



Open-source Education →

Java para programadores Mainframe

Iniciativa Globalcode

Mini-cursos Globalcode

MC1 – Introdução à plataforma Java

MC2 – Sintaxe da linguagem e orientação a objetos com Java

MC3 – Introdução à plataforma J2EE – Java 2 Enterprise Edition

MC4 – Desenvolvimento de aplicativos Web com Java

MC5 – J2EE modelando arquiteturas para demandas de 10 a mais de 10.000 usuários

MC6 – Java e mainframe: analogias, integrações e arquiteturas

MC7 – Metodologias de desenvolvimento para Java e UML

MC8 – Desenvolvimento Web com design-patterns e Struts

MC9 – Desenvolvimento de componentes Enterprise JavaBeans

MC10 – Planejamento e execução de stress-test

MC11 ao MC13 – Preparatórios para certificações Java

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
5. Java no CICs
6. Analogias de Java-Cobol
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

A Globalcode

The Developers Company

Educação treinamentos gratuitos, vídeo-aulas, palestras em empresas e universidades, cursos individuais, carreiras e serviços de consultorias pontuais e mentoring;

Pesquisa desenvolvimento de experiências com publicações em conferências internacionais - eXPerience Group -, convênio com ITA e IPEN;

Produção de software pequena fábrica de desenvolvimento de componentes Java, em expansão para 2006;

Palestrante / Instrutor

Vinicius Senger – vinicius@globalcode.com.br

- Sócio e fundador da Globalcode, foi instrutor e consultor da Sun e Oracle no Brasil;
- Trabalhou em projetos de grande porte em bancos. Começou a programar com 8 anos e trabalha com desenvolvimento de softwares profissionalmente desde os 13 anos;
- Certificações: Sun Java Programmer / Sun Enterprise Architect P1, Microsoft Certified Professional, Microsoft Certified Trainer;

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
5. Java no CICs
6. Analogias de Java-Cobol
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

O que é Java?

*Java é uma **linguagem de programação** capaz de produzir **softwares robustos** para diversos **tipos de microcomputadores** e dispositivos como celulares, palmtops, cartão inteligente, robots entre outros.*

Como surgiu?

Projeto Green

- Financiado pela SUN, 1990 – 1992
- Linguagem “OAK” para pequenos dispositivos

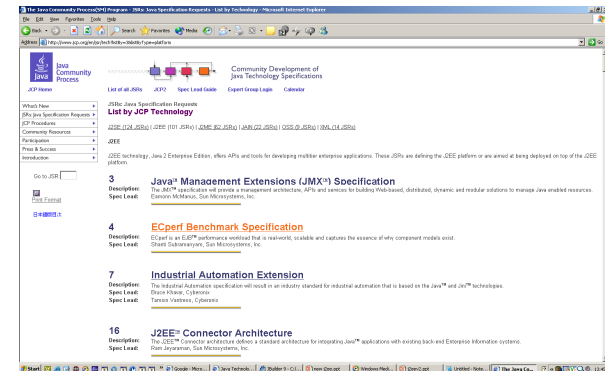
Start7 ou *7 = hardware + software

- Monitor LCD 5” colorido, Touchscreen
- Wireless, com interface PCMCIA
- Versão de UNIX rodando em menos de 1MB
- Linguagem segura, robusta, multi-plataforma, com threads, bibliotecas gráficas, interpretada, com coleta automática de lixo, distribuída, e muito mais.
- Controle remoto
- Permitia a troca de objetos em uma rede sem fio



Java Community Process

- Apesar da tecnologia ter sido criada pela Sun, ela é mantida por um consórcio de empresas chamado de Java Community Process;
- Java Community Process: estabelece processo evolutivo da linguagem e coleta opiniões de empresas, universidades e desenvolvedores;



