

2 Estado da Arte

Este capítulo tem como objetivo apresentar, de forma sucinta, o estado atual da TV Digital para dispositivos portáteis. A Seção 2.1 trata da TV Digital e dos dispositivos portáteis. A Seção 2.2 aborda os middlewares declarativos existentes, dando enfoque àqueles mais adequados ao contexto da TV Digital para dispositivos portáteis. Finalmente, a Seção 2.3 tece algumas considerações acerca dos sistemas operacionais e das plataformas de desenvolvimento existentes no escopo dos dispositivos portáteis. Um estudo mais detalhado sobre esses pontos pode ser encontrado em (Cruz, Moreno e Soares, 2008).

2.1. A TV Digital e os Dispositivos Portáteis

A TV Digital possibilita uma grande melhora na qualidade da imagem e do som dos programas ou conteúdos exibidos. Isso ocorre, dentre outros fatores, por conta da natureza da transmissão digital, que possui métodos muito mais precisos de recuperação da informação recebida se comparados aos da transmissão analógica. Dessa forma, as perdas na qualidade da imagem geradas por imprecisões ocorridas na transmissão são muito mais fáceis de serem compensadas em um sistema de TV Digital. Uma outra vantagem é que, sendo de natureza digital, o conteúdo oferecido pode ser comprimido, reduzindo a taxa de bits gerada. Isso possibilita a sua transmissão em uma banda de frequência menor do que a utilizada pela TV analógica. O fato do conteúdo ser de natureza digital também viabiliza seu armazenamento e o uso de funções de PVR (Personal Video Recoder), como pause e rewind, mesmo em programas sendo exibidos ao vivo. Com relação à redução de banda necessária, o mesmo canal de frequência hoje utilizado por uma transmissão analógica será capaz de transmitir mais de um fluxo de áudio e de vídeo, além dos fluxos de dados. O vídeo transmitido poderá possuir a definição SDTV (Standard Definition TV), conforme utilizado na transmissão analógica, ou ainda HDTV (High Definition TV), que oferece uma qualidade de imagem superior.

A transmissão de outros dados, além do áudio principal e do vídeo principal, permite o envio de aplicativos que possibilitam ao telespectador interagir com o programa que está sendo exibido. Também é possível oferecer conteúdos personalizados e mais adequados às necessidades do telespectador, o que pode contribuir, ainda, para uma mudança no modelo de negócios hoje usado.

“A TV Digital pode ser provida trazendo-se tecnologias computacionais para o âmbito da TV ou levando as tecnologias de TV para o ambiente computacional.” (Cruz, Moreno e Soares, 2008: 2).

No primeiro caso, enquadram-se, de maneira geral, as transmissões de TV Digital por radiodifusão terrestre, cabo ou satélite. Este modelo permite a disponibilização da TV Digital com interatividade e personalização do conteúdo. Se for adicionada alguma tecnologia de canal de retorno, também será possível oferecer serviços de conteúdo sobre demanda. Esse tipo de serviço permite ao telespectador escolher um programa de um acervo fornecido pela emissora para ser assistido na hora que lhe for mais conveniente, e também torna viável à emissora obter um melhor conhecimento dos seus usuários.

No caso em que a tecnologia de TV é levada ao ambiente computacional, temos a IPTV (TV over IP), onde o conteúdo televisivo é distribuído através do protocolo da Internet IP (Internet Protocol). O uso da IPTV é feito, normalmente, em redes fechadas, como em TVs por assinatura, e encontra-se mais próximo do modelo da Internet. Com o uso dessa tecnologia é possível disponibilizar a TV Digital com interatividade, personalização e serviços de conteúdo sobre demanda. Por outro lado, serviços de broadcast, normais em transmissões de TV, não são providos naturalmente com o uso do protocolo IP, onde serviços bidirecionais de unicast são mais comuns. Isso força a utilização da técnica de multicasting, pela qual a transmissão é feita de um para n endereços IP's registrados em um grupo, chamado de grupo multicast. Um problema associado a isso seria a complexidade da rede envolvida, que pode acabar aumentando muito. Uma maior discussão sobre as possibilidades da TV Digital é apresentada em (Cruz, Moreno e Soares, 2008: 1).

