

UNIVERSIDADE FEDERAL DO MARANHÃO  
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

ALEX GONÇALVES COSTA

**ESTUDO PARA INTEGRAÇÃO DE  
ARQUITETURAS PARA RECOMENDAÇÃO DE  
CONTEÚDO NA TV DIGITAL MÓVEL**

São Luís

2012

ALEX GONÇALVES COSTA

**ESTUDO PARA INTEGRAÇÃO DE  
ARQUITETURAS PARA RECOMENDAÇÃO DE  
CONTEÚDO NA TV DIGITAL MÓVEL**

Monografia apresentada ao Curso de Ciência da  
Computação da Universidade Federal do Maranhão,  
como parte dos requisitos necessários para obtenção  
do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Carlos de Salles Soares Neto

São Luís

2012

Costa, Alex Gonçalves.

ESTUDO PARA INTEGRAÇÃO DE ARQUITETURAS PARA  
RECOMENDAÇÃO DE CONTEÚDO NA TV DIGITAL MÓVEL  
/ Alex Gonçalves Costa. - São Luís, 2012.

63.p

Impresso por computador (fotocópia).

Orientador: Carlos de Salles Soares Neto.

Monografia (Graduação) - Universidade Federal do Maranhão,  
Curso de Ciência da Computação, 2012.

1. TV Digital 2. Middleware GINGA 3. Sistemas Baseado em  
Localização 4. Sensibilidade ao Contexto. I.Título.

CDU 004.7:654.19

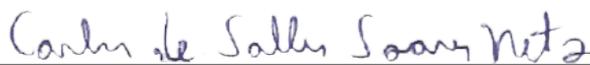
ALEX GONÇALVES COSTA

*ESTUDO PARA INTEGRAÇÃO DE ARQUITETURAS  
PARA RECOMENDAÇÃO DE CONTEÚDO NA TV  
DIGITAL MÓVEL*

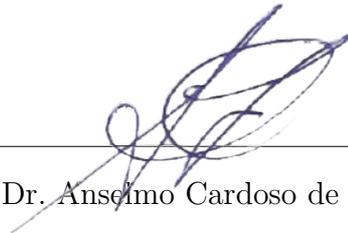
Monografia apresentada ao Curso de Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Bacharel em Ciência da Computação sob a orientação da Prof. Dr. Carlos de Salles Soares Neto.

Aprovado em 26 de junho de 2012

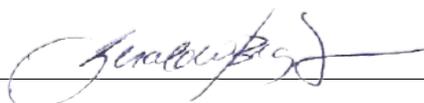
**BANCA EXAMINADORA**



Prof. Dr. Carlos de Salles Soares Neto



Prof. Dr. Anselmo Cardoso de Paiva



Prof. Ms. Geraldo Braz Júnior

*Dedico este trabalho especialmente a minha mãe,  
Maria Madalena, que jamais perdeu a fé que  
seus filhos um dia se tornassem vencedores na  
vida.*

*“Embora ninguém possa voltar atrás e  
fazer um novo começo, qualquer um pode  
começar agora e fazer um novo fim.”*

*Chico Xavier*

## AGRADECIMENTOS

À Razão da Existência, à Grande Energia do Universo, à Força que equilibra todas as forças, à Deus, à Deusa e aos Deuses. Obrigado por gerar todos os bits da minha existência.

Obrigado minha Mãe por ser maravilhosa e incansável. Aos meus familiares, pelo investimento e incentivo dado aos meus sonhos e pela educação proporcionada desde os meus primeiros passos e que transferirei às minhas filhas, Amanda e Isabele.

Ao meu irmão Alan, minha alma gêmea legítima, o qual compartilha comigo toda a carga que já carregamos para alcançarmos nossos sonhos. À minha companheira Liana que tão incentivadora, me animou quando precisei ser animado. Aos meus amigos pelo companheirismo e amizade e longos e tantos anos de convivência e aprendizado.

Ao meu orientador Carlos Salles pela paciência e por acreditar em mim incentivando-me para conclusão deste trabalho mesmo com todos os obstáculos.

A todos que colaboraram direta ou indiretamente para a concretização deste sonho. À vocês, ofereço estas palavras.

## **RESUMO**

O presente trabalho apresenta um estudo para integração de ferramentas com o objetivo de fornecer conteúdo contextualizado no cenário da TV Digital. Este trabalho propõe uma arquitetura de integração entre um framework de localização geográfica, um framework de sensibilidade ao contexto para TV Digital e a Máquina de Apresentação Ginga-NCL. Estudando cada arquitetura foi possível aproveitar módulos específicos para compor uma arquitetura combinando tais módulos.

Palavras-chave: TV Digital. Localização geográfica. Sensibilidade ao contexto. Máquina de apresentação Ginga-NCL.

## **ABSTRACT**

This paper presents a study for integration of tools in order to provide content in context in the setting of Digital TV. This paper proposes an architecture for integrating a geographic framework, a framework of context awareness for Digital TV and Display Machine Ginga-NCL. Studying each architecture could take advantage of specific modules to compose an architecture combining these modules.

Keywords: Digital TV. Geographical location. Sensitivity to context. Presentation engine Ginga-NCL.

## Lista de Figuras

2.1	Arquitetura em blocos da TV Digital. . . . .	13
2.2	Conjunto de Padrões. . . . .	14
2.3	Padrão ATSC. . . . .	15
2.4	Padrão DVB. . . . .	15
2.5	Padrão ISDB. . . . .	16
2.6	Padrão Ginga. . . . .	17
2.7	Módulo de Referência do SBTVD. . . . .	18
2.8	Arquitetura Referência do Middleware Ginga. . . . .	20
2.9	LBS como convergência de várias tecnologias. . . . .	22
2.10	Arquitetura geral de um sistema LBS. . . . .	24
2.11	Exemplo de dispositivo <i>Active Badges</i> . . . . .	27
3.1	Arquitetura geral do FRAGIL. . . . .	32
3.2	Arquitetura do PersonalTVware. . . . .	38
3.3	Arquitetura Ginga-NCL para Dispositivos Móveis. . . . .	41
4.1	Arquitetura Proposta. . . . .	47
4.2	Diagrama de Sequência da Arquitetura. . . . .	48

# Sumário

<b>Lista de Figuras</b>	<b>6</b>
<b>1 INTRODUÇÃO</b>	<b>9</b>
1.1 Objetivos . . . . .	10
1.1.1 Objetivos Gerais . . . . .	10
1.1.2 Objetivos Específicos . . . . .	11
1.2 Organização do trabalho . . . . .	11
<b>2 FUNDAMENTAÇÃO TEÓRICA</b>	<b>12</b>
2.1 TV Digital . . . . .	12
2.1.1 Arquitetura do Sistema de TV Digital . . . . .	12
2.2 Padrões de TV Digital . . . . .	13
2.2.1 Sistema Americano: ATSC . . . . .	14
2.2.2 Sistema Europeu: DVB . . . . .	15
2.2.3 Sistema Japonês: ISDB . . . . .	16
2.2.4 Padrão Brasileiro para o Middleware: Ginga . . . . .	16
2.2.5 TV Digital Móvel . . . . .	17
2.3 Middleware GINGA . . . . .	18
2.3.1 Arquitetura de Referência . . . . .	20
2.4 Sistemas Baseados em Localização . . . . .	22
2.4.1 Arquitetura de Sistemas Baseados na Localização . . . . .	23
2.4.2 Determinação da posição . . . . .	24
2.4.3 Aquisição da informação de localização . . . . .	25
2.4.4 Serviços de processamento da localização . . . . .	25

2.4.5	Serviço de Gateway . . . . .	25
2.5	Sensibilidade ao Contexto . . . . .	26
2.5.1	Computação Sensível ao Contexto . . . . .	26
2.5.2	Definição de contexto . . . . .	27
2.5.3	Classificação do contexto . . . . .	28
2.5.4	Aplicações Sensíveis ao Contexto . . . . .	29
<b>3</b>	<b>TRABALHOS RELACIONADOS</b>	<b>31</b>
3.1	FRAGIL . . . . .	31
3.1.1	RegisterService - Componente de registro do objeto móvel . . . . .	33
3.1.2	UpdatePosService - Componente de atualização da posição . . . . .	33
3.1.3	QueryService - Componente de Consultas . . . . .	36
3.1.4	EventService - Componente de Eventos . . . . .	37
3.2	PersonalTVware . . . . .	37
3.3	Ginga-NCL . . . . .	40
3.3.1	O Ambiente Ginga-NCL . . . . .	42
<b>4</b>	<b>ARQUITETURA PROPOSTA DE INTEGRAÇÃO</b>	<b>46</b>
4.1	A arquitetura . . . . .	46
4.1.1	Cenário de uso . . . . .	47
4.2	Descrição dos componentes arquiteturais . . . . .	49
4.2.1	Personal TVware . . . . .	49
4.2.2	FRAGIL . . . . .	50
4.2.3	Ginga-NCL . . . . .	50
<b>5</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>52</b>
5.1	Sugestões para Trabalhos Futuros . . . . .	52
	<b>Referências Bibliográficas</b>	<b>53</b>

# 1 INTRODUÇÃO

O Brasil, nos últimos anos atravessa a fase de implantação da tecnologia de transmissão e recepção de sinal digital. Da parte das emissoras de sinal cabe além da mudança de transmissão analógica para digital, adequa-se com equipamentos de filmagem e armazenamentos digitais. Já os telespectadores terão que adaptar seus televisores (analógicos) acoplando unidades conversoras (Unidade Receptora Decodificadora - URD) [TAVARES, 2001], também conhecida pelos termos IRD - Integrated Receiver Decoder e Set Top Box - STB [BECKER, 2009] para que possam receber o sinal digital, ou adquirir televisores portados da unidade conversora desde a fábrica.

A maioria da programação veiculada pelas emissoras de televisão aberta já é produzida, utilizando-se equipamentos de filmagem e armazenamento digitais. No entanto, como os segmentos de transmissão e de recepção continuam sendo analógicos, os usuários não percebem esse ganho de qualidade. A introdução da tecnologia digital no serviço de televisão refere-se, no momento, à digitalização desses dois segmentos, o que garantirá de imediato melhor qualidade de imagem e som para os telespectadores. [TAVARES, 2001]

A tecnologia digital, entretanto, ao superar diversas restrições da tecnologia analógica, abre várias outras possibilidades. Imagem de alta definição e som com qualidade de CD são algumas das inovações que poderão ser disponibilizadas com a introdução da televisão digital. A transmissão de vários programas em um único canal é outra novidade que, com certeza, implicará numa maior diversidade de conteúdos, atendendo a diferentes necessidades e interesses dos usuários [TAVARES, 2001]. No entanto, a mudança que parece ser mais drástica é a transformação do televisor em um equipamento interativo. Segundo especialistas do setor, essa nova aplicação modificará radicalmente o modelo de exploração da televisão aberta, viabilizando novos serviços como, por exemplo, um novo tipo de comércio, já definido como *t-commerce*, ou seja, venda de produtos por meio da televisão. Outra possibilidade que também poderá se concretizar, no curto e médio prazos, é a recepção móvel de sinais de televisão em carros, ônibus e trens e em telefones celulares das próximas gerações. [TAVARES, 2001] O uso de dispositivos portáteis no contexto da TV Digital já é uma realidade em diversas partes do mundo

[CRUZ, 2008]. Permitir acesso ao conteúdo televisivo a qualquer instante, em qualquer lugar, e até mesmo em movimento, torna a TV Digital (TVD) para dispositivos portáteis um modelo de negócios interessante.

Nesse modelo, é possível oferecer, além dos serviços de TV tradicionais, serviços com interatividade, no qual o usuário telespectador não reage apenas aos estímulos audiovisuais, mas também às aplicações interativas que despertam seu interesse. Para oferecer uma interface padronizada e um suporte de programação para a autoria dessas aplicações, os dispositivos receptores devem possuir uma camada de software, denominada middleware, usualmente padronizada por cada sistema de TV Digital particular [MORRIS & SMITH-CHAIGNEAU, 2005].

A última década trouxe grandes avanços nas tecnologias de computação e comunicação móvel, provocando uma tendência de mudança no paradigma tradicional de computação, estático e baseado em desktops, para um novo paradigma, altamente dinâmico, com constantes mudanças de ambiente causadas pela mobilidade do usuário, e caracterizado pelo uso intenso de dispositivos móveis multifuncionais, como celulares, palmtops e PDA's. Essa mudança é mais um passo em direção à Computação Ubíqua [WEISER, 1991], um novo paradigma em que sistemas e ambientes acrescidos de recursos computacionais proveem serviços e informações quando e onde sejam desejados pelos usuários. A Computação Ubíqua sugere o desenvolvimento de uma nova classe de aplicações que explora as mudanças de contexto dentro de um domínio dinâmico.

Diferentemente das aplicações tradicionais, as “aplicações móveis sensíveis ao contexto” levam em consideração em seus processamentos não apenas as entradas de dados explícitas dos usuários, mas também entradas implícitas, referentes ao contexto físico e computacional dos mesmos e dos ambientes que os cercam. Essas aplicações exploram a natureza contextual provocada pela mobilidade para produzir serviços mais flexíveis, adaptáveis e centrados no usuário.

## **1.1 Objetivos**

### **1.1.1 Objetivos Gerais**

O objetivo deste trabalho é propor uma arquitetura para ofertas de serviços baseado/sensíveis

à localização geográfica para TV Digital. Para isso, usaremos as informações geográficas de um dispositivo móvel na sensibilidade ao contexto.

### 1.1.2 Objetivos Específicos

Como objetivos específicos, temos:

1. Estudar e explicar o que TV Digital, seu funcionamento e implicações em nossas vidas;
2. Estudar e explanar o que é o Middleware Ginga e seu funcionamento;
3. Estudar e explicar o funcionamento e aplicações de Sistemas Baseados em Localização;
4. Estudar e explicar o que é Sensibilidade ao contexto;
5. Demonstrar a abordagem de trabalhos relacionados ao escopo deste trabalho e que serviram para o desenvolvimento do mesmo;
6. Propor a insteграção entre as arquiteturas relacionadas neste trabalho.

## 1.2 Organização do trabalho

Este trabalho está dividido em 5 capítulos.

No Capítulo 2 é apresentada a fundamentação teórica, na qual serão abordados quatro tópicos relacionados relevantes ao trabalho e que facilitarão a compreensão da proposta, são eles: TV Digital; Middleware GINGA; Sistemas Baseado em Localização e Sensibilidade ao Contexto.

No Capítulo 3, apresentamos trabalhos relacionados para a elaboração da proposta, são eles: FRAGIL, PersonalTVware e Ginga-NCL.

No Capítulo 4, proposta da arquitetura para TV Digital para auxiliar na recomendação de programação sensível ao contexto baseado em localização.

Por fim, no Capítulo 5 são apresentadas as considerações finais com conclusões e sugestões para trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Neste Capítulo será apresentado o conceito de TV Digital, assim como, *Middleware* Ginga, Sistemas Baseados em Localização, Sensibilidade ao Contexto e encerrando com algumas aplicações sensíveis ao contexto.

### 2.1 TV Digital

É um sistema de televisão no qual corresponde transmitir, receber e processar sinais digitais. Isso pode ocorrer por meio de televisores totalmente digitais ou em analógicas com o suporte de uma Unidade Receptora Decodificadora (URD). Com a TV digital há o aumento na qualidade do vídeo (som e imagem) para o telespectador.

