

# Estudo comparativo entre tecnologias Java: Applet e JWS.

Clara Aben-Athar B. Fernandes<sup>1</sup>, Carlos Alberto P. Araújo<sup>1</sup>

<sup>1</sup>Centro Universitário Luterano de Santarém – Comunidade Evangélica Luterana (CEULS/ULBRA) Caixa Postal 731– 68025-000 – Santarém – PA - Brasil

clara.aben-athar@hotmail.com, pedroso\_araujo@yahoo.com.br

**Resumo.** *Este artigo tem como objetivo fazer uma comparação entre Applets e software JWS (Java Web Start), ambos com características marcantes para trabalhar na World Wide Web (WWW), ou web. Este último armazena todo o programa localmente, na memória cache do computador e além de ser inicializado através de um navegador da web, conhecido como browser, pode ser iniciado por um ícone do desktop. Já o Applet é um aplicativo pequeno que exige poucos recursos de memória e por sua vez, é executado dentro de um browser. Independente da escolha entre as duas soluções é necessário que se analise o aplicativo que será criado para por em prática o que melhor atenda aos requisitos do problema.*

## 1 Introdução

Com o surgimento da Internet e sucessivamente da WWW, esse meio foi enriquecido fazendo nascerem as diversas linguagens de programação para Web. A linguagem pioneira nessa área, o HTML (*HyperText Markup Language*, que significa Linguagem de Marcação de Hipertexto), naquela época já não supria todas as necessidades que a internet exigia. Segundo Damasceno Jr. (1996), “O HTML não aceitava estruturas lógicas e não passava de um editor de texto com alguns poucos recursos adicionais.” A partir disso a linguagem JAVA, criada pela Sun Microsystems que resolveu investir em uma linguagem orientada a objetos e totalmente independente de plataforma de hardware e software, surge para apresentar interatividade em páginas web, o que a tornava moderna naquele período, surgindo assim um novo paradigma.

A princípio o Oak como era conhecido antes de se tornar Java tinha sido desenvolvido com o objetivo de ser usado na produção de sistemas embarcados. Quando surge a internet a Sun adapta essa linguagem para construir aplicativos web. Definitivamente esse novo paradigma veio para inovar em animações para internet, tornando-a mais dinâmica e divertida. “A história da Sun confunde-se com o surgimento da própria Internet” (DAMASCENO JR. 1996). Desse modo, o auge do Java se deu no mesmo período em que a web precisava de uma linguagem de programação interativa. Para isso, a Sun tornou ao Java possível a criação de *applet*, isto é, um pequeno programa que será executado quando chamado em uma página da web. Outra solução para implantar aplicativos na web é a tecnologia Java Web Star, ou JWS, que proporciona uma interface de usuário altamente interativa além de oferecer suporte para o uso do software *offline*.

Será realizado um estudo comparativo entre essas duas formas de desenvolver software para web, destacando suas principais características, de modo a ressaltar vantagens e desvantagens presentes em cada uma.

## 2 Breve História do Java

Em 1991 a Sun Microsystems começou a criar uma linguagem com base em uma já existente, C++. Os engenheiros escolheram-na por ser orientado a objetos e por gerar códigos pequenos, exatamente o que eles precisavam para implantar em pequenos aparelhos. Apesar dessas características precisavam de outra essencial, ser independente de plataforma, uma vez que deveriam ser executados em qualquer arquitetura, tais como, controle de TV, telefones, entre outros. Para solucionar esse problema utilizaram o conceito de máquina virtual. Em meio a isso nasce o antecessor de Java, o Oak.

Após várias tentativas de sucesso desenvolvendo sistema operacional, criando servidores digitais de vídeo, com a expansão da Internet a Sun se viu obrigada a adaptar essa linguagem a uma nova versão para internet. Foi então em 1995 que o Oak foi renomeado como Java.

Nesse mesmo período, a Sun cria um *web browser*, o HotJava, que tinha o recurso para executar pequenos programas Java, os applets, embutidos dentro das páginas web, o que transformou Java em uma linguagem de grande importância para o meio. Hoje os *browsers* que possuem a máquina virtual Java embutida podem executar programas dessa linguagem.