No caso da aplicação da TV Digital nos dispositivos portáteis, é preciso ficar atento às características diferenciais encontradas nesses aparelhos, das quais é possível citar:

- Uso de bateria, que exige um consumo moderado de energia. Aplicações nesse escopo podem ser sumariamente terminadas, e precisam se adequar ao fato. Um outro ponto é que o consumo de energia gerado pela aplicação precisa ser baixo;

- Limite de processamento e de memória;
- Mobilidade, com processo de handoff;
- Tamanho de tela pequeno;
- A conversa, no caso do celular, como funcionalidade principal. Qualquer outra atividade fica em segundo plano;
- Teclado limitado;
- Limite maior de banda. ” (Cruz, Moreno e Soares, 2008: 4)

Os sistemas de TV Digital e seus middlewares precisam se adaptar a essas características a fim de possibilitarem a recepção dos conteúdos digitais nos dispositivos portáteis. Entretanto, os esforços nesse sentido iniciaram-se há pouco tempo e ainda não são muitos. Um exemplo é o caso do Japão, país avançado em termos de TV Digital, que começou a migração desse serviço para os dispositivos portáteis apenas em meados de 2006.

2.2. Middlewares

“Middleware é uma camada intermediária entre o hardware/SO e as aplicações” (Cruz, Moreno e Soares, 2008: 18) que tem por finalidade facilitar o desenvolvimento das suas aplicações e torná-las independentes da plataforma de hardware e do sistema operacional do aparelho receptor. (Cruz, Moreno e Soares, 2008), no Capítulo 6, estuda vários sistemas de TV Digital encontrados na literatura e os seus respectivos middlewares. Foi feita, ainda, uma análise comparativa entre os vários middlewares, com o objetivo de determinar quais seriam as melhores opções para o escopo dos dispositivos portáteis. Esta seção apresenta, de forma resumida, esse estudo e as suas conclusões.

Em IPTV, não existe qualquer padrão de middleware definido. Hoje, o que mais se vê são soluções proprietárias, altamente especializadas e, freqüentemente, completamente fechadas. No que diz respeito ao uso dessa tecnologia em dispositivos portáteis, existe menos definição ainda. As soluções proprietárias (Cruz, Moreno e Soares, 2008: 52-63) fazem nenhuma ou pouca menção ao uso nesses dispositivos. Mesmo a pequena movimentação de padronização existente na área de IPTV não chega a tocar nesse assunto com profundidade. Sendo assim, não parece existir, em IPTV, qualquer solução interessante para a disponibilização da TV Digital em dispositivos portáteis.

Por outro lado, a TV Digital por radiodifusão terrestre, cabo ou satélite possui padrões de sistemas de TV Digital abertos e implementados, que são apresentados, com os seus respectivos middlewares, em (Cruz, Moreno e

Soares, 2008: 18-51). Os mais relevantes são apresentados aqui de forma resumida, ao mesmo tempo em que é feita a avaliação de seus usos nos dispositivos portáteis.

O MHP (Multimedia Home Platform), desenvolvido pelo fórum DVB (Digital Video Broadcast), não foi desenvolvido para o ambiente dos dispositivos portáteis. Sua especificação baseia-se no J2SE (Java 2 Standard Edition), que é uma plataforma de desenvolvimento Java feita para terminais fixos. Como descrito em (Cruz, Moreno e Soares, 2008: 12-16), J2SE não é adequado para ser usado em dispositivos portáteis com limitações de recursos. Além disso, o próprio DVB define um padrão de transmissão de TV Digital para esse tipo de aparelho, o DVB-H (Digital Video Broadcast - Handheld), que não determina o uso do MHP. Esse middleware parece ser, portanto, inadequado para o escopo dos dispositivos portáteis. Mais detalhes acerca deste middleware podem ser encontrados em (Cruz, Moreno e Soares, 2008: 18-24).

O sistema de TV Digital ACAP (Advanced Common Application Platform), criado pelo comitê internacional ATSC (Advanced Television Systems Committee), não é apropriado para a transmissão de TV Digital em dispositivos portáteis por conta do seu padrão de modulação, o 8-VSB. Como é detalhado em (Digital TV Facts, 2006), o 8-VSB não atende os requisitos mínimos necessários para a transmissão adequada aos dispositivos portáteis, que é feita sem o uso de cabos. O middleware do sistema ACAP não é implementado em nenhum aparelho desse tipo. Mais informações desse sistema de TV Digital podem ser encontradas em (Cruz, Moreno e Soares, 2008: 30).

