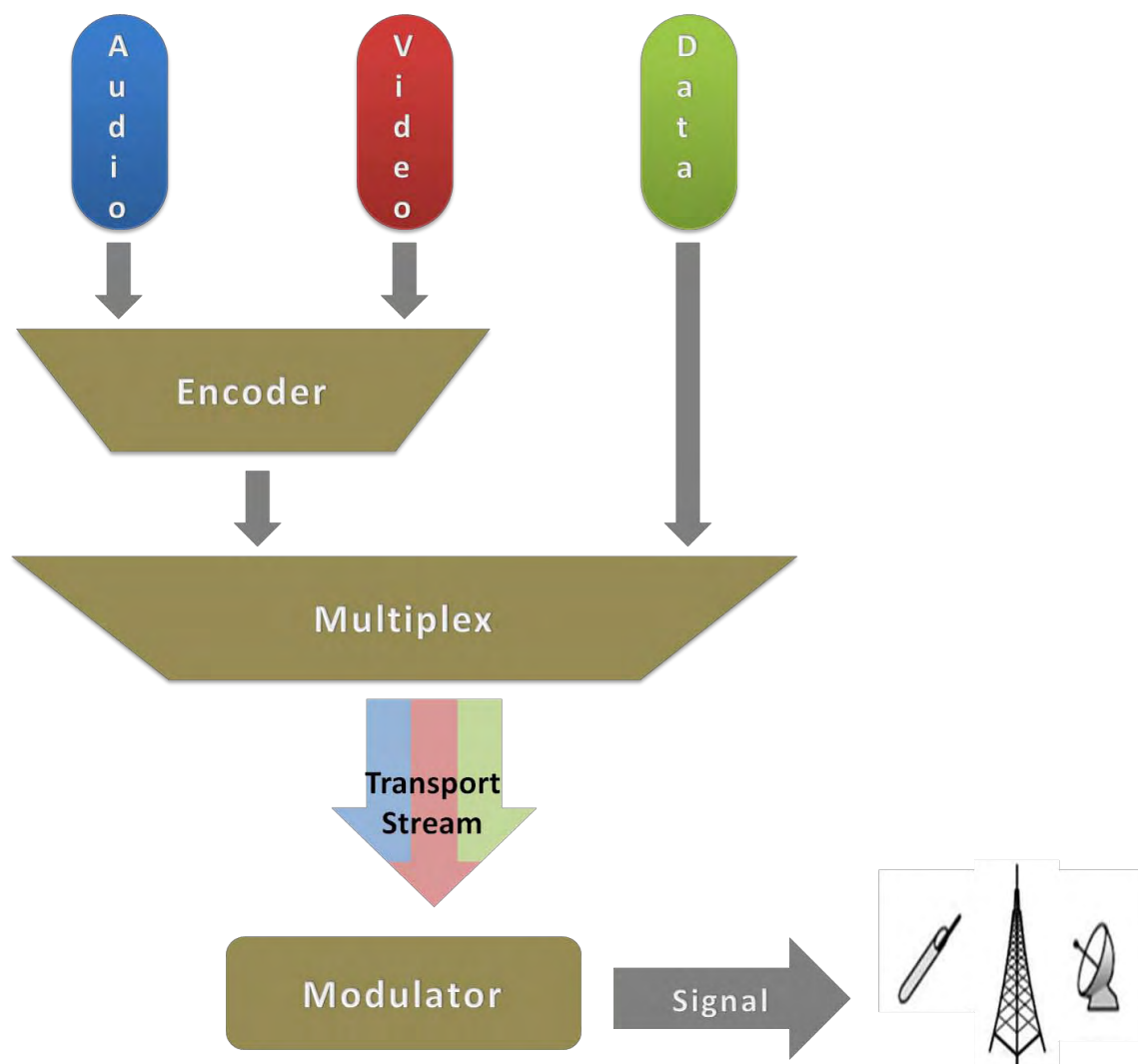


Figura 2 - Processo de geração do sinal da TVDI

O processo de **codificação** é iniciado após a captura das componentes de vídeo e áudio através de câmeras filmadoras digitais, ou através da digitalização do conteúdo, caso esteja em formato analógico. Em seguida são aplicadas técnicas e padrões de compressão da informação em formato digital, gerando fluxos elementares empacotados de áudio e vídeo, ou pacotes PES (*Packetized Elementary Stream*), sendo então enviados à etapa seguinte.

Durante o processo de **multiplexação** os pacotes de fluxos elementares de áudio e vídeo são sincronizados em uma seqüência única de pacotes. Nesta etapa são encapsuladas novas estruturas de dados contendo informações sobre programação transmitida ou contendo arquivos para a execução de aplicações interativas. Em TV Digital essa seqüência única recebe o nome de fluxo de transporte, ou MPEG-2 TS (*Transport Stream*), por ser definido no padrão MPEG-2.

O encapsulamento das informações sobre a programação é feito no Provedor de Informações de serviços, cujo funcionamento é definido pelo MPEG-2 *Systems* para a maioria dos sistemas de TVDI, além de um conjunto de tabelas adicionais e específicas a cada padrão. Mais detalhes são apresentados na subseção 2.2.2, que trata de metadados em TV Digital.

Para a difusão de aplicações interativas, o encapsulamento dos dados é feito através do módulo Gerador de Carrossel, que é baseado no protocolo DSM-CC (*Digital Storage Media Command and Control Protocol*) do padrão MPEG-2. A sua função é criar um sistema cíclico, contendo todos os arquivos da aplicação, resultando em um carrossel de dados. Dessa forma, nesse carrossel, um servidor de aplicações apresenta os dados de uma aplicação para um decodificador repetindo ciclicamente o seu conteúdo, de forma a garantir que independentemente do momento da sintonização do canal, o decodificador conseguirá receber todos os dados da aplicação para então executá-la.

A partir do processo de **modulação** do sinal, cada sistema adota estratégias específicas, que melhor refletem decisões técnicas e políticas regionais. Por fugir do escopo deste trabalho, tais conceitos e técnicas desta etapa não serão abordados. Para mais detalhes sobre o processo de modulação do sinal, consulte (FERNANDES; SOUZA FILHO; ELIAS; 2005)

A recepção do sinal da TV Digital Interativa é feita através de um receptor que pode ser embutido na própria TV ou através de um dispositivo a parte, chamado de *Set-Top-Box* (STB), cuja função é receber o sinal, decodificá-lo e exibí-lo. Por possuir tais funcionalidades, esse dispositivo assemelha-se a um computador com restrições e limitações para adaptar-se ao ambiente televisivo. Entretanto, sua arquitetura também é composta por unidades de processamento, memórias, barramentos e interfaces de entrada e saída. Além disso, para permitir a interação do telespectador através de canal de retorno, as interfaces devem ser compatíveis com o canal de interação utilizado. Dispositivos móveis, PDAs e celulares, também podem ser utilizados como receptores.

Um receptor de TVDI é construído em uma arquitetura baseada em camadas, como ilustra a Figura 3. A função de cada camada é oferecer serviços à camada superior, que por sua vez utiliza os serviços oferecidos pela camada inferior.



Figura 3 - Arquitetura de receptor de TVDI

Nessa arquitetura, o *hardware* tem a função de capturar o fluxo transmitido, enquanto o sistema operacional trata os mecanismos de transmissão e sincronização de mídias, a partir dos fluxos obtidos na camada inferior. A camada de *middleware* será abordada na subseção seguinte e, por fim, na camada de aplicações são executados aplicativos, que podem ou não requerer a interação do usuário.

2.1.1.1 Middleware para TVDI

Em TV Digital Interativa, *middleware* é uma camada de software intermediária presente no receptor. Sua função é abstrair as particularidades do sistema para aplicações e usuários, controlando as principais funcionalidades do receptor, inclusive suporte à execução aplicações interativo e ocultar toda a complexidade do hardware, software e interfaces de comunicação com sinal digital (LEITE et.al., 2005).

Dessa forma, tem-se o objetivo de garantir portabilidade entre as diferentes tecnologias da TVDI, como a fabricação de *set-top-box* (STB) em plataformas diversas e com arquiteturas de hardware diferentes, apresentando níveis de processamento, memória e comunicação variados. Nesse cenário, também é função do *middleware* reduzir essa heterogeneidade para os desenvolvedores de aplicações, fornecendo interfaces de programação padronizadas (APIs, do inglês *Application Programming Interfaces*), tornando as aplicações portáveis nos diversos tipos de STBs (SOUZA FILHO et. al.; 2007).

As principais propostas de *middleware*s são compostas basicamente por três ambientes, ou subsistemas: (i) Ambiente Imperativo, que provê o suporte a execução de aplicações escritas em linguagem procedurais (como Java, por exemplo); (ii) Ambiente Declarativo, responsável na apresentação de aplicações desenvolvidas em linguagens declarativas (como HTML e NCL, por exemplo) e, por fim, (iii) Núcleo Comum, que é o ambiente responsável por oferecer funcionalidades específicas de TV Digital, comum tanto aos ambientes imperativo e declarativo. Esses subsistemas formam uma arquitetura básica para *middleware* de TV Digital, apresentada na Figura 4, e são desenvolvidos seguindo as especificações ITU: J.200 (ITU, 2001), J.201 (ITU, 2004) e J.202 (ITU, 2003).

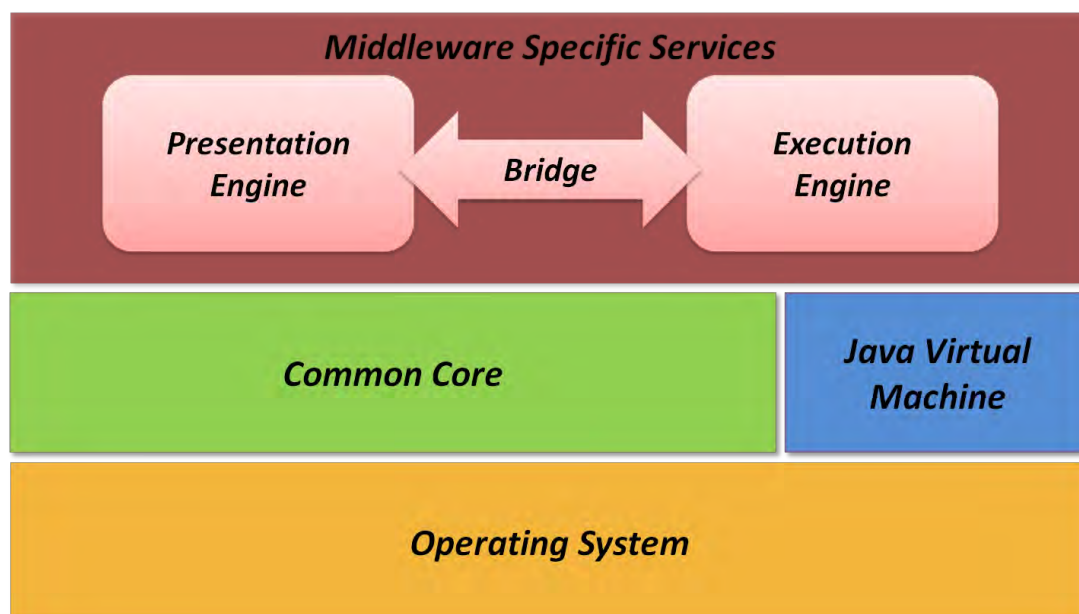


Figura 4 - Arquitetura de referência para *middleware* de TVDI

Os principais projetos de *middleware* utilizados atualmente são o MHP⁵, do padrão europeu DVB; os americanos DASE⁶, OCAP⁶ e ACAP⁶, utilizados no sistema ATSC, sendo o primeiro no sistema terrestre e os dois últimos no sistema de TV à cabo, e o japonês ARIB⁷, do sistema ISDB (MORRIS; SMITH-CHAIGNEAU; 2005) (FERNANDES; SOUZA FILHO; ELIAS; 2005). Recentemente o Brasil optou pela especificação de um *middleware* próprio, denominado Ginga, que tem sido adotado em diversos países da América Latina e será discutido na subseção seguinte.

⁵ <http://www.mhp.org/>

⁶ <http://www.cablelabs.com/opencable/>

⁷ <http://www.dibeg.org/>

2.1.1.1.1 Middleware Ginga

O Ginga é a especificação de *middleware* do SBTVD, resultado da junção do FlexTV (LEITE et. al; 2005) e MAESTRO (SOARES; 2006), *middlewares* desenvolvidos por consórcios liderados pela UFPB e PUC-RJ no projeto SBTVD, respectivamente.

O FlexTV, proposta de *middleware* procedural do projeto SBTVD, incluía um conjunto de APIs compatíveis com outros padrões além de funcionalidades inovadoras, como a possibilidade de comunicação com múltiplos dispositivos, permitindo que diferentes telespectadores pudessem interagir com uma mesma aplicação interativa a partir de dispositivos remotos. Já o MAESTRO foi a proposta de *middleware* declarativo do projeto SBTVD. Com foco em oferecer facilidade do sincronismo espaço-temporal entre objetos multimídia, utiliza a linguagem declarativa NCL (*Nested Context Language*) agregada as funcionalidades da linguagem de script Lua.

O Ginga integrou estas duas soluções, agora chamadas de Ginga-J (SOUZA FILHO; LEITE; BATISTA; 2007) e Ginga-NCL (SOARES; RODRIGUES; MORENO; 2007), tomando por base as recomendações internacionais da ITU. Desta forma, o *middleware* é subdividido em dois subsistemas principais interligados, ilustrados na Figura 5, também chamados de Máquina de Execução (Ginga-J) e Máquina de Apresentação (Ginga-NCL). A execução do conteúdo imperativo é possível através da Máquina Virtual Java (JVM, do inglês *Java Virtual Machine*). Dependendo dos requisitos da aplicação, um paradigma de programação pode ser mais adequado que o outro.

Outro aspecto importante reside no fato de os dois ambientes para execução de aplicações interativas não serem necessariamente independentes, uma vez que a **recomendação do ITU inclui uma “ponte”, que deve disponibilizar mecanismos para** intercomunicação entre os mesmos. Essa API de ponte permite que as aplicações imperativas utilizem serviços disponíveis nas aplicações declarativas, e vice-versa. Considerando que as aplicações sejam executadas um nível acima da camada dos ambientes de execução e apresentação, torna-se possível a execução de aplicações interativas híbridas, que agreguem as facilidades de apresentação e sincronização de elementos multimídias da linguagem NCL com o poder da linguagem imperativa Java.

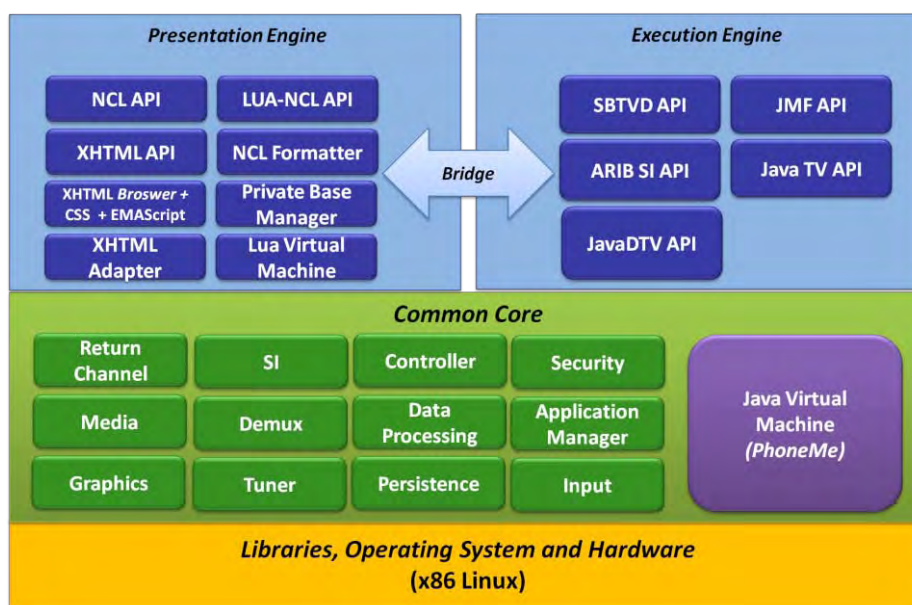


Figura 5 - Arquitetura conceitual do *middleware* Ginga

No Núcleo Comum do Ginga (*Ginga Common Core*) são definidas funcionalidades básicas em sistemas de TVDI, conforme mencionado acima, que dentre as suas principais funções, podemos citar a: exibição e controle de mídias, armazenamento, acesso a informações de serviço, sintonização de canais. Tais funcionalidades são especificadas na forma de componentes de software que se comunicam entre si para fornecer suporte aos ambientes de apresentação e execução. (KULESZA; et. al; 2010)

2.1.2 Metadados para TV Digital

Com o número crescente de informações disseminadas em todas as áreas de conhecimento e sem mecanismos para controle e limites de abrangência, torna-se necessário encontrar meios que possibilitem a criação, distribuição, busca, tratamento e consumo das informações contextualizadas às necessidades do usuário. Nesse contexto os metadados possuem papel importante, uma vez que podem ser entendidos como dados com informações adicionais que descrevem outros dados, sendo compreensíveis tanto por sistemas computacionais e seres humanos e, portanto, cabíveis às reais necessidades do usuário. De acordo com Smith e Schirling (2006), no domínio multimídia, a importância dos metadados torna-se evidente devido aos seguintes fatores:

- Necessidade de gerenciamento do conteúdo multimídia, que apresenta um ciclo de vida estendido, podendo cobrir processos e fluxos distintos, permitindo, através dos metadados a captura de dados que podem ser armazenados ou transmitidos junto com a programação.
- Necessidade de descrição e organização estrutural do conteúdo multimídia, incluindo o empacotamento de objetos multimídia, identificação de recursos e gerência de direitos autorais.
- Descrição do conteúdo, propriamente dito, uma vez que objetos multimídia podem não ser suficientemente autodescritivos para interpretação computacional, dificultando o entendimento semântico do conteúdo.
- Metadados multimídia também podem ser utilizados para regras de controle, como taxonomias, vocabulários e listas de termos, além de esquemas de classificação, permitindo interoperabilidade entre sistemas através de tipos e valores padronizados durante a descrição do conteúdo.

Segundo Alves et. al. (2006), existem dois modelos de metadados para conteúdo multimídia, em particular para TV Digital, os rígidos e flexíveis. A principal diferença entre eles está na impossibilidade de os metadados rígidos permitirem a extensão dos descritores de informação pré-definidos de uma forma que os sistemas que tratam os metadados já existentes consigam interpretar os novos descritores. Já para os flexíveis é possível a criação de novos descritores a partir dos já existentes e permitir que o sistema incorpore essas novas informações, ou seja, a extensão ocorre a medida que o conteúdo sofre modificações.

Para prover uma maneira de tratar objetos multimídia de modo que os usuários possam ter acesso otimizado ao conteúdo e demais serviços no contexto da TV Digital personalizada, é necessária uma infra-estrutura que colete, processe e entregue eficientemente tais informações que permitam melhor execução de tarefas para o usuário. Um exemplo dessas tarefas é a customização da TV ao seu perfil como telespectador. A chave para a construção de uma ferramenta de busca interativa e eficiente é a implantação de uma base de metadados que contenha atributos referentes aos conteúdos multimídia (LUGMAYR, NIIRANEN, KALLI; 2004).

Atualmente, os principais sistemas de TV Digital adotam metadados rígidos, através da Tabelas de Informações de Serviço (ABNT; 2007). No entanto, muitos serviços necessitam de informações mais detalhadas sobre o conteúdo, esbarrando nas limitações dessas tabelas. Dessa forma surgiu a necessidade

metadados flexíveis, que utilizassem estruturas passíveis de customização a serviços e cenários mais complexos (ALVES et. al; 2006). Entretanto, já existem pesquisas e recomendações na direção da utilização de um padrão flexível orientado a *broadcast* chamado de TV-Anytime (TVA; 2010). Tais propostas são apresentadas nas subseções a seguir.

2.1.2.1 Tabelas de Informações de Serviço MPEG-2 PSI/SI

As tabelas de informações de serviços são estruturas de dados utilizadas para transmissão de metadados que descrevem o conteúdo audiovisual. A estrutura dessas tabelas é definida no padrão MPEG-2 *Systems*, cuja finalidade é definir regras sintáticas e semânticas para a compressão, empacotamento e multiplexação de fluxos elementares, ou *elementary streams*, de áudio, vídeo e outros formatos de dados digitais em um fluxo de transporte, ou *transport streams* (TS), que representa um ou mais serviços de TVDI. Este padrão é atualmente adotado nos principais sistemas de TV Digital e seu funcionamento foi apresentado na subseção 2.2.1, no processo de codificação e multiplexação, durante a geração do sinal digital.

A descrição do conteúdo se dá através do mapeamento e da identificação dos fluxos elementares de mídia, via seções privadas das tabelas de informações específicas de programa, ou *Program Service Information table (PSI table)*. São quatro as tabelas PSI transmitidas em fluxos de transporte MPEG-2, sendo apenas as duas primeiras apresentadas a seguir obrigatórias:

- ***Program Association Table (PAT)***: contém a lista dos programas contidos em um TS. Os pacotes que contém a PAT possuem sempre o identificador de pacote (PID) igual a zero, a partir dos quais a tabela é reconstruída e podemos recuperar o identificador do fluxo de transporte (*transport stream id*), metadado único que identifica um TS na rede.
- ***Program Map Table (PMT)***: contém as informações referentes aos fluxos de mídia que fazem parte de um programa. Os identificadores de programas podem ser recuperados desta tabelas e utilizados como números do canal na rede.
- ***Conditional Access Table (CAT)***: possui as informações necessárias para acesso a conteúdos restritos de fluxos elementares.

- **Network Information Table (NIT):** possui o conteúdo privado e é utilizada para transporte de parâmetros físicos da rede.

Além dessas tabelas, cada sistema de TVDI especifica um conjunto particular de tabelas, chamado Tabelas SI, ou Tabelas de Informações de Serviços, que foram embora apresentem muitas similaridades, possuem diferenças que não permitem agrupá-las em um conjunto único padrão. Dessa forma, daremos destaque ao conjunto de tabelas especificadas no SBTVD, em particular para as tabelas relevantes à este trabalho, sendo todas listadas a seguir obrigatórias na transmissão do fluxo de transporte:

- **Network Information Table (NIT):** estende o conjunto de informações definidos na NIT do MPEG-2 *Systems*, e as suas características, assim como todo dado relevante sobre a sintonia dos serviços existentes.
- **Service Description Table (SDT):** contém informações sobre os serviços transmitidos em um fluxo de transporte.
- **Event Information Table (EIT):** apresenta informações cronológicas acerca dos eventos, ou programas, transmitidos em cada serviço.
- **Time Offset Table (TOT):** responsável pelo fornecimento da data, hora e fuso horário ao receptor.
- **Broadcaster Information Table (BIT):** responsável pelo fornecimento do nome do radiodifusor. Dentre as tabelas consideradas neste trabalho, esta trata-se da única cuja transmissão não é obrigatória.

2.1.2.2 TV-Anytime

O TV-Anytime (TVA) foi iniciado em 1999 propondo a criação de um padrão público e aberto para o desenvolvimento de sistemas integráveis e interoperáveis possibilitando aos desenvolvedores a manipulação do conteúdo digital aplicável nas fases de pós-produção, distribuição, consumo e interação do conteúdo de TV Digital interativa. Tal manipulação seria realizada através das operações de busca, seleção e aquisição do conteúdo, independentemente do meio onde ocorrerá a entrega.

Diferentemente das Tabelas de Informações de Serviços, o TV-Anytime é baseado em no formato XML (W3C; 2010) para representação de metadados e XML *Schema* para definição formal da estrutura e sintaxe de metadados. Além disso, o TVA também utiliza esquemas de descrição definidos na *Description Definition Language* (DDL) do padrão MPEG-7 (MARTINEZ; KOENEN; PEREIRA; 2002),

principalmente para descrição de mídias e preferências do usuário (ALVES et. al; 2006).

Do ponto de vista funcional, a TVA é um padrão de metadados orientado ao terminal de acesso do usuário. Dentre os objetivos propostos no padrão, segundo Evain e Murret-Labarthe (2003) e Lugmayr, Niiranen e Kalli (2004), destacam-se:

- Assegurar que os usuários tenham acesso a conteúdo personalizado, ou seja, de acordo com seus interesses específicos, a partir de uma grande variedade de provedores de conteúdos possíveis.
- Agregar valor ao conteúdo, permitindo ao usuário acessar e utilizar este conteúdo a qualquer momento e onde eles desejarem, sem regras de uso ou restrições de acesso.

Neste padrão, os metadados são classificados em quatro categorias: (i) descrição de conteúdo; (ii) descrição de instância; (iii) descrição de usuários, que representam os consumidores do conteúdo e; (iv) descrição de segmentações.

Na primeira categoria os metadados descrevem o programa de uma forma geral, apresentando informações básicas, descritivas (gênero, título, sinopse, etc.), audiovisuais (formatos de arquivo, atributos A/V, taxa de bits, etc.), informações sobre o programa (variações do programa), entre outros tipos de descritores definidos no padrão. Contém as seguintes tabelas a serem consideradas neste trabalho:

- ***Program Information Table:*** Contém descrições de itens do conteúdo, em geral, programas de TV. É composto de um ou mais elementos ***ProgramInformation***, um tipo de dados complexo definido pelo TVA.
- ***Group Information Table:*** Contém descrições de grupos de itens relacionados a um conteúdo. É composto de um ou mais elementos ***GroupInformation***, um tipo de dados complexo definido pelo TVA.
- ***Credits Information Table:*** Contém descrições de créditos relacionados a um conteúdo.
- ***Program Review Table:*** Contém descrição de avaliações relacionadas a um conteúdo.

Os metadados para descrição de instâncias têm como propósito a localização e seleção dos itens escolhidos, descrevendo o tipo de serviço oferecido. Informações de serviço (canal de TV) e a localização do programa (uma entrada do EPG) são

exemplos de descritores desta categoria. Contém as seguintes tabelas a serem consideradas neste trabalho:

- ***Service Information Table***: Contém informações descritivas acerca de serviços disponibilizados em um sistema. Tais informações são encapsuladas em elementos do tipo ***ServiceInformation***, definido pelo TVA.
- ***Program Location Table***: Contém registros descritivos relacionados a uma instância do conteúdo, representado genericamente pelo elemento do tipo ***Program LocationType***, definido pelo TVA e que pode ser especializado em diversos outros tipos específicos de instâncias de um conteúdo.

Os metadados para descrição de usuários determinam estruturas para definição da identificação do usuário, suas preferências e seu histórico, possibilitando o desenvolvimento de um grande número de aplicações possíveis em diversos cenários. Por fim, metadados de segmentação: descrevem associações entre metadados e fluxos de áudio e vídeo. Cada fluxo completo A/V é dividido em segmentos ou grupos de segmentos, que podem ser adicionados a metadados específicos.

É importante destacar que o TV-Anytime não define mecanismo e tecnologias aplicadas no transporte de dados, o que possibilita a sua incorporação em qualquer padrão de TV Digital. Atualmente, existe um esforço para que este padrão seja adotado como modelo de referência no padrão DVB (ALVES et. al; 2006).

2.2 A visão da Web Semântica

Durante as últimas décadas a ***World Wide Web***, também chamada de Internet revolucionou a forma de comunicação e intercâmbio de informações entre as pessoas e no mundo dos negócios. As fronteiras geográficas que antes impunham limitações e obstáculos deixaram de existir. Rapidamente, este novo meio passou a ser o principal meio de comunicação, pesquisa e ponto central de negociações.

Entretanto, apesar de todo esse crescimento, há sérios problemas associados ao seu uso, como os descritos por Antoniou e van Harmelen (2008): (i) muitas requisições para pouca precisão de resultados; (ii) resultados sensíveis ao vocabulário, ou seja, as informações encontradas estão associadas à sintática do vocabulário e não ao seu uso; (iii) resultados são páginas simples, isto é, a busca por

informações relevantes sobre vários documentos deve ser feita de forma manual, através de diversas consultas. A maioria desses problemas é decorrente das limitações da web atual, descrita por Berners-Lee, Hendler e Lassila (2001) como a ausência de expressão de significado, uma vez que a maioria do conteúdo atualmente *online* está projetado para leitura de humanos e não para manipulação computacional de acordo com seu significado.

Na visão da Web Semântica não se busca uma nova Web, mas uma extensão, que gradualmente passe a ser incorporada a Web existente, tornando os documentos mais acessíveis à interpretação e processamento automático e ao desenvolvimento de técnicas inteligentes para tirar o melhor proveito desta nova forma de representação. Com isso, documentos e sistemas são representados através de sua semântica, tornando-se ricos em informações complementares ao invés de informações ligadas sintaticamente.

A Web Semântica tem por objetivo o estabelecimento de padrões e tecnologias que permitam o processamento automático por máquinas que sejam capazes de compreender as informações expressas na Web, permitindo a descoberta de conhecimento, integração e reuso de dados, automatização de tarefas e a harmonização entre sistemas. Com o estabelecimento deste objetivo, tornam-se viáveis novas formas de: (i) busca de informação; (ii) atualização da informação; (iii) extração de informação; (iv) uso, reuso e compartilhamento da informação; (v) harmonização na forma de representação de conhecimento, permitindo o trabalho cooperativo entre computadores e pessoas; entre outros (ANTONIOU, VAN HARMELEN; 2008).

Em 2000, Berners Lee propôs uma arquitetura em camadas lógicas para a Web Semântica que ficou conhecida como “bolo de noiva”. Nesta arquitetura, ilustrada na Figura 6 são previstas as seguintes camadas: (i) *Unicode e URI*, onde são definidos conjuntos de caracteres apropriados e mecanismos para referência entre objetos, a partir do endereço; (ii) XML + NS + xmlSchema, onde estão contidos documentos autodescritivos, escritos na linguagem XML; (iii) RDF + rdfschema, que oferece primitivas de modelagem que permitem a criação e organização de objetos Web em hierarquias; (iv) Ontologia, onde são providas linguagens para construção de ontologias, como a OWL; (v) Lógica, usada para estender a linguagem de ontologia e para permitir a escrita de aplicações específicas baseadas em conhecimento; (vi) Prova, que envolve o processo de dedução como a representação e validação de provas em linguagem Web e; (vii) Confiança, que emergirá através do

uso de assinatura digital e outros conhecimentos, com base em recomendações de agentes de confiança ou agências de classificação e certificação.

