

## **Metodologias e tecnologias de software Wplex**

### **Mauro Ransolin**

Wplex Software  
Florianópolis SC  
<http://www.wplex.com.br>

### **Objetivo**

*O objetivo deste documento é apresentar um resumo da abordagem adotada pela Wplex Software a partir de 2003, para o desenvolvimento de seus produtos. Nele, fazemos um apanhado das principais tecnologias empregadas no desenvolvimento de software, das metodologias de gestão dos projetos, de como os clientes estão inseridos e participam do processo de desenvolvimento, e de como esse esforço agrega valor ao nosso negócio, resultando em maiores benefícios aos nossos clientes.*

### **Práticas de desenvolvimento**

A Wplex prima pela excelência de seus produtos e serviços, por isso sua preocupação vai além de disponibilizar softwares com as melhores regras de negócio. Utilizamos as melhores práticas de desenvolvimento e as mais modernas tecnologias da atualidade, para que nossos clientes possam ter a disposição softwares eficazes para executar seu trabalho com eficiência.

A Wplex vê em seus clientes o motivo da sua existência, por isso sua metodologia de desenvolvimento de software abrange mecanismos de aproximação e comunicação com seus clientes. Além dos tradicionais canais de comunicação como e-mail, telefone e chat-internet, a Wplex criou um programa de relacionamento onde o cliente pode opinar sobre os produtos em desenvolvimento até mesmo nas fases mais incipientes do projeto.

Para atender a estas necessidades de tecnologia e métodos, a Wplex escolheu a plataforma Java EE® para desenvolver seus softwares e incorporou em sua metodologia de desenvolvimento as melhores práticas de metodologias consagradas, como Enterprise Unified Process, eXtreme Programming e Agile Modeling.

### **Linguagem de programação**

A Wplex Software optou pela linguagem de programação Java® e hoje todos os seus softwares são desenvolvidos na plataforma Java EE® utilizando as melhores práticas de Arquitetura de Software.

## Por que Java?

Java® é considerada a melhor tecnologia para o desenvolvimento de aplicações distribuídas. Ela está presente em mais de 2 bilhões de dispositivos que vão desde servidores de missão crítica, passando por microcomputadores, impressoras, dispositivos multi-funcionais, telefones celulares, PDA's, torradeiras e refrigeradores do futuro. Até mesmo os veículos espaciais da NASA, Spirit e Opportunity que chegaram a Marte em 2004, carregam tecnologia Java® para coleta de imagens em 3 dimensões.



A adoção da plataforma Java® nos garante as seguintes vantagens:

### *Independência de banco de dados relacional*

Nossas aplicações já saem homologadas para trabalhar perfeitamente em Oracle®, Microsoft SQL Server® e PostgreSQL®. Ainda, caso o cliente tenha outro modelo de banco relacional com controle transacional, basta que esteja homologado para Java® para que nossas aplicações o utilizem.

### *Independência de plataforma e portabilidade*

A própria natureza da tecnologia Java® permite que a aplicação rode em qualquer computador em qualquer sistema operacional. Isso permite aos nossos clientes optarem pela plataforma de servidores e estações de trabalho que desejarem, utilizando sistemas operacionais como Windows®, Linux®, Unix®, MacOS X®, FreeBSD® e outros.

### *Segurança*

A tecnologia Java® possui segurança embutida, sendo que as aplicações desenvolvidas nesta plataforma são consideradas as únicas livres de vírus, que podem danificar seu disco-rígido, roubar informações de seu celular, suas informações financeiras ou outras.

### *Escalabilidade*

O fato da tecnologia Java® ser uma tecnologia orientada a objetos e distribuída permite que os componentes de software sejam distribuídos em uma arquitetura de hardware simples (Figura 1), até uma redistribuição da aplicação em uma arquitetura de hardware extremamente potente (Figura 2). Isso é altamente vantajoso caso o número de usuários ou dados da aplicação cresça muito rapidamente, evitando problemas de lentidão de acesso e processamento.

A figura abaixo representa um modelo de rede simplificado, em uma arquitetura cliente-servidor, também conhecida como two-tier (duas camadas).