O middleware DIMS (Dynamic and Interactive Multimedia Scenes), desenvolvido por uma organização internacional de padronização na área de telecomunicações, denominada 3GPP (3rd Generation Partnership Project), foi especificado para ser usado no padrão de transmissão do sistema de TV Digital MBMS (Mobile Broadcast/Multicast Service). Esse middleware foi desenvolvido especificamente para os dispositivos portáteis. Apesar disso, a sua especificação ainda se encontra em fase de “drafting” e com pouco conteúdo definido, sendo que boa parte dele ainda é muito superficial. Dessa forma, o DIMS também pode ser descartado, hoje, como uma opção de middleware para dispositivos portáteis. Um estudo mais aprofundado desse middleware e do seu respectivo sistema de TV Digital é encontrado em (Cruz, Moreno e Soares, 2008: 44-46).

O LAsER (Lightweight Application Scene Representation) é um padrão de middleware da ISO (International Organization for Standardization) desenvolvido com o objetivo de substituir o MPEG4 BIFS (Binary Format for Scene) que não

se mostra adequado para o ambiente dos dispositivos portáteis. LAsER foi desenvolvido especificamente para esse tipo de aparelho e é um superset do SVG Tiny (Scalable Vector Graphics - Tiny), que, por sua vez, é uma linguagem declarativa do W3C, desenvolvida para dispositivos com limitações de recursos. A especificação LAsER define, além da linguagem declarativa, dois tipos de linguagens de script: o application/ecmascript e o application/laserscript. O primeiro deriva do SGV, que define uma linguagem de script baseada no ECMAScript, e o segundo é usado em aplicações LAsER a fim de executar ações de edição ao vivo. Mais detalhes sobre esse middleware podem ser conferidos em (Cruz, Moreno e Soares, 2008: 40-44). Apesar de se apresentar, aparentemente, como um bom middleware de TV Digital para dispositivos portáteis, o LAsER não possui qualquer implementação sua sendo usada ou mesmo de referência, além da sua especificação ainda não estar totalmente finalizada.

HisTV é um padrão de middleware desenvolvido especificamente para dispositivos portáteis que usam o DVB-H na recepção de conteúdo. Foi criado pela GMIT, uma empresa de consultoria relacionada ao padrão DVB-H, mas é, atualmente, suportado por outras instituições. A sua especificação já está bem adiantada e já existem alguns produtos da Siemens e da Benq com sua implementação, como pode ser visto em (Cruz, Moreno e Soares, 2008: 40).

O desenvolvimento de uma aplicação HisTV deve ser feito através de um processo de três etapas: “a construção do layout da tela, o projeto da máquina de estados e a produção dos objetos.” (Cruz, Moreno e Soares, 2008: 37). A construção do layout deve ser feita com o uso de objetos espaciais predispostos na tela, os ifields. São dois os tipos de layout: o landscape (320x204) e o portrait (240x320). A aplicação deve ser desenvolvida considerando-se ambos. Pode-se perceber, portanto, que a arrumação dos objetos na tela é limitada aos ifields disponíveis e aos modos de layout.

O próximo passo no desenvolvimento de uma aplicação é a construção de uma máquina de estados que será usada na apresentação da aplicação HisTV. Essa máquina deve ser concebida com cuidado, pois o procedimento é bem complexo.

A produção do conteúdo consiste na edição dos nomes dos arquivos que representam os objetos de mídia, levando-se em conta os ifields e os estados da máquina de estados relacionados a esses objetos. Esse procedimento é, aparentemente, muito complexo, confuso e trabalhoso. A especificação cita, entretanto, a existência de um script perl que gera automaticamente os arquivos

corretamente nomeados. Isso é feito recebendo como entrada um arquivo no formato .csv, contendo informações sobre a máquina de estados, as mídias e os ifields. O uso desse script, entretanto, não é muito claro, assim como também não é claro como o arquivo no formato .csv deve ser criado. Detalhes em minúcias sobre o HisTV podem ser vistos em (Cruz, Moreno e Soares, 2008: 36-40).