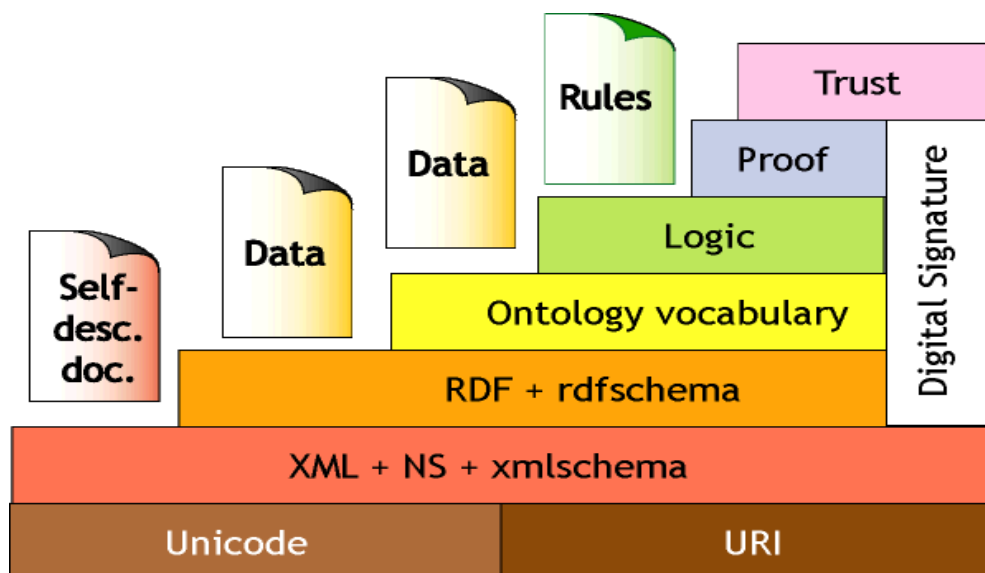


Figura 6 - Camadas lógicas da Web Semântica (ANTONIOU, VAN HARMELEN; 2008)

Há de se destacar que esta arquitetura não é proposta para modificar a atual estrutura da Web de forma imediata, mas com o objetivo de construí-la gradativamente sobre a que existe atualmente, trazendo novas contribuições aos cenários atuais.

Grande parte deste trabalho se concentra na camada de ontologia, na definição de um modelo de representação de conhecimento capaz de expressar com relevância informações sobre conteúdo multimídia transmitido em sistemas de TV Digital Interativa. Em virtude disso, as subseções nas seguintes aprofundam mais detalhadamente acerca do uso e construção de ontologias e a linguagem OWL. Maiores informações sobre as demais camadas dessa arquitetura lógica da Web, vista na Figura 6, podem ser encontradas em (BREITMAN; 2005) e (ANTONIOU, VAN HARMELEN; 2008).

2.2.1 Ontologias

O termo ontologia, do grego *ontos (ser) + logos (palavra)*, é original da filosofia e conceitualmente constitui a doutrina do ser, ou seja, é a parte da filosofia que estuda a natureza dos seres, enquanto seres. Segundo Breitman (2005), este vocábulo foi introduzido no estudo da Filosofia com o objetivo de distinguir o estudo do ser do estudo de diversos tipos de seres vivos existentes no mundo natural. Por sua vez, enquanto disciplina, o objetivo da Ontologia é estabelecer sistemas de categorização para organizar a realidade.

Na óptica na Web Semântica, uma das definições mais aceitas é a proposta por Gruber (1993), na qual ontologia é definida como uma especificação formal e explícita de uma conceitualização compartilhada de conceitos e relacionamentos que podem existir entre agentes ou comunidade de agentes. É possível interpretar essa definição da seguinte forma: **conceitualização** representa um modelo abstrato de algum fenômeno que identifica os conceitos relevantes para ele; **explícita**, significa que os elementos e suas restrições estão claramente definidos; **formal**, que a ontologia deve ser passível de processamento automático e; **compartilhada**, que reflita a noção de compartilhamento de conhecimento concensual, aceito por um grupo de pessoas.

Outras três definições bastante aceitas são propostas por Uschold e Jasper (1999) e Hendler (2001) que definem ontologia de uma forma mais abrangente sob o ponto de vista da Ciência da Computação. A terceira, mais sucinta e objetiva, é proposta pelo consócio W3C⁸.

Na primeira definição, uma ontologia pode assumir vários formatos, mas necessariamente deve incluir um vocabulário de termos e alguma especificação de seu significado. Deve abranger definições e uma definição de como os conceitos estão inter-relacionados, o que resulta na estruturação do domínio e nas restrições de possíveis interpretações de seus termos. (USCHOLD, JASPER; 1999). Tal definição está próxima ao que diz Hendler (2001) sobre o conceito utilizado nos principais círculos da Web Semântica, para o qual uma ontologia é um conjunto de termos, incluindo vocabulário, interconexões semânticas e algumas regras simples de inferência e lógica para algum domínio particular.

Já o W3C conceitualiza ontologia como a definição dos termos utilizados na descrição e representação de uma área de conhecimento. Além disso, ela deve

⁸ <http://www.w3.org/>

fornecer descrições para classes no domínio em que são aplicadas, relacionamentos entre essas classes e as propriedades ou atributos que as classes descritas possuem.

2.2.1.1 Classificação de Ontologias

Uma ontologia pode ser classificada de duas maneiras: (i) de acordo com a sua generalidade e; (ii) em relação ao tipo de informação que representam. Segundo Breitman (2005) e Nicola Guarino (1998), a classificação quanto à generalidade, ilustrada na Figura 7, pode ser feita da seguinte forma:

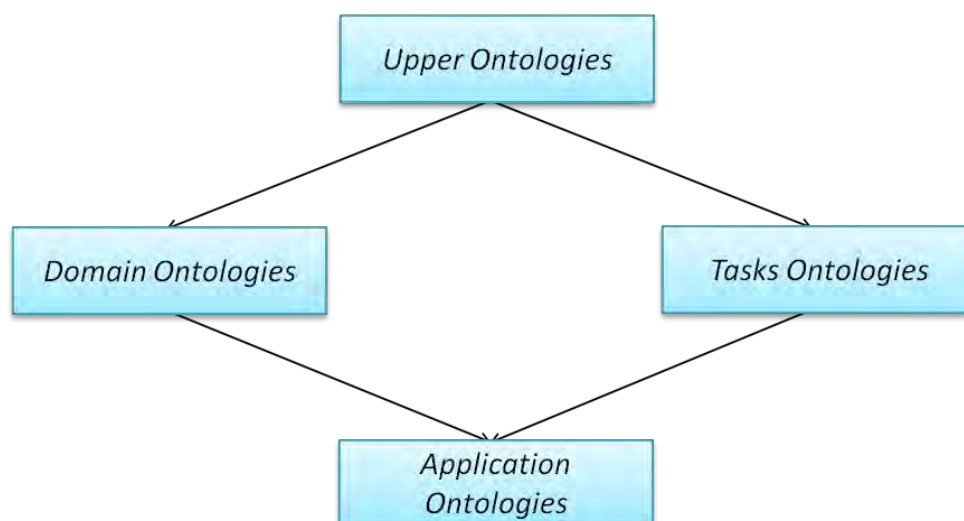


Figura 7 - Classificação de ontologias quanto à generalidade (BREITMAN; 2005)

- Ontologias de nível superior: descrevem conceitos muito genéricos, tais como espaço, tempo e eventos. Seriam, a princípio, independentes de domínio e poderiam ser reutilizados na confecção de novas ontologias.
- Ontologias de domínio: descrevem o vocabulário relativo a um domínio específico através da especialização de conceitos presentes na ontologia de alto nível.
- Ontologias de tarefas: descrevem o vocabulário relativo a uma tarefa genérica ou atividade através da especialização dos conceitos presentes na ontologia de alto nível.

- Ontologias de aplicação: são as ontologias mais específicas, que possuem conceitos relacionados a papéis desempenhados por entidades do domínio no desenvolvimento de alguma tarefa.

Já a classificação quanto ao tipo de informação representada, segundo Breitam (2005), Gómez-Perez, Fernández-Lopez e Corcho (2004), pode ser feita da seguinte forma:

- Ontologias para representação de conhecimento: capturam primitivas de representação de conhecimento e fornecem primitivas para modelagem de linguagens baseadas em frames, tais como classes, subclasses, atributos, valores e axiomas.
- Ontologias gerais e de uso comum: representam conhecimento de senso comum, podendo ser utilizado em vários domínios. Incluem um vocabulário relacionado a suas classes, eventos, espaço, causalidade, etc.
- Ontologias de topo ou de nível superior: descrevem conceitos muitos genéricos.
- Ontologias de domínio: representam conceitos que podem ser utilizados dentro de um domínio específico (médico, farmacêutico, jurídico, etc).
- Ontologias de tarefas: descrevem o vocabulário ligado a uma tarefa ou atividade específica.
- Ontologias de domínio-tarefa: são ontologias de tarefa que podem ser reutilizadas em um dado domínio, mas não em outros domínios.
- Ontologias de métodos: fornece definições para os conceitos e relacionamentos relevantes para um processo de modo a se atingir um objetivo.
- Ontologias de aplicação: são ontologias dependentes da aplicação e contém toda a informação necessária para a modelagem da aplicação em particular. São utilizadas para estender ontologias de domínio ou tarefa para uma dada aplicação.

2.2.2 Linguagens da Web Semântica

À medida que a Web Semântica foi crescendo e a necessidade de representação de informação de forma expressiva tornava-se essencial, passaram a surgir diversas linguagens com o objetivo de codificar de uma forma estruturada a

semântica dos documentos. Cada uma dessas linguagens possuía propósitos específicos e foram influenciadas fortemente pelos seus criadores, como também a época em que se propunham a ser utilizadas.

Tais linguagens podem ser trabalhadas em conjunto, de forma que uma complementa e fornece suporte a outra que apresenta maior poder de expressividade. Entretanto, elas apresentam diferenças a partir dos quais é possível estabelecer uma comparação entre elas, apresentando suas potencialidades e limitações. A seguir são apresentadas descrições básicas de algumas das principais linguagens da Web Semântica.

2.2.2.1 RDF

A linguagem *Resource Description Language* (RDF)⁹ foi idealizada para representar informações de recursos na Web, fornecendo interoperabilidade e semântica para metadados. A linguagem fornece um meio universal para expressar informações sobre recursos, além de permitir o intercâmbio entre aplicações sem a perda de significado. O modelo e a especificação do RDF foram propostos pelo consórcio W3C em 1999.

O RDF apresenta sintaxe baseada na linguagem XML e possui três princípios básicos: (i) recursos, **que são objetos, as “coisas” sobre o que se quer falar**, como autores, livros, lugares, pessoas, etc. Cada recurso possui um URI representando seu identificador universal; (ii) propriedades, que descrevem os relacionamentos entre recursos e também são identificadas por URIs e; (iii) frases, que é uma tripla objeto-atributo-valor, consistindo de um recurso, uma propriedade e um valor, respectivamente. Os valores, por sua vez, podem ser tanto recursos quanto literais (*string*, inteiro, decimal, etc.). Um exemplo de sentença RDF é apresentado na Figura 8, a seguir.

⁹ <http://www.w3.org/RDF/>

```
<rdf:RDF>
  <rdf:Description rdf:about: "CoreKTV">
    <prop: criador>
      Jônatas Araujo
    </prop: criador>
  </rdf:Description>
</rdf:RDF>
```

Figura 8 - Exemplo de sentença RDF

Nesta representação, o documento RDF utiliza-se de um elemento XML com a **tag** `rdf:RDF`. O conteúdo desse elemento é um conjunto de descrições que utilizam a etiqueta `rdf:Description`, onde cada descrição refere-se à um recurso, identificado das seguintes formas: (i) através do atributo do tipo `about`, que faz referência a um recurso existente; (ii) através do atributo do tipo `ID`, que cria um novo recurso; e (iii) sem nome, criando um atributo anônimo.

Uma sentença em RDF pode ser representada de duas formas: (i) através de uma fórmula lógica $P(x,y)$, que representa a propriedade, ou predicado binário P relacionando o objeto x ao objeto y ou; (ii) através da utilização de grafos, que fornece uma visualização gráfica, conforme ilustração da Figura 9. É composta de nós e arcos legendados. Para alguns domínios, como na Inteligência Artificial, essa representação é conhecida como uma rede semântica (BREITMAN; 2005) (ANTONIOU, VAN HARMELEN; 2008).

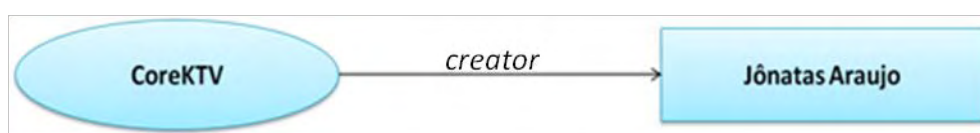


Figura 9 - Modelo de dados RDF

2.2.2.2 RDF-Schema

Sendo uma extensão do RDF, que apresenta limitado número de elementos predefinidos, o RDF-Schema é uma linguagem universal que permite ao usuário descrever recursos usando seu próprio vocabulário, ou seja, o desenvolvedor pode especificar suas classes e propriedades a partir do seu próprio vocabulário, particular ao seu domínio de aplicação. Sua especificação também foi disponibilizada pelo consórcio W3C, em 2004.

O RDF-Schema fornece um *framework* para descrição das propriedades e classes. É permitido aos recursos serem definidos como instâncias ou subclasses das classes presentes no RDF-Schema. Na Tabela 1 são apresentadas as classes essenciais e mais utilizadas do RDF-Schema para definição de classes e relacionamentos.

Tabela 1 - Classes essenciais do RDF-Schema (BREITAMN; 2005)

Classe	Descrição
rdfs:Resource	Classe de todos os recursos
rdfs:Class	Classe de todas as classes
rdfs:Literal	Classe de todos os literais, ou cadeias de caracteres
rdfs:Property	Classe de todas as propriedades
rdfs:Statement	Classe de todas as sentenças retificadas
rdfs:subClassof	Define um relacionamento de herança entre duas classes, podendo haver herança múltipla
rdfs:subPropertyof	Define um relacionamento de herança entre duas propriedades
rdfs:type	Relaciona um recurso a sua classe, sendo o recurso declarado como instância dessa classe.

Também pode ser definido um vocabulário para restrições, através das primitivas `rdfs:domain` e `rdfs:range`. A primeira especifica um domínio de uma propriedade, ou seja, a classe de recursos que pode aparecer como sujeito de uma sentença RDF. Já a primitiva `rdfs:range` especifica o alcance da propriedade, isto é, a classe de recursos que pode aparecer como objeto da sentença RDF. O uso destas primitivas é ilustrado no exemplo da Figura 10, onde a propriedade `hasGenre` define os recursos `MMContent` como domínio, através da primitiva `rdfs:domain` e `Genre` como alcance da propriedade, com uso da primitiva `rdfs:range`.

```
<rdf:Property rdf:about="hasGenre">
  <rdfs:domain rdf:resource="#MMContent"/>
  <rdfs:range rdf:resource="#Genre"/>
</rdf:Property>
```

Figura 10 - Exemplo de uso de primitivas RDF-Schema

2.2.2.3 OWL

A *Web Ontology Language* (OWL)^{10, 11} é uma linguagem de marcação, baseada em RDF/RDFS e XML, projetada para atender as necessidades das aplicações da Web Semântica e carência de expressividade semântica na conceitualização de um domínio. Ela foi lançada pelo consócio W3C como padrão para especificação de ontologias e como evolução de outra linguagem, a DAML+OIL (BREITMAN; 2005). Seu propósito é a representação de conceitos e seus relacionamentos em forma de ontologia e segundo Breitman (2005), sua aplicabilidade para Web Semântica pode ser resumida em:

- Construção de ontologias
 - Criação de uma ontologia
 - Explicitar conceitos fornecendo informações sobre os mesmos
 - Explicitar propriedades fornecendo informações sobre as mesmas
- Explicitar fatos sobre um domínio específico
 - Fornecimento de informações sobre indivíduos integrantes do domínio aplicado
- Racionalizar sobre ontologias e fatos
 - Determinar as conseqüências do que foi construído e explicitado.

A OWL oferece três diferentes sublinguagens, com cada uma fornecendo diferentes níveis de expressividade para uso por comunidades de utilizadores, sendo cada uma delas uma extensão da anterior:

- **OWL Lite**: utilizada na construção de uma hierarquia e restrições simples.
- **OWL DL**: usada quando se busca a máxima expressividade, considerando a computabilidade de sua representação. A sigla DL provém de **Description Logic**, ou lógica de descrição, o que permite a essa linguagem ser mapeada para outras linguagens de tipo lógico.

¹⁰ <http://www.w3.org/TR/2004/REC-owl-features-20040210/>

¹¹ <http://www.w3.org/TR/2009/REC-owl2-overview-20091027/>

- **OWL Full:** usada quando se pretende a máxima expressividade e liberdade sintática com uso de primitivas da linguagem OWL juntamente com as do RDF e RDF-Schema.

Dentre os elementos da linguagem, destacam-se os seguintes, considerados básicos:

- Namespaces: são declarações localizadas entre as **tags** do tipo `rdf:RDF`, permitindo aos identificadores presentes na ontologia a interpretação sem ambigüidade. Uma ontologia típica em OWL é iniciada por um conjunto de declarações de namespaces.
- Cabeçalhos: é uma coleção de sentenças sobre a ontologia agrupadas sob a etiqueta `owl:Ontology`, responsável por registrar comentários, controle de versão e pela inclusão de conceitos e propriedades de outras ontologias. Além disso, é neste elemento onde são colocados os metadados referentes ao documento.
- Classes: representam um conjunto ou coleção de indivíduos que compartilham características que os diferenciam dos demais. São utilizadas para descrever conceitos de um domínio através do elemento `owl:Class`. Um exemplo do uso deste elemento na definição de uma classe é apresentado na Figura 11, no qual é declarada a classe `MMContent`. Além disso, é utilizada o construtor `rdfs:subClassOf`, que define uma relação de hierarquia e é fundamental para construção de taxonomias. No exemplo da Figura 11, é estabelecida a relação de hierarquia onde a classe `MMContent` é subclasse de `Thing`, que em OWL é a classe raiz, de onde pertencem todas as classes de uma ontologia.

```
<owl:Class rdf:ID="MMContent">  
  <rdfs:subClassOf rdf:resource="Thing"/>  
</owl:Class>
```

Figura 11 - Declaração de classe OWL

- Indivíduos: são os membros de classes, ou seja, os objetos do mundo. Também chamados de instâncias, pertencem às classes e são relacionados a outros indivíduos e classes através de propriedades. Sua declaração é feita

usando primitivas RDF, ilustrada na Figura 12, ou de forma simplificada, como na apresenta a Figura 13.

```
<owl:Thing rdf:about="# JônatasAraujo">  
  <rdf:type rdf:resource="#Person"/>  
</owl:Thing>
```

Figura 12 - Declaração de indivíduo usando primitivas RDF

```
<Person rdf:ID="JônatasAraujo" />
```

Figura 13 - Declaração de instância em uma ontologia

- Propriedades de elementos: são utilizadas para descrever fatos em geral e podem se referir a todos os membros que pertencem a uma classe. Em OWL são relacionamentos binários. Podem ser de dois tipos: (i) propriedade de objetos, que estabelece o relacionamento entre duas classes e; (ii) propriedade de tipo de dados, que indicam um relacionamento entre instâncias de classes e literais expressos em RDF e *datatypes* do XML Schema (Breitman; 2005).
- Propriedades de restrições: são propriedades utilizadas para criar restrições, ou seja, definir limites para os indivíduos que pertencem a uma classe. Podem ser de três tipos: (i) restrições quantificadoras; (ii) restrições de cardinalidade e; (iii) restrições do tipo *hasValue* (“tem valor de”).

2.3 Conclusão

Neste capítulo foram apresentados conhecimentos teóricos levantados acerca das principais áreas de domínio sobre as quais este trabalho foi desenvolvido: (i) TV Digital Interativa, seus componentes, o Ginga, *middleware* do SBTVD, no qual o trabalho foi desenvolvido, além da utilização de metadados como fonte de dados acerca do conteúdo transmitido; (ii) a visão da Web Semântica, ontologias e linguagens da Web Semântica para construção de bases de conhecimento.

Quanto às linguagens da Web Semântica abordadas neste trabalho, elas podem ser comparadas quanto aos seus objetivos, seu grau de expressividade e sua

capacidade para descrição. Dessa forma, é possível sintetizá-las comparativamente da seguinte forma:

- Quanto aos objetivos: **(i)** o **RDF** foi concebido com o objetivo de descrever recursos, que representam objetos do mundo real e os relacionamentos entre eles; **(ii)** o **RDFS**, assim como o RDF, também foi concebido de descrever recursos, entretanto, vai além do seu antecessor, permitindo a construção de hierarquias entre os recursos e; **(iii)** a **OWL** foi construída com o intuito de ser usada na construção de ontologias, explicitando fatos de domínios e poder de racionalizar sobre ontologias e fatos. Dessa forma, é possível perceber que enquanto os objetivos das linguagens RDF e RDFS são mais restritivos a descrição, a linguagem OWL possui objetivos maiores, permitindo automatização de tarefas como a de raciocínio a partir do que já foi construído.
- Quanto à capacidade de descrição: **(i)** o **RDF** apresenta limitado poder de descrição, restrito a três elementos básicos, que podem ser construídos a partir de um vocabulário definido na linguagem; **(ii)** o **RDFS** também apresenta capacidade de descrição limitada, entretanto permite que sejam definidos vocabulários próprios do domínio da aplicação e de termos de restrição através de primitivas definidas na linguagem; **(iii)** a **OWL** supre carências encontradas nas suas anteriores, como a possibilidade de descrever negações, disjunções e conjunções, permitindo ao desenvolvedor melhor descrever recursos e tarefas do domínio da aplicação.
- Quanto ao grau de expressividade, devido as suas limitações para descrição, que restringem a capacidade de expressar os recursos, fatos e domínios, as linguagens RDF e RDFS apresentam menor grau de expressão, com a primeira sendo ainda mais limitada. Já a OWL apresenta diferentes graus de expressividade, de acordo com as suas sublinguagens, OWL Lite, OWL DL e OWL Full, que apresentam, respectivamente, graus de expressividade crescentes. Tais expressividades estão de acordo com as propostas sob as quais as linguagens foram construídas, entretanto, sempre com maiores recursos que suas antecessoras (RDF e RDFS), que podem ser utilizadas em conjunto, como construtores na descrição de recursos.

Capítulo

3

Knowledge TV

"Há conhecimento de dois tipos: sabemos sobre um assunto, ou sabemos onde podemos buscar informação sobre ele."

Samuel Johnson

3.1 Contextualização

A tecnologia da Web Semântica tem como objetivo prover um framework comum, que permita o compartilhamento e o reuso de dados nas aplicações, bem como o processamento automático de informações. Além disso, também é fornecida uma linguagem para a definição de ligações semânticas entre os dados processados e os objetos do mundo real. Durante as pesquisas realizadas nas áreas de Inteligência Artificial e TVDI, constatou-se a oportunidade de estender a visão da Web Semântica para o domínio da TV. Essa perspectiva idealizou-se como possível a partir do momento em que elementos presentes, anteriormente, apenas aos ambientes computacionais passaram a ser inseridos no ambiente de TV.

Dessa maneira, o *Knowledge TV*: Uma Camada Semântica para prover serviços na plataforma da TV Digital Interativa, projeto integrante do programa Centro de Pesquisa e Desenvolvimento em Tecnologias Digitais para a Informação e Comunicação (CTIC) do Ministério da Ciência e Tecnologia (MCT) e gerenciado pela

Rede Nacional de Ensino e Pesquisa (RNP), tem o objetivo de contribuir com o desenvolvimento de uma camada semântica para a plataforma de TVDI. Para isto, busca-se o compartilhamento das mesmas idéias e conceitos da Web Semântica, adaptando-se as suas restrições de domínio e apresentando novos caminhos na busca de um ambiente de convergência digital.

3.2 A camada semântica

Para o ambiente de TVDI, o *Knowledge TV* está inserido, conceitualmente, entre as camadas do *middleware*, ou *software* básico, e a camada de aplicações e serviços, conforme ilustrado na Figura 14.



Figura 14 - Visão do *Knowledge TV* em um ambiente de TVDI

Nesta visão simplificada da arquitetura de um ambiente de TVDI são apresentadas as seguintes camadas: **(i) Hardware**, ou Camada Física, na qual estão concentrados todos os aspectos e componentes físicos operantes em um ambiente de TVDI; **(ii) Middleware e (iii) Software Básico**, cuja responsabilidade é abstrair os detalhes e peculiaridades dos dispositivos físicos, permitindo a comunicação eficiente entre o *hardware* e as camadas superiores, tentando maximizar o desempenho do sistema; **(iv) Knowledge TV**, ou Camada Semântica, cuja finalidade é o fornecimento de modelagem semântica rica, além de bases de conhecimento contendo descrições de dados, recursos, serviços, aplicações

e relacionamentos entre eles. Tais descrições são construídas em uma linguagem formalmente padronizada, capaz de ser processada automaticamente por ambientes computacionais; **(v) Aplicações e Serviços** é a camada na qual são requisitados e utilizados os recursos das camadas anteriores, a partir de aplicações e serviços executados no terminal do usuário.

Concentrando-se na Camada Semântica, na qual este trabalho está inserido, conceitualmente, sua arquitetura é composta de cinco unidades básicas, ou componentes, apresentados na ilustração da Figura 15: **(i) Ambiente KDD para TVDI**, que comporta um ambiente de KDD (HAN; KAMBER; 2006), no qual estão inclusos um banco de dados tradicional, um modelo analítico de dados multidimensional em um *Data Warehouse* e um submódulo de tecnologias e ferramentas para análise de dados (*Data Mining* e *OLAP*); **(ii) Bases de Conhecimento**, referentes aos casos reais (instâncias) de dados armazenados, que estão modelados semanticamente, considerando a especificação presente na Subcamada de Modelagem Semântica. Neste componente, diferentes domínios de ontologias podem oferecer conceitos e taxonomias em níveis de detalhes diferenciados; **(iii) Subcamada de Modelagem Semântica**, cuja finalidade é a definição de métodos e linguagens formais com semântica bem definida, que permita a criação de instâncias e o seu processamento automático por agentes computacionais; **(iv) Raciocínio Automático**, responsável por prover todas as operações de raciocínio automático, que são executadas com base nos modelos de conhecimento fornecidos e nos mecanismos de raciocínio disponíveis na plataforma e; **(v) Serviços e Aplicações**, cujo objetivo é oferecer serviços e aplicações semânticas construídas com base na representação de conhecimento e raciocínio automáticos.

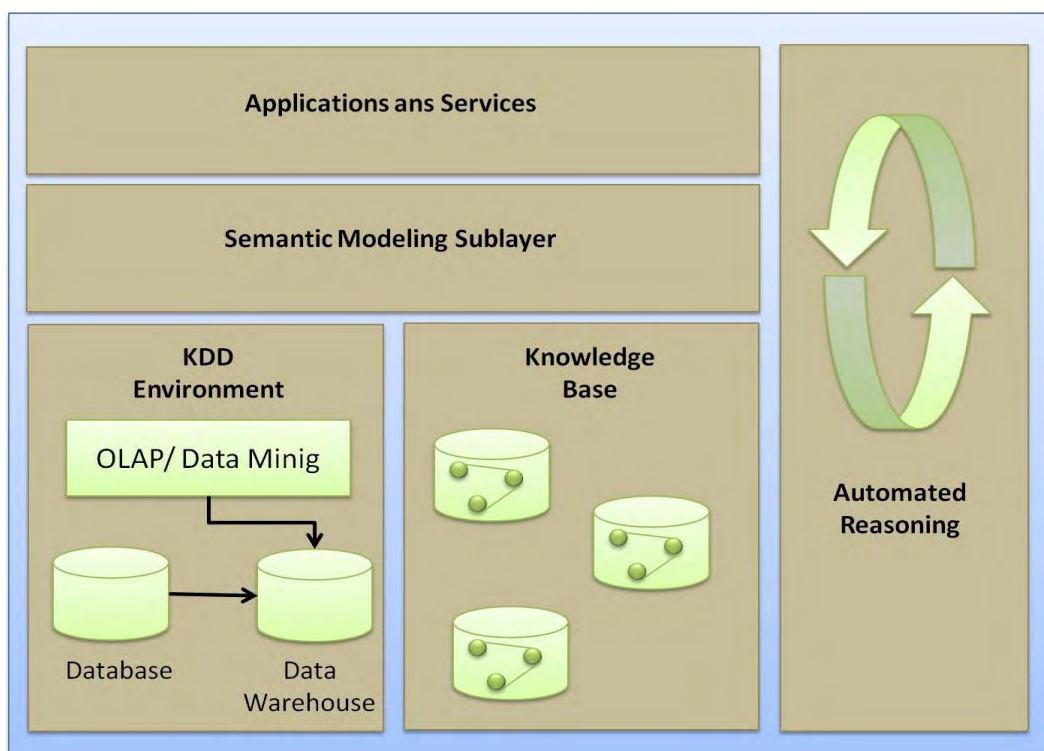


Figura 15 - Arquitetura conceitual do *Knowledge TV*

Sob a perspectiva de integração ao *middleware* e a execução das operações previstas em seus módulos, descritos acima, e que demandam razoável capacidade de desempenho durante processamento, o *Knowledge TV* foi projetado em uma arquitetura cliente-servidor, visualizada na Figura 16, na qual existe, ao menos, um módulo cliente, situado no *Set-Top-Box* do usuário e inserido dentro do núcleo comum do *middleware*, e outro módulo localizado no servidor, no qual é concentrada a maior parte do processamento de informações para dar suporte aos serviços oferecidos pela Camada Semântica.

