



Red Hat JBoss Enterprise Application Platform 7.1

Introdução ao JBoss EAP

Para uso com o Red Hat JBoss Enterprise Application Platform 7.1

Red Hat JBoss Enterprise Application Platform 7.1 Introdução ao JBoss EAP

Para uso com o Red Hat JBoss Enterprise Application Platform 7.1

Nota Legal

Copyright © 2018 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Resumo

Este documento oferece uma visão geral conceitual de alto nível do Red Hat JBoss Enterprise Application Platform, bem como de diversos tópicos em que o JBoss EAP é baseado.

Índice

CAPÍTULO 1. VISÃO GERAL DE CONCEITOS GERAIS	3
1.1. JAVA	3
1.2. SERVIDORES DE APLICATIVOS	3
1.3. JAVA EE 7	3
CAPÍTULO 2. VISÃO GERAL DO JBOSS EAP	4
2.1. SOBRE O JBOSS EAP 7	4
2.2. SUBSISTEMAS	5
2.3. ALTA DISPONIBILIDADE	5
2.4. MODOS DE OPERAÇÃO	5
CAPÍTULO 3. EXEMPLOS	6
3.1. EXEMPLO SIMPLES	6
3.2. EXEMPLO EXPANDIDO	6

CAPÍTULO 1. VISÃO GERAL DE CONCEITOS GERAIS

Antes de compreender como o Red Hat JBoss Enterprise Application Platform pode ser configurado e implantado, é necessário entender alguns conceitos importantes.

1.1. JAVA

O Java é uma linguagem de programação e uma plataforma de computação que incorpora conceitos como orientação a objetos, classes e simultaneidade. Os aplicativos Java são compilados em código de bytes e executados em uma Máquina Virtual Java (JVM).

1.2. SERVIDORES DE APLICATIVOS

Um servidor de aplicativos é um software que fornece um ambiente para execução de aplicativos web. A maioria dos servidores de aplicativos também oferecem funcionalidade para aplicativos web em execução em seu ambiente por meio de APIs. Por exemplo, um servidor de aplicativos pode fornecer uma API para conexão com um banco de dados.

1.3. JAVA EE 7

O Java EE (Java Platform, Enterprise Edition) é uma plataforma corporativa com base em padrões que oferece API e ambiente de tempo de execução para a execução e o desenvolvimento de aplicativos Java. O objetivo é aprimorar a produtividade dos desenvolvedores com recursos corporativos avançados em estruturas de fácil consumo que eliminam o texto clichê e reduzem a carga técnica. As estruturas que compõem o Java EE são extensamente testadas em combinação. O Java EE 7, com base no [JSR 342](#), é um sucessor do Java EE 6, com um enfoque principal na simplificação adicional de APIs para acessar serviços de contêiner e aumentar o suporte a tecnologias web emergentes, como o HTML5.

O Java EE 7 inclui suporte para vários perfis, ou subconjuntos, de APIs. A especificação do Java EE 7 define os perfis *full* e *web*.

Perfil web do Java EE 7

O perfil web do Java EE 7 é projetado para o desenvolvimento de aplicativos web e oferece suporte a um subconjunto das APIs definidas por tecnologias com base na web relacionadas ao Java EE 7.

Perfil full do Java EE 7

O perfil full do Java EE 7 contém todas as APIs definidas pelo Java EE 7 (incluindo todos os itens do perfil web). Ao desenvolver EJBs, aplicativos de mensagens e serviços web (em vez de aplicativos web), use o perfil full.

CAPÍTULO 2. VISÃO GERAL DO JBOSS EAP

2.1. SOBRE O JBOSS EAP 7

O Red Hat JBoss Enterprise Application Platform 7.1 (JBoss EAP 7) é uma implementação certificada das especificações dos perfis web e full do Java Enterprise Edition 7 (Java EE 7).