O HisTV tem como vantagem o fato de ter sido especificado para o ambiente dos dispositivos portáteis e de já possuir implementações da sua especificação executando nesse tipo de aparelho. Entretanto, o seu domínio de aplicações é pequeno por conta da sua interface fixa. Comparado aos outros middlewares que serão apresentados, o seu maior problema é, contudo, o fato da autoria das suas aplicações ser muito complicada, o que certamente inviabiliza a criação de aplicações por pessoas leigas.

A BML (Broadcast Markup Language) é parte integrante do middleware do padrão de TV Digital Japonês, o ISDB-T. A linguagem baseia-se, com algumas adaptações e adições, na XHTML 1.0 e possui uma curva de aprendizagem pequena. A BML possui também elementos imperativos especificados com uma linguagem de script baseada no ECMAScript. O foco do middleware BML é na apresentação do documento e na interação com o usuário. Nas aplicações de TV Digital, entretanto, o sincronismo e a adaptabilidade são fatores tão ou mais importantes, conforme (Moreno, 2006). A BML é adequada à transmissão para dispositivos fixos ou portáteis, definindo perfis da linguagem que são mais adequados a cada um dos dois casos. No Japão, já existem implementações comerciais desse middleware para terminais fixos e dispositivos portáteis.

A BML é o único middleware apresentado até agora que possui especificação finalizada com implementações nos dispositivos portáteis, e que é simples o suficiente de ser usada. Em (Cruz, Moreno e Soares, 2008: 30-36), podem ser encontradas mais informações sobre a BML.

O Ginga-NCL faz parte do Sistema Brasileiro de TV Digital (SBTVD). Esse sistema define dois ambientes para seu middleware: um ambiente declarativo (Ginga-NCL) e um ambiente imperativo (Ginga-J). Esta dissertação se concentra no Ginga-NCL, por ser o único ambiente obrigatório no perfil portátil. A linguagem usada por esse middleware é a NCL (Nested Context Language), que define dois perfis: um avançado, o Enhanced Language Profile, e um outro mais básico, o Basic Language Profile. A linguagem de script especificada para a NCL é a linguagem Lua. O Ginga-NCL possui uma curva de aprendizagem mais alta que a BML, mas, quando aprendida, é mais simples de ser usada. Ela se

distingue da BML especialmente por ter o seu foco no sincronismo, em sua forma geral, e na adaptabilidade. Em (Cruz, Moreno e Soares, 2008: 25-28), podem ser encontradas mais informações sobre o Ginga-NCL.

Dentre os middlewares apresentados, a BML e o Ginga-NCL parecem ser, atualmente, os mais relevantes para o escopo dos dispositivos portáteis. (Cruz, Moreno e Soares, 2008: 62-64) oferece mais detalhes sobre essa conclusão, bem como apresenta uma análise comparativa entre os middlewares aqui resumidos.

Ainda que se apresente como uma boa opção de middleware para os dispositivos portáteis, faltava ao Ginga-NCL uma implementação de referência para esse tipo de dispositivo que demonstrasse o seu uso na prática, o que foi realizado e está sendo apresentado nesta dissertação.

2.3. Sistemas Operacionais e suas Plataformas

No escopo dos dispositivos portáteis, encontramos Sistemas Operacionais feitos justamente para atender as características diferenciais desses aparelhos. Um desses sistemas teve que ser escolhido para a realização da implementação de referência do Ginga-NCL proposta nesta dissertação. Um estudo desses Sistemas Operacionais e das suas plataformas de desenvolvimento foi feito e apresentado em (Cruz, Moreno e Soares, 2008: 5-18) com o objetivo de levantar todas as opções existentes e permitir a escolha de uma delas. Nesse estudo, deu-se preferência a sistemas com plataforma aberta, estáveis, de baixo custo, com boa quantidade de dispositivos disponíveis para teste no mercado, que sejam usados hoje e que possam representar uma opção viável de uso no mercado futuro. Esta seção apresenta um resumo desse estudo realizado, onde foram analisados cinco sistemas operacionais: Symbian, Linux, Windows, PalmOS e BlackBerry.