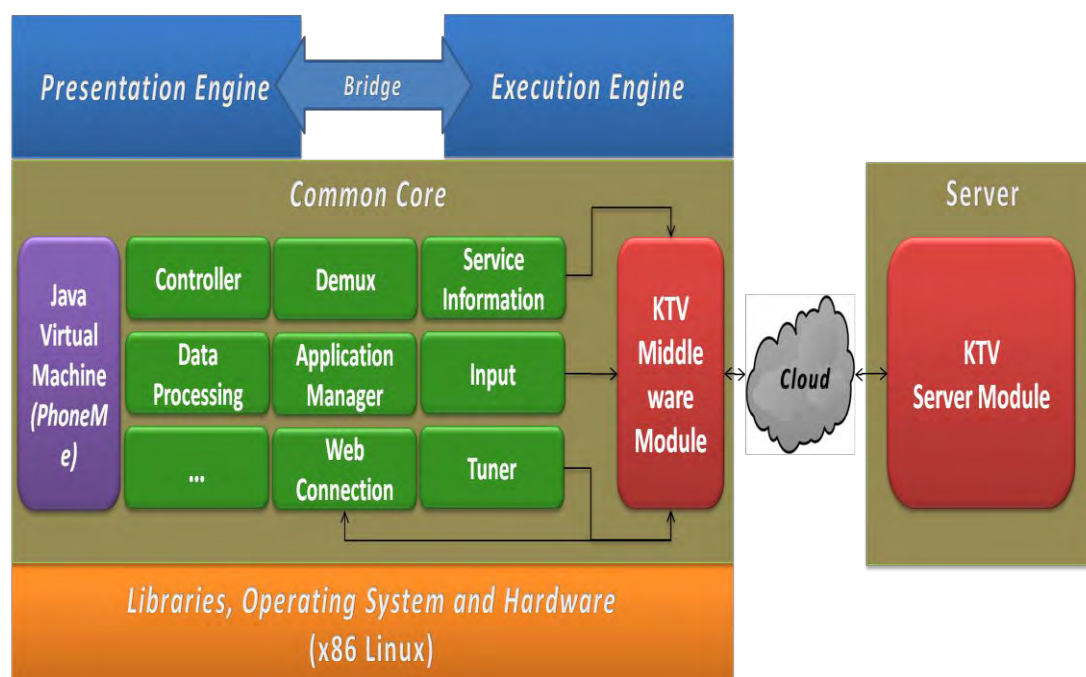


Figura 16 - Arquitetura *Knowledge TV* em um MW de TVDI

Com uma visão mais detalhada da arquitetura dos dois módulos do *Knowledge TV*, conforme ilustra a Figura 17, a seguir. As cinco unidades básicas, apresentadas na arquitetura conceitual são traduzidas em diversos componentes de processamento e armazenamento de informações, dentre os quais destacamos alguns, por estarem ligados ao escopo deste trabalho, ilustrados na cor laranja, na Figura 17. No módulo servidor, temos dois componentes: **(i) Monitor**, cuja tarefa é a captação do comportamento do usuário de TV, por meio das interações com o controle remoto e, **(ii) Provedor**, responsável pela obtenção dos metadados junto aos provedores de informação de serviço do *middleware*. No módulo servidor, destacamos os seguintes componentes: **(iii) DataCatcher**, ou Capturador de Dados, responsável por receber as informações obtidas no *middleware*, através dos agentes Monitor e Provedor, e promover uma verificação e limpeza antes de encaminhá-las ao **(ii) OWL Parser**, cuja finalidade é a reorganização dos dados sob um único formato, o OWL, possibilitando reutilização e agrupamento de conceitos na construção das ontologias de domínio e da **(iii) Ontologia de Núcleo**, cuja especificação formal é feita na **(iv) Subcamada de Modelagem Semântica**, e é responsável por manter uma base de conhecimento de conceitos e propriedades gerais acerca de conteúdo multimídia em TVDI e que possam ser

usados ou estendidos para qualquer domínio específico de um serviço ou aplicação desenvolvida ou utilizada a partir das especificações do *Knowledge TV*.

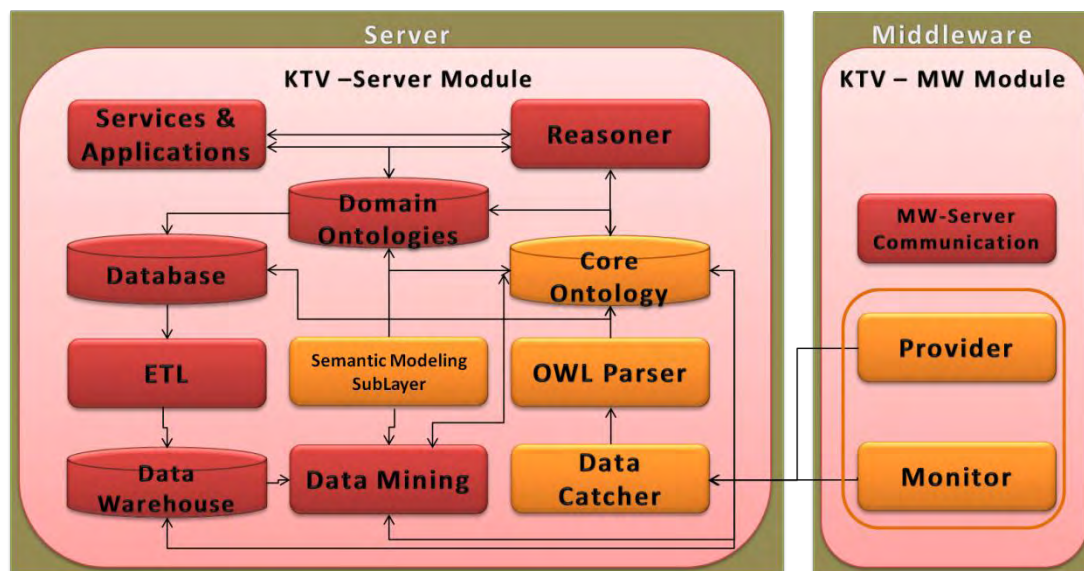


Figura 17 - Arquitetura dos módulos do *Knowledge TV*

3.3 Conclusão

Sob o ponto de vista da arquitetura conceitual (Figura 15), este trabalho contribui para a construção dos seguintes ambientes: **(i)** Subcamada de Modelagem Semântica e; **(ii)** Base de Conhecimento. Quando considerada as arquiteturas dos componentes e módulos da plataforma *Knowledge TV* (Figuras 16 e 17), as contribuições deste trabalho podem ser listadas nas seguintes atividades: **(i)** definição e desenvolvimento de um componente para captura dos dados operacionais, acerca da programação de TV, através de metadados transmitidos; **(ii)** integrar a plataforma semântica ao *middleware* Ginga ao demais componentes de núcleo comum; **(iii)** uma componente receptor de dados no módulo servidor do KTV; **(iv)** um mecanismo de tradução dos diferentes dados operacionais para um modelo único de representação, sob a perspectiva semântica; **(v)** especificação e desenvolvimento de uma base de conhecimento de núcleo, que reflita as informações e características básicas dos programas multimídia para TV, e que a partir da qual serão construídas bases de conhecimento de domínio específico.

Capítulo

4

Uma Ontologia Núcleo para o *Knowledge TV*

"O conhecimento une cada um consigo mesmo e todos com todos."

José Saramago

4.1 Visão Geral

Para fornecer suporte a diferentes serviços e aplicações baseados em conhecimento, o *Knowledge TV* prevê uma infraestrutura básica, um Ontologia Núcleo, a partir da qual deverão ser estendidos conceitos, propriedades e relacionamentos para o desenvolvimento de outras ontologias de domínio específico e de serviços e aplicações baseadas em semântica.

Sendo dividida em dois módulos, como foi visto na seção 3.2, a plataforma do *Knowledge TV* especifica a Ontologia Núcleo localizada no módulo servidor. Entretanto, o seu processo de modelagem e construção, com informações sobre o conteúdo é iniciado no *middleware*, conforme fluxo ilustrado na Figura 18.

Através dos agentes provedores e monitores do componente *Semantic Integration* é realizada a comunicação junto aos componentes do *middleware* responsáveis pelas informações de serviço e de sintonização do *middleware*, dos

quais são obtidos os metadados extraídos das Tabelas de Informação de Serviços dos padrões MPEG-2 PSI/SI e TV-Anytime, além de dados das ações do usuário relacionadas às mudanças de canal.

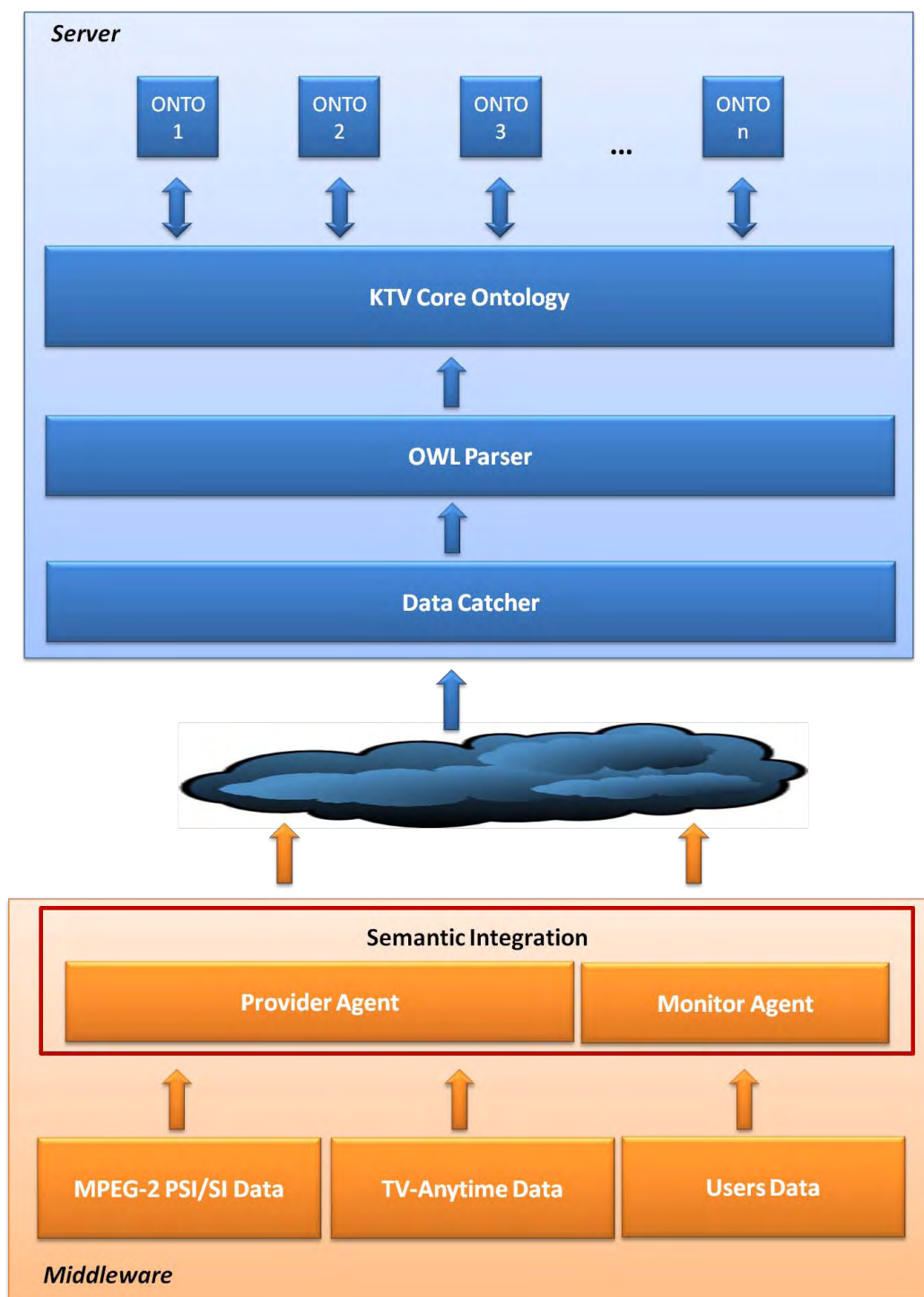


Figura 18 - Visão geral do processo de construção da Ontologia Núcleo do KTV

Os agentes possuem também a função de iniciar o processo de tradução de dados para um formato de representação único. Isso é feito com auxílio de esquemas de descrição XML, construídos de acordo com os dados presentes nas normas e especificações dos padrões de metadados considerados neste trabalho. Com isso, são geradas mensagens, contendo as informações sobre o conteúdo e usuário, que são enviadas ao servidor através dos recursos presentes na TV para comunicação com internet.

No módulo servidor, o componente Capturador de Dados, ou *DataCatcher*, funciona como a interface de entrada de dados, recebendo as mensagens e arquivos XML contendo informações descritivas acerca de conteúdo multimídia. Além de comunicável ao módulo *middleware*, este componente também interage com os serviços *web* integrados ao KTV para recepção de dados que poderão vir a alimentar a Ontologia Núcleo, ontologias de domínio ou os demais componentes da plataforma KTV. O tradutor OWL recebe os arquivos XML com a função de convertê-los e integrá-los ao formato de ontologia para em seguida ser adicionada a base de dados da CoreKTV, a qual dará suporte ao desenvolvimento de aplicações e serviços baseados em conhecimento, além de ser extensível à construção de ontologias de domínio específico.

4.2 Dados Operacionais

Os dados operacionais representam as informações que descrevem o conteúdo audiovisual transmitido, além das informações sobre o comportamento, as preferências e o histórico dos usuários da TV. Tais dados são obtidos de duas maneiras: **(i)** os referentes às descrições acerca do conteúdo audiovisual através dos metadados encapsulados nas tabelas dos padrões MPEG-2 PSI/SI e TV-Anytime, que são processados no componente “Provedor de Informações de Serviço” do *middleware* e; **(ii)** os referentes ao usuário são obtidos junto aos componente de “Sintonização”, a partir do qual é possível identificar as mudanças de canal e, dessa forma, registrar o tempo de audiência em determinado programa ou serviço.

A partir do levantamento feito acerca das tabelas de informação de serviço dos padrões MPEG-2 PSI/SI e do TV-Anytime, apresentados no capítulo 2, foram identificados grupos de metadados que possibilitaram mapear os conceitos e propriedades relevantes na construção da ontologia núcleo e representam informações importantes acerca dos programas de TV. As tabelas 2 e 3 apresentam

alguns dos metadados selecionados, considerados relevantes no contexto neste trabalho, onde podem ser localizados e o tipo de informação que representam. As tabelas completas do levantamento dos metadados podem ser encontradas no Apêndice A desta dissertação.

Tabela 2 - Metadados MPEG-2 PSI/SI

#	Metadado	Origem	Descrição
1	content_nibble_level_1	EIT	Informação de gênero
2	content_nibble_level_2	EIT	Informação de subgênero
4	content_rating	EIT/PMT	Classificação temática do conteúdo
5	age_rating	EIT/PMT	Classificação etária do conteúdo
6	event_name	EIT	Nome do conteúdo (programa)
7	short_description	EIT	Breve descrição do conteúdo
11	start_time	EIT	Horário de início do evento (programa)
12	Duration	EIT	Duração do evento (programa)
33	service_provider_name	SDT	Nome do fornecedor do serviço
34	service_name	SDT	Nome do serviço

Tabela 3 - Metadados TV-Anytime

#	Metadado	Origem	Descrição
61	Keyword	KeywordType	Define uma palavra-chave para o conteúdo
57	PersonName	TVAAGentType	Especifica o nome de uma pessoa
74	Character	CreditsItem	Especifica o nome do personagem representado pelo artista
80	Award	AwardsListItemType	Informação detalhada do prêmio
75	CategoryAward	AwardType	Especifica a categoria na qual o conteúdo foi premiado
79	YearAward	AwardsListItemType	Ano da premiação

114	MediaReview	MediaReviewType	Descreve uma crítica sobre o conteúdo
63	Genre	GenreType	Define um gênero para o conteúdo
64	GenreType	GenreType	Indica a ordem de importância do gênero que descreve o conteúdo (principal/secundário/outro)
86	ReleasedDate	ReleasedDateType	Data (dia, mês e ano) de lançamento do conteúdo

A partir deste levantamento destas tabelas, estes dados operacionais, indicados a partir da numeração na classificação (#), foram classificados inicialmente em quatro áreas básicas, apresentados na tabela 4, a seguir: **(i)** descrição de conteúdo; **(ii)** propriedade de mídia; **(iii)** unidade de tempo e; **(iv)** unidade geográfica.

Tabela 4 - Classificação dos dados operacionais

Descrição de Conteúdo	Propriedade de Mídia
1, 2, 4, 5, 6, 7, 8, 9, 14, 15, 23, 25, 26, 27, 28, 29, 31, 33, 34, 35, 38, 39, 40, 43, 44, 45, 49, 50, 51, 52, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 71, 74, 75, 76, 77, 78, 80, 81, 89, 90, 91, 92, 96, 111, 113, 114, 115, 118, 119, 120, 130, 131	12, 13, 16, 17, 18, 19, 20, 21, 22, 24, 27, 32, 46, 47, 48, 67, 68, 69, 70, 72, 73, 93, 94, 95, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 112, 121, 122, 126, 127, 128, 129
3, 36, 37, 41, 42, 83, 85, 88	10, 11, 30, 53, 54, 55, 56, 79, 82, 84, 86, 87, 116, 117, 123, 124, 125
Unidade Geográfica	Unidade de Tempo

Seguinte a esta etapa, os dados de cada área foram avaliados com objetivo de identificar a sua utilização como uma entidade, como uma propriedade de entidade ou como uma propriedade que relaciona entidades. O detalhamento do

processo de modelagem dos conceitos e propriedades da ontologia a partir destes dados é apresentado na subseção seguinte.

Para a identificação do perfil comportamental e histórico do usuário, alvo de consumo para o conteúdo da TV, foi definido um modelo de dados, apresentado na Tabela 5. Este modelo é capaz de registrar, com uma quantidade mínima de informações obtidas sem a interferência explícita, a forma com que o usuário consome o conteúdo transmitido. Dessa forma, torna-se possível a construção do perfil comportamental e histórico do usuário, sem que para isto seja necessária a sua intervenção direta para informar seus interesses ou preferências.

Tabela 5 - Modelo dados do usuário

Dado	Função
UserID	Identificador do usuário
UserAction	Identifica a ação realizada pelo usuário
TimeUserAction	Horário da ação realizada pelo usuário
Channel	Canal/Serviço consumido no momento
ProgramName	Nome do programa assistido
ProgramGenre	Gênero do programa assistido
StartTime	Horário em que o usuário começou a assistir ao programa
Duration	Período de tempo em que o usuário assistiu ao programa

4.3 Modelagem Semântica da Ontologia Núcleo

A modelagem semântica da Ontologia Núcleo constituiu-se de diferentes etapas, desde o levantamento e classificação dos dados operacionais, vistos na subseção 4.2, até a modelagem semântica do domínio e a tradução dos dados operacionais em entidades da ontologia. Ao final deste processo termos constituída a CoreKTV, ou seja, a base de conhecimento cujos conceitos, propriedades e instâncias

possam ser usados ou estendidos em qualquer domínio específico de um serviço ou aplicação desenvolvidos a partir das especificações da plataforma KTV. A partir da construção desta Ontologia Núcleo no domínio multimídia, mais especificamente na descrição de conteúdo para TVDI temos, dentre outras contribuições: **(i)** o compartilhamento da estrutura de informação, definida para descrição do conteúdo audiovisual, entre pessoas e agentes computacionais; **(ii)** a possibilidade de reuso e extensão do conhecimento deste domínio aplicável em outros domínio como recomendação de conteúdo, consultas semânticas, mineração de dados, alerta contra situações de emergência, etc; **(iii)** realizar inferência e suposições explícitas a partir do contexto em que as informações são utilizadas; **(iv)** separar o conhecimento do domínio do conhecimento operacional, ou seja, fazendo com que a informação tenha um valor semântico e deixe de representar apenas um dado, no formato sintático e; **(v)** analisar o conhecimento do domínio, permitindo identificar onde a informações podem apresentar similaridades e conceitos em comum, como por exemplo, identificar em grades de programação e programas de conteúdos semelhantes.

A partir da classificação dos dados operacionais realizada na subseção 4.2, foram identificados os seguintes conceitos e grupos de conceitos: (i) descrição de conteúdo; (ii) créditos; (iii) revisão ou crítica; (iv) alerta de emergência; (v) idioma ou linguagem falada; (vi) gênero; (vii) audiência pretendida; (viii) classificação; (ix) pessoa; (x) organização; (xi) prêmio; (xii) formato de mídia; (xiii) grupo; (xiv) meio original de distribuição; (xv) data; (xvi) horário; (xvii) continente; (xviii) país; (xix) conteúdo multimídia; (xx) usuário e; (xxi) ação do usuário. A construção de todos estes conceitos, sua hierarquia de classes e suas propriedades são explicadas detalhadamente nas próximas subseções.

4.3.1 Classes

O conceito básico a partir do qual passam a ser estabelecidos todos os outros é de Conteúdo Multimídia, ou **MMContent**, que representa uma entidade de conteúdo multimídia genérica. No contexto deste trabalho, esta classe representa o conteúdo transmitido digitalmente via radiodifusão para o terminal de TV do usuário, que pode vir a ser um serviço ou evento audiovisual, juntamente com suas informações e dados associados. Além desta, as principais classes e grupo de classes modeladas são:

- ***ContentDescription***, ou descrição de conteúdo: é a classe que representa um ponto chave agregador de diversas informações descritivas acerca do conteúdo transmitido.
- ***Credits***, ou créditos: representa os dados complementares, que embora não sejam descritivas em relação ao conteúdo, apresentam grande importância por conterem informações relativas às pessoas e organizações que contribuíram para criação do conteúdo.
- ***Review***, ou crítica: está relacionado à descrição de avaliações feitas sobre o conteúdo e que permitem ao usuário.
- ***EmergencyAlert***, ou alerta de emergência: representa uma sinalização transmitidas pelo provedor de conteúdo para informar a população de uma região a iminência de uma situação de emergência.
- ***Language***, ou idioma: descreve a língua falada ou idioma usado em dados textuais no conteúdo transmitido.
- ***Genre***, ou gênero: identifica o gênero do conteúdo transmitido. Uma característica dessa classe é a sua reutilização, uma vez que um único pode estar associado a mais de um gênero, ou possuir gênero e subgêneros.
- ***IntendedAudience***, ou grupo de audiência pretendida: representa o público alvo de um programa.
- ***Classification***, ou classificação: onde são apresentadas formas de classificar o conteúdo.
- ***Keyword***: representa uma palavra-chave representativa na descrição do conteúdo.
- ***Person***: modela a entidade pessoa, que pode ser especializada em diversos outros conceitos.
- ***Organization***: modela a entidade organização, que representa um empresa, companhia ou órgão.
- ***Award***, ou prêmio: representa uma premiação obtida ou promovida pelo conteúdo, organização ou pessoa.
- ***MediaFormat***: modela os formatos de mídia possíveis de transmissão.
- ***Group***: representa um conceito genérico identificada pelo agrupamento de entidades do mesmo tipo ou semelhantes.
- ***Origination***: representa o formato de distribuição original do conteúdo, ou seja, o meio para o qual o conteúdo foi produzido inicialmente.
- ***User***: representa a entidade usuário.

- **UserAction:** representa uma ação do usuário, que no contexto deste trabalho está limitada ao conjunto de ações que o usuário pode realizar através do controle remoto.
- **GeographicArea:** representa uma posição geográfica, como continente, país, região, estado ou cidade.
- **Time:** representa data, horário e intervalo de tempo.

4.3.2 Hierarquia de Classes

A partir da definição das classes, passou-se à etapa de agrupamento de conceitos em hierarquias, a partir das quais, ocorrem diversas especializações dos conceitos para buscar uma representação mais precisa, permitindo aperfeiçoar o uso em operações sobre as informações obtidas acerca do conteúdo. A Figura 19 apresenta uma visão completa da hierarquia de classes da Ontologia Núcleo, que será vista detalhadamente nas subseções seguintes e teve seu modelo construído usando a ferramenta *Protegé*.

A modelagem das hierarquias presentes na CoreKTV foi realizada a partir dos descritores e esquemas de descrição contidos nas normas ABNT NBR (ABNT; 2007) e da especificação do TV-Anytime (ETSI; 2009).

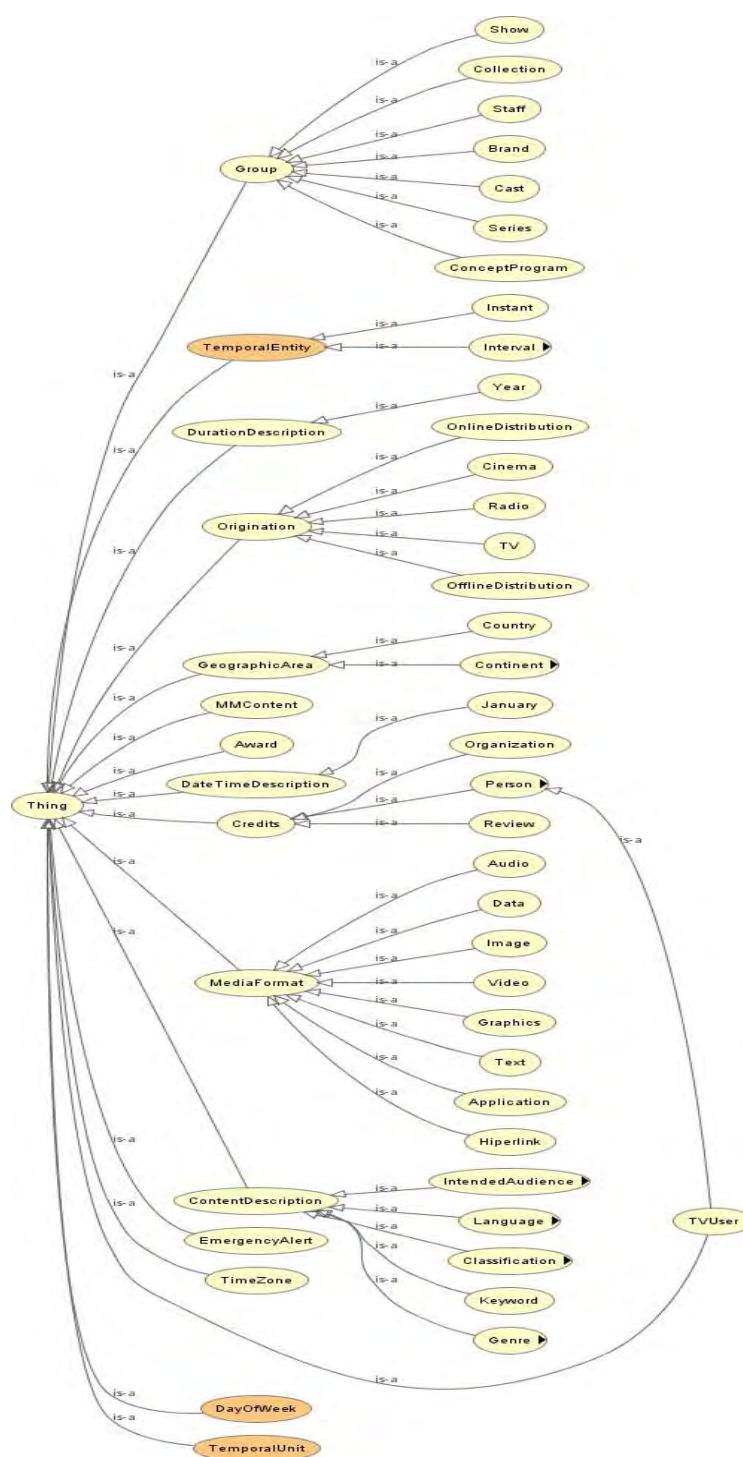


Figura 19 - Visão geral da Ontologia Núcleo do KTV (até o 3º nível)

4.3.2.1 Descrição de conteúdo

A construção da hierarquia 'Descrição de Conteúdo', a partir da classe *ContentDescription*, representa diferentes formas pelas quais um conteúdo multimídia pode ser descrito para o ambiente de TV. Para a definição desta hierarquia foi utilizado o processo de desenvolvimento proposto por Noy e McGuinness (2001), no qual são combinadas duas abordagens: (i) uma *top-down* e (ii) outra *bottom-up*.

Na primeira, foi realizada a definição dos conceitos mais genéricos ao domínio da TVDI em relação à descrição de seu conteúdo, a partir das principais classes previamente definidas e com a modelagem de outras. Dessa forma, foram identificadas como integrantes desta hierarquia as classes *Genre*, *Language*, *Classification*, *IntendedAudience* e *Keyword*, que foram definidas na etapa anterior de definição de conceitos e ilustradas na Figura 20.

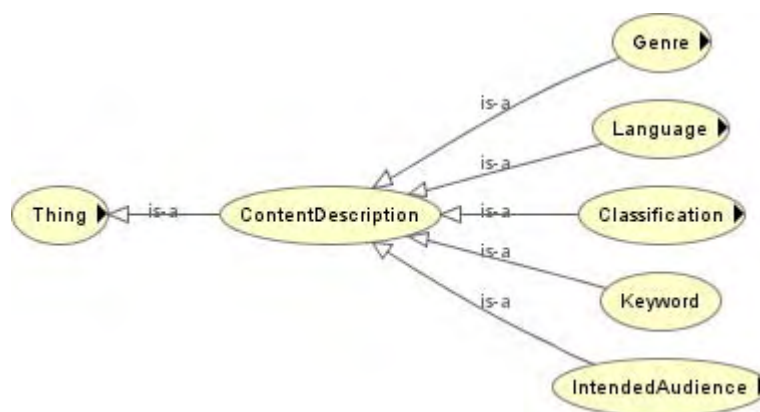


Figura 20 - Hierarquia de classes para *ContentDescription*

A abordagem *bottom-up* foi utilizada na definição das classes mais específicas, que fazem parte das hierarquias internas de *ContentDescription*:

- a) Hierarquia *Genre*, que especializa o gênero do conteúdo multimídia em diversas categorias e subcategorias, conforme ilustra a Figura 21.