A resolução da imagem fica mais larga que a atual (alta definição). O som fica mais estéreo provocando uma sensação de realismo. Mas vale lembrar que não é apenas porque há um sinal digital e pronto para desfrutar de tais melhorias que isso efetivamente ocorre. É preciso estar usando equipamento adequado, senão, o telespectador não perceberá nenhuma diferença, pois o sinal analógico continua sendo transmitido junto ao digital. Sua previsão para parar de ser transmitido é para o ano de 2016. [IDEC, 2007]

Além da melhor qualidade de som e imagem, a TV digital possibilitará o telespectador interagir com a programação dos canais. Através de um canal de interatividade o usuário poderá agir sobre serviços oferecidos pela provedora do sinal digital, o que inclui serviços comerciais, consultas, opiniões, etc. Esse novo paradigma rompe com o conceito de telespectador passivo no qual a pessoa apenas recebe informação da televisão sem interagir com a mesma.

A seguir falaremos da arquitetura de TV Digital, Padrões e TV Portátil.

#### 2.1.1 Arquitetura do Sistema de TV Digital

Assim como em um modelo de redes de computadores é também a arquitetura de um sistema de TV digital, dividido em camadas para ter autonomia, requisitando ou oferecendo

serviços das camadas vizinhas.

O modelo de arquitetura da TV Digital é mostrado na Figura 2.1. Além das funções de modulação/transmissão e demodulação/recepção de sinal já conhecidas da TV convencional (analógica), são introduzidas as funções de compressão/descompressão e o *middleware*. [FERNANDES *et al.*, 2004]

Para o sinal digital carregado de dados digitais se faz necessário uso de compressão da emissão por parte da emissora do sinal e descompressão na parte da receptora (televisor digital).

O middleware ocupa uma posição entre a camada de transporte do sinal e os aplicativos interativos de TV Digital. Observa-se que é através do *middleware* que atua o provedor de serviços interativos. [FERNANDES *et al.*, 2004]

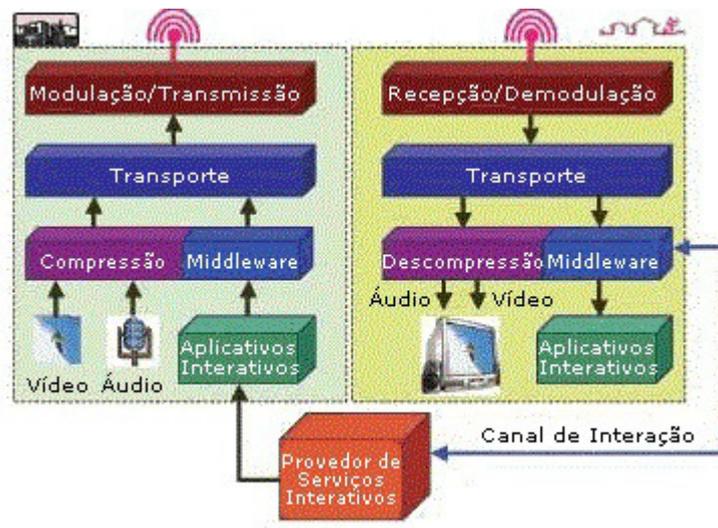


Figura 2.1: Arquitetura em blocos da TV Digital.

## 2.2 Padrões de TV Digital

A figura 2.2, abaixo, mostra como um Sistema de TV Digital é formado por um conjunto de padrões.

Este conjunto de padrões é formado pela digitalização do vídeo e do áudio, middleware responsável pela implementação de interatividade e novos serviços e a multiplexação e transmissão de sinais (modulação).

A seguir apresentamos os três principais padrões mundiais de TV Digital, os

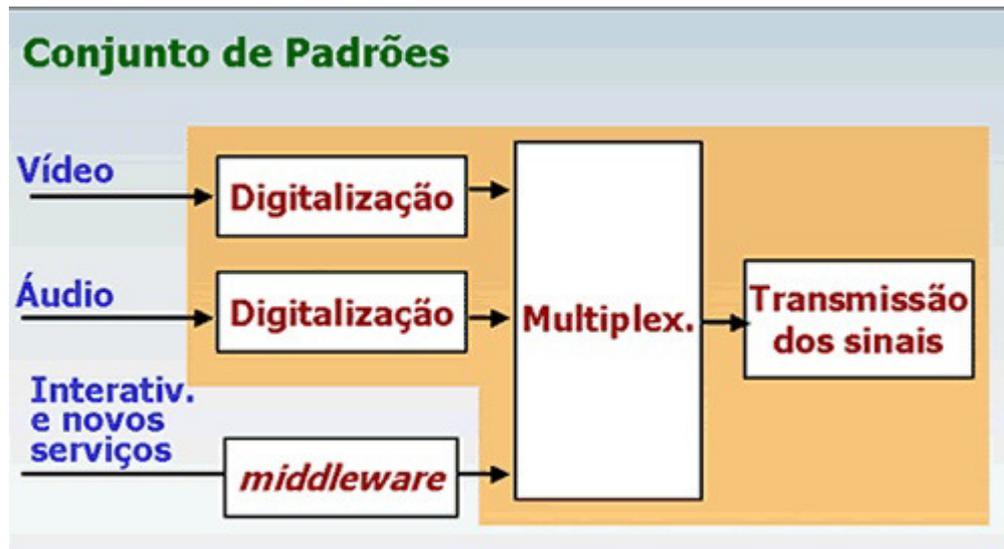


Figura 2.2: Conjunto de Padrões.

quais são: ATSC (Estados Unidos), DVB (Europa) e ISDB (Japão).

### 2.2.1 Sistema Americano: ATSC

ATSC é a sigla de *Advanced Television System Committee*. É o padrão norte-americano de TV Digital, desenvolvido a partir de 1987 por um grupo de 58 indústrias de equipamentos eletroeletrônicos. Desde outubro de 98, está em operação comercial nos Estados Unidos. Foi implantado também no Canadá e na Coreia do Sul. Tem um mercado atual de 267 milhões de televisores.

Seu desenvolvimento foi pensado para operar com conteúdo audiovisual em alta definição (HDTV). A opção do consórcio ATSC garante a melhor resolução de imagem possível. Ao mesmo tempo, restringe a capacidade de transmissão a um só programa por canal. O padrão ATSC-T (*Advanced Television System Committee - Terrestre*) é o padrão estritamente terrestre, não permitindo aplicações móveis e portáteis. Isso se dá as suas características de modulação, entrelaçamento temporal e inflexibilidade na configuração dos parâmetros de transmissão que resultam numa baixa imunidade a multipercursos, afetando assim a recepção em áreas abertas e fechadas.

Atualmente, conforme mostra Figura 2.3, esse padrão somente é utilizado com os canais de 6MHz da TV analógica. Utiliza o formato MPEG-2 para a codificação do sinal de vídeo e multiplexação de fluxos elementares, a codificação Dolby AC-3 para áudio e um sistema de modulação conhecido como 8-VSB para a camada de transporte (no caso

da radiodifusão terrestre).

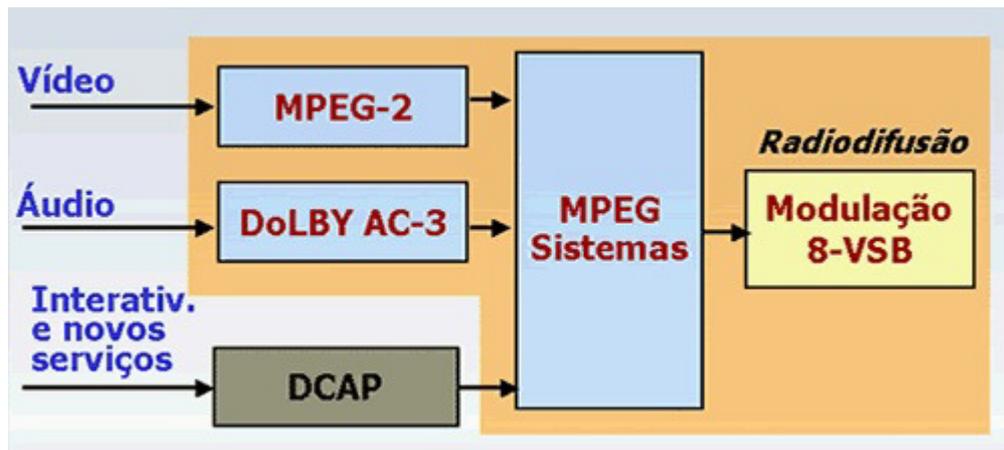


Figura 2.3: Padrão ATSC.

## 2.2.2 Sistema Europeu: DVB

O padrão europeu DVB (*Digital Video Broadcasting*) que significa Transmissão de Digital de Vídeo, mostrado na Figura 2.4. Foi desenvolvido por um consorcio europeu e é muito utilizado por países em todo o mundo. Possui cinco modos de transmissão com resoluções que vão da mais baixa até a mais alta definição.

Esse padrão suporta recepção por dispositivos móveis. Entretanto, segundo seus críticos, não funciona satisfatoriamente, principalmente no modo hierárquico, quando transmite ao mesmo tempo para televisão de alta definição e sistemas móveis.

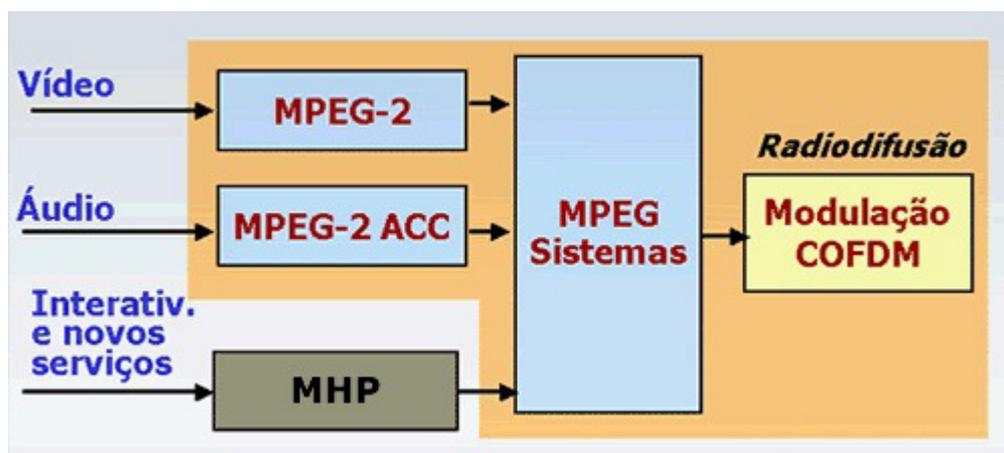


Figura 2.4: Padrão DVB.

### 2.2.3 Sistema Japonês: ISDB

O ISDB (*Integrated Services Digital Broadcasting*) é um padrão de radiodifusão de serviços multimídia desenvolvido pelo consórcio DIBEG (*Digital Broadcasting Experts Group*) no Japão e tendo como suporte, a emissora pública japonesa NHK, mostrado na Figura 2.5.

Essa característica de serviços multimídia do ISDB-T (terrestre) além de permitir uma gama de serviços, também possibilita significativos benefícios aos radiodifusores e fabricantes de receptores, se considerarmos a popularização da radiodifusão digital e do cultivo de um novo mercado.

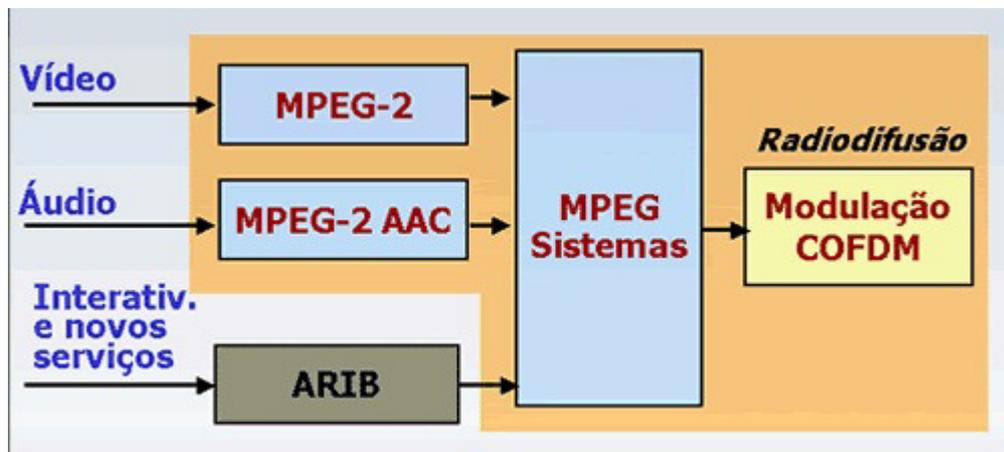


Figura 2.5: Padrão ISDB.

### 2.2.4 Padrão Brasileiro para o Middleware: Ginga

A TV Digital possibilita uma gama de novidade e melhorias, seja em qualidade de imagem e áudio ou com a introdução de interatividade e novos serviços. No entanto, estas novidades não são totalmente realizáveis simultaneamente, seja por limitações técnicas ou custo. Assim, deve se escolher quais requisitos devem ser priorizados tal qual fizeram os americanos, europeus e japoneses ao definirem seus padrões.

O padrão Ginga, que comentaremos com mais detalhes a frente, na sua definição teve seus requisitos definidos atendendo as necessidades específicas da nossa sociedade tais como: baixo custo e robustez na recepção (classes C, D e E); flexibilidade e capacidade de evolução (classes A e B); e interatividade e novos serviços (inclusão digital), conforme mostrado na Figura 2.6.

Esses atributos estão mapeados na proposta de Sistema de TV Digital Brasileiro

apresentada na figura a seguir.

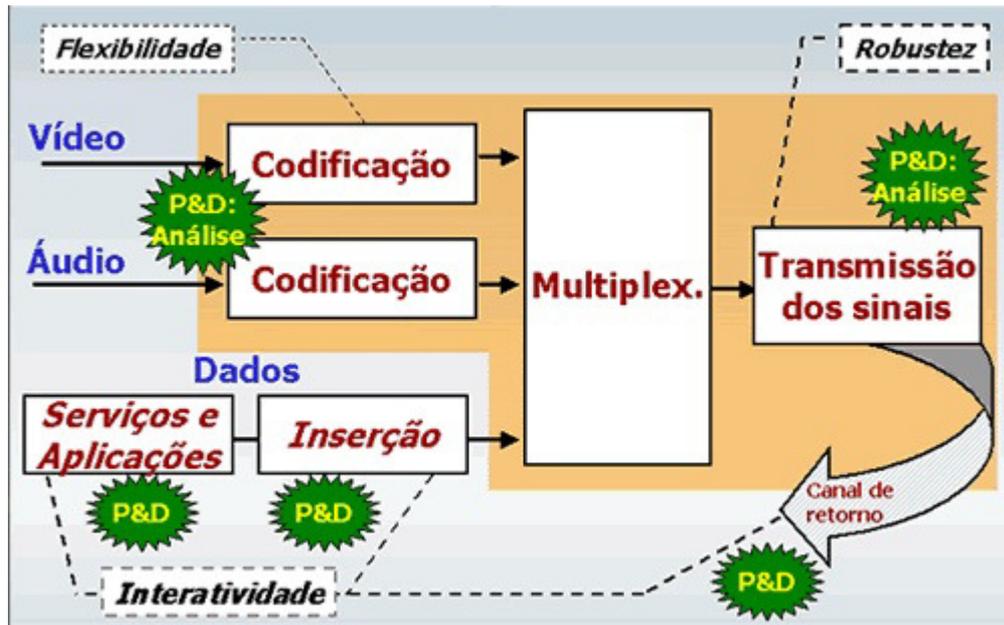


Figura 2.6: Padrão Ginga.