O BlackBerry possui pouca participação no mercado, não apresentando tendências de crescimento. É fechado e não é distribuído em dispositivos que não tenham sido desenvolvidos pela sua fabricante, a RIM (Research In Motion). Sendo assim, o BlackBerry foi desconsiderado para uma implementação de referência do Ginga-NCL. O mesmo pode-se dizer do PalmOS, que não possui nenhuma das suas versões sendo substancialmente utilizada. Ao invés disso, a Palm, criadora do PalmOS, tem investido na criação de um novo sistema, o

ACCESS Linux Platform (ACL), baseado no Linux, com o objetivo de tentar obter uma participação maior no mercado. Isso, entretanto, acaba por exaltar mais o valor da plataforma Linux em detrimento do sistema PalmOS propriamente dito.

O Windows Mobile é baseado na conhecida e muito usada plataforma Windows. Uma grande vantagem desse sistema é que o desenvolvimento das suas aplicações pode ser feito com as mesmas ferramentas usadas na versão do Windows para desktop. Por outro lado, os mesmos problemas encontrados na versão desktop, como reinicializações de sistema e vazamentos de memória, também são encontrados no Windows Mobile. Um outro problema são os custos elevados atrelados à plataforma, discutidos com mais detalhes em (Cruz, Moreno e Soares, 2008: 6-7). Um último problema é a bem conhecida existência de vírus na plataforma Windows.

Sistemas Linux para dispositivos portáteis têm surgido por conta do crescimento desse mercado. “O Linux é um sistema flexível que oferece uma grande diversidade de métodos de instalação e, por conta disso, pode funcionar em muitos sistemas heterogêneos. Uma grande vantagem do seu uso nos dispositivos portáteis é que, se for configurado corretamente, o consumo de recursos pode ser reduzido em muito.” (Cruz, Moreno e Soares, 2008: 7). Outra vantagem é o porte de uma aplicação de uma plataforma Linux fixa para uma portátil — exige-se, aparentemente, apenas que essa última possuísse a sua disposição todas as bibliotecas requeridas pela aplicação.

Apesar dessas vantagens, a configuração desse sistema é, normalmente, uma tarefa complicada. Um outro problema reside nas distribuições mais estáveis do Linux, que para dispositivos portáteis são pagas. Isso adiciona um problema de custo. Entretanto, o Linux possui uma plataforma aberta e a maioria das suas ferramentas de desenvolvimento pode ser usada sem qualquer custo.

O Linux foi um forte candidato para o desenvolvimento do middleware de TV Digital proposto neste estudo, isso por contar com uma plataforma aberta, robustez, flexibilidade, baixo custo e perspectiva de crescimento no segmento dos dispositivos portáteis. O desenvolvimento do middleware ainda seria favorecido pelo fato da implementação de referência para receptores fixos ter sido desenvolvida para esse sistema operacional. Entretanto, devido a pouca quantidade de distribuições existentes e estáveis, a falta de informação acerca das configurações exigidas e das bibliotecas disponíveis, além dos poucos dispositivos que vem com o sistema instalado, a sua utilização foi descartada.

O Symbian é um sistema operacional feito especificamente para dispositivos portáteis. É robusto e se encontra no mercado desde 1998. Foi feito

com o objetivo de ser econômico no consumo de recursos, em especial a memória e a bateria. É, também, o mais usado, possui plataforma aberta e as aplicações podem ser desenvolvidas em uma série de tecnologias diferentes, como Symbian C++, Java e Flash. Uma lista completa dessas tecnologias pode ser encontrada em (Cruz, Moreno e Soares, 2008: 5). Symbian também é o único sistema que oferece uma versão, a 9.5, com suporte à TV Digital. Por ser muito utilizada, a plataforma em questão está sujeita a vírus, mas, em razão disso, novas versões oferecendo mais segurança já foram lançadas e já são utilizadas. Embora tenha muitas vantagens, os altos custos atrelados à plataforma constituem um problema. O sistema em si não é gratuito e as ferramentas de desenvolvimento gratuitas possuem recursos limitados.

O Symbian possui uma vasta comunidade, muitas opções de desenvolvimento, muitos dispositivos no mercado com o sistema instalado e é o único que possui uma versão para TV Digital. A sua ferramenta de desenvolvimento gratuita, apesar de apresentar limitações, é estável, e a plataforma como um todo parece ser robusta. Apesar dos custos associados, o Symbian foi considerado a melhor opção para uma primeira implementação do Ginga-NCL para dispositivos portáteis. Em (Cruz, Moreno e Soares, 2008: 8-9), essa conclusão é apresentada juntamente com uma análise mais detalhada de todos os sistemas aqui apresentados.