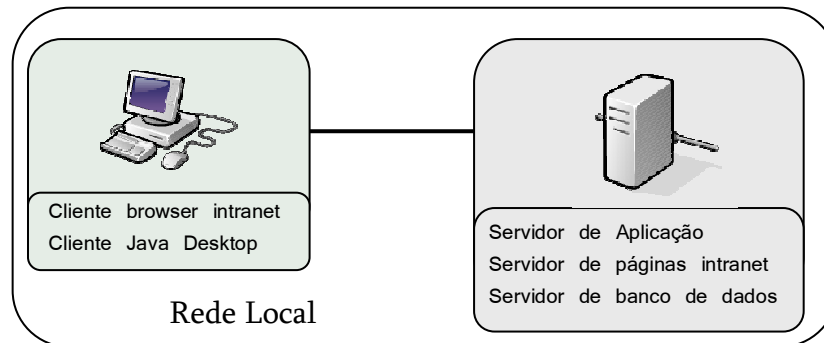


Figura 1: exemplo de rede simples

A figura abaixo representa um modelo de rede um pouco mais complexo, contendo clustering e distribuição de componentes de negócio conforme a capacidade do servidor numa arquitetura distribuída, também conhecida como n-tier (multi-camadas).

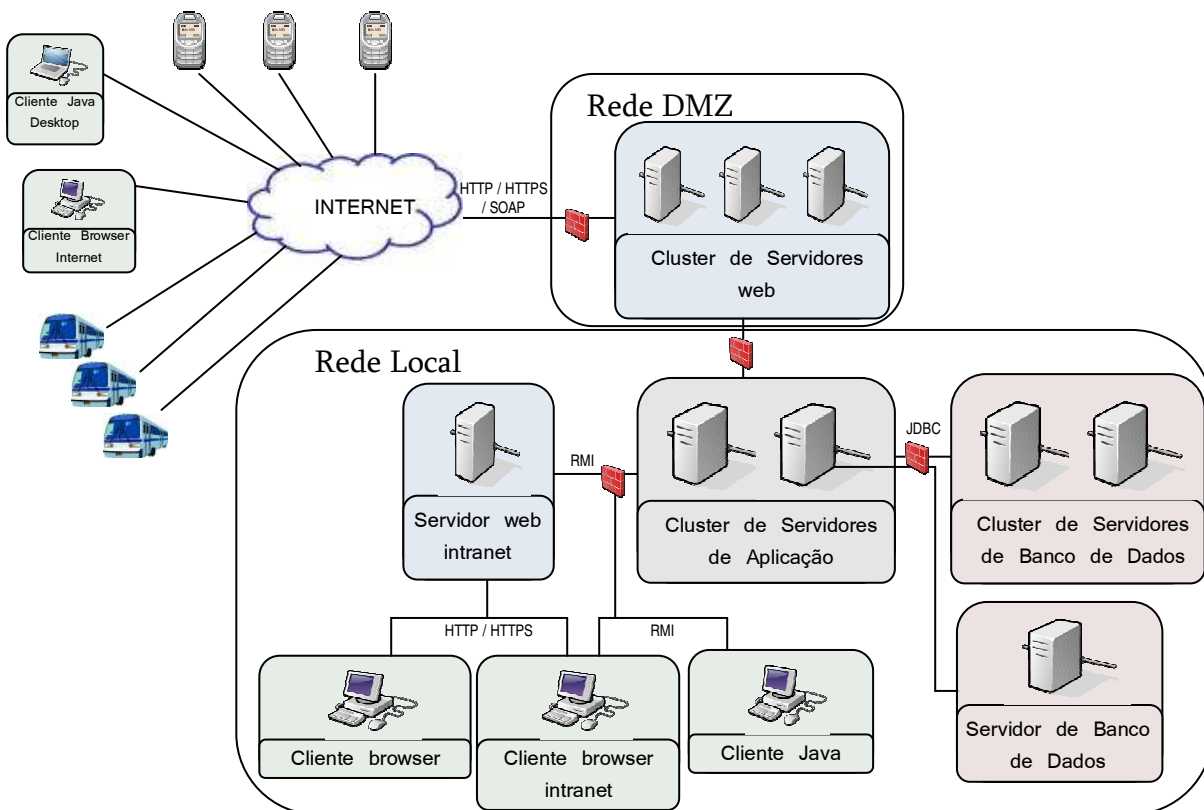


Figura 2: exemplo de rede distribuída

### Mobilidade

Java® não é apenas multiplataforma no que diz respeito a microcomputadores. A tecnologia Java J2ME® presente em PDA's, celulares domésticos e industriais, permite a consulta a dados vitais de seu negócio através destes dispositivos em qualquer lugar do planeta, além da integração com GPS embarcado.

### Transparência – código open-source

A Sun Microsystems®, proprietária da tecnologia Java® adotou a filosofia de open-source – código aberto – para a plataforma Java®. Qualquer pessoa tem acesso ao código fonte da linguagem Java®, e se desejar pode participar de uma comunidade de desenvolvedores chamada JCP – Java Community Process, onde pode participar do processo de desenvolvimento e das revisões das especificações da tecnologia Java®.

### Reuso de componentes e frameworks

Uma grande parcela dos componentes de software são componentes de infra-estrutura que deveriam ser plugáveis, ou seja, a roda não deve ser reinventada. Se alguém já fez algo que possa ser reutilizado, isso poderia estar disponível para todos através de licenças open-source. Como a plataforma Java EE® foi projetada em múltiplas camadas e sua licença