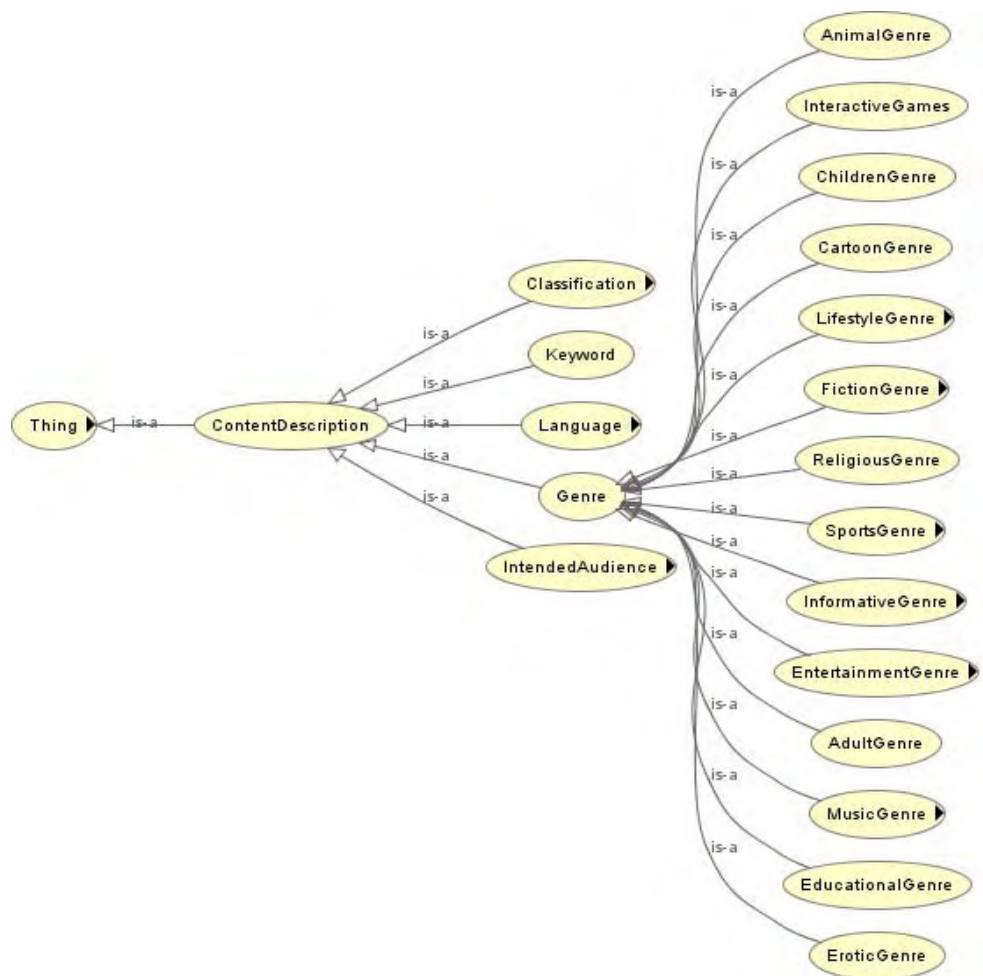


Figura 21 - Hierarquia *Genre*

Compõem esta hierarquia as seguintes classes:

- ***AdultGenre***: representa gênero de conteúdo adulto;
- ***AnimalGenre***: representa gênero de conteúdo relacionado à animais;
- ***CartoonGenre***: representa gênero de desenho animado;
- ***ChildrenGenre***: representa gênero de conteúdo infantil;
- ***EducationalGenre***: representa gênero de conteúdo relacionado a educação, cursos, treinamentos e similares de caráter formativo;
- ***EroticGenre***: representa gênero de conteúdo erótico;
- ***FictionGenre***: representa gênero de conteúdo fictício, de caráter não informativo;
- ***InformativeGenre***: representa gênero de conteúdo informativo;
- ***InteractiveGames***: representa gênero de conteúdo relacionado a jogos interativos;

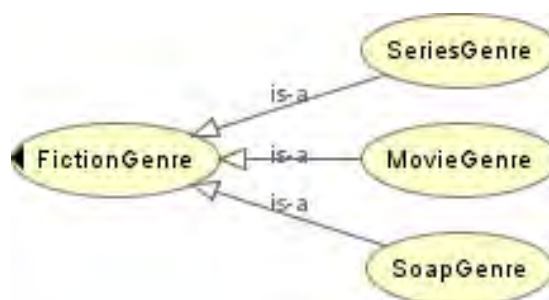
- ***LifestyleGenre***: representa gênero de conteúdo relacionado a estilos e áreas de interesses variados;
- ***MusicGenre***: representa gênero de conteúdo relacionado a estilos musicais;
- ***ReligiousGenre***: representa gênero de conteúdo de cunho religioso, ou gospel;
- ***SportsGenre***: representa gênero de conteúdo relacionado a esportes.

Há de se destacar as classes ***EntertainmentGenre***, ***FictionGenre***, ***InformativeGenre***, ***LifestyleGenre***, ***MusicGenre*** e ***SportsGenre***, que representam hierarquias internas à classe ***Genre***, sendo mais detalhadas a seguir.

A classe ***EntertainmentGenre*** apresenta a estrutura hierárquica ilustrada na Figura 22, na qual são modelados os seguintes conceitos: (i) ***RealityGenre***, no qual são representados conteúdos do gênero que retratam a realidade, como os ***reality shows***, por exemplo; (ii) ***ChatGenre***, que representa o gênero de programa onde ocorre conversas ou troca de mensagens entre os participantes e/ou usuário de TV; (iii) ***LotteryGenre***, representando o gênero de conteúdo de jogos de loteria; (iv) ***TalentGenre***, que representa gênero de conteúdo de programas de descoberta de talentos, calouros e similares; (v) ***DatingGenre***, no qual são representados conteúdos de gênero relacionado a encontros e compromissos entre pessoas; (vi) ***ContestGenre***, que representa o gênero de programas onde os participantes competem diretamente entre si; (vii) ***HumorGenre***, no qual são representados os programas do gênero humorístico; (viii) ***QuizGenre***, que representa o gênero de conteúdo relacionado a perguntas e respostas; (ix) ***CircusGenre***, que representa gênero de conteúdo relacionado à atrações circenses; (x) ***VarietyGenre***, no qual estão representados os conteúdos relacionados a programas de variedades; (xi) ***SurpriseGenre***, que representa gênero de conteúdo relacionado a programas onde são realizados sonhos de telespectadores; (xii) ***MagicGenre***, representando o gênero de conteúdo relacionado a mágicas e magia; (xiii) ***AwardsGenre***, no qual são representados os conteúdos relacionados a definição e entrega de prêmios e; (xiv) ***ComedyGenre***, que representa o gênero de conteúdos ligados a comédia.

Figura 22 - Hierarquia *EntertainmentGenre*

A hierarquia *FictionGenre*, ilustrada na Figura 23, é especializada da seguinte forma: (i) *SeriesGenre*, no qual é representado o gênero de conteúdo produzido em formato de seriados; (ii) *MovieGenre*, que representa o gênero de conteúdo produzido em formato de filme e; (iii) *SoapGenre*, relacionado ao gênero de conteúdo produzido em formato de novela.

Figura 23 - Hierarquia *FictionGenre*

A hierarquia *InformativeGenre*, ilustrada na Figura 24, é especializada nas seguintes classes: (i) *BiographyGenre*, que representa o gênero de conteúdo

relacionado à biografias pessoais; (ii) *InterviewGenre*, representando o gênero de conteúdo associado a entrevistas; (iii) *DebateGenre*, no qual é representado o gênero de conteúdo que contém debate ou discussão entre participantes; (iv) *DocumentaryGenre*, associado ao gênero de conteúdo documentário; (v) *TVNewsGenre*, que representa o conteúdo de jornais e telejornais para TV; (vi) *ReportGenre*, representando o conteúdo associado ao gênero de reportagens especiais e; (vii) *PoliticsGenre*, no qual é representado o gênero de conteúdo relacionado a política e políticos.

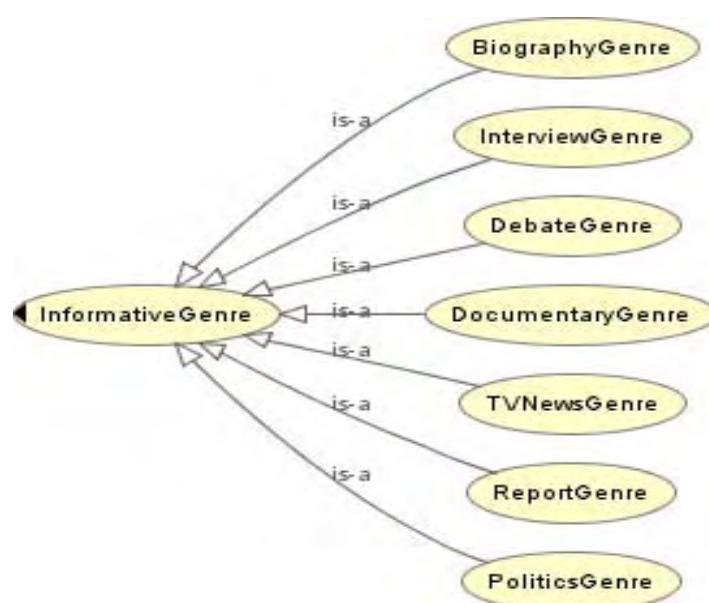


Figura 24 - Hierarquia *InformativeGenre*

A hierarquia *LifestyleGenre*, apresentada na Figura 25 apresenta as seguintes subclasses: (i) *HealthGenre*, na qual é representado o gênero de conteúdo associado à saúde e bem-estar; (ii) *FashionGenre*, representando o gênero de conteúdo relacionado à moda, vestuário e informações similares; (iii) *TVShoppingGenre*, que representa o gênero de conteúdo no qual são realizados anúncios e vendas de produtos; (iv) *HomeGenre*, modela o gênero de conteúdo relacionado à informações de atividades do lar; (v) *DrinksGenre*, na qual é representado o gênero de conteúdo informativo sobre bebidas; (vi) *TourismGenre*, onde é representado o gênero de conteúdo acerca de informações turísticas; (vii) *RuralGenre*, representando gênero de conteúdo associado às informações do ambiente rural; (viii) *CookingGenre*, que representa o gênero de conteúdo relacionado à informações culinárias e; (ix) *GardeningGenre*, onde é representado o

gênero de conteúdo que contém informações sobre jardinagem, paisagismo e similares.

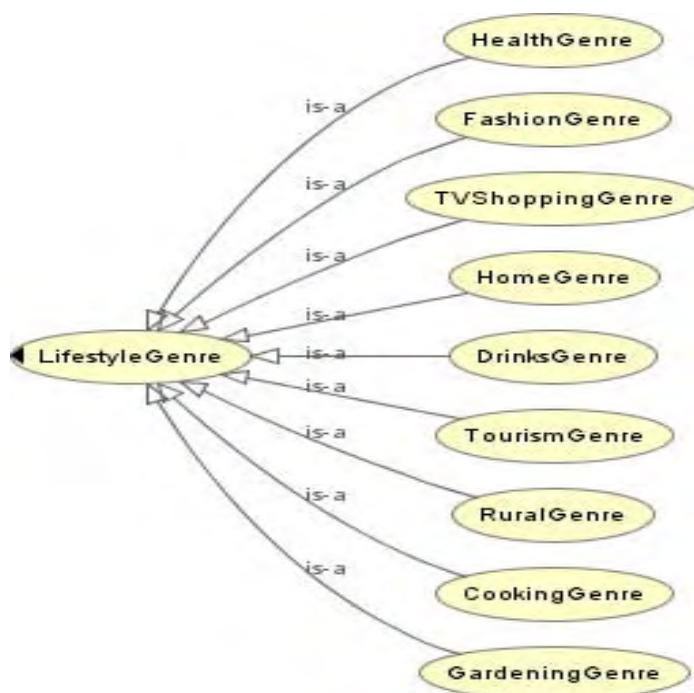


Figura 25 - Hierarquia *LifestyleGenre*

A hierarquia *MusicGenre*, ilustrada na Figura 26, é especializada nas seguintes classes, representam gêneros musicais específicos: *BackgroundMusicGenre*, *BluesMusicGenre*, *ChildrenSongMusicGenre*, *ClassicalMusicGenre*, *ClubMusicGenre*, *CountryMusicGenre*, *DanceMusicGenre*, *ElectronicMusicGenre*, *EraMusicGenre*, *EthnicMusicGenre*, *EventMusicGenre*, *FolkMusicGenre*, *GospelMusicGenre*, *HipHopMusicGenre*, *HitChartMusicGenre*, *JazzMusicGenre*, *PopRockMusicGenre*, *RapMusicGenre*, *ReggaeMusicGenre*, *ReligiousMusicGenre*, *RequestedSongsMusicGenre*, *SoulMusicGenre*, *SpokenMusicGenre*, *TraditionalMusicGenre*, *UrbanMusicGenre*, *WesternMusicGenre*, *WorldMusicGenre*.

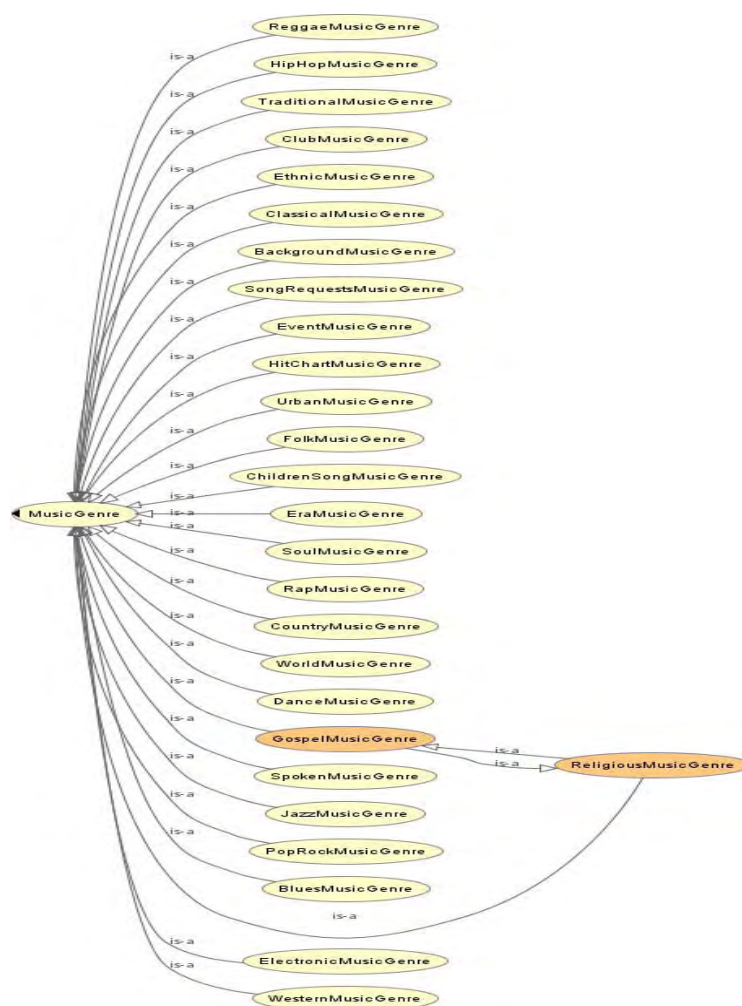


Figura 26 - Hierarquia *MusicGenre*

A hierarquia *SportsGenre*, ilustrada na Figura 27 é especializada nos seguintes conceitos, que representam informações sobre o esporte específico: *AdventurySportsGenre*, *AirSportsGenre*, *AthleticsSportsGenre*, *BicycleSportsGenre*, *CastingSportsGenre*, *CyclingSportsGenre*, *DisableSportsGenre*, *DogRacingSportsGenre*, *EquestrianSportsGenre*, *FencingSportsGenre*, *GymnasticsSportsGenre*, *GolfSportsGenre*, *MaccabiSportsGenre*, *MindSportsGenre*, *ModernPentathlonSportsGenre*, *MotorSportsGenre*, *SocialSportsGenre*, *SomboSportsGenre*, *StrenghtBasedSportsGenre*, *TraditionalGamesSportsGenre*, *WinterSportsGenre*, *MartialSportsGenre*, *RacketSportsGenre*, *TeamSportsGenre*, *WaterSportsGenre*.

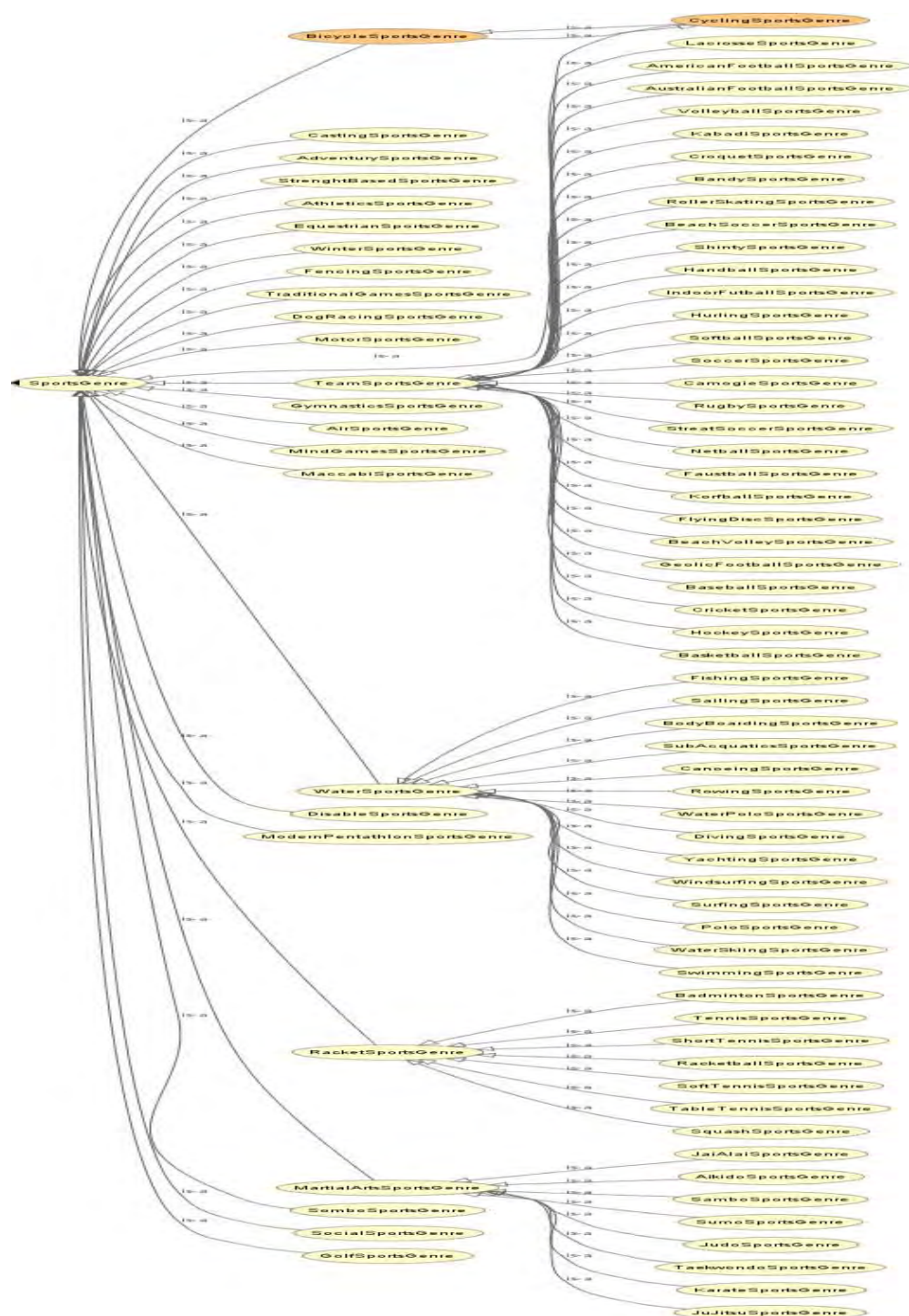


Figura 27 - Hierarquia *SportsGenre*

Nesta hierarquia, destacam-se as seguintes classes, por apresentarem hierarquias internas:

• *MartialSportsGenre*: *AikidoSportsGenre*, *JaiAlaiSportsGenre*, *JuJitsuSportsGenre*, *JudoSportsGenre*, *KarateSportsGenre*, *SamboSportsGenre*, *SumoSportsGenre*, *TaekwondoSportsGenre*.

• *RacketSportsGenre*: *BadmintonSportsGenre*, *RacketballSportsGenre*, *ShortTennisSportsGenre*, *SoftTennisSportsGenre*, *SquashSportsGenre*, *TableTennisSportsGenre*, *TennisSportsGenre*.

• *TeamSportsGenre*: *AmericanFootballSportsGenre*, *AustralianFootballSportsGenre*, *BandySportsGenre*, *BaseballSportsGenre*, *BasketballSportsGenre*, *BeachSoccerSportsGenre*, *BeachVolleySportsGenre*, *CamogiesSportsGenre*, *CricketSportsGenre*, *CroquetSportsGenre*, *FaustballSportsGenre*, *FlyingDiscSportsGenre*, *GeolicFootballSportsGenre*, *HandballSportsGenre*, *HockeySportsGenre*, *HurlingSportsGenre*, *IndoorFootballSportsGenre*, *KabadiSportsGenre*, *KorfballSportsGenre*, *LacrosseSportsGenre*, *NetballSportsGenre*, *RollerSkatingSportsGenre*, *RugbySportsGenre*, *ShintySportsGenre*, *SoccerSportsGenre*, *SoftballSportsGenre*, *StreatSoccerSportsGenre*, *VolleyballSportsGenre*.

• *WaterSportsGenre*: *BodyBoardingSportsGenre*, *CanoeingSportsGenre*, *DivingSportsGenre*, *FishingSportsGenre*, *PoloSportsGenre*, *RowingSportsGenre*, *SailingSportsGenre*, *SubAcquaticsSportsGenre*, *SurfingSportsGenre*, *SwimmingSportsGenre*, *WaterPoloSportsGenre*, *WaterSkiingSportsGenre*, *WindsurfingSportsGenre*, *YachtingSportsGenre*.

b) Hierarquia *Language*, que especializa a descrição do idioma utilizado em um conteúdo multimídia em subcategorias, conforme ilustração da Figura 28. Compõem esta hierarquia as seguintes classes:

- *CaptionLanguage*: representa a descrição do idioma usado no sistema de *closed caption*.
- *SubtitleLanguage*: representa a descrição do idioma utilizado em legendas, nos conteúdos em que elas são disponibilizadas.
- *SpokenLanguage*: representa a descrição do idioma falado no conteúdo.

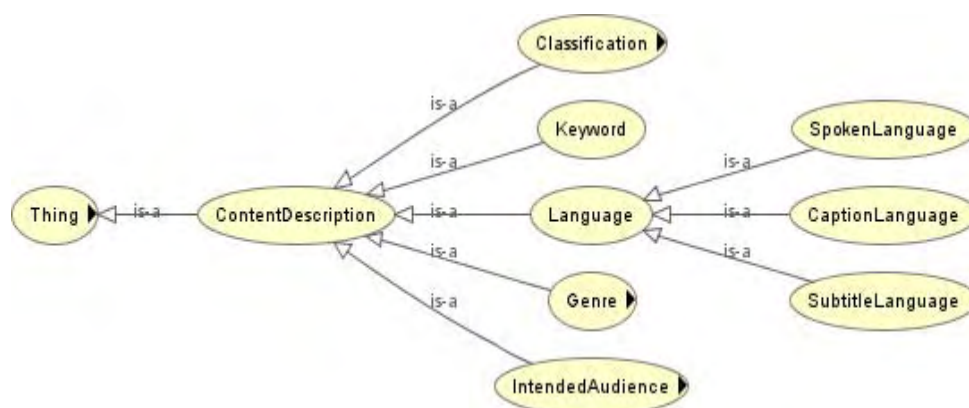


Figura 28 - Hierarquia *Language*

c) A Hierarquia *Classification*, ilustrada na Figura 29, descreve a forma como um conteúdo multimídia pode ser classificado, é especializada nos seguintes conceitos:

- *ClassificationByTheme*: representa a classificação do conteúdo acerca da temática abordada.
- *ClassificationByAge*: representa a classificação do conteúdo de acordo com a faixa etária indicativa recomendada.
- *ClassificationByUsedLanguage*: representa a classificação do conteúdo relacionada a linguagem falada, escrita ou visual (cenas) usada no conteúdo.
- *NoClassification*: quando não há descrição disponível.

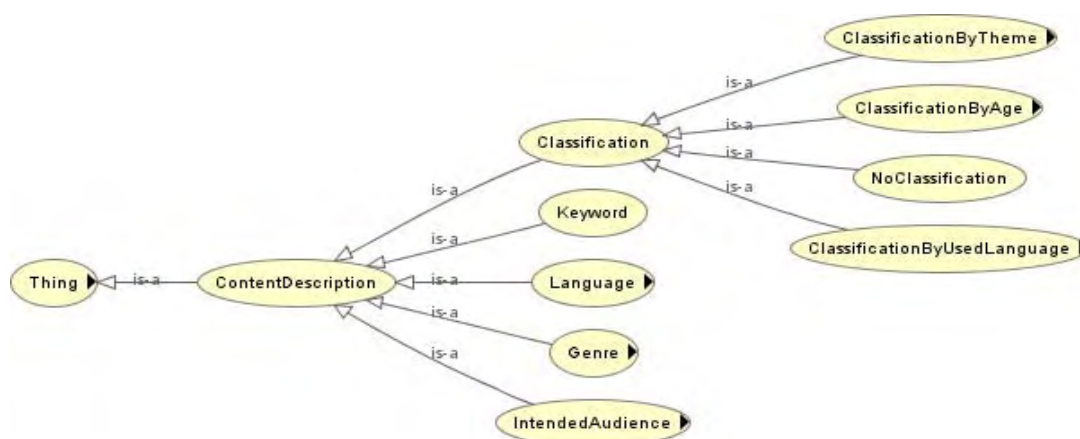


Figura 29 - Hierarquia *Classification*

As classes *ClassificationByTheme*, *ClassificationByAge* e *ClassificationByUsedLanguage*, por sua vez, apresentam hierarquias internas que especializam os conceitos adotados, permitindo a melhor descrição da classificação do conteúdo

multimídia. A Figura 4.30 ilustra a estrutura hierárquica partindo da classe *Classification* até as classes mais específicas desta hierarquia, que são:



Figura 30 - Visão completa da hierarquia *Classification*

- *ClassificationByTheme*: *BloodTheme*, *SexTheme*, *DrugsTheme*, *DistressTheme*, *ViolenceTheme*, *NudityTheme*;
 - *ClassificationByAge*: é uma estrutura hierárquica na qual a classe mais especializada, *AllAges*, representa o conteúdo disponível para todas as idade e a classe mais genérica, *18Years*, representa o conteúdo restrito as pessoas adultas ou maiores de 18 anos. A partir dessa classe seguem-se as *16Years*, *14Years*, *12Years*, *10Years*, até chegar ao ponto final da hierarquia.
 - *ClassificationByUsedLanguage*: *MildSwearWordsLanguage*, *StrongLanguage*, que é especializada em *VeryStrongLanguage* e, *OffensiveLanguage*, que é especializada em *RacistLanguage*, *SexistLanguage*, *HomophobicLanguage*.
- d) A Hierarquia *IntendedAudience*, ilustrada na Figura 31, descreve a forma como um conteúdo multimídia pode ser descrito quanto ao objetivo de Audiência, ou seu público-alvo. Está hierarquia é especializada nos seguintes conceitos:

- **AgeGroupAudience:** representa o público-alvo de acordo com a idade dos usuários da TV.
- **FamiliarStageAudience:** representa o público-alvo de acordo com o formato familiar ao qual os usuários de TV pertencem.
- **SocialGroupAudience:** representa o público-alvo de acordo com a classe social da qual os usuários de TV fazem parte.
- **GeographicAudience:** representa o público-alvo associado à uma área geográfica.
- **GenderAudience:** representa o público-alvo de acordo com o sexo.
- **EducationStandard:** representa o público-alvo associado ao nível educacional que os usuário de TV possuem.
- **SpecificGroup:** representa grupos específicos aos quais o conteúdo multimídia pode ter como alvo.

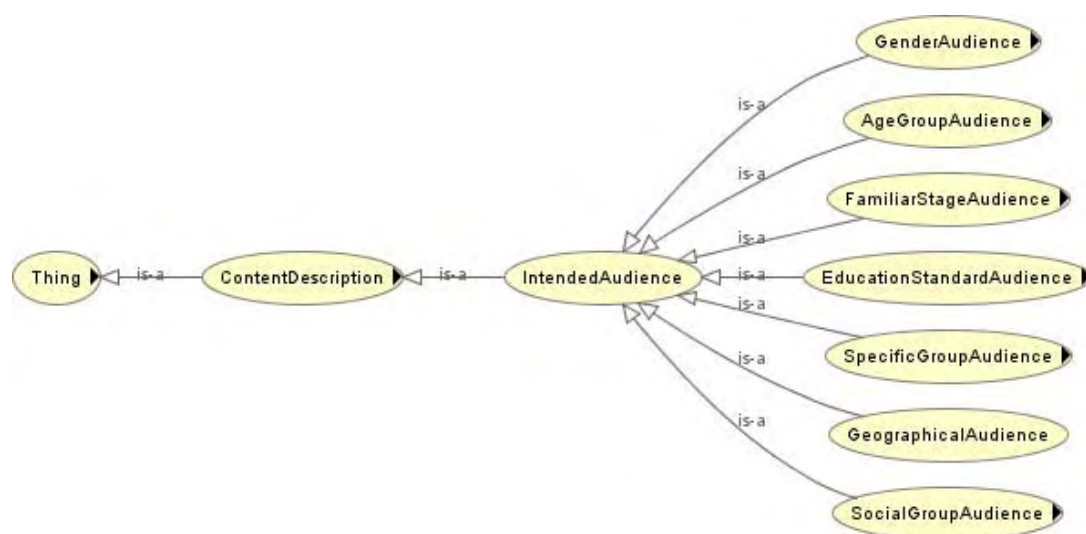


Figura 31 - Hierarquia *IntendedAudience*

Entretanto, esta hierarquia contém o seguinte conjunto de hierarquias internas, ilustradas nas Figuras 32 à 37:

- **AgeGroupAudience:** *Age0-3Years, Age4-7Years, Age8-13Years, Age14-15Years, Age16-17Years, Age18-24Years, Age25-34Years, Age35-44Years, Age45-54Years, Age55-64Years, Age65+Years.*

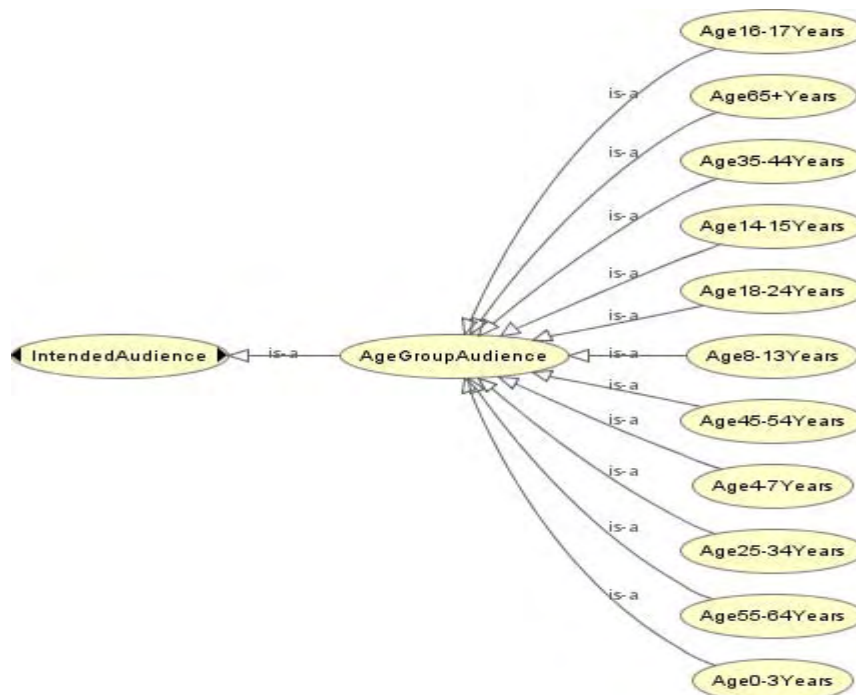


Figura 32 - Hierarquia *AgeGroupAudience*

- *EducationalStandardAudience*: *NoSchool*, *PreSchoolEducation*, *PrimaryEducation*, *SecondaryEducation*, *TertiaryEducation*, *PosGraduation*.