Uma vez tendo escolhido o sistema, é necessário optar por uma das suas plataformas de desenvolvimento. Um estudo abrangente foi feito em (Cruz, Moreno e Soares, 2008: 9-18) sobre as duas principais plataformas do Symbian: a plataforma nativa do Symbian OS e a plataforma JavaME (Java Micro ou Mobile Edition).

Na plataforma nativa do Symbian OS, a linguagem Symbian C++, que é baseada na linguagem C++, é usada no desenvolvimento das aplicações. Symbian C++ é uma linguagem complexa, mas ela permite ao programador ter mais controle sobre o código da sua aplicação, facilitando, com isso, a criação de otimizações mais refinadas. O poder de expressão da linguagem Symbian C++ também é muito elevado, o que permite ao programador desenvolver tanto aplicações simples quanto complexas (Symbian Ltd, 2007a). Uma outra vantagem a ser considerada é que realizar o desenvolvimento na plataforma nativa do Symbian OS implica na possibilidade do uso direto de API's nativas do sistemas, que são muito otimizadas. O desenvolvimento nessa plataforma também permite ao programador ter acesso completo aos recursos de hardware. Todos esses são fatores importantes e requeridos na implementação de um

middleware de TV Digital. Maiores detalhes sobre a plataforma nativa do sistema operacional Symbian são discutidos em (Cruz, Moreno e Soares, 2008: 10-12).

A plataforma JavaME (Micro ou Mobile Edition), por sua vez, é um subconjunto da J2SE (Java Technology, 2007) e foi desenvolvida para atender aos requisitos dos dispositivos portáteis e móveis de pequeno a grande porte. A plataforma JavaME oferece uma linguagem simples de desenvolvimento que é bem conhecida, a Java, e oferece uma portabilidade que, na prática, não existe. A linguagem de programação Java é simples de ser usada, mas, por conta disso, ela remove parte do controle que o programador tem sobre o código da sua aplicação, impossibilitando-o de realizar otimizações mais refinadas. Entretanto, o maior fator negativo contra a JavaME é a impossibilidade de acesso aos recursos de hardware por meio da sua linguagem. Toda funcionalidade existente no dispositivo deve ser acessada através de uma API JavaME, que, quando não existe, inviabiliza o uso da funcionalidade em questão. No que diz respeito à TV Digital, a comunidade Java ainda está preparando a JSR 272, uma API Java que oferecerá suporte às funcionalidades de TV Digital. Quando a especificação em questão estiver terminada, novos aparelhos com a sua API precisarão ser lançados antes que novas aplicações que façam uso dessa API possam ser desenvolvidas. Maiores detalhes sobre a plataforma JavaME são discutidos em (Cruz, Moreno e Soares, 2008: 12-16).

O que se pôde concluir é que a plataforma nativa do Symbian OS reúne todas as características necessárias para o desenvolvimento do middleware Ginga-NCL para dispositivos portáteis, enquanto que as limitações do JavaME inviabilizam esse projeto. Portanto, para o objetivo proposto, optou-se pela plataforma de desenvolvimento do Symbian OS. Essa resolução e uma análise mais detalhada dessas duas plataformas são apresentadas em (Cruz, Moreno e Soares, 2008: 17-18).

Nessa plataforma, é preciso, ainda, escolher uma UI (User Interface) de desenvolvimento. As UIs determinam as características específicas do sistema Symbian, e são associadas às famílias de dispositivos. Como exemplo, tem-se o S60 Symbian SDK (Software Development Kit), usado para implementar aplicações compatíveis com a família de dispositivos S60 da Nokia. Na implementação do Ginga-NCL para dispositivos portáteis, foi dada prioridade ao uso de APIs Symbian genéricas. Entretanto, parte da implementação é necessariamente específica a uma UI.

A UI escolhida foi a S60, com o SDK que oferece suporte a versão 9.2 do sistema operacional Symbian, a mais recente disponibilizada para

desenvolvimento. Essa escolha foi feita devido à disponibilidade de documentação, a uma ampla comunidade que oferece suporte ao desenvolvimento e à grande quantidade de dispositivos Nokia S60 disponíveis no mercado, o que possibilitaria testes em dispositivos com mais facilidade e rapidez. Neste trabalho, quando qualquer API for descrita, ela será sempre genérica, a não ser que o contrário seja dito.