é open-source, isso deixa de ser tese para ser realidade. Tarefas como controle de comportamento de telas, gravação de dados, log de eventos, entre muitos outros já possuem diversos frameworks gratuitos disponíveis na Internet, o que diminui o custo de desenvolvimento e propriedade do software. Na Figura 3 observamos o modelo multi-camadas Java EE®.

Java Compiler	Java Debugger	Javadoc	JPDA
Java Web Start		Java Plug-In	
Swing		AWT	
Sound	Input Methods	Java 2D	Accessibility
RMI	JDBC	JNDI	CORBA
XML	Logging	Beans	Locale Support
Preferences	Collections	JNI	Security
Lang	Util	New I/O	Networking
Java Hotspot Client Compiler		Java Hotspot Server Compiler	
Java Hotspot VM Runtime			
Solaris	Linux	Windows	Other

Figura 3: arquitetura multi-camadas Java EE®

A Wplex utiliza em seus softwares os mais modernos frameworks para cada camada da plataforma Java EE®. O quadro abaixo apresenta alguns destes frameworks e em qual camada da arquitetura se encontram.

<b>Camada</b>	<b>Framework</b>	<b>Utilidade</b>
Swing, AWT	Spring RCP JGoodies	Maior controle do comportamento das interfaces Componentes para elegância de interfaces
Java 2D	GEF	Maior controle do comportamento de gráficos 2D
JDBC	Hibernate	Efetua a comunicação da aplicação com qualquer banco de dados com homologação JDBC
XML, Logging, Util, Beans, Locale Suport e outros	Jakarta Commons	Bibliotecas de utilitários que facilitam a vida do programador
Javadoc	Apache Ant	Gera de forma automática toda a documentação da aplicação, além de executar vários outros tipos de tarefas automáticas