Visualizando os requisitos apresentados, o que se enxerga é um esforço em Pesquisa e Desenvolvimento (PD) ao analisar e selecionar os padrões de codificação e transmissão dos sinais e desenvolver o middleware de maneira à futura inserção de novos serviços e aplicações.

### 2.2.5 TV Digital Móvel

Os brasileiros são os que mais assistem TV móvel (via smartphone, PC ou laptop) na América Latina. A informação é de um estudo global realizado pela Motorola Mobility com 9 mil consumidores em 16 mercados, que revela que 34

Os laptops ainda são o meio mais utilizado para assistir TV entre os equipamentos móveis citados na pesquisa, com quase 60

O principal gatilho dessa demanda por vídeo móvel é a experiência de entretenimento do consumidor, que busca TV móvel, TV social, serviços de casa conectada e serviços em nuvem personalizados.

**TV social** Os números ainda demonstram uma alta tendência para o crescimento da TV social no País e classifica essa tecnologia como um grande nicho de oportunidades, tanto para as operadoras quanto para seus anunciantes. O estudo reforça a

questão de que as redes sociais mudaram a experiência de ver televisão. De acordo com o levantamento, os brasileiros gastam seis horas por dia em redes sociais, número semelhante à média global e de outros países da América Latina. Desses, 43

**Objetivo** O estudo teve como objetivo identificar os hábitos de consumo de vídeo dos telespectadores e dar uma compreensão dos tipos de serviços disponíveis aos consumidores e suas respectivas preferências. Os entrevistados são da Argentina, Austrália, Brasil, China, França, Alemanha, Japão, México, Rússia, Cingapura, Coreia do Sul, Suécia, Turquia, Emirados Árabes, Reino Unido e EUA. (Fonte: Tela Viva)

## 2.3 Middleware GINGA

Se pensarmos num esquema de camadas tal qual na Figura 2.7, podemos dizer que o *middleware* é uma destas camadas posicionada entre as aplicações e a infraestrutura (hardware e sistema operacional), como ilustrado no Modelo de Referência do Sistema Brasileiro de TV Digital Terrestre. O *middleware* para aplicações de TV digital compreende de máquinas de execução das linguagens oferecidas, e bibliotecas de funções, que permitem o desenvolvimento rápido e fácil de aplicações. [SOARES, 2009]

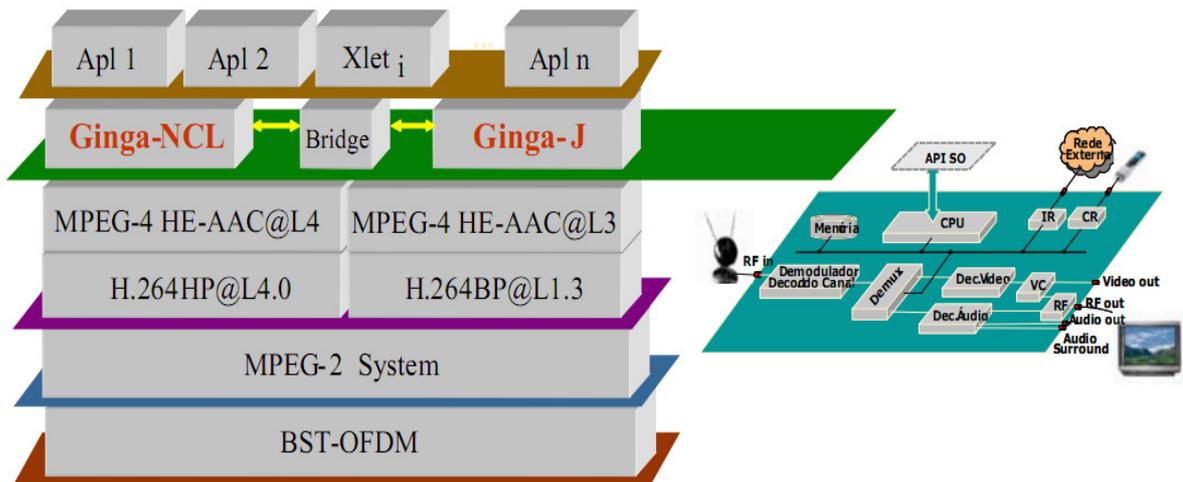


Figura 2.7: Módulo de Referência do SBTVD.

As aplicações de TVD (TV Digital) podem ser divididas em um conjunto de aplicações declarativas e um conjunto de aplicações imperativas. Quem define a que conjunto pertence uma aplicação é a entidade inicial da mesma, isto é, aquela que dispara a aplicação, ou seja, se essa entidade é codificada segundo uma linguagem declarativa

ou imperativa logo logo o conjunto também o é. Mas é importante dizer que aplicações imperativas podem conter entidades declarativas e vice-versa, a entidade inicial é que as caracteriza.

Linguagens declarativas destacam a descrição declarativa de uma tarefa, ao invés de sua decomposição sequencial de um algoritmo, tal qual fazem as descrições imperativas. Tarefas descritas de forma declarativa são mais fáceis de serem concebidas e entendidas, pois são de mais alto nível de abstração, sem ser necessário um programador especialista, como é o caso nas tarefas descritas de forma imperativa. Entretanto, uma linguagem declarativa tende para um determinado domínio de aplicações e define um modelo específico para esse domínio. Geralmente quando uma tarefa coincide com o modelo da linguagem declarativa, é melhor desenvolvê-la em linguagem declarativa.

Linguagens imperativas, em geral, são bem definidas, de propósito geral e com um elevado custo. Do contrário da declarativa, as imperativas usualmente exigem um programador especialista, geralmente colocam em risco a portabilidade de uma aplicação. Porém, quando o objetivo de uma tarefa não coincide com o da linguagem declarativa, então é melhor desenvolvê-la em linguagem imperativa.

Pode-se dizer que os middlewares para TV digital fornecem suporte para o desenvolvimento tanto seguindo o paradigma declarativo quanto o imperativo. No caso do sistema japonês, a entidade inicial de uma aplicação é sempre declarativa, mas as outras entidades podem ser codificadas segundo o paradigma imperativo. Já o caso do sistema americano e europeu, há suporte tanto para aplicações declarativas, quanto para aplicações imperativas, mas em ambos os casos, entidades podem ser definidas mesmo adotando um paradigma diferente da entidade inicial.

O ambiente declarativo no *middleware* possui suporte necessário às aplicações declarativas, tal qual o ambiente imperativo para aplicações imperativas. No caso do padrão brasileiro, os dois ambientes são exigidos nos receptores fixos e móveis, enquanto para receptores portáteis apenas o ambiente declarativo é necessário.

O Sistema Brasileiro de TV Digital Terrestre (SBTVD) trouxe como principal inovação seu *middleware*, denominado Ginga. Ginga é uma qualidade dos brasileiros que lembra movimento e atitude, facilmente visível naquilo que fazem. Ginga é flexibilidade, adaptação, qualidades inerentes ao *middleware* brasileiro.

### 2.3.1 Arquitetura de Referência

A arquitetura do Ginga pode ser dividida em três módulos principais: Ginga-CC, Ginga-NCL e Ginga-J, como mostra a Figura 2.8. Os dois últimos módulos compõem a camada de Serviços Específicos do Ginga. Ginga-J é o subsistema lógico do middleware Ginga responsável pelo processamento de aplicações imperativas escritas utilizando a linguagem Java. A especificação desse subsistema caberá à Norma ABNT NBr 15606-4. Ginga-NCL é o subsistema lógico do middleware Ginga responsável pelo processamento de aplicações declarativas NCL. NCL (*Nested Context Language*) e sua linguagem de script Lua compõem a base para o desenvolvimento de aplicações declarativas no SBTVD. A especificação desse subsistema cabe às Normas NBR 15606-2 e ABNT NBR 15606-5, e é o assunto da Seção 3.3.

Para prover as funcionalidades comuns ao suporte dos ambientes declarativo, Ginga-NCL, e imperativo, Ginga-J, há o subsistema lógico Ginga-CC (Ginga Common Core). Essa arquitetura garante que apenas o módulo Ginga-CC precise ser adaptado à plataforma onde o Ginga será embarcado. Assim, Ginga-CC provê um nível de abstração da plataforma de *hardware* e sistema operacional, acessível através de APIs (*Application Program Interfaces*) bem definidas.

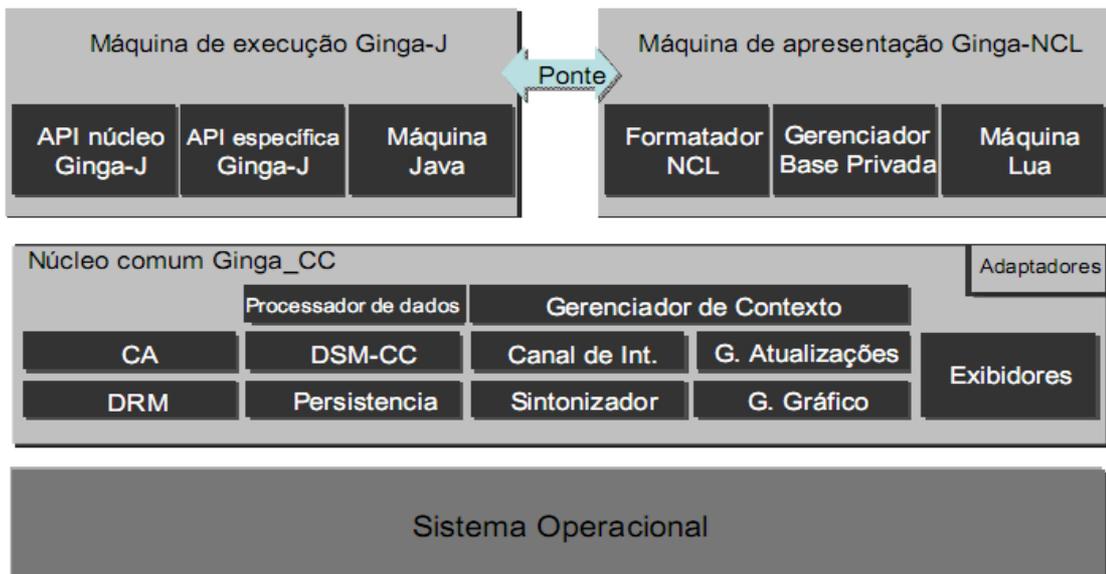


Figura 2.8: Arquitetura Referência do Middleware Ginga.

Na Figura 2.8, um conjunto comum de exibidores monomídia faz parte dos componentes do Ginga-CC. Eles são exibidores de áudio, vídeo, texto e imagem, in-

cluindo entre eles o exibidor MPEG-4/h.264, implementado por *hardware*. O acesso a tais exibidores se dá através de adaptadores, responsáveis por notificar eventos de apresentação e seleção (interação do usuário). As características de tais exibidores são definidas na Norma ABNT NBr 15606-1. O Gerenciador Gráfico é o responsável pelo gerenciamento do modelo conceitual do plano gráfico de apresentação. É ele que define o plano de exibição do vídeo principal h.264, os planos de exibição dos outros objetos de mídia que compõem uma aplicação TVD, e como esses planos se superpõem. A Norma ABNT NBr 15606-1 é responsável também por tal definição.

Todo o acesso a dados obtidos através do canal de retorno (ou Canal de Interatividade) é também de responsabilidade do Ginga-CC. As diversas possibilidades de canal de interatividade são especificadas na Norma ABNT NBR 15607.

Ainda na Figura 2.8, os componentes DSM-CC e Processador de Dados oferecem o suporte para obtenção de dados, obtidos através de carrosséis de objetos DSM-CC, especificados na Norma ABNT NBr 15606-3. O componente de Persistência é o encarregado pelo gerenciamento de armazenamento de dados requisitados pelas aplicações; enquanto o componente Sintonizador é o responsável pela sintonização e controle do canal de rádio frequência.

Os demais componentes do Ginga-CC são opcionais e dependem da implementação particular de cada receptor. O Gerenciador de Contexto é o encarregado de colher informações do dispositivo receptor, informações sobre o perfil do usuário e sua localização, e torná-las disponíveis ao Ginga-NCL e Ginga-J, para que eles possam efetuar adaptação de conteúdos ou da forma como conteúdos deverão ser apresentados, conforme determinado pelas aplicações.

Por fim, ao Gerenciador de Atualizações cabe o controle das atualizações de todo o *software* residente e do *middleware* Ginga, durante o ciclo de vida de um dispositivo receptor. Os componentes CA (*Conditional Access*) e DRM (*Digital Right Management*) são os responsáveis por determinar os privilégios de acesso às diversas mídias que compõem uma aplicação (programa) TVD.

## 2.4 Sistemas Baseados em Localização

Sistemas ou serviços baseados em localização (LBS - *Location-based Systems or Services*) proveem aos usuários, através de uma rede de comunicação móvel, um conjunto de serviços ou funcionalidades baseadas na sua posição ou localização geográfica. Os LBS podem ser vistos como a convergência de várias novas tecnologias (NICTs - *New Information Communication Technologies*) [SHIODE, 2002], entre elas: sistemas de comunicação móvel, tecnologias de localização e dispositivos portáteis, com a Internet, Sistemas de Informação Geográfica (SIGs) e banco de dados espaciais, como mostrado na Figura 2.9.

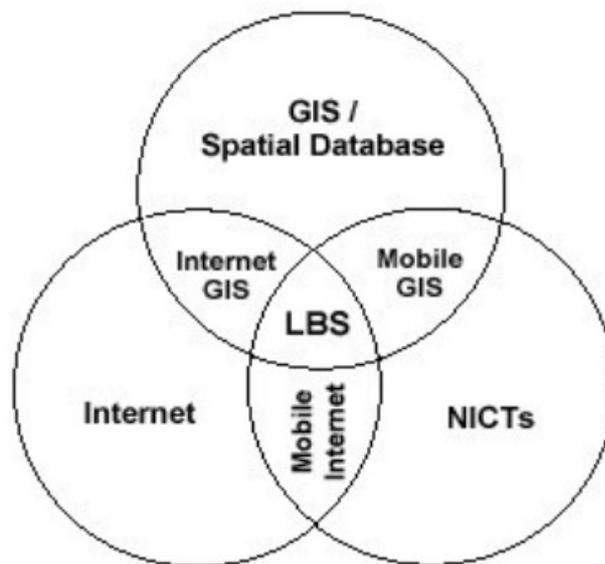


Figura 2.9: LBS como convergência de várias tecnologias.

A mobilidade dos usuários é uma das principais características de sistemas LBS, graças a dispositivos portáteis como laptops, PDAs e principalmente telefones celulares. Poder saber a localização de pessoas, objetos e fenômenos a qualquer momento, com aplicações que são sensíveis à posição do usuário através de uma grande variedade de dispositivos via Internet e rede sem fio, nos leva imensos benefícios para diversos setores da sociedade, tais como o comércio (ofertas de produtos e serviços) e os consumidores (rastreamento e navegação).