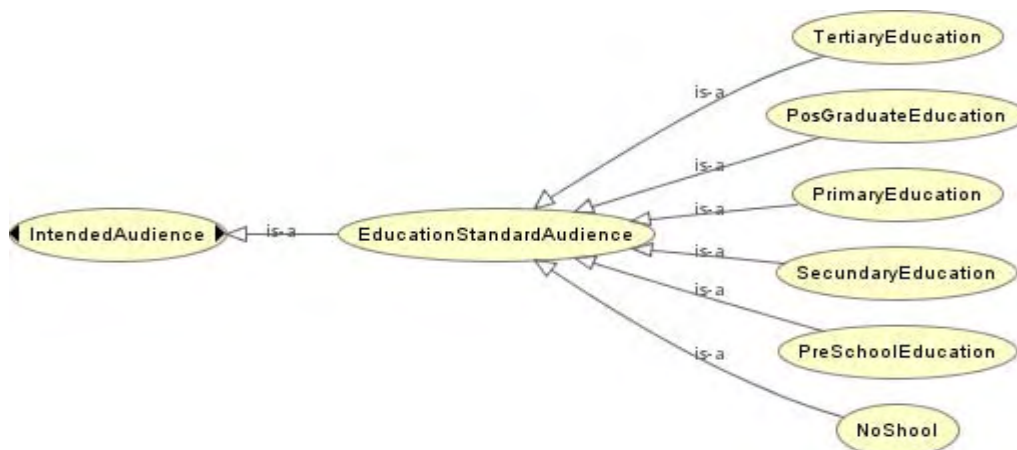


Figura 33 - Hierarquia *EducationStandardAudience*

- *FamiliarStageAudience*: *CoupleFamiliarStage*, *Retired*, *Single-Familiar* e *ChildrenFamiliarStage*, que é especializado em *Chil-dreno-3Years*, *Children4-7Years*, *Children8-15Years* e *Chil-dren16+Years*.

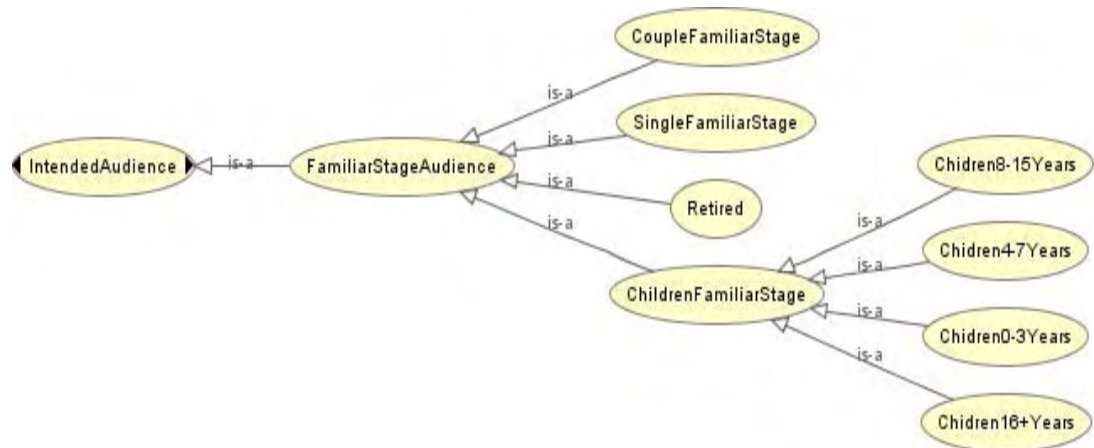


Figura 34 - Hierarquia *FamiliarStageAudience*

- *GenderAudience*: *FemaleGendeAudience* e *MaleGenderAudience*.

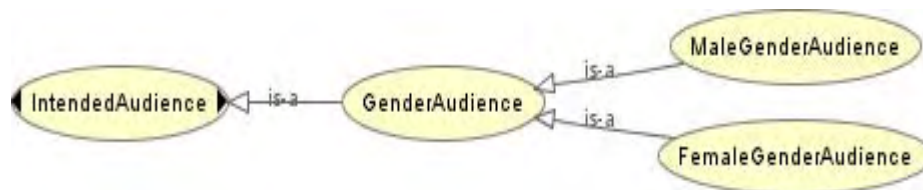


Figura 35 - Hierarquia *GenderAudience*

- *SocialGroupsAudience*: *ClassA*, *ClassB*, *ClassC*, *ClassD* e *ClassE*.

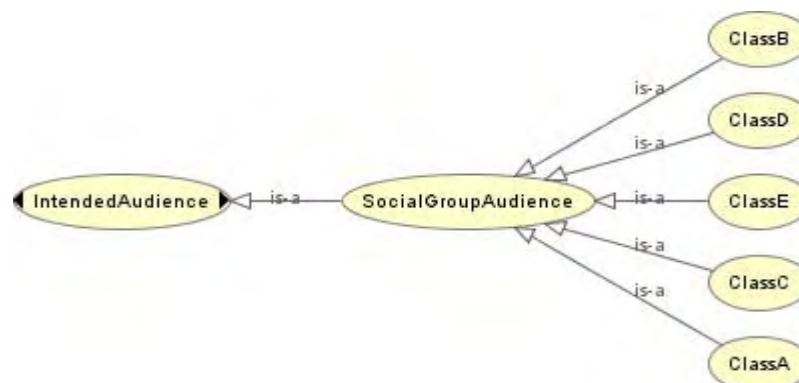


Figura 36 - Hierarquia *SocialGroupAudience*

- *SpecificGroupAudience: EthnicGroup, ImigrantGroup, Indigenous-Group, ReligiousGroup, StudentsGroup e OtherSpecificGroup.*

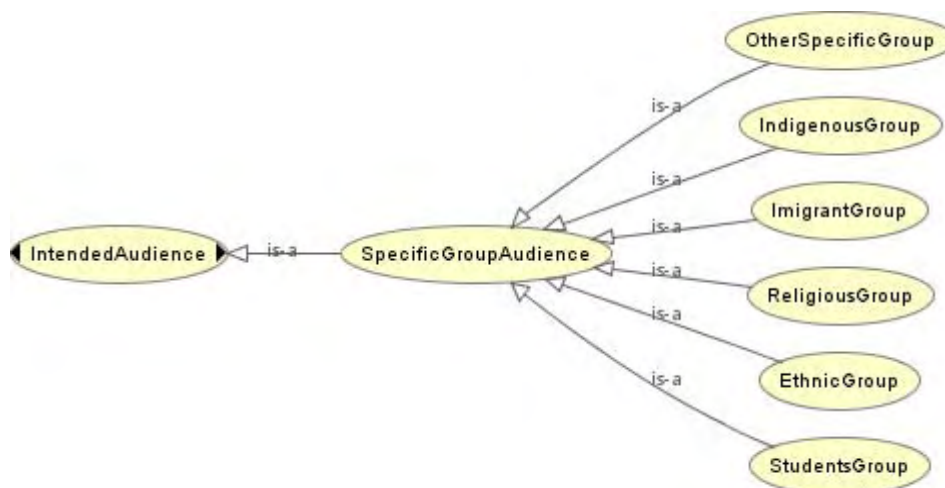


Figura 37 - Hierarquia *SpecificGroupAudience*

A hierarquia de classes que tem por base *ContentDescription*, cuja visualização completa pode ser observada na Figura 38, é a mais representativa quando relacionada a descrição do conteúdo sob a óptica de utilização e alvo do usuário de TV. O nível de expressividade atingido permite construir diversos cenários de uso, nos quais esta ontologia pode ser usada ou estendida para domínios de uso específico, como os cenários apresentados no capítulo 5.

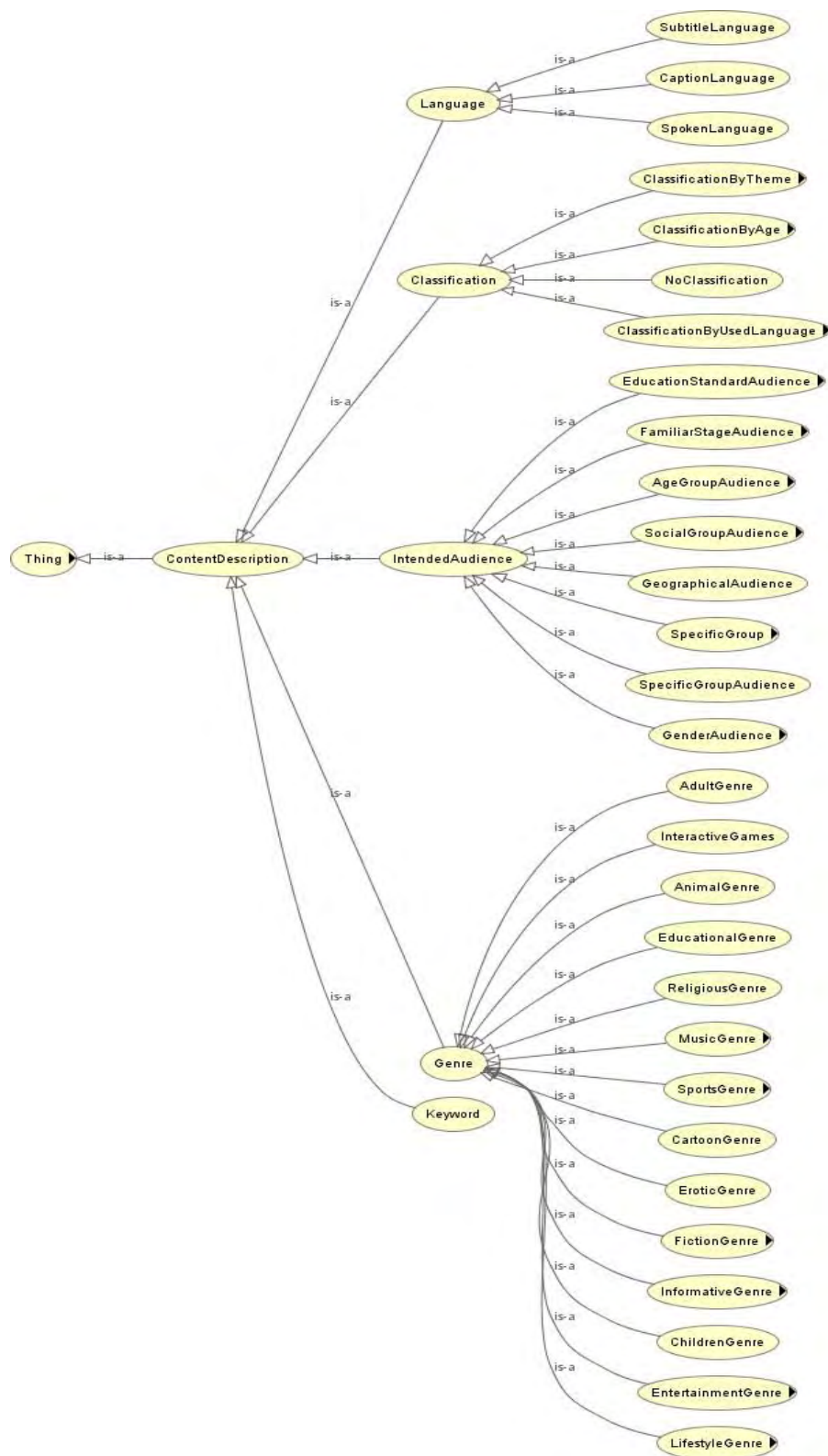


Figura 38 - Visão completa de hierarquia *ContentDescription* (até o 3º nível)

4.3.2.2 Créditos

A constituição da estrutura hierárquica de **Credits**, ilustrada na Figura 39, fundamentou-se inicialmente no agrupamento dos conceitos comuns de pessoa, revisão e organização, representadas nas classes **Person**, **Review** e **Organization**, considerando a forma como eles são utilizados com relação ao uso em conteúdo multimídia. Tais informações constituem um grupo de dados importantes do conteúdo multimídia, por descrever entidades do mundo real que participam do processo de criação e produção, sendo muitas vezes utilizadas como critérios de escolha pelo usuário no momento em que ele decide o que conteúdo consumir.

A classe **Person** também constitui uma base hierárquica, da qual podem ser especializados os seguintes conceitos, modelados por apresentarem relação direto processo de criação do conteúdo: **Author**, **Actor**, **Director**, **Producer** e **Singer**. Além dessas, há ainda a classe **TVUser**, que representa o usuário que participa da avaliação do conteúdo, através da escrita de revisões.

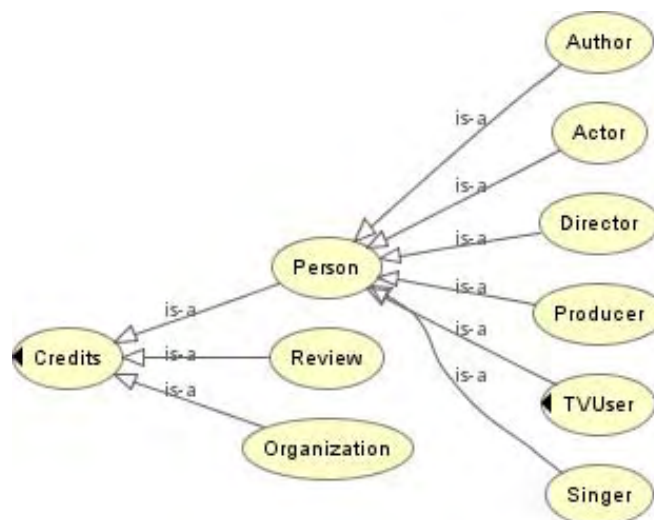


Figura 39 - Hierarquia *Credits*

4.3.2.3 Formato de Mídia

Um conteúdo é composto de, ao menos, um formato de mídia, podendo também ser formado por um conjunto de componentes de mídia de formatos diferentes. O objetivo a hierarquia **MediaFormat**, ilustrada na Figura 40, é especializar o a descrição do conteúdo sob o ponto de vista operacional, ou seja, a partir do ponto de vista em que o conteúdo é visto como matéria prima para

utilização em um propósito específico, juntamente com suas propriedades e características inerentes a cada formato. Compõem esta hierarquia as seguintes classes: *Application*, *Audio*, *Data*, *Graphics*, *Hiperlink*, *Image*, *Text* e *Video*.

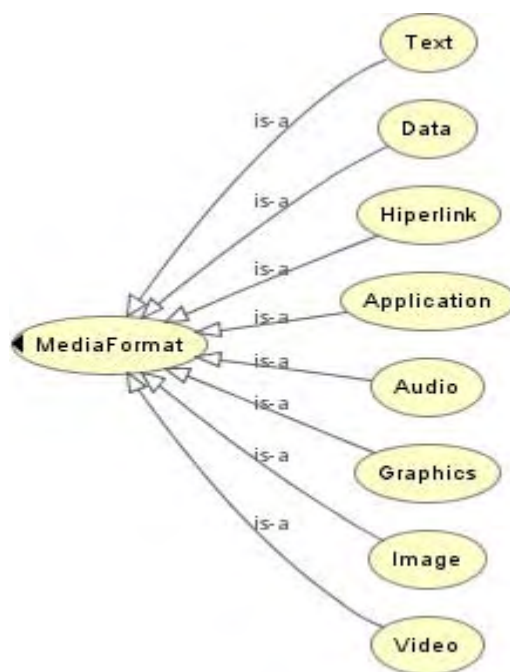
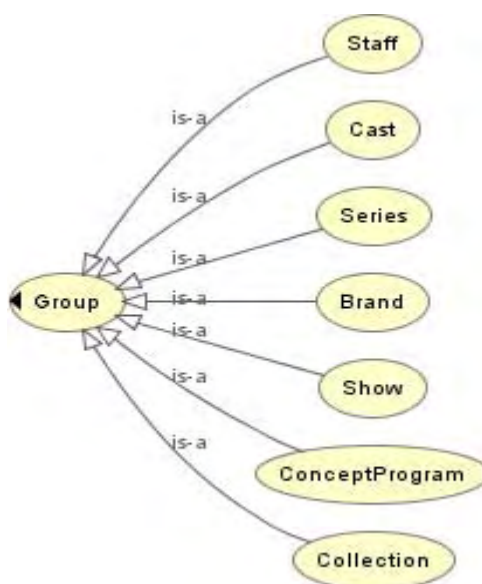


Figura 40 - Hierarquia *MediaFormat*

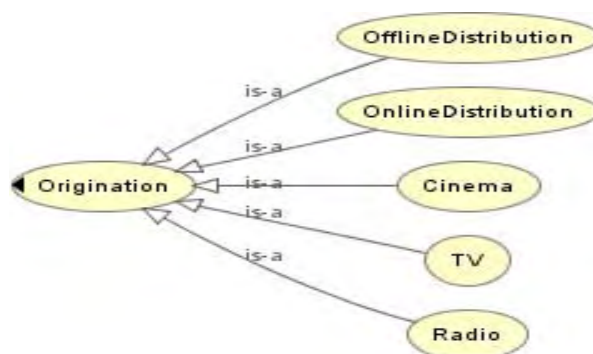
4.3.2.4 Grupo

A hierarquia *Group*, apresentada na Figura 41, tem como fundamento o conceito de que um conteúdo multimídia pode ser parte integrante de um conjunto de conteúdos associados a partir de características específicas. Além disso, o conceito de grupo também se propõe ao agrupamento de informações do corpo técnico responsável pela criação e produção do conteúdo. Fazem parte desta hierarquia as seguintes classes: *Brand*, *Cast*, *Collection*, *ConceptProgram*, *Series*, *Show* e *Staff*.

Figura 41 - Hierarquia *Group*

4.3.2.5 Distribuição Original

Esta hierarquia, que tem por base a classe *Origination*, conforme ilustrado na Figura 42, tem o propósito de representar o modelo de distribuição inicialmente adotado para o conteúdo multimídia. Os seguintes conceitos estão inseridos nesta estrutura hierárquica: *Cinema*, *Radio*, *TV*, além das classes *OfflineDistribution* e *OnlineDistribution*, que representam formas genéricas de distribuição, que podem ser estendidas para uso em domínio específico.

Figura 42 - Hierarquia *Origination*

4.3.2.6 Área Geografica

A hierarquia *GeographicArea*, ilustrada na Figura 4.3, representa a organização geopolítica na qual os continentes e países estão inseridos. Dessa forma, o modelo permite sua utilização e extensão nos mais variados propósitos de uso.

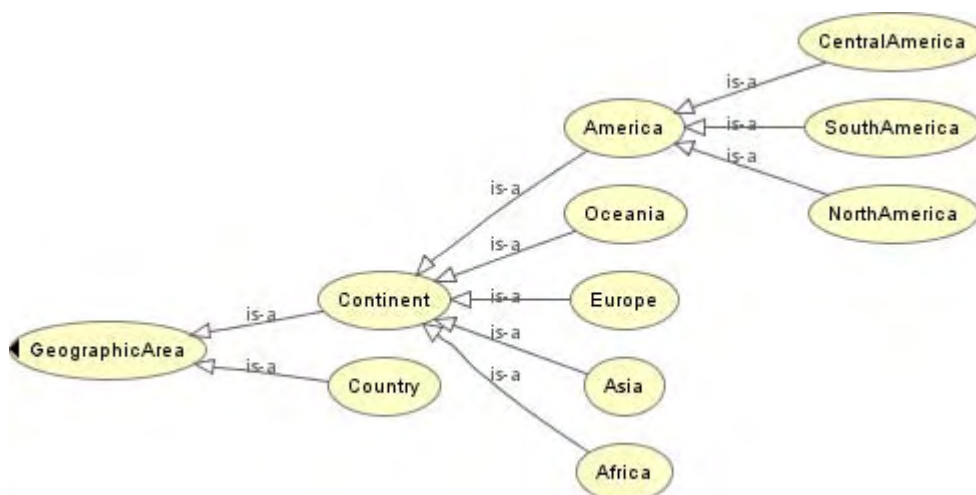


Figura 43 - Hierarquia *GeographicArea*

São modeladas as classes *Country* e *Continent*. A primeira representa a entidade país e a segunda, a entidade continente, sendo especializadas nos seguintes conceitos, correspondentes à realidade da estrutura geopolítica: *Africa*, *Asia*, *Europe*, *Oceania* e *America*, que agrupa as classes *CentralAmerica*, *NorthAmerica* e *SouthAmerica*.

4.3.3 Propriedades

Para estabelecer a conexão entre conceitos que retratassem as ligações entre as entidades no mundo real foram definidas as seguintes propriedades:

1. *hasActor*: possui as classes *MMContent* e *Cast* como domínio e *Actor* como contradomínio. Tem equivalência funcional com a propriedade *hasActress*.
2. *hasActress*: possui as classes *MMContent* e *Cast* como domínio e *Actor* como contradomínio. Sua diferença com a propriedade *hasActor* ocorre da instância sobre a qual é aplicada, que varia de acordo com sexo.

3. *hasAgeClassification*: possui a classe *MMContent* como domínio e *ClassificationByAge* como contradomínio.
4. *hasAuthor*: possui a classe *MMContent* como domínio e *Author* como contradomínio.
5. *hasAward*: apresenta as classes *MMContent*, *Author*, *Actor*, *Singer*, *Producer* e *Director* como domínio e *Award* como contradomínio.
6. *hasCast*: apresenta a classe *MMContent* como domínio e *Cast* como contradomínio.
7. *hasCaptionLanguage*: apresenta a classe *MMContent* como domínio e *CaptionLanguage* como contradomínio.
8. *hasClassification*: apresenta a classe *MMContent* como domínio e *Classification* como contradomínio.
9. *hasCredits*: apresenta a classe *MMContent* como domínio e *Credits* como contradomínio.
10. *hasDepictedLocation*: apresenta a classe *MMContent* como domínio e *GeographicArea* como contradomínio.
11. *hasDescription*: apresenta a classe *MMContent* como domínio e *ContentDescription* como contradomínio.
12. *hasDirector*: apresenta a classe *MMContent* como domínio e *Director* como contradomínio.
13. *hasGenre*: apresenta a classe *MMContent* como domínio e *Genre* como contradomínio.
14. *hasIntendedAudience*: apresenta a classe *MMContent* como domínio e *IntendedAudience* como contradomínio.
15. *hasKeyword*: apresenta as classes *MMContent* e *EmergencyAlert* como domínio e *Keyword* como contradomínio. Possui equivalência funcional com a propriedade *hasTopic*.
16. *hasLanguage*: apresenta a classe *MMContent* como domínio e *Language* como contradomínio.
17. *hasLocation*: apresenta a classe *MMContent* como domínio e *Location* como contradomínio.
18. *hasLocationAlertFor*: apresenta a classe *EmergencyAlert* como domínio e *GeographicArea* como contradomínio.
19. *hasLocationDistribution*: apresenta a classe *Origination* como domínio e *GeographicArea* como contradomínio.

-
20. *hasLocationProduction*: apresenta a classe *MMContent* como domínio e *Location* como contradomínio.
 21. *hasLocationRelease*: apresenta a classe *MMContent* como domínio e *Location* como contradomínio.
 22. *hasMediaFormat*: apresenta a classe *MMContent* como domínio e *MediaFormat* como contradomínio.
 23. *hasMusicStyle*: apresenta a classe *Singer* como domínio e *MusicGenre* como contradomínio.
 24. *hasProducer*: apresenta a classe *MMContent* como domínio e *Producer* e *Organization* como contradomínios.
 25. *hasReviewer*: apresenta a classe *Review* como domínio e *Person* como contradomínio.
 26. *hasSinger*: apresenta a classe *MMContent* como domínio e *Singer* como contradomínio.
 27. *hasSpokenLanguage*: apresenta a classe *MMContent* e *Audio* como domínios e *SpokenLanguage* como contradomínio.
 28. *hasStaff*: apresenta a classe *Organization* como domínio e *Staff* como contradomínio.
 29. *hasSubgenre*: apresenta a classe *MMContent* como domínio e *Genre* como contradomínio.
 30. *hasSubtitleLanguage*: apresenta a classe *MMContent* como domínio e *SubtitleLanguage* como contradomínio.
 31. *hasSynopsis*: possui a classe *MMContent* como domínio e *string* como tipo de dados.
 32. *hasTextLanguage*: apresenta a classe *MMContent* e *Text* como domínios e *TextLanguage* como contradomínio.
 33. *hasTitle*: possui a classe *MMContent* como domínio e *string* como tipo de dados.
 34. *hasThemeClassification*: apresenta a classe *MMContent* como domínio e *ClassificationByTheme* como contradomínio.
 35. *hasUsedLanguageClassification*: apresenta a classe *MMContent* como domínio e *ClassificationByUsedLanguage* como contradomínio.
 36. *hasTopic*: apresenta as classes *MMContent* e *EmergencyAlert* como domínios e *Keyword* como contradomínio. Possui equivalência funcional com a propriedade *hasKeyword*.

-
37. *isActressOf*: possuía a classe *Actor* como domínio e a classe *MMContent* como contradomínio. Possui equivalência funcional com a propriedade *isActorOf*.
38. *isActorOf*: possuía a classe *Actor* como domínio e a classe *MMContent* como contradomínio. Possui equivalência funcional com a propriedade *isActressOf*.
39. *isDescriptionOf*: possuía a classe *ContentDescription* como domínio e a classe *MMContent* como contradomínio.
40. *isEmergencyAlertTransmittedInto*: possuía a classe *EmergencyAlert* como domínio e a classe *MMContent* como contradomínio.
41. *isGroupPartOf*: possuía a classe *MMContent* como domínio e a classe *MMContent e Series* como contradomínio. Esta propriedade deve ser usada quando o conteúdo multimídia representa um segmento ou episódio de um grupo de conteúdo ou um conteúdo completo.
42. *isLocationAlertFor*: possuía a classe *EmergencyAlert* como domínios e a classe *GeographicArea* como contradomínio.
43. *isMediaComponentOf*: possuía a classe *MediaFormat* como domínio e a classe *MMContent* como contradomínio.
44. *isOriginallyDistributedFor*: possuía a classe *MMContent* como domínio e a classe *Origination* como contradomínio.
45. *isProvidedBy*: possuía a classe *Award* como domínio e a classe *Organization* como contradomínio.
46. *isProductionLocationOf*: possuía as classes *Country e Continent* como domínios e a classe *MMContent* como contradomínio.
47. *isReleaseLocationOf*: possuía as classes *Country e Continent* como domínios e a classe *MMContent* como contradomínio.
48. *isReviewOf*: possuía a classe *Review* como domínio e a classe *MMContent* como contradomínio.
49. *isWrittenBy*: possuía a classe *Review* como domínio e a classe *Person* como contradomínio.

4.3.4 Usando a Ontologia *OWL Time*

Informações de tempo como horário de início ou duração de um programa, a data em que uma nova série começa a ser transmitida ou é encerrada, o ano em que um filme ou artista ganhou um prêmio, dentre outras informações temporais

são bastante comuns de serem requisitadas. A partir de necessidades como estas, juntamente com os dados operacionais categorizados como “Unidade de Tempo”, tornou-se essencial que a CoreKTV também fornecesse meios de estabelecer a descrição temporal associada ao conteúdo multimídia.

Entretanto, como a descrição de informações de tempo é bastante comum em diversos outros domínios, após elencar os conceitos que deveriam ser fornecidos na Ontologia Núcleo, buscou-se encontrar uma ontologia que apresentasse objetivos semelhantes e, com isso, equivalência de conceitos. Desta forma, fica reforçada uma das premissas da Web Semântica, na qual a reutilização e extensão de conhecimento e, com isso facilitar a integrabilidade entre sistemas.

A ontologia *OWL Time (W3C; 2006)* foi desenvolvida pelo Grupo de Trabalho para Desenvolvimento da Web Semântica, e se propõem a descrever conteúdo e propriedades temporais de páginas e serviços da *Web*. Ela fornece um vocabulário para expressão de fatos sobre relações topológicas entre instantes e intervalos, juntamente com informações sobre data, horário e duração. A sua utilização junto à CoreKTV passa por reuso de dois conceitos básicos: o de descrição de data e duração de intervalo de tempo, além de propriedades que estabelecem a conexão com conceitos relacionados à conteúdo multimídia da CoreKTV. Assim, temos o uso das seguintes classes:

- *DateTimeDescription*: representa a descrição de uma data com horário;
- *DurationDescription*: representa a descrição de um intervalo de tempo ou seqüência temporal.

Além da definição das classes, também foram estendidos os campos de *year, month, day, hour, minute* e *second*, correspondendo aos atributos de ano, mês, dia, hora, minuto e segundo, respectivamente. Com isso, a partir da reuso dessas classes e atributos, ou propriedades de dados, foram definidas as seguintes propriedades, cujas finalidades estão na ligação entre os conceitos definidos pela CoreKTV e as duas classes descritas acima.

1. *hasDuration*: possui a classe *MMContent* como domínio e *DurationDescription* como contradomínio.
2. *hasAwardYear*: possui a classe *Award* como domínio e a classe *DateTimeDescription* como contradomínio.
3. *hasAwardDate*: possui a classe *Award* como domínio e a classe *DateTimeDescription* como contradomínio, considerando apenas o campo *year*.