As principais versões do JBoss EAP são bifurcações do projeto comunitário [WildFly](#) em determinados momentos em que ele atingiu o nível de completude de recursos desejado. Depois desse ponto, ocorre um extenso período de testes e produtividade em que o JBoss EAP é estabilizado, certificado e aprimorado para uso de produção. Durante a vida útil de uma versão principal do JBoss EAP, recursos selecionados do projeto comunitário podem ser escolhidos e inseridos por backport em uma série de lançamentos secundários de aprimoramento de recursos, dentro da mesma família de versão principal. Por exemplo, o JBoss EAP 7.1 é uma bifurcação de um ramo de trabalho do WildFly 11.

O JBoss EAP oferece opções preconfiguradas para recursos como clustering de alta disponibilidade, mensagens e cache distribuído. Ele também permite que os usuários escrevam, implantem e executem aplicativos usando vários serviços e APIs fornecidos pelo JBoss EAP.

O JBoss EAP inclui uma estrutura modular que permite a ativação de serviços somente quando necessário, melhorando a velocidade inicialização. O console de gerenciamento baseado na web e a interface de linha de comando de gerenciamento (CLI) tornam desnecessária a edição de arquivos de configuração XML e adicionam recursos de script e automatização de tarefas. Além disso, o JBoss EAP inclui APIs e estruturas de desenvolvimento para a construção rápida de aplicativos Java EE seguros e escaláveis.

Tabela 2.1. Recursos do JBoss EAP

Recursos	Descrição
Conformidade com o Java EE 7	Certificado para os perfis web e full do Java Enterprise Edition 7.
Domínio Gerenciado	Gerenciamento centralizado de várias instâncias de servidor e hosts físicos, enquanto que um servidor autônomo permite uma única instância de servidor. Gerenciamento de grupos por servidor de configuração, implantação, associações de soquete, módulos, extensões e propriedades do sistema. Gerenciamento centralizado e simplificado de segurança de aplicativos (incluindo domínios de segurança).
Console de gerenciamento e CLI de gerenciamento	Novas interfaces de gerenciamento de servidor autônomo ou de domínio. A CLI de gerenciamento também inclui um modo de lotes que pode utilizar scripts e automatizar tarefas de gerenciamento. A edição direta dos arquivos de configuração XML do JBoss EAP não é recomendada.

Recursos	Descrição
Layout de diretórios simplificado	O diretório de módulos contém todos os módulos de servidores de aplicativos. Os diretórios de servidor autônomo e de domínio contêm os artefatos e os arquivos de configuração para implantações autônomas e em domínio, respectivamente.
Mecanismo de carregamento de classe modular	Os módulos são carregados e descarregados conforme a necessidade. Isso melhora o desempenho, oferece benefícios de segurança e reduz os tempos de inicialização e reinicialização.
Gerenciamento de fonte de dados otimizado	Os drivers de banco de dados são implantados como outros serviços. Além disso, as fontes de dados são criadas e gerenciadas com o uso do console de gerenciamento e da CLI de gerenciamento.

2.2. SUBSISTEMAS

Muitos dos recursos e das APIs que são expostos a aplicativos implantados no JBoss EAP são organizados em subsistemas. Esses subsistemas podem ser configurados por administradores para fornecer comportamentos diferentes, dependendo do objetivo do aplicativo. Por exemplo, se um aplicativo exigir um banco de dados, uma fonte de dados pode ser configurada no subsistema **datasources** e acessada pelo aplicativo depois de implantada no servidor ou domínio do JBoss EAP.

2.3. ALTA DISPONIBILIDADE

A alta disponibilidade (HA) no JBoss EAP refere-se a várias instâncias do JBoss EAP funcionando em conjunto para fornecer aplicativos mais resistentes a flutuações de tráfego, carga de servidores e falha de servidor. O HA incorpora conceitos como dimensionamento, balanceamento de carga e tolerância a falhas.