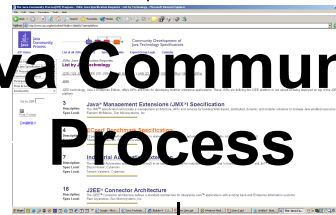
Java Community Process

Empresas

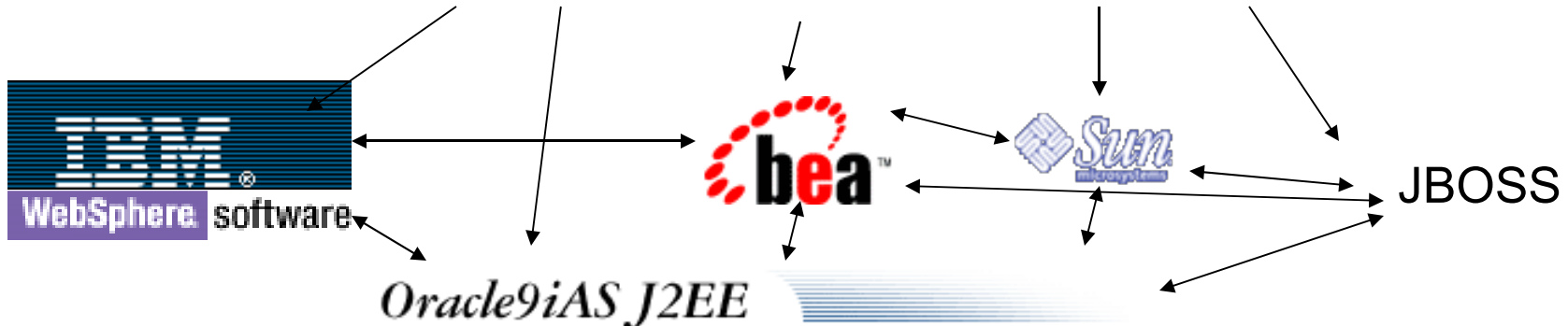
Universidades

Experts

**Java Community
Process**



Java Specification Request (JSR)
Especificações públicas



Java Community Process

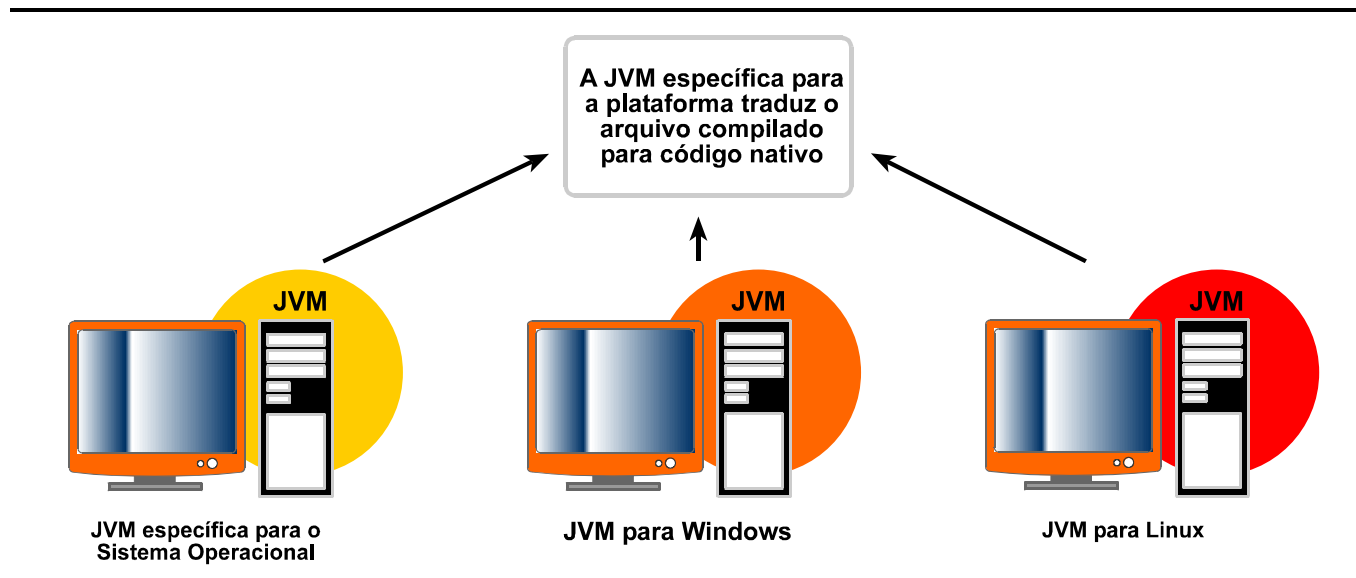
Conclusão: a linguagem e tecnologia Java não estão sob domínio de uma só empresa, garantindo assim maior vida útil para as soluções desenvolvidas.