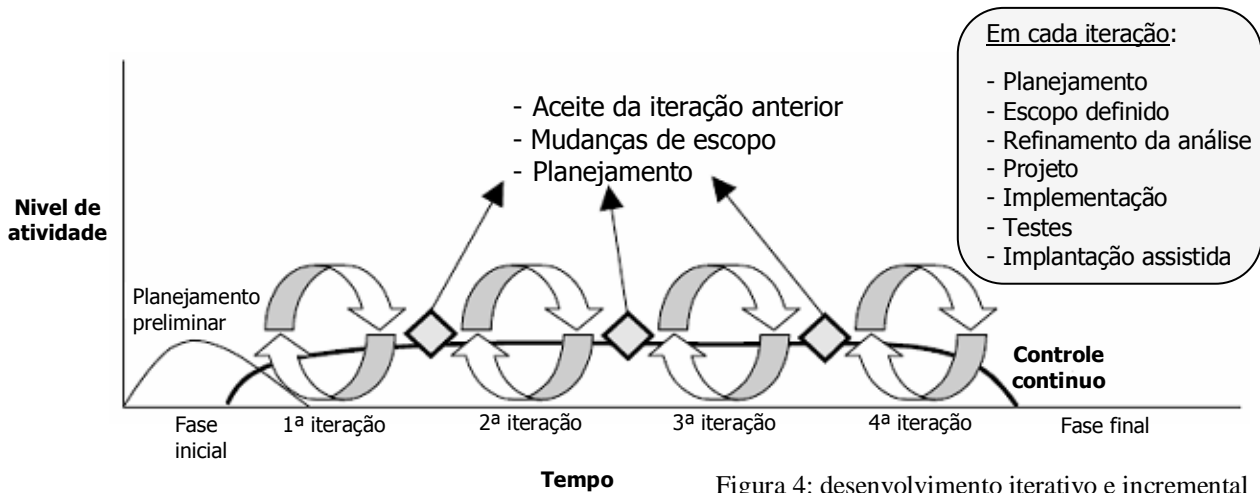
Ainda, a Wplex se beneficia de vários outros frameworks, componentes e ferramentas desenvolvidas em Java®, ou de licença open-source, tais como: Xdoclet®, Struts®, Jasper Reports®, JUnit®, Cactus®, Spring®, Eclipse®, SQuirreL SQL®, Bugzilla®, dotProject®, CVS®, JBoss®, entre outros.

## **Metodologia e Processo de Desenvolvimento**

Um projeto centrado em arquitetura que possui diversos ciclos rápidos de especificação, implementação, teste, validação e implantação pode produzir resultados mais precisos e de menor custo que seu desenho e planejamento integral.

A Wplex desenvolve seus softwares num processo conhecido como 'desenvolvimento iterativo e incremental' (Figura 4). O surgimento de uma nova aplicação, ou o desenvolvimento de um conjunto de novas funcionalidades, não possui apenas as 4 fases estanques de um projeto tradicional de software, que são: análise, projeto, desenvolvimento e testes, pois sabemos que o mundo não é estático e que nossos clientes acompanham esta evolução através do aprimoramento contínuo de seu negócio. Assim, nossas aplicações são desenvolvidas incrementalmente, passando diversas vezes pelas 4 fases do desenvolvimento até serem concluídas. Ainda, conferimos voz ativa a nosso cliente, portanto adicionamos uma quinta fase: Implantação com Acompanhamento Continuado (ou Implantação Assistida). Isso não significa apenas que atentamos pela fase de implantação, mas também que efetuamos

ajustes, aperfeiçoamentos e mudanças no código da aplicação, para ajustá-la perfeitamente ao ambiente do cliente.



O ciclo de desenvolvimento iterativo e incremental é um dos paradigmas que rege a metodologia de desenvolvimento nomeada Unified Process (Processo Unificado), também faz parte do conjunto de princípios e práticas das metodologias derivadas de Agile Process (Processos Ágeis), como Agile Modeling (Modelagem Ágil) e eXtreme Programming (Programação Extrema).

A Wplex formulou uma Metodologia de Desenvolvimento e Gestão de Projetos baseada nos paradigmas centrais destas metodologias e nos seus conjuntos de princípios e práticas. Como exemplo, citamos alguns destes:

#### [Programação de testes unitários](#)

A prova real de que uma funcionalidade está correta é codificar os testes pelos quais esta funcionalidade deve passar. Novas funcionalidades não devem impactar negativamente nas já existentes, portanto os testes já existentes devem continuar funcionando.

#### [Projeto centrado em arquitetura](#)

A arquitetura do coração do sistema é quem determina a qualidade do produto final e seu tempo de vida. Uma aplicação bem arquitetada terá um custo menor de manutenção, uma taxa menor de erros e permitirá uma distribuição dos componentes compatível com o poder de processamento dos servidores.