Em um LBS o ponto central está em obter a informação necessária para determinar a localização do usuário móvel e, baseado nessa informação, oferecer serviços sensíveis a esse contexto de utilização do sistema.

A localização representa uma posição no espaço e no tempo, que pode ser

mensurada tendo como base suas coordenadas em um sistema de referência espacial e temporal particular [OGC, 2004]. Para sistemas sensíveis à localização de um usuário, são necessárias tecnologias para determinar a localização dos clientes e processar a informação de localização.

Assim, considerando a forma como a localização do usuário é obtida, cita-se três gerações de LBS distintas [DIBDIN, 2001]. A primeira geração caracteriza-se pelo fato de que o usuário deveria fornecer, manualmente, para a aplicação, a informação de onde está localizado. Para isso poderia ser em forma de um código postal (CEP) e endereços determinados por nome de cidades, ruas, quadras. Na segunda geração, a localização do dispositivo passa a ser determinada de maneira automática. Por fim, na terceira e atual geração das aplicações LBS, a informação de localização é determinada automaticamente e com um alto grau de precisão, porém a principal diferença para com segunda geração é a pro-atividade, ou seja, a possibilidade de um serviço ser ofertado sem que o usuário solicite-o explicitamente.

Segundo [DESHPANDE & BORRIELLO, 2002], o estado dos sistemas baseados na localização, é marcado pela grande gama de tecnologias de localização de posição, pela necessidade do uso de diferentes tecnologias de localização para diferentes ambientes e, por isso, a falta da chamada tecnologia de localização ideal, isto é, a falta de uma tecnologia que atenda a todos os ambientes e usos de maneira ótima.

### **2.4.1 Arquitetura de Sistemas Baseados na Localização**

Para sistemas sensíveis à localização de um usuário, deve ser considerada uma estrutura básica de componentes que deem suporte à localização, solicitações, consultas e ao gerenciamento da informação geográfica e contextual. E são necessárias tecnologias para determinar a localização dos clientes e processar a informação de localização.

Como a mobilidade dos usuários é o fator principal, é preciso localizá-los de forma que no momento em que uma solicitação/consulta for realizada, saibam-se quais os dados devem ser enviados, levando-se em consideração a posição atual do cliente. Também é necessária uma estrutura que receba e processe as solicitações dos clientes e envie a informação pedida para o mesmo através de uma rede sem fio. Na Figura 2.10 pode-se observar uma arquitetura genérica para sistemas baseados na localização do usuário.

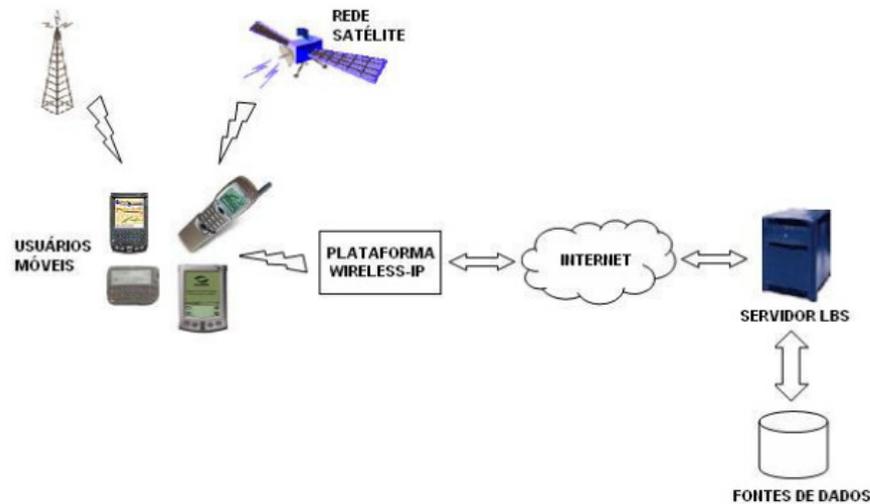


Figura 2.10: Arquitetura geral de um sistema LBS.

Pode-se verificar da arquitetura geral que entre os componentes necessários para aplicações LBS, destacam-se os componentes para a determinação da posição, aquisição da informação sobre a localização, serviços de processamento da localização e o serviço de *Gateway*.

## 2.4.2 Determinação da posição

Em um LBS, os componentes de determinação da posição são responsáveis em obter a posição dos dispositivos móveis em tempo real, conseguindo isto através de *hardware* e software tanto para identificar quanto para rastrear a localização dos usuários. O equipamento de determinação da posição (PDE - Position-Determining Equipment) é o *hardware* básico identificar a localização dos dispositivos móveis [MAPINFO, 2002]. Existem duas maneiras básicas para conseguir a posição de um dispositivo móvel, são elas: soluções baseadas em rede (*network-based*) e soluções baseadas no dispositivo (*handset-based*).

As soluções baseadas em rede (*network-based*) usam um equipamento de localização colocado nas estações rádio-base (ERB' s) do sistema. Desta maneira, fica a cargo da infraestrutura da rede todo o processamento matemático necessário para a determinação das coordenadas dos dispositivos. Por sua vez, as soluções baseadas nos dispositivos (*handset-based*) implementam grande parte da tecnologia nas unidades móveis e que utilizam como base o uso da tecnologia GPS (*Global Positioning System*).

### 2.4.3 Aquisição da informação de localização

A informação contida na localização se refere tanto ao conteúdo que descreve características relacionadas a uma posição geográfica quanto à própria posição geográfica. Assim, um sistema de localização pode fornecer vários tipos de informação, aqui denominada: física ou simbólica e absoluta ou relativa. Entende-se por física a localização geográfica ocupada pelo usuário. Por exemplo, o GPS oferece posições físicas: um prédio está situado a 47°39'17" N, 122° 18' 23" W e 20,7 metros de altitude. Em compensação, a localização simbólica utiliza um posicionamento abstrato de onde algo está localizada como, por exemplo, na sala, na rua em frente, no andar de cima. Sistemas que oferecem posicionamento físico podem ser estendidos para suportar uma localização simbólica correspondente e utilizar o posicionamento físico para determinar um conjunto de informações simbólicas.

A informação absoluta possui uma matriz de referência única para especificar a localização de um determinado objeto, expressa, em latitude, longitude e altitude. Adicionalmente, pode-se ter um atributo de direção complementando a localização absoluta. Já no sistema relativo, cada objeto pode ter a sua própria matriz de referência [HIGHTOWER & BORRIELLO, 2001].

### 2.4.4 Serviços de processamento da localização

Para disponibilizar serviços baseados na informação geográfica, deve haver processamentos, de forma a satisfazer as consultas dos usuários oferecendo uma visão amigável e contextualizada dessas informações. Esses componentes podem oferecer serviços tais como: serviços de consulta e visualização, serviços de geocodificação (procurar a localização física a partir de um endereço específico) e geocodificação reversa, serviços de planejamento de rotas e serviços de geração e visualização de mapas (transformação da informação de localização em mapas e imagens).

### 2.4.5 Serviço de Gateway

*Gateway* é um *middleware* que fica entre os serviços de processamento e os dispositivos móveis. Seu papel é fazer a comunicação entre os serviços oferecidos pelo servidor o dispositivo do usuário. Portanto, o *gateway* tem como objetivos obter a atual posição

de um objeto móvel a partir de um servidor de localização, receber e processar as solicitações advindas do usuário e encaminhá-las para um servidor Web ou outra aplicação e, por fim, converter as respostas em um formato que possa ser visualizada, levando-se em consideração o sistema operacional e as limitações dos diferentes dispositivos móveis.

## 2.5 Sensibilidade ao Contexto

### 2.5.1 Computação Sensível ao Contexto

A computação sensível ao contexto é um paradigma computacional que se propõe a permitir que as aplicações tenham acesso e tirem proveito de informações que digam respeito às computações que realizam, buscando otimizar seu processamento. Tem a ver com a capacidade dos sistemas computacionais conseguirem vantagem das informações ou condições existentes em um ambiente dinâmico para adicionar valor aos serviços ou executar tarefas mais complexas.

Assim, além de lidar com entradas explícitas, a computação sensível ao contexto também considera informação de contexto capturadas por meio de sensores, ou seja, entradas implícitas, tais como: localização, recursos e infra-estrutura disponíveis, preferências do usuário, atividade do usuário, número de dispositivos, tipo de dispositivo, carga computacional [CHEN & KOTZ, 2000].

Os sistemas computacionais sensíveis ao contexto começaram a ser desenvolvidos em 1992 com a aplicação proposta por [WANT *et al.*, 1992] o *Active Badge Location System*, esse sistema conseguia determinar a localização atual do usuário, a qual era usada para direcionar as ligações telefônicas para o telefone mais próximo, fazia uso de tecnologia de infravermelho. A Figura 2.11 mostra exemplos de dispositivos *Active Badges*. Alguns trabalhos a respeito de sensibilidade à localização foram desenvolvidos ao longo dos anos 90, como se pode ver em [ABOWD *et al.*, 1997], nesse período a localização do usuário era o atributo de contexto mais usado.

A localização é um aspecto-chave para os sistemas com mobilidade, pois a localidade influencia significativamente no contexto disponibilizado para os mecanismos de adaptação. [YAMIN, 2004].

A computação sensível ao contexto, nos últimos anos, tem recebido maior



Figura 2.11: Exemplo de dispositivo *Active Badges*.

atenção, principalmente em função do desenvolvimento da computação móvel e do aparecimento de uma nova geração de dispositivos móveis. Porém, a construção do suporte à sensibilidade ao contexto para as aplicações apresenta inúmeros desafios, os quais se relacionam especialmente a obtenção, modelagem, armazenamento, distribuição e monitoramento do contexto.

## 2.5.2 Definição de contexto

A primeira vez que o termo *context-aware* (sensibilidade ao contexto) apareceu foi em [SCHILIT & THEIMER, 1994]. Neste trabalho o contexto é descrito como o local, as identidades de pessoas próximas e os objetos e mudanças para esses objetos. O contexto refere-se como a localização do usuário, o ambiente, a identidade e o tempo, segundo [RYAN *et al.*, 1997]. Assim como [CHEN, 2002] que defende a inclusão do Tempo (hora do dia, da semana, do mês e a estação do ano) como mais uma categoria de contexto e introduz o conceito de Histórico de Contexto e a necessidade de armazenamento de informações contextuais como fonte de tomada de decisões e construção de aplicações sensíveis ao contexto. Para [DEY, 2000] o contexto é definido como o estado emocional

do usuário, o foco de atenção, a localização e a orientação, a data e o tempo, os objetos e as pessoas no ambiente do usuário. Já em [HULL *et al.*, 1997] o contexto é descrito como os aspectos da situação corrente.

A definição de [DEY & ABOWD, 2000] é bastante citada, na qual, o contexto é compreendido como qualquer informação que pode ser usada para caracterizar a situação de uma entidade, considerando-se por entidade uma pessoa, um lugar ou um objeto que é levado em conta para a interação entre um usuário e uma aplicação, incluindo os próprios usuário e aplicação. [DEY, 2000] também destaca que os contextos mais relevantes para um ambiente computacional são: a localização, a identidade, o tempo e a atividade de uma entidade.

### 2.5.3 Classificação do contexto

Uma das formas para classificação é através do dimensionamento do contexto. Para [PREKOP & BURNETT, 2003] e [GUSTAVSEN, 2002] essas dimensões são denominadas de externa e interna, de modo semelhante, [HOFER *et al.*, 2002] refere-se a contexto físico e lógico. A dimensão externa (ou física) significa que o contexto pode ser capturado ou mensurado através de sensores de *hardware*, por exemplo, localização, luz, som, movimento, temperatura. No caso da dimensão interna (ou lógica) é explicitada pelo usuário ou capturada monitorando a interação do mesmo, por exemplo, tarefas, contexto de trabalho, processos de negócio, estado emocional do usuário, perfil.

A maior parte dos sistemas sensíveis ao contexto aborda o uso de fatores de contexto externo, pois fornecem dados úteis, tais como, informação de localização, também bastante utilizado naqueles sistemas. Além disso, atributos externos são fáceis de serem monitorados, pois a grande gama tecnologias de sensoriamento facilita o acesso a estes atributos. Mas, em [BUDZIK & HAMMOND, 2000], é possível conferir o uso de atributos lógicos, no qual o projeto fornece ao usuário informações relevantes obtidas a partir do acesso a páginas *Web* e documentos, trabalhando na perspectiva de descoberta de recursos e agentes de informação.

Quando lidamos com contexto podemos identificar entidades, tais como: lugares (salas, prédios), pessoas (indivíduos, grupos) e coisas (objetos físicos, recursos computacionais) [DEY *et al.*, 2001]. Cada entidade pode ser descrita através de vários atributos, como: identidade (cada entidade tem um único identificador), localização (uma

entidade possui uma localização e proximidades), status (corresponde às propriedades intrínsecas de cada entidade, por exemplo, temperatura e iluminação de uma sala, processos sendo executados em um dispositivo) e tempo (usado para precisamente definir a situação, ordenação de eventos).

#### 2.5.4 Aplicações Sensíveis ao Contexto

A seguir são apresentados exemplos de aplicativos que implementam o conceito de sensibilidade ao contexto:

##### **Shopping Assistant (ATT Bell Labs, 1994)**

Usa a localização do cliente na loja para guiar o comprador pela loja, por exemplo, mostra a localização de certos itens, indica promoções, faz uma comparação de preços etc. Para usuários cativos e usuários anônimos, que permitem (ou não) o uso de seus perfis ou preferências de consumo.

##### **Cyberguide (Georgia Tech, 1997)**

Fornece ajuda a turistas, como: informações sobre caminhos/rotas/atrações, informações adicionais sobre as atrações visitadas (história, vida do artista). Permite deixar comentários sobre os itens visitados para futuros usuários. Automaticamente compila um diário de visitas e a partir desta informação, faz sugestões sobre lugares que possam interessar. Localização indoors (usando dispositivos com infravermelho) e outdoors (GPS).

##### **Conference Assistant (Georgia Tech, 1999)**

Usa informação de contexto sobre atividade do usuário, seus interesses, sua localização, hora e o programa da conferência para auxiliar o participante de uma conferência. Assim, sugerir sessões/palestras que deve assistir. Na sala de apresentação, é mostrado o nome do palestrante, título da apresentação, etc. Os slides e todos os comentários, perguntas, anotações são gravados (em áudio e vídeo) para posterior consulta. Localização *Indoors* (usando *active badges*).

##### **GUIDE (Lancaster, 1999)**

Guia para turistas da cidade de Lancaster. Usa 802.11 (Wi-Fi) para comunicação e detecção de localização (conexão com um *Access Point*). As células são distantes

e sem interseção (*handover* desabilitado) permitindo melhor identificação de localização. Cada célula possui um servidor da célula, que difunde informações sobre atrações locais para usuários naquela célula. Dispositivo é um TabletPC TeamPad (21x15cm, 850g), e acesso através da Web.

## 3 TRABALHOS RELACIONADOS

Neste capítulo são apresentados trabalhos que contribuíram para o desenvolvimento da proposta deste trabalho, sendo nesta ordem: FRamework de Gerenciamento da Informação de Localização, FRAGIL; PersonalTVware; e Ginga-NCL.