Vantagens do Java

- É independente de sistema operacional, banco de dados, servidor Web, IDE's, etc. Na prática isso significa que podemos programar softwares com diferentes combinações;
- Linguagem moderna, segura, orientada a objetos e altamente preparada para computação em rede;
- Alto índice de adoção de negócios críticos;
- Diversificação de um mesmo tipo de produto Java devido ao consórcio e padrões J.C.P.;

Máquinas Virtuais

- Responsável por prover a flexibilidade de sistema operacional para um software Java



Máquinas Virtuais

- Interpretação de código inteligente;
- Mantém código mais acessado permanentemente traduzido para código nativo;
- Gerencia alocação e desalocação de memória.
- Class loader - carrega arquivos .class para memória
- Segurança do código – responsável por garantir a não execução de códigos maliciosos (ex: applets)
- Verifica integridade do arquivo .class - Bytecode verifier.

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
5. Java no CICs
6. Analogias de Java-Cobol
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

Java no Mainframe

- Neste tópico devemos apresentar um histórico de como a tecnologia Java entrou no mainframe além de mostrar de forma macroscópica as possibilidades de uso de Java no mainframe.

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
5. Java no CICs
6. Analogias de Java-Cobol
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

Arquiteturas Mainframe c/ baixa plataforma

- Apresentaremos as formas mais utilizadas mercado para integração entre mainframe e baixa plataforma, tais como:
 - Integração com servidor TCP/IP proprietário;
 - Integração via MQ-Series;
 - Integração com CICS Connector;
 - Captura de telas;
 - FTP;

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
- 4. Java no Mainframe Unix System**
5. Java no CICs
6. Analogias de Java-Cobol
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

Unix System

- Devemos apresentar tecnicamente como funciona o Java rodando no Unix System do mainframe;

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
- 5. Java no CICs**
6. Analogias de Java-Cobol
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

Java no CICs

- Devemos apresentar tecnicamente como funciona o Java rodando no CICs;

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
5. Java no CICs
- 6. Analogias de Java-Cobol**
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

Java - Cobol

- Analogias entre:
 - Estrutura Vs. Classes;
 - Procedural Vs. POO;
 - funções Vs. métodos;
 - if, switch, while e for;
 - alocação de memória;
 - acesso a arquivos;
 - acesso a banco de dados;

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
5. Java no CICs
6. Analogias de Java-Cobol
- 7. Analogias de Java-Assembler**
8. Analogias de arquiteturas J2EE-Mainframe / CICs

Java – Assembler

- Analogias entre:
 - Estrutura Vs. Classes;
 - Procedural Vs. POO;
 - funções Vs. métodos;
 - if, switch, while e for;
 - alocação de memória;
 - acesso a arquivos;
 - acesso a banco de dados;

Agenda

1. Introdução à tecnologia Java
2. Introdução à Java no Mainframe
3. Arquiteturas de integração com baixa plataforma
4. Java no Mainframe Unix System
5. Java no CICs
6. Analogias de Java-Cobol
7. Analogias de Java-Assembler
8. Analogias de arquiteturas J2EE-Mainframe / CICs

Java – Assembler

- Este tópico tem o objetivo de apresentar as terminologias Mainframe comparadas com a terminologias Java, tais como:
 - Processo batch;
 - EJB Vs. Transação;
 - Procedure / Proc;
 - Processo assíncronos;
 - Outras terminologias adicionais;