4. *hasDepictedDate*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio.
5. *hasDepictedYear*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio, considerando apenas o campo *year*.
6. *hasFinishDate*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio.
7. *hasFinishTime*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio, considerando apenas os campos *hour*, *minute* e *second*.
8. *hasLocalOffsetTime*: possui a classe *GeographicArea* como domínio e a classe *DurationDescription* como contradomínio.
9. *hasLocalTime*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio.
10. *hasProductionDate*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio.
11. *hasProductionDate*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio, considerando apenas o campo *year*.
12. *hasReleaseDate*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio.
13. *hasReleaseYear*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio, considerando apenas o campo *year*.
14. *hasStartDate*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio.
15. *hasStartTime*: possui a classe *MMContent* com domínio e a classe *DateTimeDescription* como contradomínio, considerando apenas o campo *year*.

4.4 Componente *Semantic Integration*

O componente chamado *Semantic Integration*, mostrado na Figura 18 representa o ponto de início da integração entre a plataforma tradicional da TVDI e a infraestrutura do *Knowledge TV*. A sua principal função é realizar a captura de

dados e informações que sejam processadas e utilizadas em diversos serviços e aplicações baseadas em conhecimento, desenvolvidas a partir da plataforma do KTV. Para isso, ele está inserido no núcleo comum do *middleware* Ginga, conforma arquitetura ilustrada na Figura 44, e mantém comunicação direta com os componentes de sintonização e provedor de informações de serviço.

Apesar de não estar presente na norma que descreve uma arquitetura sugerida para o *middleware* Ginga, a criação deste componente tornou-se essencial para a construção da infraestrutura semântica no modo cliente-servidor. Tal necessidade foi levantada a partir de dois pontos: **(i)** atualmente, não há especificação no Ginga de unidades capazes de fornecer semântica às informações obtidas do fluxo, com isso, foi necessário especificar agentes que se comunicam com demais componentes do *middleware* para captar dados; **(ii)** as restrições físicas existentes no terminal tornam inviável embarcar toda a infra-estrutura do KTV, tornando necessária a definição de uma unidade de processamento inicial, o *Semantic Integration*, no ambiente do *Set-Top-Box*. A arquitetura do componente foi desenvolvida a partir do guia de desenvolvimento de componentes do projeto GingaCDN (LAVID; 2010), disponível para implementação no modelo FlexCM (MIRANDA FILHO *et. al.*; 2007).

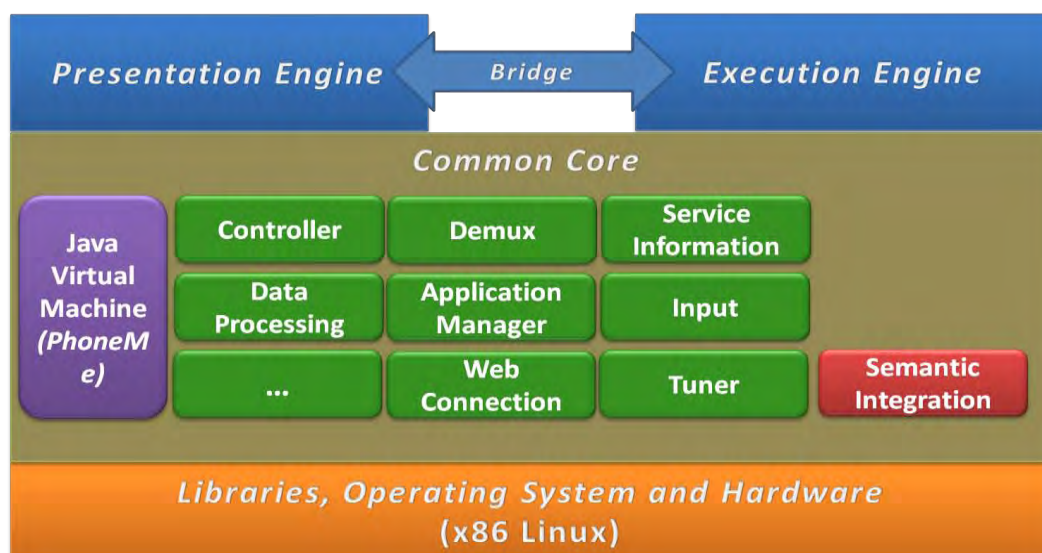


Figura 44 - Componente *Semantic Integration* na arquitetura do *middleware* Ginga

Conceitualmente, o componente *Semantic Integration* é constituído por duas unidades básicas denominadas de agente Provedor e Monitor, responsáveis por

estabelecer a comunicação com os demais componentes do *middleware* para obtenção dos dados. Eles são os responsáveis pela recuperação de todas as informações relacionadas ao conteúdo e aos usuários.

A integração entre o componente e o ambiente de componente é realizada pela interface *ISemanticAgent*, que também é responsável por fornecer métodos comuns para encapsulamento das informações em arquivos XML, além de representar o meio de comunicação entre o componente e quem faz chamadas aos seus métodos. Os cabeçalhos da API do *Semantic Integration* podem ser encontrados no Apêndice B desta dissertação e também representam parte das contribuições deste trabalho.

4.4.1 Agente Provedor

O Agente Provedor tem a função de recuperar as informações sobre a programação transmitida. Para isso, podem ser usadas duas fontes de recursos: (i) os metadados transmitidos junto ao fluxo e processados no componente de Informação de Serviços do *middleware* e; (ii) informações da Web, que podem ser adquiridas através de busca em repositórios de dados. Observe que a convergência semântica entre TVDI e Web apenas é possível se ambas as tecnologias forem baseadas nos mesmos princípios semânticos.

Como vista na Figura 45, o Agente Provedor é especializado em dois outros agentes que correspondem as principais conceitos da Ontologia Núcleo, correspondentes as classes mais genéricas das hierarquias internas à classe *Thing*. Suas funções estão relacionadas às seguintes ações:

- ***ProviderAgent***: Classe responsável por fornecer métodos para busca e recepção de dados a partir da Web que possam enriquecer a descrição do conteúdo, abstraindo as questões relativas ao canal de comunicação. Os métodos definidos nesta classe devem poder ser usados por objetos das classes *MMContent* e *ContentDescription* para recuperar e incluir informações à instâncias das classes. O uso de consultas e requisições de informação em repositórios da Web não está incluso no escopo deste trabalho e, embora prevista como fonte de recursos na arquitetura do componente de integração semântica, ela deve ser construída como classe interna, de acordo com a plataforma ou sistema Web se conectará para realizar as consultas.

- **MMContent:** classe responsável pelo estabelecimento de regras a serem transformadas em propriedades da ontologia. Cada objeto desta classe gera uma instância que representa o conteúdo a ela associado, juntamente com uma coleção de objetos *ContentDescription*.
- **ContentDescription:** classe que obtém junto ao componente do *middleware* toda a descrição do conteúdo transmitido.

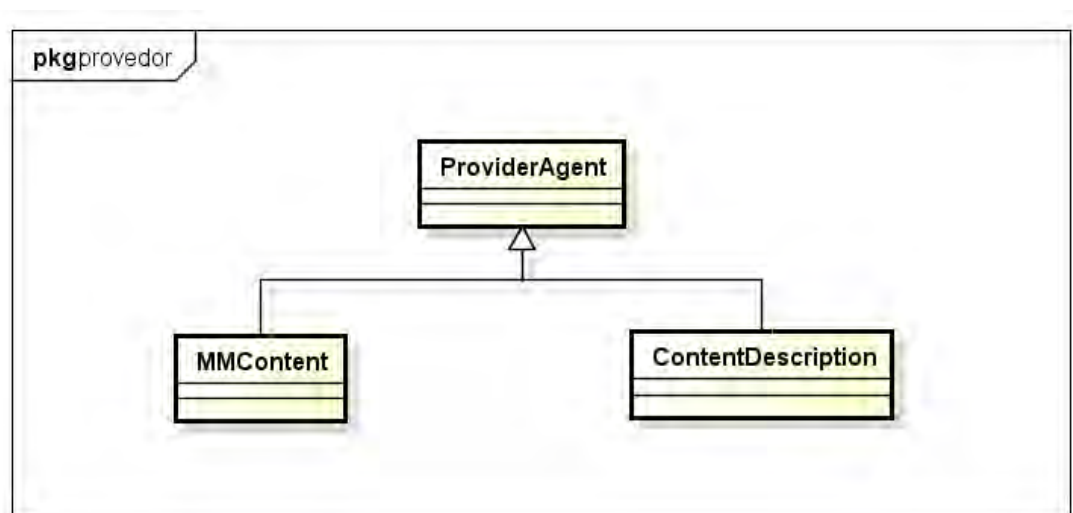


Figura 45 - Estrutura do Agente Provedor

4.4.2 Agente Monitor

O agente Monitor tem como função principal o monitoramento do comportamento do usuário em relação ao conteúdo exibido na TV. Para isso, ele obtém informações básicas sobre os programas assistidos e registra todas as interações realizadas através do controle remoto. Tais informações são enviadas periodicamente ao servidor, onde está construída e é atualizada uma ontologia para o perfil de uso da TV, capaz de refletir o padrão comportamental e histórico de uso da TV e a partir disto, identificar padrões de programas que possam ser interessantes aos usuários da TV.

Para refletir as diferentes informações acerca do usuário, foi definida a classe *UserMonitorAgent*, cuja função é monitorar as mudanças de canal a partir de integração junto ao componente de Sintonização do *middleware* e buscar junto ao componente de Informação de Serviço informações mínimas sobre o programa, que ajudarão a identificar o perfil de audiência atual do usuário. Além disso, tal classe

fornece método definindo intervalo de tempo mínimo para que um registro seja efetuado acerca das ações do usuário. As informações obtidas com esta classe refletem o modelo de dados do usuário definidos na tabela 5 (Modelo de dados do usuário).

4.5 Capturador de Dados

O componente *DataCatcher*, ou Capturador de Dados, tem a função de receber os dados e informações que serão usadas nas operações dos componentes do KTV – Módulo Servidor. Em outras palavras, o Capturador de Dados representa o ponto de entrada na plataforma. Sua arquitetura, conforme ilustração da Figura 46, é relativamente simples, consistindo de duas interfaces e um diretório para armazenamento de arquivos, o *XML Files Directory*.

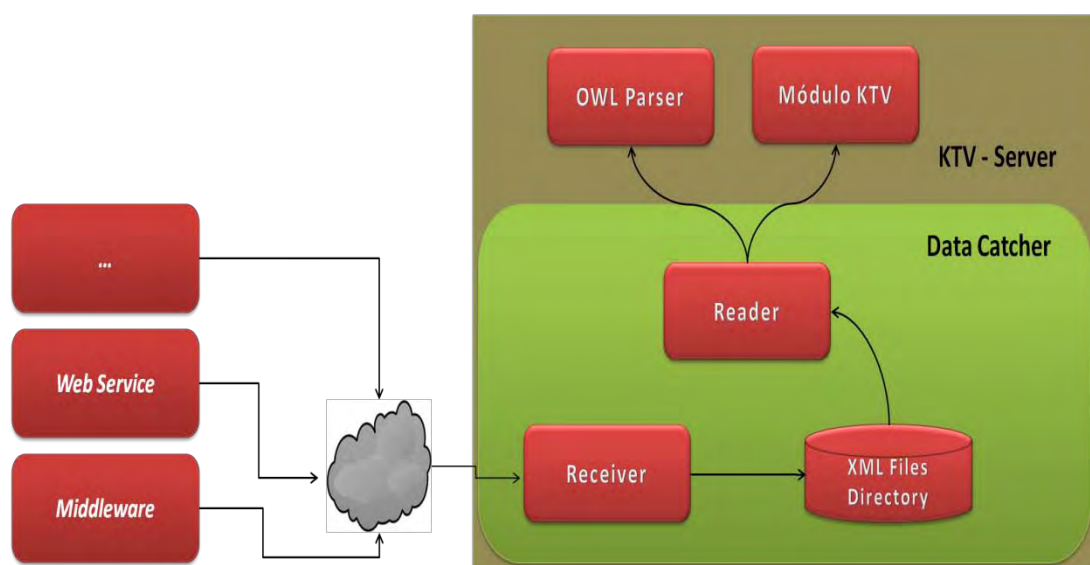


Figura 46 - Arquitetura do componente Capturador de Dados

As duas interfaces presentes no componente têm por objetivo:

- **Receiver**, ou Receptor: obter os arquivos transmitidos na rede e endereçados ao KTV- Módulo Servidor. Uma vez que o formato de arquivo utilizado é XML, fica abstraída a origem, bem como a forma como transmitida a mensagem. Tal abstração permite, por exemplo, que além da comunicação com o *middleware*, outros serviços da *web* ou aplicações possam se comunicar com a plataforma, enviando arquivos contendo informações a serem armazenadas e utilizadas para operações ou componentes específicos do KTV.

- **Reader**, ou Leitor: Recuperar os arquivos XML presentes no *XML File Directory*, enviando ao solicitante uma cópia das informações, para evitar situações onde o mesmo recurso está sendo usado por dois ou mais solicitantes.

4.6 OWL Parser

Com a função de realizar a tradução para formato de ontologia das informações que se encontram no arquivo XML, recebidos no *DataCatcher*, o *OWL Parser* do módulo servidor do *Knowledge TV* apresenta uma arquitetura simplificada, conforme ilustra a Figura 47.

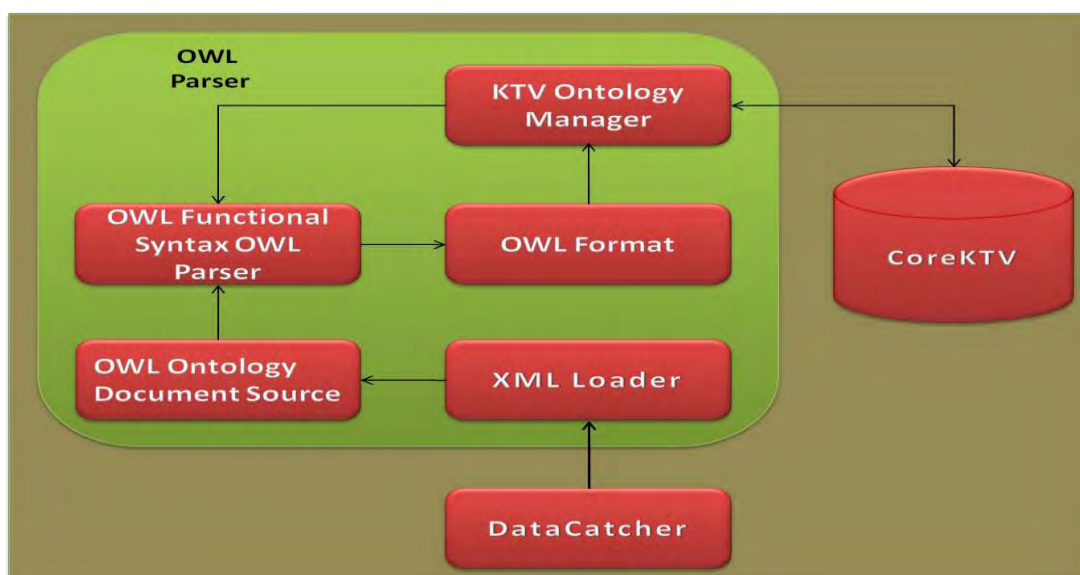


Figura 47 - Arquitetura do componente *OWL Parser*

Durante a especificação e construção dessa unidade de processamento foi usada a OWLAPI¹², é uma API Java e implementação de referência para criação, manipulação e serialização de ontologia na linguagem OWL. Sua distribuição está sob as licenças LGPL e Apache, além de ser desenvolvida com código aberto. As classes descritas na arquitetura da Figura 48 têm as seguintes funcionalidades:

- **XML Loader**: interface responsável requisitar os arquivos XML do diretório de dados, via comunicação com a classe *Reader DataCatcher*. Ela funciona

¹² <http://owlapi.sourceforge.net/>

como um ouvinte que se registra para receber sempre que houver novos arquivos com informações a serem traduzidas para o formato de ontologia ou incorporadas à uma ontologia existente. Um exemplo de arquivo XML de entrada no XML Loader, contendo as informações sobre o conteúdo é apresentado na Figura 48.

- **OWL Ontology Document Source:** Interface da OWL API que é implementada para o processamento de documento para obtenção de ontologia. Neste caso, ela é implementada para processar um recurso em formato de arquivo XML ou *strings* contendo dados desse arquivo.

```
<DOCTYPE mmcontent SYSTEM "mmcontent.dtd">
<mmcontent contentName="Brasil x Argentina" contentID="1234">
  <description>
    <contentDescription>
      <genre>Sports</genre>
      <subgenre>Soccer</subgenre>
      <classification>
        <ByAge>All Ages</ByAge>
      </classification>
    </contentDescription>
  </description>
  <credits>
    <award name="Copa Rocca">
      <awardProvider>
        <organization name="Confederação Sulamericana de Futebol" id="CSF"/>
      </awardProvider>
    </award>
  </credits>
  <geographicArea>
    <country name="Brasil" id="BRA">
      <city>Belem</city>
    </country>
  </geographicArea>
  <mediaFormat>
    <video>H.264</video>
  </mediaFormat>
  <origination>TV</origination>
  <origination>Radio</origination>
  <dateTime name="startTime">
    <date>
      <day>14</day>
      <month>September</month>
      <year>2011</year>
    </date>
    <time>
      <hour>21</hour>
      <minute>50</hour>
      <second>00</second>
    </time>
  </dateTime>
  <dateTime name="finishTime">
```

Figura 48 - Documento de entrada XML descrevendo um conteúdo multimídia

- **KTV Ontology Manager:** Classe desenvolvida como uma extensão de *Ontology Manager*, da OWL API, com a função de gerenciar as operações que manipulam a ontologia. É realizado, através desta classe, o carregamento

da ontologia já existentes na base, bem como o armazenamento, após sua manipulação.

- **OWL Parser:** Interface da OWL API cuja função é realizar o *parsing* de um recurso de dados específico para uma ontologia OWL.
- **KTV OWL Ontology Format:** Classe extendida de OWL Ontology Format, da OWL API, que representa o resultado final da tradução, com uma ontologia seguindo as especificações da CoreKTV.

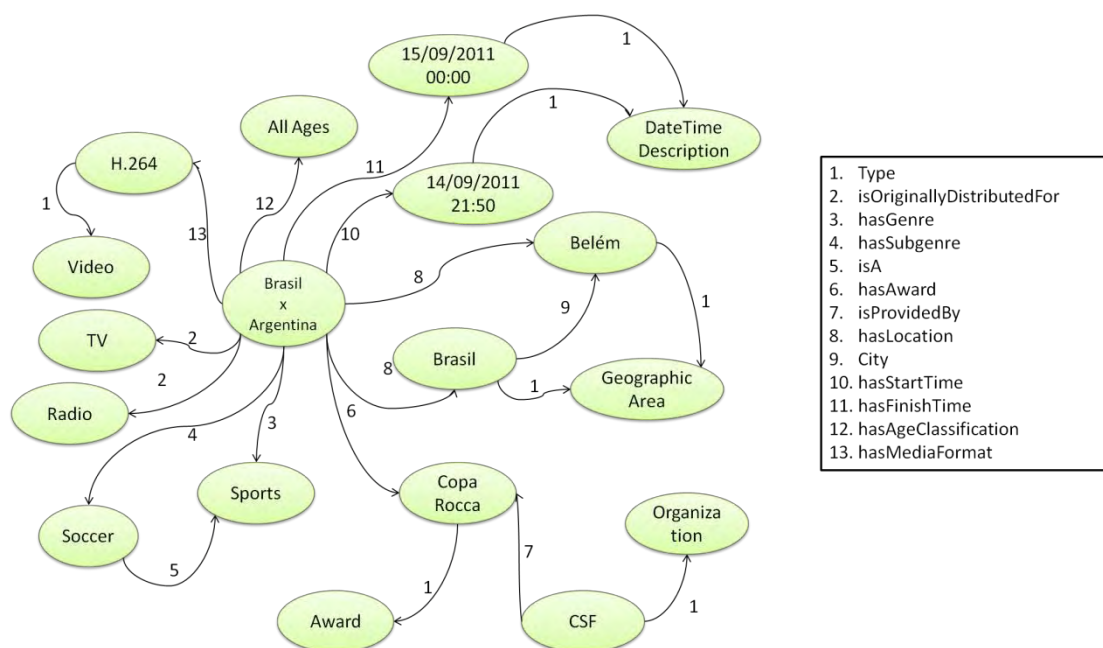


Figura 49 - Arquivo XML traduzido para o formato de ontologia

A Figura 49 apresenta um trecho do formato final, em ontologia, do arquivo XML de entrada observado na figura anterior. Nela é possível observar instâncias das classes e propriedades definidas no modelo semântico, apresentado na subseção 4.3

4.7 Conclusão

Neste capítulo foi apresentado todo o processo de especificação e construção da CoreKTV, desde a forma como é realizada a integração junto ao *middleware* Ginga até o formato de representação da informação que descreve o conteúdo de TV, após a tradução do formato sintático para o formato em ontologia, integrada à plataforma *Knowledge TV*.

Capítulo

5

Cenário de uso: Recomendação de conteúdo em TVDI

“Quem quer vencer um obstáculo deve armar-se da força do leão e da prudência da serpente.”

Píndaro

5.1 Motivação

Em meio às inúmeras mudanças ocorridas no ambiente de TV, o usuário tem se deparado com uma diversidade de opções de conteúdo, programas e serviços aos quais tem acesso. Com isso, a simples ação de encontrar ou buscar um programa desejado torna-se uma atividade demorada e pouco eficiente, principalmente quando considerada a possibilidade de centenas de canais disponíveis. Uma vez estabelecido esse problema, também se passa a enfrentar os mesmos desafios característicos de sistemas da internet, onde os resultados interessantes ao usuário são obtidos junto a muitas outras informações sem utilidade ou descartáveis.

Uma forma de aperfeiçoar a experiência do usuário neste novo cenário da TVDI, permitindo-lhe o direcionamento aos programas de seu interesse, passa pelo

desenvolvimento de uma infraestrutura que suporte o desenvolvimento de sistemas de recomendação (RESNICK, VARIAN; 1997). Esses sistemas visam auxiliar o usuário no processo de busca por conteúdo que sejam de sua preferência, promovendo uma TV personalizada segundo seus interesses, comportamento e histórico de uso, desconsiderando o conteúdo que venha a ser irrelevante ao usuário. Além disso, diante de um contexto de convergência digital, como se apresenta o atual ambiente de TVDI, onde o conteúdo televisivo também é distribuído e consumido em sistemas da internet, torna-se necessária a adoção de conceitos e tecnologias que harmonizem e tornem uniforme a utilização dos serviços e produtos disponíveis.

No presente cenário se propõe a construção de um guia de programação eletrônico, ou EPG, de forma personalizada através do fornecimento de recomendações, considerando padrões de comportamento e ações do usuário junto ao conteúdo da TV. Através dele, busca-se explorar o máximo os recursos da CoreKTV, com o objetivo de promover uma experiência individualizada na forma de consumir conteúdo televisivo.

5.2 Recomendação de conteúdo

A origem dos sistemas de recomendação pode ser atribuída a um processo social e natural, presente na comunicação entre pessoas. Esse processo consiste de consultas e sugestões entre familiares, amigos e colegas de trabalho em geral, com o propósito de reduzir o tempo gasto para um objetivo, em particular, ser alcançado (GOLDBERG et. al., 1992) (RESNICK; VARIAN, 1997). A partir da popularização da internet, sistemas de recomendação passaram a ser largamente utilizados, como por exemplo, em portais de notícias, com a sugestão ao leitor de notícias relacionadas ao mesmo assunto de uma que esta sendo lida, e em lojas de comércio eletrônico, na indicação de produtos e serviços para venda ao consumidor.

Diversos aspectos são considerados na construção de um sistema de recomendação, como a forma pelo qual é feito o gerenciamento do perfil do usuário, o nível de personalização que se deseja obter, as técnicas de filtragem empregadas no sistema para processar os dados e gerar as saídas, além das estratégias adotadas na apresentação das recomendações ao usuário. Neste cenário proposto, a arquitetura foi concebida no formato cliente-servidor, com o objetivo de reduzir a carga de processamento no terminal do usuário e, reduzindo o tráfego de

informações na rede, uma vez que a CoreKTV e os serviços básicos oferecidos na plataforma *Knowledge TV*, que faz uso do mesmo padrão arquitetural, estão concentrados no módulo servidor. O estabelecimento da comunicação entre os módulos cliente e servidor é definido através de chamadas de procedimento remoto, ou RPC, definidas para comunicação entre processos dos serviços Web da plataforma *Knowledge TV*.

5.2.1 Gerência de informações do usuário

A gerência de informações do usuário (GIU) em um sistema de recomendação reflete-se em definições de como o perfil do usuário é construído e como suas informações são coletadas. Gauch *et. al.* (2007) apresenta tais definições das seguintes formas: (i) quanto à construção, o perfil pode ser dinâmico, com as informações modificadas continuamente, ou estático, no qual os dados mantidos na base apresentam um ciclo de vida relativamente longo; (ii) quanto a forma de coleta de informações, pode ocorrer por meios explícitos, através da intervenção direta do usuário, ou de forma implícita, onde os dados são obtidos via monitoramento das ações e comportamento do usuário.

No cenário proposto foram determinadas as técnicas de coleta implícita e o uso de perfis estáticos pelas razões: (i) o ambiente de TV apresenta-se, tradicionalmente, com um nível bem menor de interatividade do usuário quando comparado ao ambiente da Internet, dessa forma, tal atividade de o usuário inserir informações pessoais ou de seu interesse, comuns na coleta explícita, poderia acarretar na perda de interesse e conseqüentemente abandono do uso das recomendações; (ii) o dispositivo para interação na TV, o controle remoto, é reduzido se comparado aos dispositivos de computadores pessoais ou *laptops*, tornando-se ainda maior a necessidade de interação mínima do usuário para geração de informações; (iii) como se deseja obter informações sobre o padrão comportamental ao longo do tempo, o perfil estático se apresenta como a melhor escolha, uma vez que as informações do usuário são persistidas na plataforma do *Knowledge TV*.

Em termos funcionais, a unidade de gerência de informações do usuário é responsável pela recuperação das informações do usuário junto a CoreKTV. A especificação de tais funções é ilustrada no diagrama de caso de uso da Figura 50.

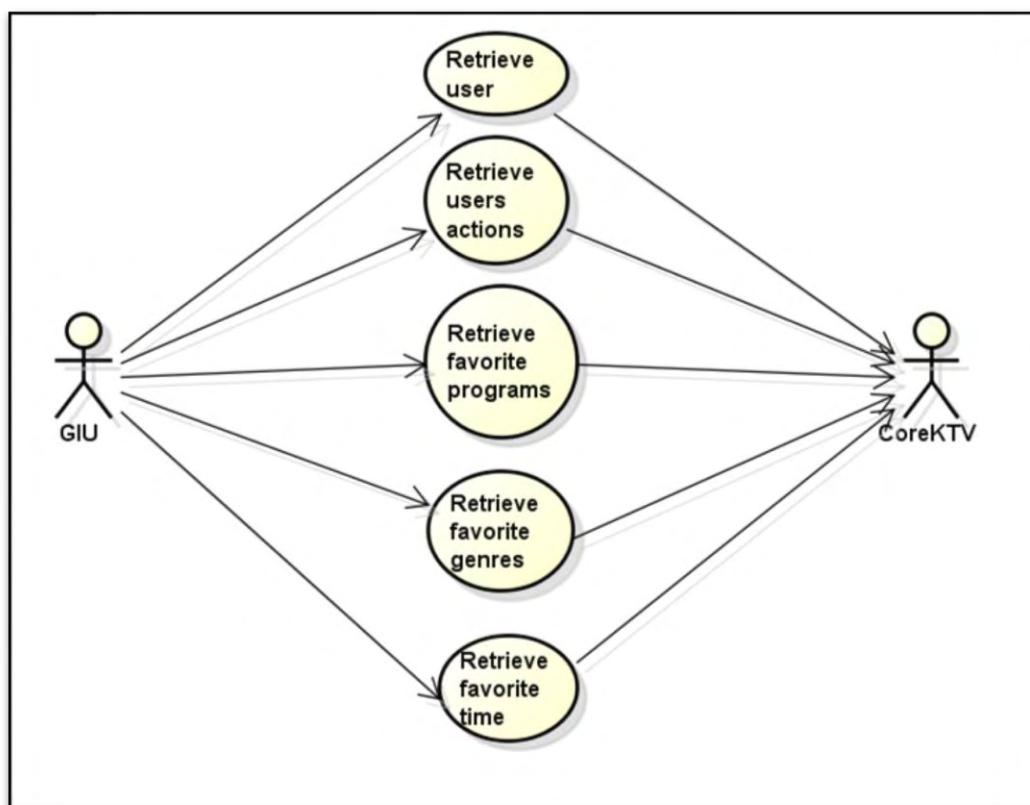


Figura 50 - Diagrama de caso de uso da Gerência de Informação do Usuário (GIU)

O diagrama de casos de uso da figura 50 expõe os seguintes casos de uso:

- 1) Recuperar usuário: o ator GIU, Gerência de Informação do Usuário, requisita ao ator CoreKTV os usuários presentes na base de dados. É retornada uma lista com os identificadores dos usuários presentes na base.
- 2) Recuperar ações do usuário: o ator GIU requisita ao ator CoreKTV o histórico de ações de um usuário, passando como parâmetros o identificador do usuário, um horário de início e um horário de fim. É retornada uma lista de triplas, contendo a ação do usuário, o horário em que foi ela feita e o canal em que estava sendo assistido no momento da ação.
- 3) Recuperar programas favoritos: o ator GIU requisita ao ator CoreKTV os programas favoritos do usuário, passando como parâmetro o identificador do usuário. É retornada uma lista com os nomes do programas de maior audiência por parte do usuário, ou seja, os programas que ele tem o hábito de assistir.
- 4) Recuperar gêneros favoritos: o ator GIU requisita ao ator CoreKTV os gêneros preferidos do usuário, em geral associados aos programas que

assiste com frequência. É passado o identificador do usuário e retornada uma lista com os principais gêneros associados aos programas assistidos e registrados na base. Há de se destacar que também são considerados os subgêneros, os quais apresentam um maior peso na descrição, por representar de forma mais específica uma área de interesse do usuário.

- 5) Recuperar horários preferidos: o ator GIU requisita ao ator CoreKTV a faixa de horário preferida do usuário, passando o seu identificador como parâmetro. Deve ser retornada uma lista com as faixas de horários cujo usuário tem o hábito regular de assistir TV.