### 3.1 FRAGIL

A arquitetura do FRamework de Gerenciamento da Informação de Localização, FRAGIL, busca possibilitar a reutilização de funcionalidades de gerenciamento da informação de localização oferecendo diversos protocolos para a atualização dessa informação, permitindo a obtenção da mesma através do dispositivo móvel ou de dispositivos da rede e oferecendo diversas consultas espaciais úteis em aplicações LBS. [MONTEIRO, 2005]

Nesta arquitetura observamos que o *framework* é organizado seguindo um modelo em camadas, sendo três camadas: Location Server, ClientAPI e ApplicationAPI; e um componente de dados DataAdapter, como apresentado na Figura 3.1.

O Location Server (LS) é gerenciador da informação de localização dos dispositivos móveis, oferecendo para a aplicação LBS e seus clientes serviços relacionados à posição de tais objetos. Para isso, o LS é composto por componentes responsáveis pela execução de tarefas específicas, tais como: definição e controle dos protocolos de atualização da informação de localização dos dispositivos móveis (UpdatePosService), cadastro de um novo dispositivo móvel a ser monitorado (RegisterService), elaboração de consultas (QueryService), controle de eventos relacionados ao posicionamento de um ou mais objetos (EventService) e manutenção da informação de localização.

A camada ClientAPI serve como interface de comunicação do cliente móvel com o LS, permitindo ações de solicitação de registro do dispositivo móvel e a atualização de sua localização, de acordo com o protocolo de atualização escolhido.

Já a camada ApplicationAPI é uma interface para comunicação entre o LS e o servidor LBS que contém a semântica da aplicação. O DataAdapter é o componente para que o *framework* seja independente de SGBD e das linguagens de consulta espa-

cial, as quais, geralmente, são específicas de cada arquitetura proprietária, componente este, inspirado na arquitetura do *framework* IGis [MIRANDA, 2002]. O DataAdapter é responsável pela tradução de consultas realizadas no padrão OpenGIS para a linguagem específica de cada sistema de banco de dados. No trabalho proposto é feito uso principalmente das duas primeiras camadas: Location Server (lado servidor) e ClientAPI (lado cliente).

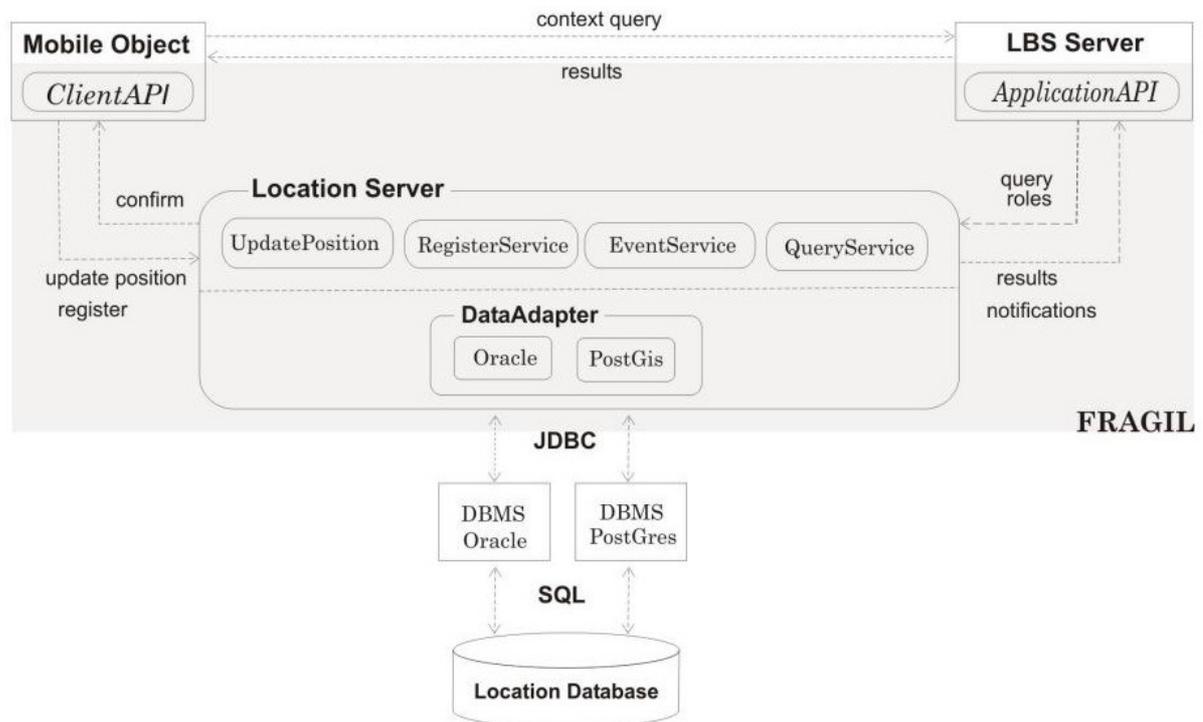


Figura 3.1: Arquitetura geral do FRAGIL.

O LS é composto pelos seguintes componentes: o UpdatePosService, que permite a atualização da posição atual dos objetos móveis; o RegisterService, que administra o cadastro de um novo objeto a ser monitorado e é responsável pela definição das políticas de atualização a serem utilizadas de forma a otimizar o acesso às informações de localização; o QueryService, que fornece uma série de consultas baseadas na posição dos dispositivos móveis e o EventService, que permite o cadastro e o controle dos eventos relacionados ao posicionamento de um ou mais objetos. Nas subseções seguintes descreveremos com detalhes os componentes internos do LS. [MONTEIRO, 2005]

### 3.1.1 RegisterService - Componente de registro do objeto móvel

Para que a posição de um objeto possa ser gerenciada é necessário que este seja registrado junto ao módulo de gerenciamento da informação de localização. Para tanto, algumas informações iniciais deverão ser fornecidas, tais como: posição inicial, tempo inicial e tipo de política de atualização a ser adotada.

Cada objeto móvel que deseja ser monitorado deverá, através da ClientAPI, enviar uma mensagem registrando-se no Location Server. Esse, por sua vez, cria uma sessão de monitoramento para o objeto e o envia uma mensagem de confirmação do serviço. A iniciativa de solicitar o registro de um objeto também poderá ser feita por um servidor nos casos em que o objeto não tem a capacidade de obter o seu posicionamento sozinho, dependendo de um sistema de posicionamento baseado na rede. A partir desse momento, um identificador será atribuído ao objeto, permitindo que seja referenciado tanto pelos módulos da aplicação quanto pelo componente de gerenciamento de localização.

Há duas formas de armazenar a posição de um determinado objeto, dependendo do tipo de aplicação que o usuário estiver interessado em desenvolver. A primeira, seria ter para cada objeto uma única entrada no banco de dados que representaria a provável posição atual daquele objeto específico. A segunda, seria para cada objeto móvel existisse um conjunto de entradas no banco de dados, definindo sua posição espacial num tempo  $t$ . Essa última abordagem é essencial para aplicações que necessitam obter a localização dos usuários tanto no presente quanto no passado, um exemplo seriam as aplicações de rastreamento. Tal abordagem é a que atualmente encontra-se disponibilizada no *framework*.

### 3.1.2 UpdatePosService - Componente de atualização da posição

Para controlar a transmissão da informação de localização, existem diferentes protocolos de atualização que podem ser adotados. Tais protocolos usam várias propriedades especiais da informação de localização para transmiti-las da forma mais eficiente possível, objetivando minimizar o número de mensagens para atualizar o servidor, sem perder, contudo, o efetivo grau de precisão desejado. A localização do dispositivo móvel é determinada diretamente por um sistema de posicionamento acoplado ao próprio dispositivo ou está armazenada em outro servidor de localização.

Segundo [LEONHARDI & ROTHERMEL, 2002], os protocolos de atualização são classificados em três classes: *querying*, *reporting* e *combined protocols*, onde cada classe possui um número típico de variantes. Cada um desses protocolos possui propriedades características e que se modificam para um dado ambiente ou para certos requisitos da aplicação.

Um protocolo é classificado como *querying* se o servidor decide quando solicitar a informação de localização do objeto móvel. Nesse caso, o objeto móvel pode ser implementado de forma simples, pois não necessitará processamentos lógicos complicados ou fazer análise extensas de informações sobre o seu estado. Isso torna possível o uso de dispositivos de pequeno porte.

Na forma mais simples, denominada *simple querying*, o servidor solicita a informação de localização do objeto móvel sempre que a aplicação necessitar de tal informação e, portanto não necessitaria ficar armazenada no servidor. Isso implica em um alto grau de precisão da informação de localização, mas também em um grande número de troca de mensagens, se a mesma for requerida de forma frequente. O tempo de resposta do servidor é também comparativamente alto, uma vez que o mesmo deverá contatar o cliente a cada nova solicitação. Por outro lado, esse protocolo deve ser utilizado se as configurações de privacidade do usuário não permitirem que sua localização seja armazenada em qualquer servidor de localização a não ser no seu próprio dispositivo.

Uma otimização do protocolo anterior, em que o servidor armazena uma cópia da última atualização transmitida, é conhecida como *cached querying*. Nesse caso, quando a informação de localização de um objeto é solicitada, o servidor analisa se a informação que possui é satisfatória para a aplicação e a retorna, caso contrário comporta-se como no *simple querying* e solicita a informação ao objeto móvel. Por fim, está disponível a técnica em que, a cada intervalo de tempo determinado  $t$ , o servidor solicita a localização ao objeto móvel, a esse protocolo denominamos *periodic querying*.

Outra classe de protocolos que merece destaque são os *reporting*. Esta é a classe de protocolos atualmente suportada pelo *framework*. No caso desses protocolos, a iniciativa da atualização parte do objeto móvel. Este sabe qual foi a última informação enviada para armazenamento no servidor e, portanto conhece a informação armazenada nele.

Uma mensagem de atualização é enviada sempre que uma comparação com a

posição atual atinge algum parâmetro de configuração predeterminado como, por exemplo, tempo e distância. O servidor pode assim calcular o máximo de incerteza do valor armazenado a partir do valor limite parametrizado. Com esse protocolo, o servidor sempre responde às solicitações das aplicações consultando diretamente o valor atualizado em sua base. Dessa forma, ele somente pode retornar a informação de localização com a precisão determinada pelo valor limite parametrizado, ainda que a aplicação solicite uma informação mais precisa. O tempo de resposta do servidor é relativamente curto, já que o mesmo não necessita entrar em contato com a fonte. Tal protocolo é usualmente mais eficiente que a classe de protocolos anteriormente citados se a informação de localização é requerida frequentemente ou se o servidor administra a ocorrência de eventos baseados na posição de um ou mais objetos.

Na versão denominada *simple reporting*, o objeto móvel atualiza a localização no servidor toda vez que o sistema de posicionamento determinar uma nova localização para o objeto móvel. O número de mensagens enviadas depende da quantidade de atualizações feitas pelo sensor que está sendo utilizado, podendo chegar a níveis elevados. Já no *time-based reporting*, a informação é transmitida periodicamente, depois de decorrido certo intervalo de tempo  $t$ . A taxa de atualização é fixa e não depende do comportamento do objeto móvel, o qual garante uma certeza temporal, mas não espacial da informação. Se o objeto move-se de forma lenta ou não se move, então uma diferença mínima ou inexistente é observada entre a informação real e a informação contida no servidor. Por outro lado, se o objeto está se movendo de forma rápida, dependendo do intervalo de tempo utilizado, o número de mensagens enviadas pode não ser suficiente para garantir uma elevada precisão da posição.

Em outro tipo de protocolo, classificada como *distance-based reporting*, a fonte envia uma mensagem de atualização toda vez que uma distância geográfica entre a posição atual e a última posição enviada para o servidor superar um valor limite  $d$ . Como esse protocolo envia um número maior de mensagens se o objeto estiver se locomovendo rapidamente e um número menor de mensagens se o mesmo estiver se deslocando de forma lenta ou estiver parado, ele é mais eficiente para objetos que realizam movimentos esporádicos entre períodos de imobilidade. Uma alternativa para tornar os protocolos acima mais eficientes, seria combiná-los.

Enquanto os protocolos do tipo *querying* não podem ser ajustados para diferentes características de mobilidade de objetos móveis, o *reporting* não considera a taxa de

consulta e a precisão requisitadas pelas aplicações. Com uma combinação de protocolos, no qual se integra o *distance-based reporting* e o *cached querying*, ambas características podem ser alcançadas. Similar ao *distance-based reporting*, o cliente deverá enviar uma mensagem de atualização ao servidor para obter uma dada precisão espacial  $d$ . Se por ventura a informação armazenada no servidor não satisfizer ao nível de precisão exigido por uma determinada consulta, então o servidor enviará uma mensagem para a fonte a fim de obter sua localização atual como ocorre no *cached querying*.

### 3.1.3 QueryService - Componente de Consultas

O FRAGIL possui a capacidade de responder a uma variedade de questionamentos com relação ao posicionamento dos objetos que estão registrados, assim um serviço de consultas é suportado através de um componente do framework denominado QueryService e disponibilizado aos usuários através de uma API específica, à qual chamamos ApplicationAPI. A seguir descreveremos as consultas que podem ser realizadas através da ApplicationAPI.

Uma questão trivial neste tipo de sistemas é determinar a posição atual de um objeto móvel. Aqui, utilizamos uma funcionalidade chamada positionQuery, cujo parâmetro é o identificador de um objeto móvel e a resposta é a posição deste no instante atual. Adotamos que a posição de um objeto móvel é dada por um sistema de coordenadas tridimensionais, sendo constituído pela tríade  $x$ ,  $y$  e  $z$ .

Em casos em que a comunicação com objeto móvel estiver indisponível, o usuário poderá indicar se irá receber uma informação nula ou se receberá a última posição atualizada pelo objeto em questão. Uma variação dessa funcionalidade é a possibilidade de se verificar a posição de um objeto em relação a um tempo  $t$  no passado. Assim como na funcionalidade anterior, o usuário poderá definir qual resposta ele deverá obter caso não haja registro da posição para o momento requisitado.

Outro tipo de consulta que pode ser oferecida é aquela em que o cliente informa uma área geográfica como parâmetro e obtém como resposta todos os objetos que estão atualmente inseridos dentro da região especificada. Estendendo esta definição, podemos querer saber quais os objetos que encontravam-se dentro de uma determinada área em um tempo passado  $t$ .

Uma informação bastante valiosa quando estamos monitorando objetos móveis,

seria conhecer a trajetória destes, isto é, obter o conjunto de posições ocupadas pelo objeto em relação a um intervalo de tempo determinado. Devido a uma série de fatores o objeto pode não ter sido monitorado durante todo o intervalo de tempo indicado na consulta. Por esse motivo, a união das trajetórias retornadas pela consulta pode não representar um caminho contínuo percorrido pelo objeto.

Por fim, através de uma funcionalidade de aproximação, os clientes poderão verificar a proximidade de objetos com relação a uma posição específica ou com relação a posição de um outro objeto móvel, podendo ainda definir qual a quantidade de objetos é adequada para os propósitos da consulta.

### 3.1.4 EventService - Componente de Eventos

Outro componente do *framework* é o registro de eventos baseados na posição dos objetos móveis em relação a outros objetos móveis ou em relação a uma área geográfica. Inicialmente, o servidor de aplicações (LBSS) poderá, através da ApplicationAPI, registrar os eventos denominados RegionEvent e DistanceEvent no LS. Ao registrar um RegionEvent o usuário será notificado quando o objeto especificado entrar ou sair de uma área pré-determinada. Já no DistanceEvent a notificação é enviada quando um objeto especificado estiver a uma certa distância de outro objeto móvel.