Neste particular Quinteiro (2006) registrou que:

A Sun considera o sucesso do Java na Internet como sendo o primeiro passo para utilizá-lo em decodificadores da televisão interativa, em dispositivos portáteis e outros produtos eletrônicos de consumo - exatamente como o Java tinha começado em 1991. Sua natureza portátil e o projeto robusto permitem o desenvolvimento para múltiplas plataformas, em ambientes tão exigentes como os da eletrônica de consumo.

De posse disso, pode-se concluir que Java é um universo vasto de tecnologias possuindo características fundamentais. Sendo assim, constitui-se de uma linguagem de programação orientada a objetos, possui portabilidade, ou seja, é independente de plataforma, é *multithreaded* o que permite a execução de múltiplas tarefas concorrentes, tem um *automatic garbage collector*, um mecanismo de gerenciamento de memória que a Java Virtual Machine (JVM) acomoda. Além de apoiar os recursos de rede, pois suporta uma extensa biblioteca de rotinas que facilitam a cooperação com protocolos TCP/IP, como HTTP e FTP, e ainda possuem segurança em sua execução, por controlar programas via rede com restrições.

De acordo com Jundl Junior (2007, p. 22),

Atualmente a linguagem está organizada em três segmentos principais:

- JavaMe (Java Micro Edition)- Destinado a pequenos dispositivos computacionais móveis, tais como celulares, PDAs e set-top boxes. É composto de máquina virtuais otimizadas para ambientes mais restritos, especificações de funcionalidades e uma API mais compacta.
- JavaSE (Java Standard Edition)- Integra os elementos padrão da plataforma e permite o desenvolvimento de aplicações de pequeno e médio porte. Inclui todas as APIs consideradas de base, além da máquina virtual padrão.
- JavaEE (Java Enterprise Edition)- Voltada para o desenvolvimento de aplicações corporativas complexas. Adiciona APIs específicas aos elementos-padrão da plataforma.

Com isso, constata-se que Java é uma linguagem ampla com várias opções para desenvolvimento de software com ambientes de execução diferentes, em diversas plataformas.

### 3 Tecnologias Java

Java tornou-se popular pelo seu uso na Internet e hoje possui seu ambiente de execução presente em *web browser*, celular, cartões inteligentes, entre outras tecnologias. Java continua crescendo e oferece qualidade, performance e segurança.

Dois tecnologias Java de grande relevância para o desenvolvimento de aplicativos interativos para web são o Applet e o JWS, ambos apresentam valores distintos para diferentes tipos de desenvolvedores e usuários.

#### 3.1 Applet

“Um applet é um programa escrito na linguagem de programação Java que pode ser incluído em uma página HTML, da mesma forma que uma imagem está incluída em uma página.” (Sun Microsystems) De fato podemos ter em uma página da WWW apenas um applet ou obter toda utilidade possível da página, como por exemplo, possuir texto e imagens. Resumidamente, um applet tem por objetivo ser um programa pequeno e deve ser executado através de um navegador web.

Java applets foram introduzidas na primeira versão da plataforma em 1995. Para executar um applet é necessário que o *browser* tenha a JVM embutida, ou seja, deve ser capacitado para interpretar e executar programas em Java, podendo rodar em um *browser* ou em um applet *viewer* usando a máquina virtual. Como no início nem todos os *browser* davam suporte a eles, foi criado o applet *viewer* para facilitar no desenvolvimento das aplicações, proporcionando a exibição apenas do applet, para fins de teste.

Aliada a uma extensa e versátil biblioteca de programação a plataforma Java comporta APIs (*Applications Programming Interfaces*) que facilitam no desenvolvimento de aplicativos. Conforme Roos (2007) “Uma Interface entre Aplicativo e Programação é um conjunto de instruções e padrões de programação para acesso a um aplicativo de software baseado na Web, ou ferramenta de Web”. De acordo com isso, Ross acrescenta que “a API é lançado para o público de modo que outros criadores de software possam desenvolver produtos acionados por esse serviço.”

Programas desenvolvidos em Java possuem garantia de segurança em sua aplicação já que o ambiente de *runtime* do Java, assim como o ambiente criado para os programas Applets são seguros, pois não possibilitam o acesso aos arquivos do sistema local, protegendo-os de danos. Somente a máquina virtual que determina quais arquivos e com quais dispositivos podem acessar o programa.