5.2.2 Unidade de filtragem de informação do conteúdo

A unidade de filtragem de informações do conteúdo (UF) está relacionada ao processo de captura e filtragem de dados sobre os itens, que no domínio considerado trata-se de informações sobre o conteúdo multimídia. Na literatura podem ser encontradas diferentes técnicas de filtragem da informação aplicadas à recomendação, como os apresentados por Adomavicius e Tuzhilin (2005), Reategu e Cazella (2005) e Peis *et.al.* (2008). Entretanto, destacamos as técnicas de filtragem baseada em conteúdo e baseada em ontologias, por se melhor se enquadrarem na problemática do domínio de recomendação em TV e por melhor fazerem uso dos recursos presentes na CoreKTV.

Em termos funcionais para o cenário proposto, trata-se da unidade de processamento onde serão geradas as recomendações propriamente ditas, através de agrupamentos de instâncias da Ontologia Núcleo da CoreKTV que representam e apresentam informações acerca da programação de TV com o conhecimento obtido acerca do usuário. Tal unidade é desenvolvida no módulo servidor da plataforma Knowledge TV e obtém as informações do usuário via comunicação com a GIU, presente no terminal. A especificação de funções previstas nesta unidade é ilustrada no diagrama de caso de uso da Figura 51.

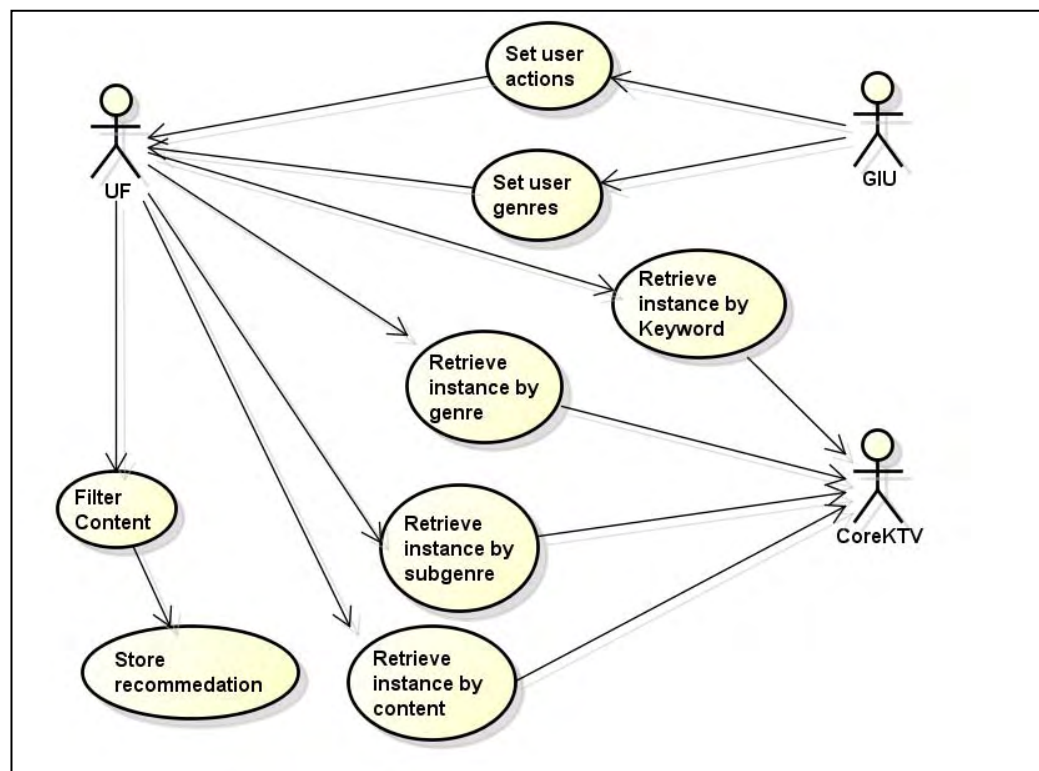


Figura 51 - Diagrama de caso de uso da Unidade de Filtragem (UF)

No diagrama de casos de uso da Figura 51, são expostas as seguintes operações, relacionadas à Unidade de Filtragem:

- 6) Setar ações de usuário: o ator UF recebe do ator GIU lista de parâmetros do usuário relativos à suas ações, como identificador, horários preferidos e lista de ações habitualmente realizadas nestes horários. Tais parâmetros serão usados no processo de filtragem para geração de recomendações baseadas no horário e ações realizadas.
- 7) Setar gêneros do usuário: o ator UF recebe do ator GIU lista de parâmetros que correspondem aos gêneros preferidos do usuário. A partir destes parâmetros será realizada a filtragem do conteúdo que apresenta temática, gênero e palavras chaves similares.
- 8) Filtrar instâncias por palavra-chave: o ator UF requisita ao ator CoreKTV a filtragem de todas as instâncias da base de conhecimento que estão associadas a palavra-chave passada como parâmetro.
- 9) Filtrar instância por gênero: o ator UF requisita ao ator CoreKTV a filtragem de todas as instâncias da base de conhecimento que estão associadas ao gênero passado como parâmetro.

- 10) Filtrar instâncias por subgênero: o ator UF requisita ao ator CoreKTV a filtragem de todas as instâncias da base de conhecimento que estão associadas ao subgênero passado como parâmetro. Há de se destacar que esta filtragem apresenta um maior grau de especialização e, conseqüentemente precisão que a anterior, por gênero.
- 11) Recuperar instâncias de conteúdo: o ator UF requisita ao ator CoreKTV a recuperação da lista de instâncias de conteúdo resultantes dos processos de filtragem anteriormente descritos. Tais instâncias representam os programas de potencial interesse por parte do usuário, uma vez que eles apresentam informações similares às capturadas como seu padrão de consumo.
- 12) Associar conteúdo: o ator UF realiza a operação de agrupar as instâncias de conteúdo recuperadas de acordo com informações de dia e horário. O resultado desta operação é a geração de listas de recomendações por dia e horário descritos como horário de início da transmissão.
- 13) Persistir recomendação: as listas de recomendação são persistidas na base de dados e associadas ao perfil do usuário.

5.2.3 Nível de Personalização

Segundo Schafer (2001), o nível de personalização esta relacionado com a precisão e corretude que se deseja obter com uma recomendação, considerando aspectos do quão útil ela é ao usuário, de acordo com seus interesses e com o momento em que é feita. Podem ser atingidos os seguintes níveis de personalização: (i) não-personalizado, quando uma recomendação é direcionada à todos os usuários; (ii) efêmero, quando uma recomendação é gerada levando em conta informações de momento do usuário e; (iii) persistente, quando as recomendações são privativas e armazenadas no perfil de cada usuário.

O nível de personalização adotado é de recomendações persistentes, uma vez que depois de geradas elas são persistidas e enviadas à aplicação quando requisitadas. Além disso, as recomendações refletem características individuais do comportamento de cada indivíduo, sendo, portanto, privativas e de baixa relevância se considerada para todos os públicos.

5.2.4 Estratégia de recomendação

Diferentes estratégias podem ser adotadas para abordar o usuário com uma recomendação. Reategui e Cazella (2005) sugerem os métodos de (i) recomendação por associação de conteúdo; (ii) avaliação de usuários; (iii) análise de ações; (iv) lista de recomendações e; (v) recomendações próprias.

A estratégia de recomendação adotada neste cenário foi construída baseada na de associação de conteúdo e a de recomendações próprias em uma unidade denominada Unidade de Recomendação (UR) no módulo cliente. A primeira estratégia permite personalizar com base em descrições e atributos do conteúdo e a última por possibilitar o oferecimento de recomendações totalmente dedicadas ao usuário, refletindo fielmente as informações contidas no seu perfil. Desta forma, com o uso das duas estratégias torna-se possível obter os melhores resultados na abordagem das recomendações ao usuário, uma vez que pode ser destacada tanto a personalização a partir das informações do seu perfil, quanto das informações do conteúdo.

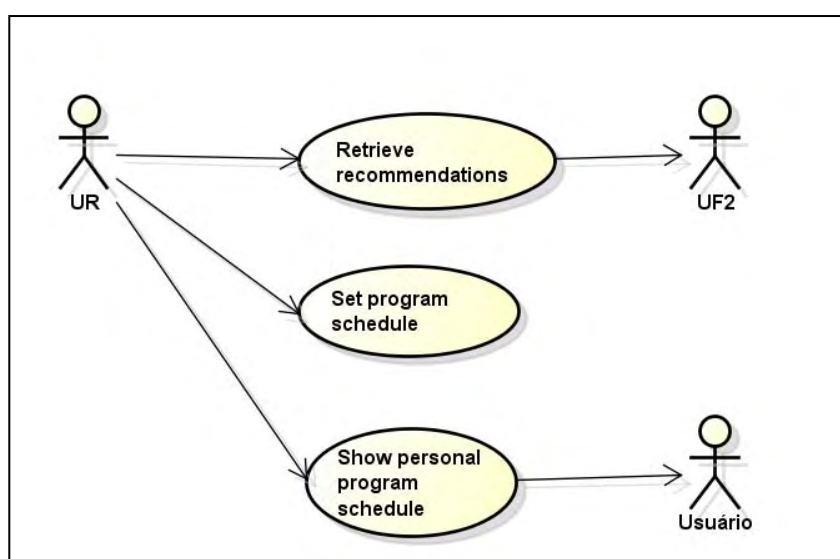


Figura 52 - Diagrama de caso de uso da Unidade de Recomendação (UR)

No diagrama de caso de uso da Figura 52 são apresentadas as principais operações da Unidade de Recomendação.

- 14) Recuperar recomendações: o ator UR requisita ao ator UF a recuperação de todas as recomendações a partir dos parâmetros passados (identificador do usuário, data de início, data de fim).

- 15) Montar grade de programação: o ator UR, a partir das recomendações recuperadas monta a grade de programação personalizada ao usuário.
- 16) Exibir grade personalizada: o ator UR, a partir da ação do usuário para visualizar a grade de programação, exibe as recomendações disponíveis para o dia.

5.3 Conclusão

Neste cenário de uso para a infraestrutura CoreKTV, desenvolvida neste trabalho, foi proposta a construção de um EPG personalizado baseado nos conceitos de sistemas de recomendação. Tal domínio, bastante comum em sistemas da Web, apresentou-se com diversas peculiaridades, que quando aplicadas ao domínio da TV Digital Interativa tornaram-se ainda mais complexos, como o gerenciamento de perfis de usuário, o desenvolvimento e uso de diferentes abordagens para filtragem da informação e a forma de apresentação das recomendações, por exemplo. Entretanto, buscou-se simplificar os problemas e minimizar as complexidades com objetivo de apresentar uma solução viável, destacando o grande potencial de uso da infraestrutura proposta neste trabalho juntamente com a plataforma em que ela está inserida, a *Knowledge TV*, como a utilização das informações obtidas do usuário via componente *Semantic Integration* e a recuperação de informações a partir da Ontologia Núcleo. Com isso, a partir da especificação deste cenário de uso, a sua validação passa a obedecer o cronograma do projeto *Knowlege TV*, juntamente como a sua integração aos demais serviços da plataforma KTV.

Capítulo

6

Trabalhos Relacionados

“O raciocínio é a escolha do melhor, por comparação. E o melhor nem sempre é igual para cada indivíduo. Por isso, as lógicas são diferentes.”

Alvaro Granha Loregian

6.1 Infraestrutura semântica para TVDI

Nesta seção são apresentados os principais trabalhos encontrados na literatura acerca de infraestrutura semântica para TVDI nos últimos anos. A análise feita nestes trabalhos busca elencar as principais similaridades e diferenças entre eles e a abordagem proposta na CoreKTV.

6.1.1 Araujo e Ricarte (2010)

Araujo e Ricarte (2010) propõem uma abordagem com integração de ontologias existentes em um ambiente de transmissão e recepção de TV Digital aberta em redes de transmissão terrestre e via satélite. O objetivo do trabalho é integrar as informações de metadados multimídia com ontologias voltadas para a

descrição de conhecimento de domínios específicos existentes na Internet. Tal abordagem é a que apresenta mais similaridades com a proposta na CoreKTV, por considerar as fontes de dados transmitidas tradicionalmente via *broadcast*, além do formato de TV-Anytime, adotando um mapeamento para construção de uma base de conhecimento. Outro ponto em comum é a possibilidade de obtenção de dados na Web, representado no trabalho pelo repositório de informações sobre filmes IMDB¹³, enquanto a CoreKTV propõem a recepção de arquivos descritivos em XML, oriundos de outros serviços, captados via *DataCatcher*.

Entretanto, as abordagens se diferenciam na modelo ontológico, uma vez que a CoreKTV provê uma representação única para o conteúdo transmitido na TV, não havendo distinção se a fonte foi o TVA ou as tabelas de informação de serviço do MPEG-2, enquanto que o modelo proposto por Araujo e Ricarte apresenta hierarquias internas, embora na mesma ontologia, para diferenciar os formatos de metadados. Outro ponto a ser destacado é a arquitetura de integração ao *middleware*, considerado por Araujo e Ricarte o local onde é realizado todo o mapeamento e processamento dos dados, enquanto a CoreKTV prevê essa tarefa no módulo servidor. Em relação a forma como é feita a coleta de informações e tradução para a ontologia, os autores não apresentam informações detalhadas.

6.1.2 Aroyo *et. al* (2007)

Aroyo *et. al.* (2007) apresenta o SenSee, um *framework* semântico, cujo objetivo é fornecer acesso personalizado a conteúdo de TV em ambientes de *cross-media*, nos quais os telespectadores possam buscar os programas que melhor atendam seus interesses. Assim como a CoreKTV, o SenSee se baseia fortemente no uso de ontologias para representação do conteúdo de TV, considerando diferentes formatos de metadados como fontes de informação.

As diferenças entre a CoreKTV e o SenSee residem quanto aos aspectos estruturais, objetivos e uso de metadados rígidos. Enquanto a proposta de Aroyo *et. al.* considera a Internet como mecanismo de busca e captura dos metadados que descrevem a programação, independente da forma como ocorre a transmissão audiovisual ocorre, a CoreKTV mantém dois meios de obter tais recursos: (i) através da integração com o *middleware*, via componente *Semantic Integration* e; (ii)

¹³ <http://www.imdb.com/>

também através da web, em comunicação com o módulo servidor do KTV, via *DataCatcher* ou via serviço de consulta semântica (LINO *et. al.*; 2011), que usa a própria estrutura do *KnowledgeTV*. Em decorrência desta abordagem de usar apenas recursos provenientes da Web, o SenSee desconsidera as tabelas de informação de serviço baseadas no padrão MPEG-2, também chamadas de metadados rígidos e consideradas a forma tradicional, ou original, de transmissão de metadados que descrevem o conteúdo da TV em um formato sintático. Tais informações sintáticas são consideradas para a construção e representação do conteúdo na CoreKTV. Por fim, enquanto a infraestrutura semântica proposta neste trabalho tem o objetivo de fornecer uma base de conhecimento para o desenvolvimento de serviços básicos a serem usados em diversos domínios de aplicações, enquanto Aroyo *et. al.* destaca a busca por acesso personalizado à TV como o objetivo do SenSee.

6.1.3 Butkus e Petersen (2007)

A abordagem proposta por Butkus e Petersen (2007) tem por objetivo modelar o conteúdo de TV a partir dos esquemas de descrição do TV-Anytime para uso em filtragem de conteúdo de EPGs ou ESGs de acordo com preferências de usuários. É proposta a identificação parcial de similaridades entre os programas a partir de atributos de gênero dos diferentes domínios dos programas. O estudo se concentra apenas na fonte de dados da BBC, que provê dados encapsulados em TV-Anytime.

Apesar do aspecto comum da modelagem a partir dos esquemas de descrição do TVA, Butkut e Petersen não propõem um modelo no qual os recursos possam ser instanciados, como ontologias ou bases de conhecimento, concentrando o trabalho apenas no estabelecimento de similaridades semânticas entre itens da grade de programação da BBC.

6.2 Cenário de Recomendação de Conteúdo

Nesta seção são elencados alguns dos principais trabalhos que tratam da recomendação de conteúdo em TVDI, o cenário de uso proposto nesta dissertação para uso da CoreKTV. Há de se destacar o primeiro trabalho da subseção 6.2.1, por além do cenário de recomendação também propõem uma infraestrutura baseada em

conhecimento para TVDI e, dessa forma também foi analisado sob os aspectos da seção anterior.

6.2.1 Blanco-Fernandez *et. al.* (2006)

Blanco-Fernandez (2006) apresenta a ferramenta AVATAR – *AdVanced Telematic search of Audiovisual contents by semantic Reasoning*, desenvolvida para o *middleware* MHP, projetada para um ambiente no qual o STB esta permanentemente conectado com a Internet. Assim como o cenário levantado neste trabalho, o AVATAR também se propõe a utilização de técnicas baseadas em conhecimento e para recomendação de conteúdo em TV Digital.

Dentre as interseções estão a utilização de ontologias para representação de conhecimento no domínio da TV, coleta de informações sobre o usuário e a utilização de similaridades semânticas no processo de filtragem baseada em conteúdo.

Entretanto, as diferenças residem inicialmente no projeto, uma vez que a AVATAR é uma ferramenta imperativa desenvolvida para ser executada na camada de aplicações para o *middleware* MHP, diferentemente do cenário proposto, que utiliza unidades de processamento projetadas para estarem integradas ao núcleo comum do Ginga. A abordagem deste trabalho prevê o processamento das recomendações no ambiente servidor para em seguida envio ao terminal, ou seja, a alta carga de processamento é externa ao ambiente do receptor. Além disso, outro fator diferencial nesta abordagem é a monitoração no usuário para levantamento do seu perfil de uso da TV sem a necessidade de identificação explícita e coleta de informações pessoais, como ocorre na proposta de Blanco-Fernandez.

6.2.2 Ávila (2010)

O trabalho *RecommenderTV*, proposto por Ávila (2010) apresenta algumas interseções com a abordagem apresentada neste trabalho, como a adição de um componente ao núcleo comum do *middleware* Ginga, uma arquitetura baseada em agentes responsáveis pela execução das operações de coleta de informações do usuário e do conteúdo, além da ausência de informações explícitas do usuário.

No entanto, as principais diferenças para o cenário levantado neste trabalho residem no processo de recomendação. Enquanto o *RecommenderTV* realiza todo o processo no terminal do usuário e utiliza apenas técnicas de mineração de dados, aplicadas às informações presentes nas tabelas SDT e EIT, obtidas a partir do provedor de informações de serviço do *middleware*, este trabalho adota a estratégia baseada em conhecimento, fazendo uso das ligações semânticas presentes na Ontologia Núcleo do KTV, enviando as informações sobre o conteúdo e usuários para terem as recomendações processadas no ambiente servidor e reenviadas ao terminal. Outra diferença encontrada trata-se da não utilização de padrões que promovem a interoperabilidade entre sistemas e o compartilhamento de informações, diferentemente da CoreKTV que além das tabelas de informação de serviços do MPEG-2 PSI/SI, também utiliza o padrão TV-Anytime como fonte de dados para recomendação de conteúdo, além dos conceitos e técnicas da Web Semântica.

6.2.3 Lucas e Zorzo (2009)

Em Lucas e Zorzo (2009) é apresentado um sistema de Recomendação Personalizada para TV Digital (RePTVD). Este sistema foi projetado e desenvolvido na camada de aplicações do *middleware*, através da linguagem Java, utilizando principalmente a API JAVATV.

O RepTVD coleta as informações da interação do usuário de forma implícita, armazenando em um formato específico do Weka, um pacote que contém algoritmos para mineração de dados. Para o processo de recomendação, o RepTVD coleta informações das tabelas SDT e EIT, e aplica regras de associação entre tais informações e os dados registrados no histórico de comportamento do usuário

O cenário de recomendação proposto nesta dissertação diferencia-se do RepTVD principalmente na sua abordagem para integração junto ao *middleware* e no modelo de recomendação proposto. Enquanto o RepTVD utiliza algoritmos de mineração de dados, este trabalho propõe a incorporação de técnicas semânticas, através do uso da Ontologia Núcleo do KTV, uma abordagem para estender *middleware* Ginga com a adição de um novo elemento de núcleo comum para coleta de informações do usuário e sobre o conteúdo, e remover os custos de processamento das recomendações no terminal do usuário, tornando esta tarefa uma função do módulo servidor do *Knowledge TV*.

6.2.4 Zhang *et. al* (2005)

O sistema *Personalized TV Guide System*, proposto por Zhang *et. al.* (2005) apresenta uma solução para extensão do modelo MHP, do padrão europeu de TV Digital. Segundo os autores, foram realizadas alterações no *middleware*, no intuito de permitir que o a solução fosse embarcada comercialmente.

Este sistema é baseado em camadas, uma inserida no núcleo comum e outra na camada de aplicações Java, cuja função é suportar o desenvolvimento de aplicações procedurais em conformidade com o MHP. A primeira, por sua vez, tem a função de estender as funcionalidades básicas do *middleware*, sendo dividido em três módulos: (i) *SI Engine*, cuja responsabilidade é obter as informações de serviços, através das tabelas de informação de serviços; (ii) *Profile Acquisition*, para coletar as informações da interação do usuário e; (iii) *Recommending Engine*, que adota a técnica de filtragem baseada em conteúdo, utilizando algoritmos de mineração de dados para realizar esta etapa.

Existem interseções entre a proposta de Zhang *et. al.* e o trabalho desenvolvido nesta dissertação, tratando-se como mais relevante a proposta de extensão de *middleware* através da adição de um novo elemento ao núcleo comum, além de outras similaridades como a utilização de métodos implícitos para coleta de informações do usuário e do conteúdo.

No entanto, existe uma diferença considerável quando relacionadas às abordagens utilizadas para o processo de recomendação. Enquanto a abordagem de Zhang *et. al.* adota apenas a filtragem baseada em conteúdo, através de algoritmos de mineração de dados, o cenário de recomendação levantado neste trabalho propõe-se a adotar técnicas baseadas na representação de conhecimento, utilizando os recursos disponíveis na Ontologia Núcleo. Com isto, este trabalho avança no estado da arte ao propor a utilização padrão TV-Anytime em harmonia com outros padrões como formato para descrição do conteúdo da TV, juntamente com a utilização das atuais Tabelas PSI/SI do MPEG-2 em uma única base de conhecimento em conformidade com as premissas da Web Semânticas.

6.3 Conclusão

Neste capítulo foram apresentados os principais trabalhos encontrados na literatura sobre o domínio do problema e solução apresentados na CoreKTV, traçando um paralelo entre os pontos comuns e divergentes das abordagens propostas. Foram também apresentados trabalhos para comparação do cenário de uso proposto, o de recomendação de conteúdo. No quadro resumo da tabela 6 é apresentado um sumário das comparações realizadas.

Tabela 6 - Quadro comparativo de trabalhos relacionados

Requisitos	Infraestrutura semântica para TVDI			Cenário de recomendação de conteúdo				CoreKTV
	Araujo e Ricarte (2010)	Aroyo <i>et. al.</i> (2007)	Butkus e Petersen (2007).	Blanco-Fernandez <i>et. al.</i> (2006)	Ávila (2010)	Lucas e Zorzo (2009)	Zhang <i>et. al.</i> (2005)	
Integração com <i>middleware</i>	X				X	X	X	x
Integração com a Internet	X	X	X	x				x
Metadados Rígidos	X			x	x	x	x	x
Metadados Flexíveis	X	X	X	x				x
Suporte a Aplicação		X		x			x	
Suporte à Serviços Básicos					x	x		x
Informações Semânticas	X	X	X	x				x
Identificação Explícita (Registro/Login)				x		x		
Identificação Implícita					x		x	x
Histórico do Comportamento					x	x	x	x
Informações Socio-Demográficas				x				x
Preferências e Interesses				x		x		x
Filtragem baseada em Conteúdo		X		x		x		x
Filtragem Semântico		X		x				x
Técnicas de Data Mining					x	x	x	

Capítulo

7

Conclusões

“O êxito da vida não se mede pelo caminho que você conquistou, mas sim pelas dificuldades que superou no caminho.”

Abraham Lincoln

7.1 Contribuições

Há de se destacar as seguintes contribuições, obtidas a partir do desenvolvimento deste trabalho:

- A análise do domínio, o levantamento dos requisitos e definição de um modelo minimalista capaz de representar as ações do usuário, permitindo construir seu comportamento como consumidor do conteúdo televisivo transmitido de forma totalmente implícita, sem que haja qualquer interferência explícita para coleta de informações;
- O levantamento dos principais conjuntos de metadados (rígidos e flexíveis) relacionados a informações consideradas de importância para usuário, de forma que a granularidade dos dados permita identificar, junto ao modelo do usuário, possíveis interesses ocultos na grade de programação dos diversos canais, serviços e programas transmitidos;

- O levantamento dos requisitos e definição de uma API de componente de núcleo para *middleware* de TVDI, o *Semantic Integration*, compatível às normas J200, J201 e J202, permitindo que seu desenvolvimento possa ser realizado em diferentes padrões de TVDI, constituindo, desta forma um avanço no estado da arte nesta área;
- A definição de um modelo de documento para armazenamento e transmissão das informações obtidas acerca do conteúdo de TV. Há de se destacar a proposta de descrição desse modelo, uma vez que ela é baseada em na linguagem de marcação XML, o que o torna integrável e de fácil comunicação e processamento entre diferentes sistemas computacionais;
- A definição de um modelo semântico, a ontologia de núcleo para o Knowledge TV, consistindo de um rico conjunto de hierarquia de classes, passíveis de uso e extensão em diversos domínios, em especial a área de multimídia e TVDI. A principal contribuição deste modelo é a capacidade de ser extensível e integrável com outros recursos da Web, uma vez que a ontologia foi construída a partir de padrões estabelecidos, além de apresentar uma das principais premissas da Web Semântica na forma de reutilização dos recursos já existentes a partir de outra ontologia, como a ontologia *OWL Time*.
- A especificação de componente genérico para captura e recepção de dados para a plataforma *Knowledge TV*, capaz de ser integrados a diferentes plataformas, como sistema de TVDI e sistemas Web, por exemplo;
- A definição de um mecanismo de tradução (*parser*) para construção de ontologias a partir de documentos baseados em XML.

7.2 Trabalhos Futuros

Como trabalho futuro mais direto está a integração da CoreKTV aos demais serviços que estão sendo desenvolvidos na plataforma do Knowledge TV, como o MKTV, serviço de mineração de dados baseado em conhecimento e o serviço de consultas semânticas aplicado ao domínio multimídia. Além destes, outro trabalho futuro a ser desenvolvido em breve será a integração junto a rede de parceiros do Knowledge TV, como a plataforma JCollab, com objetivo de desenvolver telejornalismo colaborativo em ambientes de convergência digital.

Como trabalhos a longo prazo, já existem algumas discussões no grupo acerca de expandir a pesquisa no domínio em que este trabalho foi desenvolvido e

no qual foi proposto o cenário de uso. Há de ser considerado nos próximos trabalhos o alto grau de complexidade envolvendo o desenvolvimento de sistemas de recomendação semânticos para TVDI, o que demandou muitos esforços na pesquisa e desenvolvimento. Com isso, foi possível identificar diferentes linhas de investigação com potencial a ser explorado em trabalhos futuros, dentre as quais:

- Identificação e gerenciamento de perfil do usuário;
- Integração com outras plataformas e dispositivos, como *smartphones, palms, ipads*, etc.
- Integração com redes sociais, permitindo maior captação de dados, além de possibilitar o desenvolvimento com uma base maior e distribuída de usuários;
- Inclusão de outros formatos de metadados e repositórios da Web como fontes de recursos;
- Levantar requisitos de privacidade e segurança para os dados dos usuários;
- Adição de filtros colaborativos para trabalhar as potenciais semelhanças entre comportamentos de usuários;
- Avaliação e usabilidade das recomendações;
- Investigação de técnicas de recomendação de conteúdo aplicadas ao domínio comercial, potencializando a publicidade personalizada
- Investigação e desenvolvimento de técnicas para prevenção de situações de emergências ou de risco através do ambiente convergente.

Enfim, com a convergência entre estas diferentes áreas se abrem um enorme leque de possibilidades para pesquisa e desenvolvimento de soluções nos mais diferentes graus de complexidade e utilidade.

7.3 Considerações Finais

Neste trabalho foi apresentada uma abordagem de infraestrutura baseada em conhecimento para modelagem de conteúdo multimídia em TV Digital Interativa, denominada CoreKTV. Tal abordagem, inserida no contexto do Knowledge TV, provê a integração das plataformas de TVDI e Web sob os conceitos e técnicas da Web Semântica. Dentre os benefícios trazidos a partir da abordagem e da construção da infraestrutura apresentados estão:

- Integração entre duas plataformas distintas, a emergente TVDI e a WEB, que apresentam características e níveis de complexidades diferentes;

- Harmonização com padrões internacionalmente estabelecidos pela indústria e academia, a partir do uso de metadados flexíveis para descrição de conteúdo multimídia, como o TV-Anytime e linguagens da Web Semântica para construção da base de conhecimento, como o RDF, RDFS e a OWL e a XML para documentos de comunicação entre os módulos da infraestrutura proposta;
- Interoperabilidade entre sistemas e agentes computacionais, permitida a partir do uso dos padrões supracitados;
- Compartilhamento de informações, através da base de conhecimento, permitindo o uso, reuso, extensão e integração com outras bases de conhecimento;
- Ampla possibilidade de automatização de operações por meio de raciocínio automático;
- Auxílio na extração, classificação e organização da informação na plataforma da TVDI, estruturando seu conteúdo.

Dentre as dificuldades encontradas durante a pesquisa e desenvolvimento do trabalho destacam-se dois aspectos:

- A junção dos principais domínios sobre os quais a pesquisa ocorreu resultou em um universo bastante vasto para investigação, com diversas possibilidades de temas e soluções passíveis de serem desenvolvidas. Isto tornou difícil definir claramente os limites do escopo do trabalho, uma vez que diversas tais possibilidades estavam naturalmente ligadas em outras. Por fim, optou-se por restringir-se a integração dos conceitos da Web Semântica à plataforma de TVDI.
- A total integração ao projeto *Knowledge TV*, no qual ocorreram problemas de ordem burocrática e de definição de cronograma, o que resultou na necessidade de mudanças e restrições no escopo deste trabalho, para que fosse possível estar de acordo com o cronograma do referido projeto. Entretanto, os objetivos contidos nos trabalhos estão de acordo e obedeceram ao cronograma, encontrando-se, atualmente em processo de integração com os demais serviços da plataforma KTV, como o serviço de consultas semânticas e o de mineração de dados.