#### [Pair-programming](#)

A programação em duplas torna o código melhor estruturado, de mais fácil manutenção além de minimizar a ocorrência de falhas.

### *Mudanças incrementais*

Uma grande mudança em uma aplicação deve ser promovida através de uma série de mudanças pequenas e incrementais, cada qual com sua homologação e aceite.

### *Feedback rápido*

Quanto mais tempo se passa sem um feedback, maior o custo de consertar um erro.

### *Propriedade coletiva*

Todos os desenvolvedores do time são responsáveis pelos modelos de software e pelo código, assim todos tem capacidade e responsabilidade para aplicarem o seu melhor conhecimento em todas as fases do projeto.

### *Documentação através de modelos*

Os modelos de software devem ser simples e o suficientemente corretos para transmitirem o entendimento do negócio. Documentação excessiva gera burocracia e retrabalho.

### *Software é o objetivo principal*

Software funcionando é sua melhor medida de progresso. 10 em cada 10 clientes preferem ter a aplicação instalada e funcionando ao invés de todos os seus manuais e nenhum software.

A Wplex acredita e segue estes princípios com dedicação e prazer.

## **Interatividade com o cliente**

O modo como a Wplex procura interagir com seus clientes tem como objetivo minimizar um dos grandes problemas normalmente encontrados no mercado, que é a questão da adaptação dos sistemas ao cliente.

O processo de desenvolvimento de software da Wplex, por sua natureza, disponibiliza canais de comunicação com o cliente, permitindo que este participe efetivamente durante o ciclo de desenvolvimento dos sistemas. Nossos principais clientes tem acesso a aplicação e opinam sobre ela até mesmo nos momentos mais incipientes do seu ciclo de vida, que chamamos de estágio 'Alpha'. Funciona da seguinte forma:

1. Disponibilizamos em um site, quinzenalmente, uma nova versão dos sistemas com suas últimas atualizações efetuadas;
2. Ainda no site, através do controle de versões, o cliente verifica quais foram as alterações efetuadas na aplicação;
3. O cliente pode efetuar o download desta última versão da aplicação via tecnologia Java Web Start®. A aplicação cliente acessa servidores web, de aplicação e dados hospedados na Wplex;
4. O cliente brinca com a aplicação, executando todas as funcionalidades que desejar. Seus dados ficam armazenados de forma que todos os clientes participantes se beneficiem das experiências desta comunidade;
5. Caso o cliente encontre um erro ou tenha uma sugestão de melhoria ele tem a sua disposição uma aplicação chamada Bugzilla®, onde pode registrar sua solicitação. Esta aplicação notifica via e-mail o responsável pelo projeto que toma as devidas providências. Assim que uma providência é tomada o usuário recebe um e-mail sobre a resolução tomada. Todos os erros e sugestões, assim como as resoluções ficam armazenadas no banco de dados do Bugzilla®, podendo serem consultadas a qualquer momento por qualquer usuário cadastrado.

Desta forma temos a convicção que as novas versões das nossas aplicações, quando apresentadas ao mercado, atenderão integralmente as necessidades dos clientes.

## Conclusões

A Wplex Software investe em tecnologia estado-da-arte para proporcionar aos seus clientes uma experiência diferenciada no uso de sistemas de informação. Concluindo,

- A escolha da tecnologia Java® proporciona aos clientes um custo menor de implantação dos sistemas, alta flexibilidade na escolha de estrutura de rede e hardware, além da possibilidade de redução de custo na aquisição de softwares de infra-estrutura como banco de dados, servidores de internet e até mesmo de sistemas operacionais.
- O cliente é peça-chave para o sucesso dos produtos. O processo de desenvolvimento e gestão de projetos da Wplex confere voz ativa ao cliente. Com a participação do cliente no ciclo de desenvolvimento, torna-se possível atender às suas necessidades em maior plenitude e menor prazo.

## A Wplex Software

A Wplex Software é uma empresa de software baseada em Florianópolis SC e tem como competência essencial o desenvolvimento de ferramentas para otimização do uso da frota e tripulação por empresas operadoras de transporte urbano e aéreo.

---

*Elaborado em 2006.*