## 3.2 PersonalTVware

A arquitetura proposta pelo PersonalTVware [SILVA *et al.*, 2010] visa sensibilidade ao contexto com o objetivo de oferecer suporte ao desenvolvimento, de forma modular e flexível, de aplicações para recomendação personalizada baseada em contexto. Esta arquitetura é composta por dois subsistemas: dispositivo do usuário (lado cliente) e provedor de serviços (lado servidor). O subsistema dispositivo do usuário poderá ser implementado em um *set-top box*, um computador portátil ou telefone celular, ou ainda TV Móvel com um *middleware* embarcado, no nosso caso, o Ginga [SOUZA, 2007]. A comunicação entre os subsistemas será sob o canal de retorno (ou interação) de forma bi-direcional por meio de uma interface de serviço baseada em Web Services, conforme mostra a Figura 3.2.

No lado do cliente, ou seja, no dispositivo do usuário, o módulo Gerenciador de Recomendação faz a interface entre as aplicações clientes e os demais módulos da arquite-

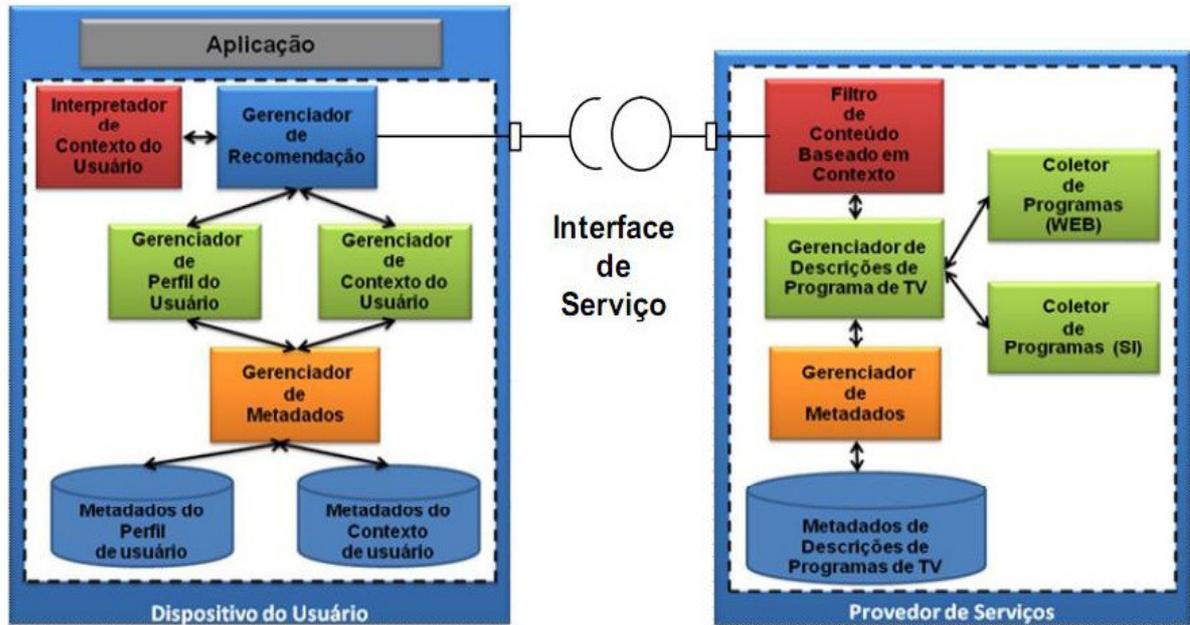


Figura 3.2: Arquitetura do PersonalTVware.

tura, sendo o responsável pelo gerenciamento do processo de recomendação, coordenando os demais módulos do sistema. Se o usuário quer saber quais os programas de TV foram recomendados, pode através de uma requisição ao aplicativo para que execute o processo de recomendação personalizada considerando o contexto. Além disso, a realimentação de relevância também fica a cargo deste módulo, permitindo ao usuário a possibilidade de avaliar dentre os programas de TV que foram recomendados quais considera relevantes ou irrelevantes [BLANCO, 2004]. Assim, nas futuras recomendações o filtro refinará ainda mais, pois as restrições de escolhas impostas pelo filtro podem ser expandidas.

O módulo Gerenciador de Contexto do Usuário é responsável pelo acesso, aquisição de forma implícita e conversões das informações do contexto atual do usuário, o que deverá gerar um histórico de contextos passados. São obtidas informações tais como sua identidade, dia e hora de acesso, localização, e como acessa recomendação, por meio de um dispositivo fixo, móvel ou portátil. Tais informações serão representadas por meio de XML Schemas baseados em um modelo de contexto especificado. O módulo Gerenciador de Perfil do Usuário é responsável pelo acesso e aquisição de forma explícita das informações que constituem o perfil do usuário. Por meio de seus componentes o usuário poderá especificar informações tais como dados pessoais (nome, idade, sexo, ocupação) e preferências (programa de TV, diretor, ator, assuntos, entre outras). As informações do perfil serão descritas de acordo com as especificações de metadados

dos padrões TV-Anytime [Forum, 2007] e MPEG-7 [MPEG-7, 2007], tornando a representação padronizada e estruturada.

Para garantia de privacidade e segurança, o perfil do usuário será armazenado no dispositivo do usuário. O módulo Interpretador de Contexto do Usuário é responsável por inferir preferências implícitas por canais e gêneros de programas de TV a partir da interpretação das informações de contexto atuais e passadas, obtidas por meio do módulo Gerenciador de Contexto do Usuário. A abordagem utilizada para executar a inferência é baseada na utilização de um conjunto de regras. Uma abordagem visa a estrutura de contexto e a outra a recorrência de históricos.

A técnica de raciocínio baseado em regras de [WATTERMAN, 1985] é a utilizada, pois permite definir, de forma flexível estruturas condicionais que refletem as relações entre as dimensões contextuais. Por exemplo, se usuário de identidade 001, requisita recomendação no domingo, às 20h, em sua casa, por meio de um set-top box então deverá ter interesse por programas de TV do canal HDTV e do tipo filme de comédia. Também é possível ter mais de um gênero associado a uma condição de contexto. Assim, um conjunto de regras ou preferências contextuais pode ser definido explicitamente pelo usuário de acordo com as variáveis de contexto: local, dia, horário de interação e o tipo de dispositivo de acesso.

No que se refere à tarefa de inferência de preferências implícitas a partir do histórico de contextos passados, é possível por meio da técnica de raciocínio baseado em caso [LORENZI & RICCI, 2005]. Neste tipo de técnica a preferência para um novo contexto é inferida baseado em contextos (casos) similares ocorridos no passado.

Por conta da limitação dos recursos computacionais dos dispositivos de acesso, o módulo Filtro de Conteúdo Baseado em Contexto está localizado no provedor de serviços. Este módulo é responsável pela filtragem dos programas de TV que provavelmente serão relevantes para o usuário considerando seu contexto atual. O processo de filtragem explora variáveis contextuais (dia, horário), perfil do usuário, suas preferências implícitas inferidas e as descrições dos conteúdos dos programas de TV. A técnica de filtragem de informação empregada tem como base a técnica de Filtragem Baseada em Conteúdo [ZHANG & ZHENG, 2005].

Encontra-se também do lado do servidor, o módulo Gerenciador de Descrições Programas de TV que é responsável pela consulta e inserção de informações referentes aos

programas de TV. Tais informações também são descritas de acordo com as especificações de metadados do padrão TV-Anytime [Forum, 2007]. Os módulos de coleta de programas deverão ser utilizados para capturar informações referentes aos programas de TV de fontes externas como a WEB e do SI (*Service Information*).

O SI é organizado em forma de tabelas que incluem metadados sobre os canais (serviços), programas de TV (eventos), data e horário de exibição, duração, entre outros [LUGMAYR, 2004]. O coletor WEB permite ao administrador do sistema por meio de uma interface WEB, a submissão de metadados referentes aos programas de TV. No caso do coletor SI, é o módulo responsável por extrair os metadados das tabelas SI provenientes de cada emissora de TV, a partir de uma unidade receptora (*set-top box*) do sinal de TV conectada ao sistema. Assim, ambos os módulos são responsáveis pela atualização automática da base de metadados de programas de TV, por meio da submissão dos metadados extraídos ao módulo Gerenciador de Descrições de Programas de TV.

Verifica-se que nos dois subsistemas, o módulo Gerenciador de Metadados fornece suporte ao demais módulos da arquitetura sendo responsável pela recuperação, armazenamento e validação dos metadados. Desta forma, este módulo atua como um mediador entre os demais módulos e a base de dados, fornecendo um conjunto de métodos que permitem a manipulação dos metadados de forma transparente.

### 3.3 Ginga-NCL

A arquitetura modular do Ginga-NCL para dispositivos portáteis, que é dividida em dois subsistemas lógicos: o Núcleo Ginga e a Máquina de Apresentação Ginga-NCL, como ilustra a Figura 3.3. [CRUZ, 2008]

Na sessão 2.2 já trata do Middleware Ginga e sua arquitetura de referência. Nesta sessão o foco é dado às aplicações para dispositivos móveis. No Núcleo Ginga, o módulo Sintonizador é responsável por receber conteúdo de TV Digital para portáteis transmitido por provedores de conteúdo. As aplicações interativas podem chegar ao dispositivo portátil multiplexadas nesse conteúdo, ou por outra interface de rede (por exemplo, canal de dados oferecido pela operadora de celular, conexão Bluetooth ou IEEE 802.11 etc.). No primeiro caso, a aplicação é obtida através de um processamento, realizado pelo módulo Processador de Dados, sobre o conteúdo recebido. No segundo caso,



Figura 3.3: Arquitetura Ginga-NCL para Dispositivos Móveis.

um componente de Transporte foi definido para controlar protocolos e interfaces de rede e atender a demanda da Máquina de Apresentação por conteúdo e aplicações. Para gerenciar o armazenamento das aplicações de TV e o conteúdo que essas aplicações referenciam, o módulo de Persistência foi definido.

Ainda no Núcleo Ginga, o módulo Exibidores é responsável por prover, à Máquina de Apresentação, o decodificador adequado para a apresentação de um conteúdo específico. O módulo Gerenciador Gráfico foi definido para realizar o controle espacial da renderização de objetos de acordo com o especificado pelas aplicações de TV. Para atender às especificações da norma brasileira [TVDT, 2007] e [TVDT, 2008], que define Lua como a linguagem script para NCL, uma máquina Lua é agregada ao Núcleo. Os módulos definidos na arquitetura podem ser atualizados de forma independente. As atualizações podem ser enviadas pelo provedor de conteúdo ou obtidas através do módulo de Transporte. Cabe ao módulo Gerenciador de Atualizações realizar o procedimento de atualização, sem interromper o funcionamento do middleware. Finalmente, um Gerenciador de Contexto foi definido para gerenciar as informações sobre o sistema embarcado no dispositivo e sobre o perfil do usuário telespectador.

Na arquitetura apresentada na Figura 3.3, o Núcleo Ginga é responsável por oferecer os serviços anteriormente mencionados à Máquina de Apresentação Ginga-NCL. A

Máquina de Apresentação Ginga-NCL é um subsistema lógico capaz de iniciar e controlar aplicações NCL. O núcleo da Máquina de Apresentação NCL é o Formatador. Esse módulo é responsável por receber e controlar as aplicações NCL entregues pelo Núcleo Ginga. Ao receber uma aplicação NCL, o Formatador solicita uma tradução das especificações NCL em estruturas de dados adequadas à apresentação das aplicações. Essa tradução é realizada pelo módulo Conversor. O resultado da tradução é então agrupado em uma estrutura de dados denominada Base Privada.

Em seguida, o módulo Escalonador é solicitado, pelo Formatador, para orquestrar a apresentação da aplicação NCL. Para permitir que cada conteúdo seja exibido de forma adequada, o Escalonador solicita que o Gerenciador de Exibidores instancie cada Exibidor de mídia apropriado, de acordo com o tipo de conteúdo a ser exibido em um dado instante. Com o intuito de padronizar a comunicação entre a Máquina de Apresentação e a API de decodificação de conteúdo dos exibidores, um componente, denominado Adaptadores, foi definido na arquitetura.

A exibição de um conteúdo deve ser realizada em regiões do dispositivo, conforme especificado pelo autor da aplicação NCL. O módulo Gerenciador de Leiaute foi definido para fazer a associação do conteúdo, tratado por um exibidor específico, a essas regiões.

O módulo Gerenciador de Bases Privadas é responsável por gerenciar todas as Bases Privadas criadas pelo Formatador, assim como processar os comandos de edição ao vivo mencionados na Seção 2. Ao processar um comando de edição, mudanças sobre as estruturas presentes nas Bases Privadas podem ser realizadas pelo Gerenciador de Bases Privadas [TVDT, 2007]. Finalmente, um Gerenciador de Contexto NCL foi definido para realizar adaptações na apresentação das aplicações NCL, de acordo com as informações providas pelo Núcleo Ginga e/ou pela própria aplicação NCL.

### 3.3.1 O Ambiente Ginga-NCL

Ginga-NCL é a inovação totalmente brasileira do SBTVD. O ambiente tem por base a linguagem NCL (uma aplicação XML) e sua linguagem de *script* Lua. Os ambientes declarativos dos sistemas americano (ACAP-X), europeu (DVB-HTML) e japonês (BML-ARIB) têm por base a linguagem XHTML. XHTML carrega o legado de tecnologias anteriormente desenvolvidas para navegação textual. Em sentido contrário, aplicações para TVD são

usualmente centradas no vídeo. Além disso, o modelo da linguagem XHTML tem o foco no suporte à interação do usuário telespectador. Outros tipos de relacionamentos, como relacionamentos de sincronização espaço-temporal e relacionamentos para definição de alternativas (adaptação de conteúdo e de apresentação), são usualmente definidos através de uma linguagem imperativa, no caso de todos os três sistemas citados na linguagem ECMAScript.

Diferente das linguagens baseadas em XHTML, NCL define uma separação bem demarcada entre o conteúdo e a estrutura de uma aplicação, provendo um controle da ligação entre o conteúdo, sua apresentação e leiaute.

O modelo da linguagem NCL visa um domínio de aplicações mais amplo do que o oferecido pela linguagem XHTML. NCL visa não apenas o suporte declarativo à interação do usuário, mas o sincronismo espacial e temporal em sua forma mais ampla, tratando a interação do usuário como um caso particular. NCL visa também o suporte declarativo a adaptações de conteúdo e de formas de apresentação de conteúdo, o suporte declarativo a múltiplos dispositivos de exibição e a edição/produção da aplicação em tempo de exibição, ou seja, ao vivo. Esses são também os focos da maioria das aplicações para TV digital, o que torna NCL a opção preferencial no desenvolvimento da maioria das aplicações de TVD. Para os poucos casos particulares, como por exemplo, quando a geração dinâmica de conteúdo é necessária, NCL provê o suporte de sua linguagem de *script* Lua. Alternativamente, as APIs da ponte com o Gingga-J podem ser usadas acionando o suporte imperativo oferecido pela linguagem Java já citados no capítulo 2.