Há de se destacar que durante o desenvolvimento deste trabalho, inserido dentro do projeto *Knowledge TV*, além dos resultados práticos, também foram

alcançados resultados na forma de publicação de trabalhos em eventos científicos de âmbito nacional e internacional. Tais resultados foram:

- Araújo, J. P. C.; Lino, N. C. Q.; ***User Profiling in Digital TV: A Semantic Modelling Approach. In: Summer School for Multimedia Semantics 2009***, Poster Session, Koblenz, Germany, 2009.
- Lino, N.; Araújo, J. P. C.; Lemos, G. and Siebra, C.; Aspectos Semânticos e Convergência Digital (Web e TV Digital). Proceedings of 2a. Conferência Web W3C Brasil (W3C Web.br 2010), Belo Horizonte, Brasil, 2010.
- Lino, N. C. Q.; Araújo, J. P. C. ; Anabuki, D.; Patrício Junior, J. C.; Batista, M.; Nóbrega, R.; Amaro, M.; Siebra, C. A. and Souza Filho, G. L. de.; ***Knowledge TV.Proceedings of the 9th EuroITV Conference*** (EuroITV 2011), Lisbon, Portugal, 2011.

Capítulo

8

Referências Bibliográficas

“Pedras no caminho?

Guardo todas, um dia vou construir um castelo...”

Fernando Pessoa

ABNT NBR 15603-1:2007. Televisão Digital Terrestre - Multiplexação e Serviços de Informação (SI) - Parte 1: Serviços de informação do sistema de radiodifusão. 2007.

ABNT NBR 15606-1. 2008. Televisão digital terrestre — Codificação de dados e especificações de transmissão para radiodifusão digital – Parte 1: Codificação de dados. 2008

Adomavicius, G. e Tuzhilin, A. Toward the Next Generation of Recommender Systems: A survey of the State-of-the-Art and Possible Extensions. Knowledge and Data Engineering, IEEE Transactions, Volume 17, Issue 6, page(s): 734– 749, June 2005.

Alves, L. G. P.; Kulesza, R.; Silva, F. S.; Juca, P. e Bressan, G. Análise Comparativa de Metadados em TV Digital. In: XXIV SBRC / WTVD 2006 - II Workshop de TV Digital, 2006, Curitiba. Anais do XXIV SBRC, 2006.

Antoniou, G.; Harmelen, F.; A Semantic Web Primer. 2nd edition. MIT Press, Cambridge Massachussetes, 2008.

Araújo, J. P. C.; Lino, N. C. Q.; *User Profiling in Digital TV: A Semantic Modelling Approach.* In: *Summer School for Multimedia Semantics 2009*, Poster Session, Koblenz, Germany, 2009.

Araujo, R. C. ; Ricarte, I. L. M. . Ontologias para integração de metadados da indústria de broadcast multimídia no contexto da TV digital. Revista de radiodifusão, v. 4, p. 48, 2010.

Aroyo, L.; Bellekens, P.; Björkman, M.; Houben, G.; Akkermans, P.; Kaptein, A.; SenSee Framework for Personalized Access to TV Content. European Conference on Interactive TV (EuroITV 2007), 156-165, 2007.

Ávila, P. M. *RecommenderTV:* Suporte ao Desenvolvimento de Aplicações de Recomendação para o Sistema Brasileiro de TV Digital. Dissertação de Mestrado. Universidade Federal de São Carlos, Brasil, 2010.

Berners-Lee, T.; Lassila, O.; Hendler, J.; The Semantic Web. *Scientific American*, 284 (5), 2001, pp.34-43.

Blanco-Fernández, Y., Pazos-Arias, J. J., Gil-Solla, A. Ramos-Cabrer, M., López-Nores, M.; *AVATAR: An Improved Solution for Personalized TV based on Semantic Inference;* in IEEE Transactions on Consumer Electronics, vol. 52, No. 1, p. 223-231, Fev/2006.

Breitman, K. K. . Web Semântica: O Futuro da Internet. 1. ed. Rio de Janeiro: LTC - Livros Tecnicos e Cientificos Editora S.A., 2005. v. 1. 190 p.

Butkus, A.; Petersen, M.; Semantic Modelling Using TV-Anytime Genre Metadata. In: Proceedings of EuroITV'2007: Interactive TV: a Shared Experience. Springer Berlin / Heidelberg. pp.226-234.

Evain, J. P.; Murret-Labarthe, H.; TV-Anytime Phase 1 – a decisive milestone in open standards for Personal Video Recorders. EBU Technical Review; Julho, 2003.

Fernandes, J. Souza Filho, G. L.; Elias, G. Introdução à Televisão Digital Interativa: Arquitetura, Protocolos, Padrões e Práticas. JAI – Jornadas de Atualização da Informática. XXIV Congresso da SBC, 2004.

Freire Filho, S. L. de M. FlexCM: Um Modelo de Componentes para Sistemas Adaptativos. Departamento de Informática, Universidade Federal da Paraíba. 2008. p. 74, Dissertação (Mestrado).

Gauch, S., Speretta, M., Chandramouli, A. e Micarelli A. User Profiles for Personalized Information Access. The Adaptive Web: Methods and Strategies for Web Personalization, Springer Berlin/Heidelberg, p. 54-89, 2007.

Goldberg, D., Nichols, D., Oki, B. M. and Terry, D. Using Collaborative Filtering to Weave an Information Tapestry. Communications of the ACM 35(12), pp. 61-70, 1992.

Gómez-Pérez, A.; Fernandez-Pérez, M.; Corcho, O.; Ontological Engineering. Springer Verlag, 2004.

Gruber, T. R.; A translation approach to portable ontology specifications. Knowledge Acquisition – 5: 1999-220.

Guarino, N.; Formal Ontology and Information Systems. In: Proceedings of FOIS'98. Formal Ontology in Information Systems, Trento. 1998.

Han, J. and Kamber, M. Data Mining Concepts and Techniques, 2ª Edição, Editora Elsevier, Reino Unido.

Handler, J. Agents and the Semantic Web. In IEEE Intelligent Systems Journal, Volume 16, Issue 2, pp: 30-37 2001.

IBGE - Instituto Brasileiro de Geografia e Estatística. Síntese de Indicadores Sociais. Brasil, 2008.

ITU. 2001. ITU-T Recommendation J.200: Worldwide common core – Application environment for digital interactive television services. 2001.

— **2003.** ITU-T Recommendation J.202: Harmonization of procedural content formats for interactive TV applications. 2003.

— **2004.** ITU-T Recommendation J.201: Harmonization of declarative content format for interactive television applications. 2004.

Jasper, R.; Ushold, M.; A framework for understanding and classifying ontology applications. Proceedings of the 12th International Workshop on Knowledge Acquisition Modeling and Management (KAW'99). Banf, Canada, 1999.

Jones, G. A., Defilippis, J. M., Hoffman, H., e Williams, E. A. Digital Television Station and Network Implementation. Proceedings of the IEEE, 94 (1), pp. 22-36. 2006.

Laboratório de Aplicações de Video Digital (LAVID). Projeto Ginga Code Develop Network (GingaCDN). Disponível em <http://gingacdn.lavid.ufpb.br/> 2010. Acessado em setembro de 2011.

Leite, L. E. C., et al. 2005. FlexTV – Uma Proposta de Arquitetura de Middleware para o Sistema Brasileiro de TV Digital. Revista de Engenharia de Computação e Sistemas Digitais. 2005, Vol. 2, pp. 29-50.

Lino, N. C. Q.; Araújo, J. P. C.; Lemos, G. and Siebra, C.; Aspectos Semânticos e Convergência Digital (Web e TV Digital). *Proceedings of 2a. Conferência Web W3C Brasil (W3C Web.br 2010)*, Belo Horizonte, Brasil, 2010.

Lino, N. C. Q.; Araújo, J. P. C. ; Anabuki, D.; Patrício Junior, J. C.; Batista, M.; Nóbrega, R.; Amaro, M.; Siebra, C. A. and Souza Filho, G. L. de.; *Knowledge TV. Proceedings of the 9th EuroITV Conference (EuroITV 2011)*, Lisbon, Portugal, 2011.

Lucas, A. S.; Zorzo, S. D. Personalização para Televisão Digital utilizando a estratégia de Sistema de Recomendação para ambientes multiusuário. In: XXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC), Recife, Brasil, 2009.

Lugmayr, A.; Niiranen, S. and Kalli, S. Digital Interactive TV and Metadata: Future Broadcast Multimedia. Springer-Verlag, New York, New York, USA, 2004.

Martinez, J. M., Koenen, R., Pereira, F. *MPEG-7: the generic Multimedia Content Description Standard*. IEEE Multimedia, Volume 9, Issue 2, pp: 78-82, 2002.

MPEG-2 - ISO/IEC 13818-1.2008. Information technology - Generic coding of moving pictures and associated audio information - Part 1: Systems. 1992

Noy, N.; McGuinness, D.; Ontology Development 101. A guide for creating your first ontology. KSL Technical Report, Stanford University, 2001.

Peis, E., Morales-del-Castillo, J. M., Delgado-López, J. A. Semantic Recommender Systems. Analysis of the state of the topic (on line). "Hipertext.net", num. 6, 2008. Disponível em <http://www.hipertext.net>.

Reateguil, E. ; Cazella, S. C. . Minicurso: Sistemas de Recomendação. In: ENIA – Encontro Nacional de Inteligência Artificial, 2005, São Leopoldo, Sociedade Brasileira de Computação. 2005.

Resnick, P. and Varian, H. R. 1997. Recommender Systems. Communications of the ACM. ACM 40, 3 (Mar. 1997), 56-58.

Schafer, J. B. MetaLens: A framework for Multi-Source Recommendations. PhD Thesis. University of Minesota, Mineapolis, USA, 2001.

Smith, J. R.; Schirling, P. Metadata Standards Roundup. In: IEEE MultiMedia, Volume 13 , Issue 2 , pp: 84 - 88 . 2006

Soares, L. F. G. 2006. MAESTRO: The Declarative Middleware Proposal for the SBTVD. Proceedings of the 4th European Interactive TV Conference. 2006.

Soares, L. F. G., Rodrigues, R. F. e Moreno, M. F. 2007. Ginga-NCL: the Declarative Environment of the Brazilian Digital TV System. Journal of the Brazilian Computer Society. 2007, Vol. v12, pp. 37-46.

Souza Filho, G. L. de, Leite, L. E. C. e Batista, C. E. C. F. 2007. Ginga-J: The Procedural Middleware for the Brazilian Digital TV System. Journal of the Brazilian Computer Society. 2007, Vol. v12, pp. 47-56.

Souza Filho, G. L. de, Leite, L. E. C., Batista, C. E. C. F. Ginga-J: The Procedural Middleware for the Brazilian Digital TV System. In: Journal of the Brazilian Computer Society, no 4, Vol 12, (ISSN 0104-6500) pp. 47-56. Março, 2007. Porto Alegre, RS, Brasil.

TVA – TV-Anytime Forum; Disponível em <http://www.tv-anytime.org>, Acessado em Setembro de 2011.

W3C – World Wide Web Consortium. Disponível em <http://www.w3.org/TR/owl-time/>>. Acessado em Setembro de 2011.

Zhang, H.; Zheng, S.; Yuan, J. A personalized TV guide system compliant with MHP. In: Consumer Electronics, IEEE Transactions, Volume 51, Issue 2, pp: 731-737, 2005.

Apêndice

A

Tabelas de Dados Operacionais

A.1 Tabela de metadados MPEG-2 PSI/SI

#	Metadado	Origem	Descrição
1	content_nibble_level_1	EIT	Informação de gênero
2	content_nibble_level_2	EIT	Informação de subgênero
3	country_code	EIT/PMT	Informação de país
4	content_rating	EIT/PMT	Classificação temática do conteúdo
5	age_rating	EIT/PMT	Classificação etária do conteúdo
6	event_name	EIT	Nome do conteúdo (programa)
7	short_description	EIT	Breve descrição do conteúdo
8	sh_language_code	EIT	Indica o idioma da descrição do conteúdo
9	event_id	EIT	Identificador único do evento (programa)
10	start_time	EIT	Horário de início do evento (programa)

11	Duration	EIT	Duração do evento (programa)
12	stream_content	EIT/PMT	Especifica o tipo do fluxo (áudio, vídeo ou dados)
13	component_type	EIT/PMT	Especifica o tipo do componente de áudio, vídeo ou dados
14	component_description	EIT/PMT	Descrição em texto do fluxo do componente
15	cd_language_code	EIT/PMT	Indica o idioma da descrição do componente
16	audio_component_type	EIT	Especifica o tipo do componente de áudio
17	audio_stream_type	EIT	Especifica o tipo do fluxo de áudio
18	audio_multilanguage	EIT	Indica se há mais dois idiomas
19	audio_quality_indicator	EIT	Indica modo de qualidade do áudio
20	audio_sample_rate	EIT	Indica a frequência de amostragem
21	audio_language_1	EIT	Identifica o primeiro idioma
22	audio_language_2	EIT	Identifica o segundo idioma
23	audio_description	EIT	Descrição do componente de áudio
24	video_encode_format	EIT	Indica o formato de codificação de vídeo
25	series_id	EIT	Identificador da série
26	series_repeat_label	EIT	Fornece rótulo de identificação do programa
27	series_program_pattern	EIT	Fornece padrão de transmissão do programa
28	series_ep_number	EIT	Número do episódio da série
29	series_last_ep_number	EIT	Número do último

			episódio da série
30	series_expire_date	EIT	Data limite do seriado
31	series_name	EIT	Nome da série
32	service_type	SDT	Especifica o tipo do serviço
33	service_provider_name	SDT	Nome do fornecedor do serviço
34	service_name	SDT	Nome do serviço
35	service_id	SDT/EIT	Identificador do serviço
36	service_countries_avability_1	SDT	Lista os países onde o serviço está disponível
37	service_countries_avability_2	SDT	Lista os países onde o serviço não está disponível
38	service_list	NIT/BIT	Lista de serviços transmitidos pela rede/radiodifusor
39	network_name	NIT	Nome da rede
40	network_id	NIT	Identificador da rede
41	state_area_code	NIT/PMT	Estado alvo para transmissão de informação de emergência
42	microregion_area_code	NIT/PMT	Microregião alvo para transmissão de informação de emergência
43	signal_level	NIT/PMT	Corresponde ao sinal de alarme de emergência especificado pelos órgãos responsáveis
44	avc_video_profile_idc	PMT	Exibe o perfil do fluxo de vídeo AVC
45	avc_video_level_idc	PMT	Mostra o nível do fluxo de vídeo AVC
46	avc_video_still_present	PMT	Indica se vídeo contém imagens estáticas
47	avc_video_24h_picture	PMT	Indica se o vídeo contém imagens 24 horas

48	aac_audio_type	PMT	Indica o tipo do áudio transmitido
49	program_number	PMT	Identificador de um programa na tabela
50	broadcaster_name	BIT	Nome do radiodifusor
51	broadcaster_id	BIT	Identificador do radiodifusor
52	broadcast_view_property	BIT	Informa se a indicação do usuário para o nome do radiodifusor é apropriado ou não
53	local_time_offset	TOT	Informa a diferença de horário em relação ao UTC-3 na faixa de ± 12
54	utc-3_time	TOT	Horário no formato UTC-3

A.2 Tabela de metadados TV-Anytime

#	Metadado	Origem	Descrição
55	TimePoint	TVATimeType	Designa um ponto no tempo
56	Duration	TVATimeType	Designa um período no tempo
57	PersonName	TVAAgentType	Especifica o nome de uma pessoa
58	PersonNameIDRef	TVAAgentType	Elemento usado para indentificar um PersonName na CIT
59	OrganizationName	TVAAgentType	Especifica o nome de uma organização
60	OrganizationNameIDRef	TVAAgentType	Elemento usado para indentificar um OrganizationName na CIT
61	Keyword	KeywordType	Define uma palavra-chave para o conteúdo
62	KeywordType	KeywordType	Indica a ordem de importância da palavra-chave que descreve o conteúdo (principal/secundário/outro)
63	Genre	GenreType	Define um gênero para o

			conteúdo
64	GenreType	GenreType	Indica a ordem de importância do gênero que descreve o conteúdo (principal/secundário/outro)
65	Synopsis	SynopsisType	Define o tamanho da sinopse (curta/média/longa)
66	SynopsisType	SynopsisType	Define a sinopse do conteúdo
67	RelatedMaterial	RelatedMaterialType	Referencia outras propriedades de mídia relacionadas ao conteúdo AV descrito
68	HowRelated	RelatedMaterialType	Natureza do relacionamento entre conteúdo e propriedade AV
69	Format	RelatedMaterialType	Define tipo do arquivo da propriedade relacionada
70	MediaLocator	RelatedMaterialType	Define localização da propriedade de mídia
71	PromotionalText	RelatedMaterialType	Fornecer texto promocional sobre a ligação (usado como atrativo adicional)
72	PromotionalMedia	RelatedMaterialType	Possibilita uso de informação não-textual, como logotipos, gráficos, etc.
73	SourceMediaLocator	RelatedMaterialType	Define localização da mídia cuja descrição é associada
74	Character	CreditsItem	Especifica o nome do personagem representado pelo artista (usado em conjunto com TVAAgentType).
75	CategoryAward	AwardType	Especifica a categoria na qual o conteúdo foi premiado
76	NomineeAward	AwardType	Especifica o candidato que venceu na categoria
77	RecipientAward	AwardType	Especifica a pessoa/organização promotora do prêmio
78	TitleAward	AwardsListItemType	Nome/Título do prêmio

79	YearAward	AwardsListItemType	Ano da premiação
80	Award	AwardsListItemType	Informação detalhada do prêmio
81	ShortTitleType	ShortTitleType	Define um título de tamanho máximo de 80 caracteres
82	CreationDate	CreationCoordinatesType	Define a data de criação de um conteúdo (TVATimeType)
83	CreationLocation	CreationCoordinatesType	Define a região onde foi criado o conteúdo ('region code')
84	DepictedDate	DepictedCoordinatesType	Define a data retratada no conteúdo
85	DepictedLocation	DepictedCoordinatesType	Define a localização retratada no conteúdo
86	ReleasedDate	ReleasedDateType	Data (dia, mês e ano) de lançamento do conteúdo
87	ReleasedYear	ReleasedDateType	Ano de lançamento do conteúdo
88	ReleasedLocation	ReleasedInformationType	País onde o conteúdo foi lançado
89	Title	BasicContentDescriptionType	Define o título do conteúdo (<i>full</i>)
90	MediaTitle	BasicContentDescriptionType	Define uma propriedade de mídia como título do conteúdo (imagem, por exemplo)
91	TVAParentalGuidance	BasicContentDescriptionType	Contém a classificação de país para o programa
92	Language	BasicContentDescriptionType	Define o idioma do conteúdo
93	CaptionLanguage	BasicContentDescriptionType	Define a linguagem de informação de <i>caption</i> incluída no conteúdo
94	SignLanguage	BasicContentDescriptionType	Define a linguagem de sinais incluída no conteúdo
95	BitRate	BitRateType	Indica a taxa de bits para um conteúdo
96	AudioLanguage	AudioLanguageType	Indica o idioma do áudio

		ype	
97	AudioCoding	AudioAttributesType	Formato de codificação do áudio
98	NumChannels	AudioAttributesType	Indica o número de canais de áudio (Mono/Stéreo/Multi)
99	AudioMixType	AudioAttributesType	Tipo da mixagem do áudio
100	FrameRate	VideoAttributesType	Indica a taxa de quadros do vídeo
101	VideoCoding	VideoAttributesType	Formato de codificação do vídeo
102	VideoScan	VideoAttributesType	Indica o vídeo scan
103	VideoHorizontalSize	VideoAttributesType	Dimensão horizontal do vídeo (em pixels)
104	VideoVerticalSize	VideoAttributesType	Dimensão vertical do vídeo (em pixels)
105	VideoAspectRatio	VideoAttributesType	Indica o vídeo aspect
106	VideoColor	VideoAttributesType	Indica o formato de cores do vídeo (preto e branco, colorido, etc)
107	CaptioningCoding	CaptioningAttributesType	Indica o formato de <i>captioning</i>
108	FileFormat	AVAttributesType	Define o formato da instância do arquivo
109	FileSize	AVAttributesType	Define o tamanho do arquivo
110	Ratio	RationType	Especifica a taxa no formato “h:v”
111	BasicDescription	ProgramInformation/GroupInformation	Descrição básica do conteúdo/grupo de conteúdo
112	AVAttributes	ProgramInformation	Descrição dos atributos de áudio e vídeo do conteúdo
113	NumOfItems	GroupInformation	Informa a quantidade total de elementos do grupo de conteúdo

114	MediaReview	MediaReviewType	Descreve uma crítica sobre o conteúdo
115	ServiceInformationName	ServiceInformation	Informa o nome do serviço
116	ValidFrom	ServiceInformation	Data e horário a partir do qual o serviço é válido
117	ValidTo	ServiceInformation	Data e horário limite da validade do serviço
118	ServiceInformationOwner	ServiceInformation	Informa nome do produtor do conteúdo
119	ServiceGenre	ServiceInformation	Informa o gênero do serviço
120	ServiceLanguage	ServiceInformation	Informa o idioma falado no serviço
121	Logo	ServiceInformation	Logotipo da rede, como uma imagem ou <i>jingle</i>
122	ServiceRelatedMaterial	ServiceInformation	Referência a outro material relacionado ao serviço
123	PublishedStartTime	ScheduleEvent	Horário anunciado para início do programa
124	PublishedEndTime	ScheduleEvent	Horário anunciado para término do programa
125	PublishedDuration	ScheduleEvent	Duração anunciada para o programa
126	Live	ScheduleEvent	Indica que a transmissão é ao vivo
127	Repeat	ScheduleEvent	Indica que é uma repetição
128	FirstShowing	ScheduleEvent	Indica que é a primeira exibição
129	LastShowing	ScheduleEvent	Indica que é a última exibição
130	InstanceDescription	ScheduleEvent/BroadcastEvent	Descreve uma instância do conteúdo
131	ServiceIDRef	BroadcastEvent	Identifica o serviço ao qual um evento de broadcast é transmitido

Apêndice

B

API do componente de núcleo *Semantic Integration*

B.1 ISemanticAgent.h

O cabeçalho apresentado abaixo representa a interface principal do componente *Semantic Integration* no modelo de componentes FlexCM. Após sua implementação, o componente poderá ser instanciado, recuperado e conectado à arquitetura de execução do *middleware* OpenGinga através do ambiente de execução do FlexCM.

```
#ifndef GINGA_SEMANTICINTEGRATION_ISEMANTICAGENT_H
#define GINGA_SEMANTICINTEGRATION_ISEMANTICAGENT_H

#include <string>
#include <list>
#include <iostream>
#include <fstream>

#include "iunknow.h"

#define IID_ISemanticAgent "94c8c731-aaaf-4f9f-a1fd-405aa52b435d"

namespace semanticintegration{
```

```
/** \brief Interface ISemanticAgent é a definição do agente
 * responsável pelo tratamento das informações capturadas junto ao
 * provedor de informações de serviço do middleware e do
 * sintonizador.
 */

class ISemanticAgent: public virtual flexcm::interface::IUnknow{

public:

    virtual ~ISemanticAgent() = 0;

    /**
     *
     */
    virtual list<MMContent> getMMContents() = 0;

    /**
     *
     */
    virtual MMContent getMMContent(string name) = 0;

    /**
     *
     */
    virtual UserMonitorAgent getUserInformation() = 0;

    /**
     *
     */
    virtual fstream createMessage(MMContent content) = 0;

    /**
     *
     */
    virtual fstream createMessage(UserMonitorAgent user) = 0;

    /**
     *
     */
    virtual bool sendMessage( fstream file) = 0;

    /**
     *
     */
    virtual fstream receiveMessage() = 0;

};
}
#endif
```

B.2 MMContent.h

```

#ifndef GINGA_SEMANTICINTEGRATION_MMCONTENT_H
#define GINGA_SEMANTICINTEGRATION_MMCONTENT_H

#include <string>
#include <list>

namespace semanticintegration{

/**
 * A classe MMContent representa uma instância de um programa
 * multimídia na TV, ou um evento presente em um serviço.
 * As informações recuperadas nesta classe são de transmissão
 * obrigatória, exceto objetos instância da classe
 * ContentDescription, cujas informações são transmitidas de modo
 * opcional.
 */
class MMContent: public ProviderAgent{

public:

    /**
     *
     */
    virtual ~MMContent() = 0;

    /**
     *
     */
    virtual string getMMContentName() = 0;

    /**
     *
     */
    virtual string getMMContentSynopsis() = 0;

    /**
     *
     */
    virtual ContentDescription getDescription() = 0;

};
}
#endif

```

B.3 DescriptionContent.h

```

#ifndef GINGA_SEMANTICINTEGRATION_CONTENTDESCRIPTION_H
#define GINGA_SEMANTICINTEGRATION_CONTENTDESCRIPTION_H

#include <string>
#include <list>

```

```
namespace semanticintegration{

/**
 * A classe ContentDescription encapsula informações que descrevem
 * o conteúdo transmitidos para a TV.Ela deve esta obrigatóriamente
 * relacionada à um objeto MMContent, que representa a entidade de
 * um programa de transmitido.
 */
class ContentDescription: public ProviderAgent{

public:

    /**
     * Destrutor virtual
     */
    virtual ~ContentDescription() = 0;

    /**
     * Construtor recebendo um objeto MMContent como parâmetro.
     */
    virtual ContentDescription(MMContent content) = 0;

    /**
     *
     */
    virtual list<string> getGenres()= 0;

    /**
     *
     */
    virtual list<string> getSubgenres()= 0;

    /**
     *
     */
    virtual list<string> getKeywords() = 0;

    /**
     *
     */
    virtual string getClassificationByAge()= 0;

    /**
     *
     */
    virtual string getClassificationByTheme() = 0;

    /**
     *
     */
    virtual string getClassificationByUsedLanguage() =0;

    /**
     *
     */
    virtual string getAgeGroupAudience() = 0;
```

```
/**
 *
 */
virtual string getEducationStandardAudience() = 0;

/**
 *
 */
virtual string getFamiliarStageAudience() = 0;

/**
 *
 */
virtual string getGenderAudience() = 0;

/**
 *
 */
virtual string getGeographicalAreaAudience() = 0;

/**
 *
 */
virtual string getSocialGroupAudience() = 0;

/**
 *
 */
virtual string getSpecificGroupAudience() = 0;

/**
 *
 */
virtual list<string> getCaptionLanguage() = 0;

/**
 *
 */
virtual list<string> getSpokenLanguage() = 0;

/**
 *
 */
virtual list<string> getSubtitleCaptionLanguage() = 0;

/**
 *
 */
virtual int getEmergencyAlertLevel() = 0;

/**
 *
 */
virtual list<string> getActors() = 0;

/**
 *
 */
virtual list<string> getAuthors() = 0;
```

```
/**
 *
 */
virtual list<string> getDirectors() = 0;

/**
 *
 */
virtual list<string> getProducers() = 0;

/**
 *
 */
virtual list<string> getSingers() = 0;

/**
 *
 */
virtual list<string> getReviews() = 0;

/**
 *
 */
virtual list<string> getAudioFormats() = 0;

/**
 *
 */
virtual list<string> getVideoFormats() = 0;

/**
 *
 */
virtual list<string> getHiperlinks() = 0;

/**
 *
 */
virtual list<string> getOriginalDistribution() = 0;

/**
 *
 */
virtual string getStartTime() = 0;

/**
 *
 */
virtual string getDuration() = 0;

/**
 *
 */
virtual string getReleasedDate() = 0;

/**
 *
 */
```

```

    virtual string getReleasedDate() = 0;

    /**
     *
     */
    virtual string getExpireDate() = 0;

    /**
     *
     */
    virtual string getTransmissionType() = 0;

    /**
     *
     */
    virtual string getBroadcasterName() = 0;

    /**
     *
     */
    virtual list<string> getAudioAttributes() = 0;

    /**
     *
     */
    virtual list<string> getVideoAttributes() = 0;
};
}
#endif

```

B.4 UserMonitorAgent

```

#ifndef GINGA_SEMANTICINTEGRATION_USERMONITORAGENT_H_
#define GINGA_SEMANTICINTEGRATION_USERMONITORAGENT_H_

#include <string>
#include <stdint.h>

namespace semanticintegration{

/**
 * A classe UserMonitorAgent é responsável pela coleta de
 * informações sobre da interação do usuário através das mudanças de
 * canal, registrando a sua audiência para determinadas
 * programações e horários.
 *
 */

class UserMonitorAgent{

public:

```

```
/**
 * Destrutor virtual.
 */
virtual ~UserMonitorAgent() = 0;

/**
 * Recupera o identificador do usuário junto ao terminal. O
 * identificador é gerado automaticamente pelo sistema para
 * permitir a identificação de padrões de comportamento
 * semelhantes ao longo do tempo e assim identificar
 * um perfil de uso do usuário.
 *
 * \return o identificador o usuário
 */
virtual uint8_t getUserID() = 0;

/**
 * recupera uma ação do usuário
 */
virtual enumKeyAction getUserKeyAction() = 0;

/**
 * Recupera o nome do canal
 */
virtual std::string getChannelName() = 0;

/**
 * Método para verificação de tempo mínimo de audiência do
 * usuário em um canal
 */
virtual bool timeout() = 0;

/**
 * Define se a audiência do canal deve ou não ser registrada.
 *
 * \para timeout indica se o tempo mínimo para registro de
 * audiência
 * foi atingido.
 */
virtual bool registerAudience(bool timeout) = 0;

/**
 * Recupera o nome do programa
 */
virtual std::string getProgramName() = 0;

/**
 * Recupera o gênero do programa
 */
virtual list<string> getProgramGenre() = 0;

/**
 * Recupera o horario em que o usuario comecou a assistir o
 * programa
 */
virtual get getStartTimeAudience() = 0;
```

```
/**
 * Recupera a duração de tempo de audiência do usuário no
 * programa
 */
virtual uint64_t getDurationAudience() = 0;

private:

/**
 * Define o conjunto de ações que podem ser realizadas pelo
 * usuário
 */
enum enumKeyAction {START, END, CHANNEL_UP, CHANNEL_DOWN,
SET_CHANNEL};

};

}

#endif /* USERMONITORAGENT_H_ */
```