A segurança da execução de códigos Java na web consiste no conceito de *sandbox*, a qual determina um programa ser executado em um ambiente particular, possuindo suas permissões e proibições.

A Sun (2009) descreve em sua API que a “classe Applet deve ser a superclasse de qualquer aplicação que está a ser incorporado em uma página da Web ou visualizado pelo Applet Java *Viewer*.” Detalhadamente pode ser explicado como uma classe principal herdar a classe applet, por principal entende-se a classe executora.

Na prática isso é mais fácil de ser entendido. Primeiramente para criar uma aplicação Java Applet é preciso estender, ou melhor, herdar a classe Applet utilizando

sua API gráfica, para o melhor desempenho do software aplicativo. Após todo processo de desenvolvimento é necessário comprimir os binários do projeto em um formato Java Archive (JAR) e referenciar o software criado em uma página Web, criando uma *tag* em HTML, informando que o conteúdo comprimido em JAR é um Applet para que o *Web Browser* possa executar com a JVM. “Um Applet Java pode ser executado localmente na máquina do usuário, sem a necessidade do *browser*, bastando ter o *player* de Applet Java instalado” (RAMALHO, 2009).

Para implantação de um applet é necessário ter conhecimentos em relação às restrições de segurança já comentada anteriormente.

Como vantagens, os Applets possuem um mecanismo em que a máquina cliente é quem executa efetivamente parte da aplicação Java. Dessa forma há necessidade de que ela possua maior velocidade de processamento e capacidade de memória. Evitando sobrecarregar o servidor, este por sua vez se concentra apenas em receber e responder às solicitações do cliente. De modo geral, clientes e servidor executam a aplicação. Applets possuem uma intensa segurança, não submetendo a máquina cliente a riscos.

De acordo com o que já foi visto em applets o código Java é carregado pela JVM a partir do servidor Web. Portanto, via internet este tempo de download pode ser significativo, e essa é uma grande desvantagem. Porém, em uma rede local este tempo de carregamento não se mostra tão problemático, mas o objetivo não é esse. A idéia principal de um applet é ser executado na WWW.

### 3.2 Java Web Start (JWS)

Java Web Start é uma tecnologia que implanta um aplicativo Java através da Internet com apenas um clique. A IBM (2009) explica como funciona esse mecanismo:

O JWS permite que você implemente aplicativos Java independentes através da Internet com um único clique. Seus usuários clicam em um link para um arquivo XML com uma extensão Java Network Launch Protocol (.jnlp) e o JWS se encarrega do restante. Utilizando o JNLP, o JWS faz o download dos arquivos que precisa, os armazena em cache e ativa o aplicativo.

O trabalho da Web Start é exatamente fazer com que a aplicação funcione, encaminhando-a por meio da tecnologia JNLP (*Java Network Launch Protocol*) que descreve como serão realizados os processos. O armazenamento da descrição fica em um XML, onde este contém informação sobre a localização dos pacotes JAR. Quando o JNLP é executado, é iniciado o download dos pacotes jar necessários para o funcionamento da aplicação na máquina do usuário. Logo após é executada a aplicação em uma janela nativa do sistema operacional. Caso a aplicação necessite de permissões mais avançadas do sistema, o Java irá perguntar para o usuário se ele permite essa interação (RAMALHO; ARAUJO JR., 2009).

Dessa forma, a JWS tem como objetivo fazer o download do aplicativo evitando que o usuário passe por procedimentos complicados de instalação, além de verificar se há alguma versão mais recente do software. Na sua primeira utilização automaticamente é feita a transferência de todos os arquivos necessários, salvando-os em cache o que proporciona uma melhor interatividade em conexões lentas. O primeiro *download* é mais demorado. Isso pode ser considerado uma desvantagem, entretanto, sua posterior ativação é quase instantânea, pois todos os recursos necessários já estão disponíveis localmente (em cache), tornando-se uma vantagem.

Caso haja alguma versão já existente na máquina do cliente, esta estará disponível para ser executada a qualquer momento, por meio de ícones do *desktop* ou no

menu Iniciar (Microsoft Windows), através de um navegador da web clicando em um link ou ainda a partir do gerenciador de aplicativos integrados. Independente da maneira utilizada para invocar a aplicação, cada vez que for iniciada, a JWS irá conectar-se com o servidor web para verificar se há alguma versão atualizada do software aplicativo.