Como a NCL tem uma separação mais acurada entre o conteúdo e a estrutura de uma aplicação, ela não define nenhuma mídia por si. Ao contrário, ela define a “cola” que prende as mídias em apresentações multimídia. NCL apenas define como objetos de mídia são estruturados e relacionados, no tempo e no espaço. Como uma linguagem de cola, ela não restringe ou prescreve os tipos de conteúdo dos objetos de mídia. Nesse sentido, podemos ter objetos de imagem, de vídeo, de áudio, de texto, de código imperativo (Xlet e Lua, no SBTVD), entre outros, como objetos de mídia NCL. Quais objetos de mídia têm suporte depende dos exibidores de mídia que estão acoplados ao formatador NCL (na verdade, que têm suporte no Gingga-CC, veja na Figura 3.3). No SBTVD, um desses exibidores é o decodificador/exibidor MPEG-4, implementado em *hardware* no receptor de televisão digital. Dessa forma, o vídeo e o áudio MPEG-4 são tratados como todos os demais objetos de mídia que podem ser relacionados utilizando NCL; em outras

palavras, eles são simplesmente parte de uma aplicação de TVD.

Outro objeto de mídia NCL que deve obrigatoriamente ser suportado pelo Ginga-NCL é o objeto de mídia baseado em XHTML. A NCL não substitui, mas embute documentos (ou objetos) baseados em XHTML. Como acontece com outros objetos de mídia, qual linguagem baseada em XHTML tem suporte em um formatador NCL é uma escolha de implementação e, portanto, depende de qual navegador XHTML, incorporado no formatador NCL (na verdade suportado pelo Ginga-CC), atua como exibidor dessa mídia. Como consequência, é possível ter navegadores BML, DVB-HTML e ACAP-X individualmente embutidos em um exibidor de documento NCL. É possível, ainda, ter todos eles. Assim, aplicações declarativas desenvolvidas para aqueles sistemas também executariam com o suporte oferecido pelo middleware Ginga. Resta mencionar que as Normas ABNT NBR 15606-2 e ABNT NBR 15606-5 definem apenas um conjunto de funcionalidades básicas para XHTML e suas tecnologias derivadas como obrigatórias. A escolha de outras funcionalidades adicionais é opcional.

Voltando nossa atenção para a Figura 3.3, o componente Formatador NCL tem como responsabilidade orquestrar toda a execução de uma aplicação NCL, garantindo que os relacionamentos espaço-temporais definidos pelo autor da aplicação sejam respeitados. A máquina de execução Lua é responsável pelo processamento do código imperativo Lua. Lua é uma linguagem de programação imperativa eficiente, rápida e leve, projetada para estender aplicações. Lua combina uma sintaxe simples para programação imperativa com construções poderosas para descrição de dados, baseada em tabelas associativas e em semântica extensível. Lua é tipada dinamicamente, é interpretada e tem gerenciamento automático de memória, com coleta de lixo incremental. Essas características fazem de Lua uma linguagem ideal para configuração, automação (*scripting*) e prototipagem rápida (geração rápida de aplicações). Lua é uma das linguagens de *script* mais eficientes; muito mais rápida do que ECMAScript, e com um *footprint* de memória bem menor; Lua é, em média, sete vezes mais rápida e com um uso de memória 40 vezes menor. Lua é hoje a linguagem mais importante na área de entretenimento.

O Formatador NCL trata de aplicações recebidas pelo Ginga-CC e depositadas em uma estrutura de dados chamada "base privada". Existe uma base privada por canal de radio-frequência. Cabe ao componente Gerenciador de Bases Privadas a tarefa de receber comandos para ativação e manipulação dessas aplicações. Como anteriormente mencionado, no Ginga-NCL, uma aplicação de TVD pode ser gerada ou modificada ao

vivo (em tempo real), através de comandos de edição. O conjunto de comandos de edição, especificados na Norma ABNT NBR 15606-2, pode ser dividido em três grupos.

O primeiro grupo de comandos é responsável pela ativação e desativação de uma base privada, ou seja, a habilitação de aplicações de um determinado canal de TV. Em uma base privada, aplicações NCL podem ser ativadas, pausadas, retomadas e desativadas, por meio de comandos bem definidos pertencentes ao segundo grupo de comandos. O terceiro grupo define comandos para modificações de uma aplicação ao vivo.

Finalmente o Ginga-NCL oferece suporte a múltiplos dispositivos de entrada e saída. Tal facilidade declarativa - juntamente com os comandos de edição ao vivo, únicos do sistema brasileiro - provê suporte para o grande domínio de aplicações interativas de TVD que se apresenta: as aplicações para as chamadas TV em comunidade (*Community* ou Social TV), em que uma comunidade de usuários cria ao vivo, sobre o conteúdo e as aplicações recebidas, novas aplicações (geração de novos conteúdos e informações personalizados), que são trocadas entre seus membros para exibição em tempo real ou sob demanda.

## 4 ARQUITETURA PROPOSTA DE INTEGRAÇÃO

O objetivo deste trabalho é propor uma abordagem na qual possa promover sensibilidade no cenário da Televisão Digital Móvel. Para isso, combinando os conceitos de sistema baseado em localização (Location Based Service - LBS) usando o Framework FRAGIO [MONTEIRO, 2005], empregando conceitos de sensibilidade ao contexto para TVD extraídos de PersonalTVware [SILVA *et al.*, 2010] e o middleware Ginga com suporte a dispositivos móveis Ginga-NCL [CRUZ, 2008].

### 4.1 A arquitetura

A arquitetura apresentada na Figura 4.1 vem propor um modo de disponibilizar a Televisão Digital Móvel abordada em Ginga-NCL [CRUZ, 2008], adicionando recursos para localização de dispositivos móveis estudados em [MONTEIRO, 2005] e a contextualização do conteúdo transmitido pelo provedor, abordado em [SILVA *et al.*, 2010]. Podendo utilizar como meio de transmissão tanto o sinal de TVD quanto operadora de celular.

A arquitetura se divide em três outras arquiteturas já mencionadas na sessão 3. Nesta sessão abordaremos as funcionalidades destas arquiteturas com a finalidade da integração das mesmas com o objetivo de disponibilizar conteúdo contextualizado para TVD Móvel. Na figura 4.1 podemos visualizar que a arquitetura se baseia em uma estrutura cliente/servidor, ou seja, composto basicamente por dois lados: um provedor de serviços (emissora de TVD) e o dispositivo móvel. Estas são características comuns tanto no FRAGIL quanto no PersonalTVware. Já o Ginga-NCL fica inteiramente embarcado no dispositivo móvel.

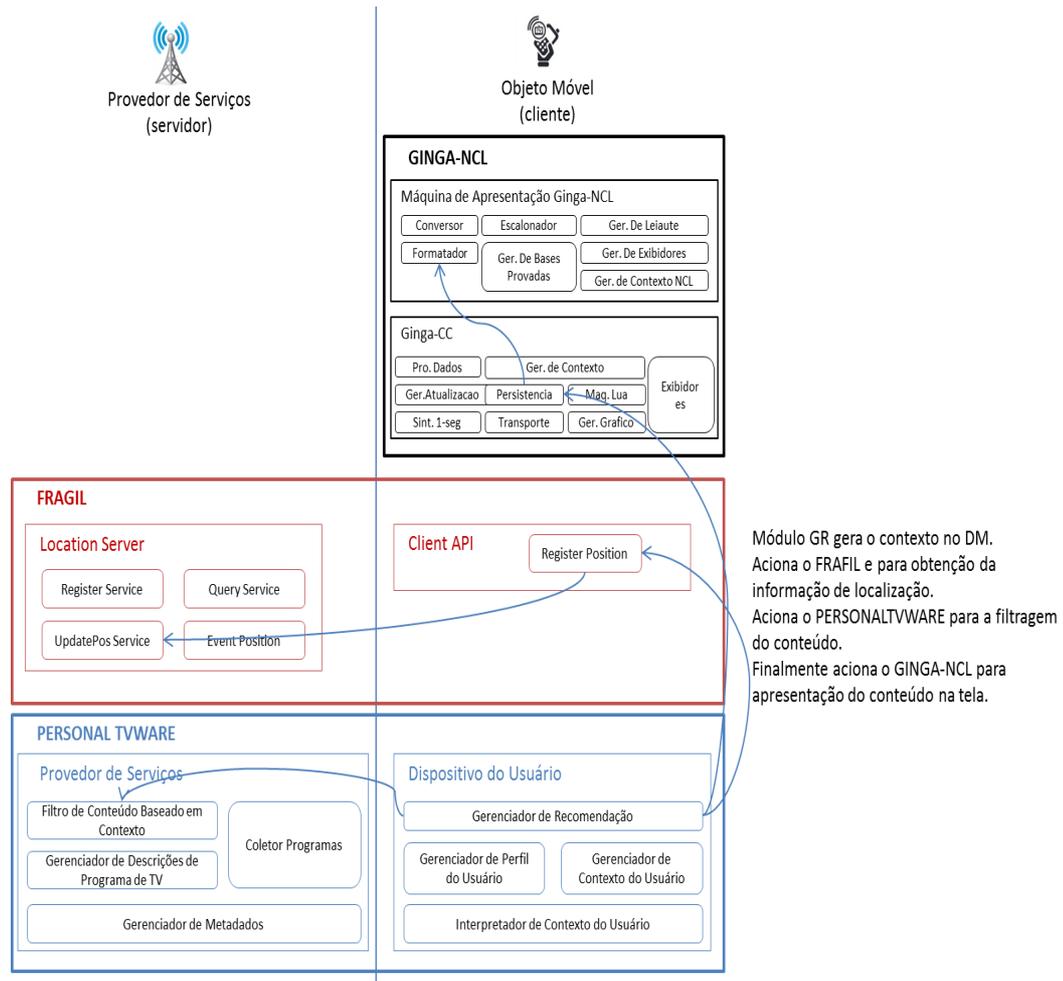


Figura 4.1: Arquitetura Proposta.

### 4.1.1 Cenário de uso

Considerando o seguinte cenário para ilustrar o uso da arquitetura: “*Marcos Pedro, está na fila do supermercado, sábado às 09h, no dia 5/5/12. Ele terá que esperar pelo menos 30min até ser atendido pelo caixa. Para passar o tempo, ele confere em seu celular quais recomendações na programação há para ele naquele momento.*”

Se considerarmos as variáveis implícitas de contexto observadas na situação, destacaremos: Marcos Pedro (quem, identidade), sábado às 09h do 5/5/12 (quando, tempo), no supermercado (onde, localização), na fila (o que, atividade) e usando celular (como, dispositivo móvel). E também variáveis explícitas capturadas das informações que constituem o perfil do usuário, seguem: Marcos Pedro (nome), 23 anos (idade), masculino (sexo), engenheiro de pesca (ocupação), não fumante, bebe ocasionalmente; e suas preferências: esporte nacional e local (programa de TV), campeonato brasileiro de futebol (notícias), esporte olímpico, cinema nacional, cinema americano, notícias locais.

Na Figura 4.2 temos o diagrama de sequência que demonstra como cada módulo age na arquitetura para o cenário descrito acima.

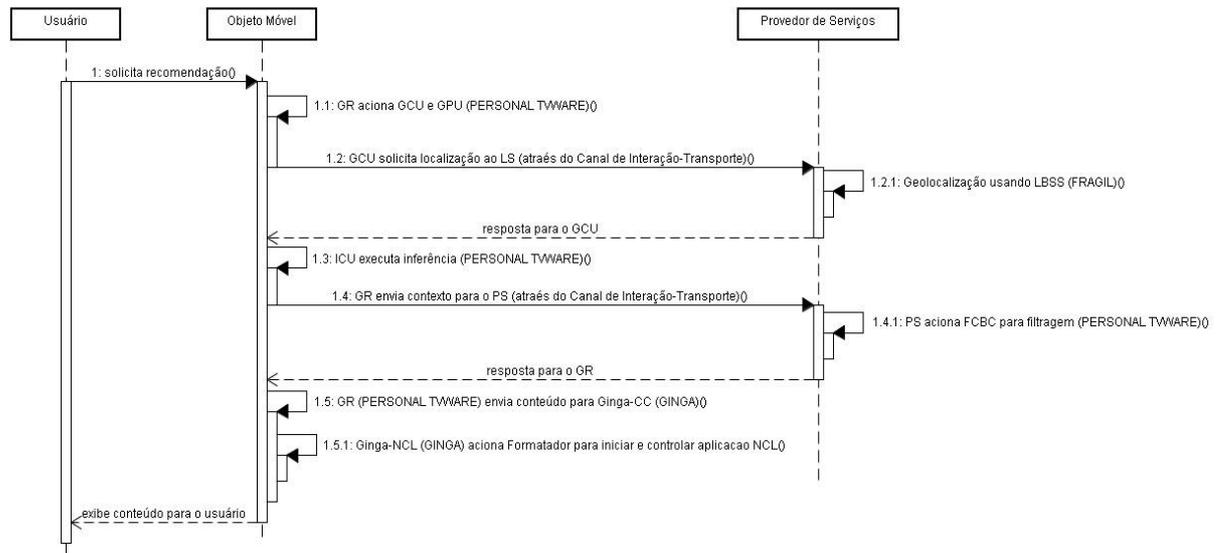


Figura 4.2: Diagrama de Sequência da Arquitetura.

- Usuário solicita recomendações através do Gerenciador de Recomendações (GR);
- O GR aciona o Gerenciador de Contexto do Usuário (GCU), variáveis contextuais;
- O GR aciona o Gerenciador de Perfil do Usuário (GPU), informações de perfil;
- O GR solicita ao Location Server (LS) a informação de localização do usuário;
- Após o Dispositivo Móvel já ter se registrado no LS, este poderá responder a solicitação. Para isso fornece a posição x,y,z e aciona o serviço de geocodificação para tradução das coordenadas num endereço X de linguagem natural. Então envia a resposta para o GCU;
- O GR aciona o Interpretador de Contexto do Usuário para fazer a inferência de preferências por programas considerando o passado e o contexto atual;
- O GR envia o contexto para o Provedor de Serviços (PS);
- O PS aciona a Filtragem de Conteúdo Baseado em Contexto (FCBC) para filtrar baseado nas variáveis contextuais e descrições de conteúdos. Então envia resposta do conteúdo já filtrado para o GR;

- O GR envia o conteúdo para o Ginga-CC. O Ginga-CC armazena as aplicações e conteúdos através do módulo Persistência;
- A Máquina de Apresentação Ginga-NCL aciona o módulo Formatador para iniciar e controlar as aplicações NCL e então exibe o conteúdo recomendado ao usuário.

## 4.2 Descrição dos componentes arquiteturais

Nas sessões seguintes descreveremos quais são e o que fazem os componentes na arquitetura proposta desse trabalho.

### 4.2.1 Personal TVware

Através do Gerenciador de Recomendações o Dispositivo Móvel (DM) pode solicitar recomendações da Emissora de sinal de TV Digital. Para isso fará uso de captura local e remota de informações realizado respectivamente pelos módulos Gerenciador de Contexto de Usuário junto ao Gerenciador de Perfil de Usuário e o módulo Coletor de Programas. Estas capturas servem para combinar informações implícitas e explícitas com conteúdo de TV e web. O módulo Interpretador de Contexto do Usuário faz inferência de preferências implícitas por canais e gêneros de programas de TV Digital, para isso interpreta informações de contextos passados e o contexto atual.

No emissor de serviços, o módulo Filtro de Conteúdo Baseado em Contexto executa a filtragem explorando as variáveis contextuais capturadas no dispositivo e descrições dos programas de TV as quais são organizadas no Gerenciador de Descrição de Programas. A técnica empregada para a filtragem tem como base a técnica de Filtragem Baseada em Conteúdo de [ZHANG & ZHENG, 2005].

O módulo Gerenciador de Metadados fornece suporte ao demais módulos da arquitetura sendo responsável pela recuperação, armazenamento e validação dos metadados. Presente em ambos os lados da arquitetura, este módulo serve uma interface entre os módulos e a base de dados permitindo a manipulação dos metadados de forma transparente. Para que se realize a contextualização é preciso de uma variável contextual fundamental que é a localização. A seguir descrevemos como conseguimos tal informação usando o framework FRAGIL.

### 4.2.2 FRAGIL

Para gerenciar a informação de localização dos Dispositivos Móveis, utilizamos o Location Server. Para que o Location Server fornecer a localização do Dispositivo Móvel, este através do ClientAPI, primeiro precisa se registrar por meio do componente Register Service que administra o cadastro de novos dispositivos a serem monitorados. Para isso, algumas informações devem ser consideradas, tais como: posição inicial, tempo inicial e tipo de política de atualização a ser adotada para otimizar o acesso às informações de localização.

Já registrado, o Dispositivo Móvel passa a ser monitorado e receber atualizações de sua posição através do componente UpdatePosService considerando a política e protocolo escolhido para a atualização. Para consultas aciona-se o componente Query Service que atende a solicitações para determinar informações básicas como a posição atual do dispositivo móvel ou informações de outros dispositivos na região em que o dispositivo se encontra. Nos casos em que o dispositivo se encontra sem comunicação, o usuário poderá indicar se irá receber uma informação nula ou se receberá a última posição atualizada pelo objeto em questão. Com o componente Event Service o usuário poderá se registrar em eventos de região (RegionEvent) e de distância (DistanceEvent) para respectivamente ser notificado de eventos quando entrar ou sair de uma área pré-determinada e notificações de quando o dispositivo estiver a uma certa distância de outro dispositivo móvel.

Agora que o Dispositivo Móvel já se registrou no Location Server e está sendo monitorado e assim fornecer sua localização ao Gerenciador de Recomendações para filtragem de conteúdo, falta apresentar tal conteúdo na tela do Dispositivo Móvel. A seguir mostramos como tal tarefa é feita por meio do Ginga-NCL.

### 4.2.3 Ginga-NCL

O Ginga-CC ou Ginga Comom Core ou Núcleo Ginga através do módulo Sintetizador é capaz de receber o conteúdo da TV Digital para dispositivos portáteis, caso este conteúdo seja multiplexado então o processamento é realizado pelo módulo Processador de Dados, mas se caso seja outra interface de rede (por exemplo, IEEE 802.11) então o módulo de Transporte é acionado para controlar os protocolos e interfaces de rede. Para o armazena-

mento é utilizado o módulo Persistência.

Na Máquina de Apresentação Ginga-NCL o componente Formatador é o módulo principal. É ele quem coordena os outros módulos para a realização das apresentações na tela do dispositivo móvel. O Formatador é responsável por solicitar ao módulo Conversor que traduza as especificações NCL na estrutura adequada para apresentação das aplicações e então agrupar numa estrutura de dados chamada Base de Dados Privada gerenciada pelo módulo Gerenciador de Base Privada.

É papel também do Formatador solicitar ao módulo Escalonador que orquestre a apresentação da aplicação NCL. Para garantir que cada conteúdo seja exibido adequadamente, o Escalonador solicita ao Gerenciador de Exibidores uma instância para cada Exibidor de mídia apropriado a aplicação.

Por fim, o Formatador solicita ao Gerenciador de Leiaute que associe a região da tela já especificada pela aplicação a um exibidor para então o conteúdo ser exibido ao usuário.

## 5 CONSIDERAÇÕES FINAIS

A construção de um mecanismo para recomendação de conteúdo no cenário da TV Digital Móvel sensível ao contexto apresenta diversos desafios os quais destacam-se principalmente a obtenção, modelagem, armazenamento, processamento, transporte e monitoramento do contexto. Entre estes desafios, a obtenção e o processamento do contexto, em especial uma variável contextual fundamental neste cenário: a localização, pois a mobilidade do usuário é fator fundamental.

Neste trabalho apresentamos uma integração de arquiteturas para possibilitar a recomendação de conteúdo considerando a mobilidade do usuário. Embora ainda esteja em fase inicial, podemos destacar os esforços conceituais obtidos para integrar uma arquitetura de Sistema de Localização Geográfica no âmbito da TV Digital. Nesta perspectiva foi proposto o uso conjunto de um LBS (FRAGIL), um Gerenciador de Contexto (PersonalTVware) e uma Máquina de Apresentação NCL (Ginga-NCL) para dar suporte à sensibilidade ao contexto na TV Digital.

Podemos resumir as contribuições da integração em: (i) suporte a localização geográfica, (ii) monitoramento de dispositivos móveis e de seus eventos, (iii) captura local e remota das informações contextuais tanto implícitas quanto explícitas, (iv) processamento das informações contextuais captadas, (v) recomendações baseadas em contexto e (vi) filtragem de conteúdo baseado em contexto.

### 5.1 Sugestões para Trabalhos Futuros

Para trabalhos futuros, sugere-se o desenvolvimento de protótipos dos módulos constituintes da integração entre as arquiteturas, tornando possíveis experimentos num ambiente real contendo vários usuários interagindo com diversos dispositivos móveis em diferentes contextos e solicitando recomendação.

## Referências Bibliográficas

- [ABOWD *et al.*, 1997] ABOWD, G., ATKESON, C., HONG, J., LONG, S., KOOPER, R., & PINKERTON, M. 1997. *Cyberguide: A mobile context-aware tour guide*. Wireless Networks, [S.l.], v.3, n.5, 1997.
- [BECKER, 2009] BECKER, V. 2009. *Concepções e desenvolvimento de aplicações de interativas para televisão digital*. Dissertação de mestrado - UFSC, Florianópolis, 2006.
- [BLANCO, 2004] BLANCO, Y. 2004. *Avatar: Advanced telematic search of audiovisual contents by semantic reasoning*. In Proceedings of the Personalization of Future TV Workshop.
- [BUDZIK & HAMMOND, 2000] BUDZIK, J., & HAMMOND, K. 2000. *User interactions with everyday applications as context for just-in-time information access*. In: INTELLIGENT USER INTERFACES 2000, 2000. Proceedings. . . ACM Press, 2000.
- [CHEN, 2002] CHEN, G. 2002. *A survey of context-aware mobile computing research*. Dartmouth College: Department of Computer Science, 2002.
- [CHEN & KOTZ, 2000] CHEN, G., & KOTZ, D. 2000. *A survey of context-aware mobile computing research*. Dept. of Computer Science, Dartmouth College, 2000.
- [CRUZ, 2008] CRUZ, V. M. 2008. *Ginga-ncl para dispositivos portáteis*. Dissertação de mestrado - PUC-Rio, Departamento de Informática, 2008.
- [DESHPANDE & BORRIELLO, 2002] DESHPANDE, NIKHIL, & BORRIELLO, GAETANO. 2002. *Location-aware computing*. Creating Innovative and Profitable Applications and Services, Intel Research Development, 25 ago. 2002.
- [DEY & ABOWD, 2000] DEY, A., & ABOWD, G. 2000. *Towards a better understanding of context and context-awareness*. Workshop on the what, who, where, when and how of context-awareness at CHI 2000, [S.l.], Abril 2000.

- [DEY *et al.*, 2001] DEY, A., SALBER, D., & ABOWD, G. 2001. *A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications*. Human-Computer Interaction, [S.l.], v.16, 2001.
- [DEY, 2000] DEY, A. K. 2000. *Providing architectural support for building context-aware applications*. [S.l.]: Georgia Institute of Technology, 2000.
- [DIBDIN, 2001] DIBDIN, PETER. 2001. *Where are mobile location based services?* CM316 Multimedia Systems Paper, dec. 2001.
- [FERNANDES *et al.*, 2004] FERNANDES, J., LEMOS, G., & SILVEIRA, G. 2004. *Introdução à televisão digital interativa: Arquitetura, protocolo, padrões e práticas*. JAI-SBC, Salvador, 2004.
- [Forum, 2007] FORUM, TV-ANYTIME. 2007. *Tv-anytime*. Part 3: Metadata, 2007 ETSI TS102 822-3-1 v1.4.1 Technical Specification.
- [GUSTAVSEN, 2002] GUSTAVSEN, R. 2002. *Condor - an application framework for mobility-based context-aware applications*. [S.l.], 2002.
- [HIGHTOWER & BORRIELLO, 2001] HIGHTOWER, JEFFREY, & BORRIELLO, GAETANO. 2001. *A survey and taxonomy of location systems for ubiquitous computing*. Computer, vol. 34, n° 8, pp. 57-66, IEEE Computer Society Press, ago. 2001.
- [HOFER *et al.*, 2002] HOFER, T., SCHWINGER, W., PICHLER, M., LEONHARTSBERGER, G., & ALTMANN, J. 2002. *Context-awareness on mobile devices - the hydrogen approach*. [S.l.], 2002.
- [HULL *et al.*, 1997] HULL, R., NEAVES, P., & BEDFORD-ROBERTS, J. 1997. *Towards situated computing*. INTERNATIONAL SYMPOSIUM ON WEARABLE COMPUTERS, 1997.
- [IDEC, 2007] IDEC, INSTITUTO BRASILEIRO DE DEFESA DO CONSUMIDOR. 2007.
- [LEONHARDI & ROTHERMEL, 2002] LEONHARDI, A., & ROTHERMEL, K. 2002. *Architecture of a large scale location service*. Proc. of the 22nd Int.Conf. on Distributed Computing Systems (ICDCS), Vienna, Austria, 2002.

- [LORENZI & RICCI, 2005] LORENZI, F., & RICCI, F. 2005. *Case-based recommender systems*. In: John Wang. (Org.). *The Encyclopedia of Data Warehousing and Mining*. I ed. Hershey: Idea Group Reference, pp 124-128.
- [LUGMAYR, 2004] LUGMAYR, A. 2004. *Digital interactive tv and metadata: Future broadcast multimedia*. Springer.
- [MAPINFO, 2002] MAPINFO. 2002. *Mobile location services*. Disponível em: <[www.mobileinfo.com/wireless/img/new](http://www.mobileinfo.com/wireless/img/new)>
- [MIRANDA, 2002] MIRANDA, RODRIGO VILAR; BAPTISTA, CLÁUDIO S.; ALMEIDA RODRIGO R.; CATÃO BRUNO; PAZINATTO EDER. 2002. *Igis: um framework para sistemas de informações geográficas em n-camadas usando um sgbd objeto-relacional*. SBC GeoInfo 2002, Caxambu, Brasil, dez. 2002.
- [MONTEIRO, 2005] MONTEIRO, ERICH FARIAS. 2005. *Um framework para o gerenciamento da informação de localização*. Dissertação - Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão UFMA, São Luis-MA, 2005.
- [MORRIS & SMITH-CHAIGNEAU, 2005] MORRIS, S., & SMITH-CHAIGNEAU, A. 2005. *Interactive tv standards: A guide to mhp, ocap, and javatv*. Focal Press, 2005.
- [MPEG-7, 2007] MPEG-7. 2007. *Iso/iec 15948-1*. MPEG-7 Overview.
- [OGC, 2004] OGC, OPEN GIS CONSORTIUM REFERENCE. 2004. *The impact and penetration of location-based services*. OpenGIS Location Services (OpenLS): Core Services. 16 Jan. 2004. Disponível em: <<http://www.opengis.org/>>.
- [PREKOP & BURNETT, 2003] PREKOP, P., & BURNETT, M. 2003. *Activities, context and ubiquitous computing*. Special Issue on Ubiquitous Computing Computer Communications, [S.l.], v.26, n.11, 2003.
- [RYAN *et al.*, 1997] RYAN, N., PASCOE, J., & MORSE, D. 1997. *Enhanced reality fieldwork: the contextaware archaeological assistant*. Computer Applications in Archaeology, [S.l.], 1997.
- [SCHILIT & THEIMER, 1994] SCHILIT, B., & THEIMER, M. 1994. *Disseminating active map information to mobile hosts*. IEEE Network, [S.l.], 1994.

- [SHIODE, 2002] SHIODE, NARUSHIGE. 2002. *The impact and penetration of location-based services*. [Centre for Advanced Spatial Analysis, University College London, May 2002. Disponível em: <<http://eprints.ucl.ac.uk/archive/00000246/01/Paper50.pdf>>.
- [SILVA *et al.*, 2010] SILVA, FÁBIO SANTOS, ALVES, LUIZ GUSTAVO PACOLA, & BRESSAN, GRAÇA. 2010. *Personalware: Uma proposta de arquitetura sensível ao contexto para suporte a recomendação personalizada de conteúdo no cenário da tv digital interativa*. Laboratório de Arquitetura e Redes de Computadores (LARC) Escola Politécnica da Universidade de São Paulo (EPUSP).
- [SOARES, 2009] SOARES, LUIZ FERNANDO GOMES. 2009. *Tv interativa se faz com ginga*. Departamento de Informática - Universidade Católica do Rio de Janeiro, 2009.
- [SOUZA, 2007] SOUZA, F. 2007. *Ginga-j: The procedural middleware for the brazilian digital tv system*. Journal of the Brazilian Computer Society, v12, n4.
- [TAVARES, 2001] TAVARES, WALKYRIA M. LEITÃO. 2001. *Implantação da televisão digital no brasil*. Consultora Legislativa da Área XIV Comunicação Social, Telecomunicações, Sistema Postal, Ciência e Tecnologia.
- [TVDT, 2007] TVDT, TELEVISÃO DIGITAL TERRESTRE. 2007. *Codificação de dados e especificações de transmissão para radiodifusão digital parte 2: Ginga-ncl para receptores fixos e móveis linguagem de aplicação xml para codificação de aplicações*. Setembro 2007.
- [TVDT, 2008] TVDT, TELEVISÃO DIGITAL TERRESTRE. 2008. *Codificação de dados e especificações de transmissão para radiodifusão digital parte 5: Ginga-ncl para receptores portáteis linguagem de aplicação xml para codificação de aplicações*. Março de 2008.
- [WANT *et al.*, 1992] WANT, R., HOPPER, A., FALCÃO, V., & GIBBONS, J. 1992. *The active badge location system*. ACM Transactions on Information Systems, [S.1.], 1992.
- [WATTERMAN, 1985] WATTERMAN, D. A. 1985. *A guide to expert systems*. Addison Wesley.
- [WEISER, 1991] WEISER, M. 1991. *The computer for the twenty-first century*. Scientific American, pp. 94-10, September, 1991.

- [YAMIN, 2004] YAMIN, A. 2004. *Arquitetura para um ambiente de grade computacional direcionado às aplicações distribuídas, móveis e conscientes do contexto da computação pervasiva*. Tese (Doutorado em Ciência da Computação) Instituto de Informática, UFRGS, Porto Alegre, RS.
- [ZHANG & ZHENG, 2005] ZHANG, H., & ZHENG, S. 2005. *Personalized tv program recommendation based on tv-anytime metadata*. In: Proc. of the Ninth International Symposium on Consumer Electronics, (ISCE 2005).