Sobre a segurança, assim como os applets, o JWS é executado em um ambiente restrito, o *sandbox*, possuindo acesso limitado aos arquivos do sistema e à rede também. Apesar das restrições é uma maneira fácil e robusta de implantar aplicativos no *browser*. É bastante eficaz e proporciona uma interface interativa, dá suporte a uso *offline* e requer baixo requisito de banda larga além de possuir segurança em sua aplicação.

JWS surgiu em 2001 na versão 1.2.2 da plataforma Java, é instalado com o SDK (*Software Development Kit*) e está incluso no *Java Runtime Environment* (JRE) como parte do JavaSe. Por isso, quando o aplicativo é baixado é feita uma busca dessas ferramentas necessárias, para que ele seja capaz de funcionar.

*Java Web Start* assegura a versão mais recente do aplicativo, bem como a versão correta do JRE. Toda vez que a aplicação é iniciada, o software JWS acessa de volta o servidor para verificar se há uma nova versão atualizada disponível. Além disso, é fácil de configurar, é independente de navegador web, possui uma configuração de *sandbox* com menos restrições o que permite aplicações mais complexas. Possuindo como desvantagem a inadequação de se comunicar com o *browser*.

A partir da teoria é observado que inúmeras vantagens estão presentes em ambas as aplicações JWS e Applets, por isso será feita uma comparação entre elas para esclarecer qual a melhor forma de implantar aplicativos na Web.

## 4 Conclusão

Tanto Applets quanto *Java Web Start* são maneiras de implementar aplicativos através da linguagem de programação Java, de forma a usufruir das vantagens sobre as características da orientação a objetos. Ambas possuem a web para disponibilizar o aplicativo. A principal diferença entre essas duas abordagens está na experiência do desenvolvedor, na escolha que ele defina como correta para determinada aplicação.

As duas tecnologias Java apresentadas possuem características semelhantes, pois são desenvolvidas em uma mesma linguagem de programação, classificada como orientada a objetos, independente de arquitetura, segura, portátil, de alto desempenho, interpretável, *multithreads* (múltiplas linhas de execução) e dinâmico.

Quando é exigido que o aplicativo tenha interatividade mútua com uma página web, a solução pode ser a escolha para o desenvolvimento com applets. Mas, se a aplicação deve ser independente do navegador (*browser*), o melhor recurso utilizado é JWS para implantação. Existem várias diferenças entre essas tecnologias, porém segundo a Sun esta é considerada como a diferença fundamental entre elas.

De qualquer forma para o desenvolvimento de um sistema para web é melhor estudar detalhadamente o problema para adequar a solução apropriada.

## 5 Referências

Damasceno Jr, A. Aprendendo JAVA: Programação na Internet. 3ª ed. São Paulo: Érica, 1996.

IBM. “Implementando um Aplicativo Baseado em Eclipse com Java Web Start”. Disponível em: <<http://www.ibm.com/developerworks/br/library/os-eclipse-jws/section3.html>>. Acesso em: 2 dez. 2009.

Jandl Junior, P. de JAVA: Guia do Programador. 1ed. São Paulo: Novatec, 2007.

Quinteiro, L. “História da Linguagem Java”. Disponível em <<http://www.htmlstaff.org/ver.php?id=4383>>. Acesso em: 13 dez. 2009.

Ramalho, G. L. e Araújo Júnior, M. B. “Um estudo comparativo de tecnologias Web para Games”. Disponível em: <[www.cin.ufpe.br/~tg/2009-1/mbaj.pdf](http://www.cin.ufpe.br/~tg/2009-1/mbaj.pdf)>. Acesso em: 16 dez. 2009.

Ross, D. “O que é uma API?”. Disponível em <<http://informatica.hsw.uol.com.br/conferencia-api1.htm>>. Acesso em: 16 dez. 2009

Sun Microsystems. “Applets”. Disponível em <<http://java.sun.com/applets>>. Acesso em: 16 dez. 2009.

\_\_\_\_\_. “API da Classe Applet”. Disponível em: <<http://java.sun.com/j2se/1.4.2/docs/api/java/applet/Applet.html>>. Acesso em: 16 dez. 2009.