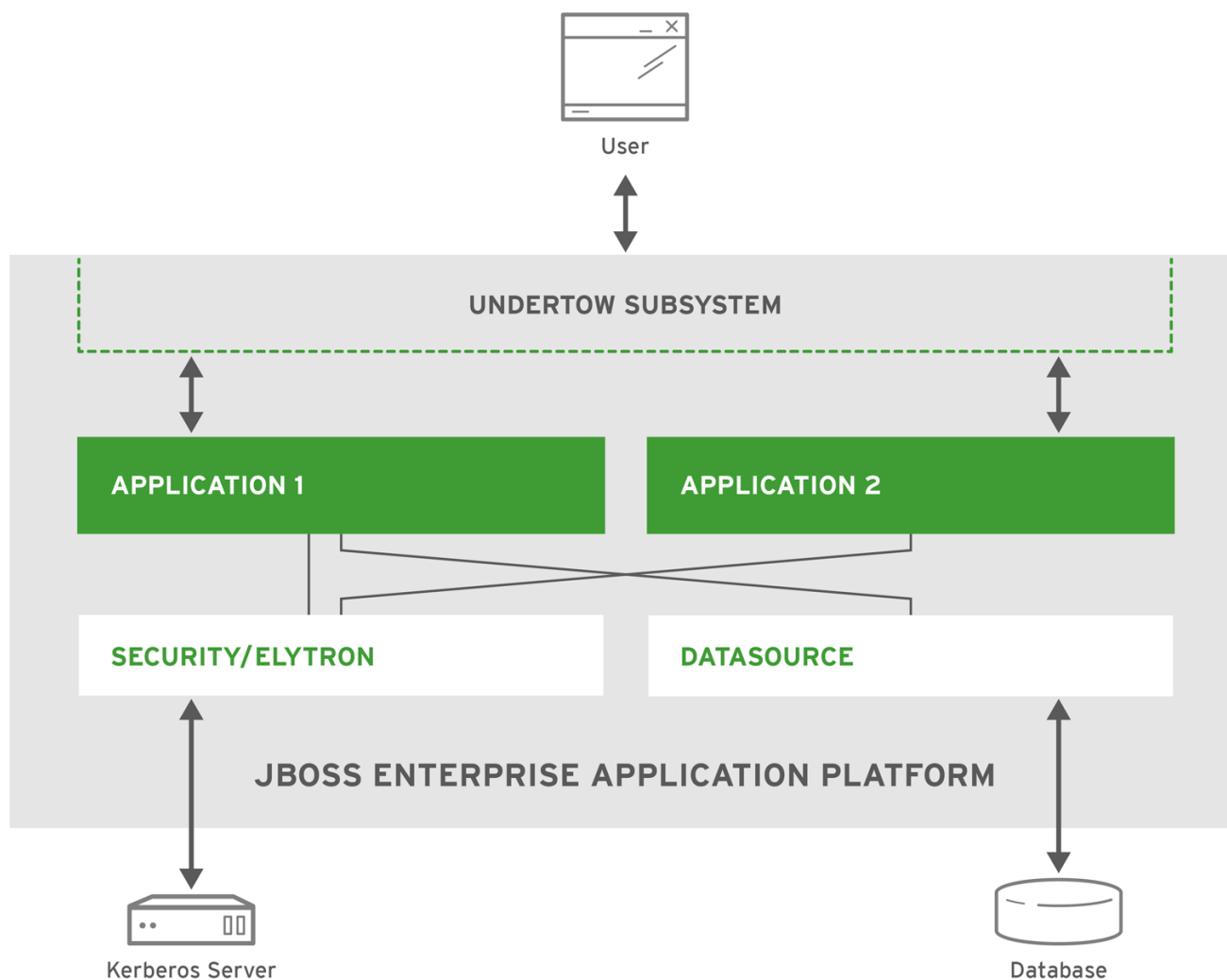
2.4. MODOS DE OPERAÇÃO

Além de fornecer funcionalidade e APIs aos seus aplicativos, o JBoss EAP tem poderosos recursos de gerenciamento. Eles variam de acordo com o modo de operação usado para iniciar o JBoss EAP. O JBoss EAP oferece um modo de operação de *servidor autônomo* para o gerenciamento de instâncias independentes e um modo de operação de *domínio gerenciado* para gerenciamento de grupos de instâncias a partir de um único ponto de controle.

CAPÍTULO 3. EXEMPLOS

Abaixo se encontram vários exemplos para ilustrar como o JBoss EAP funciona e como se encaixa em diferentes ambientes.

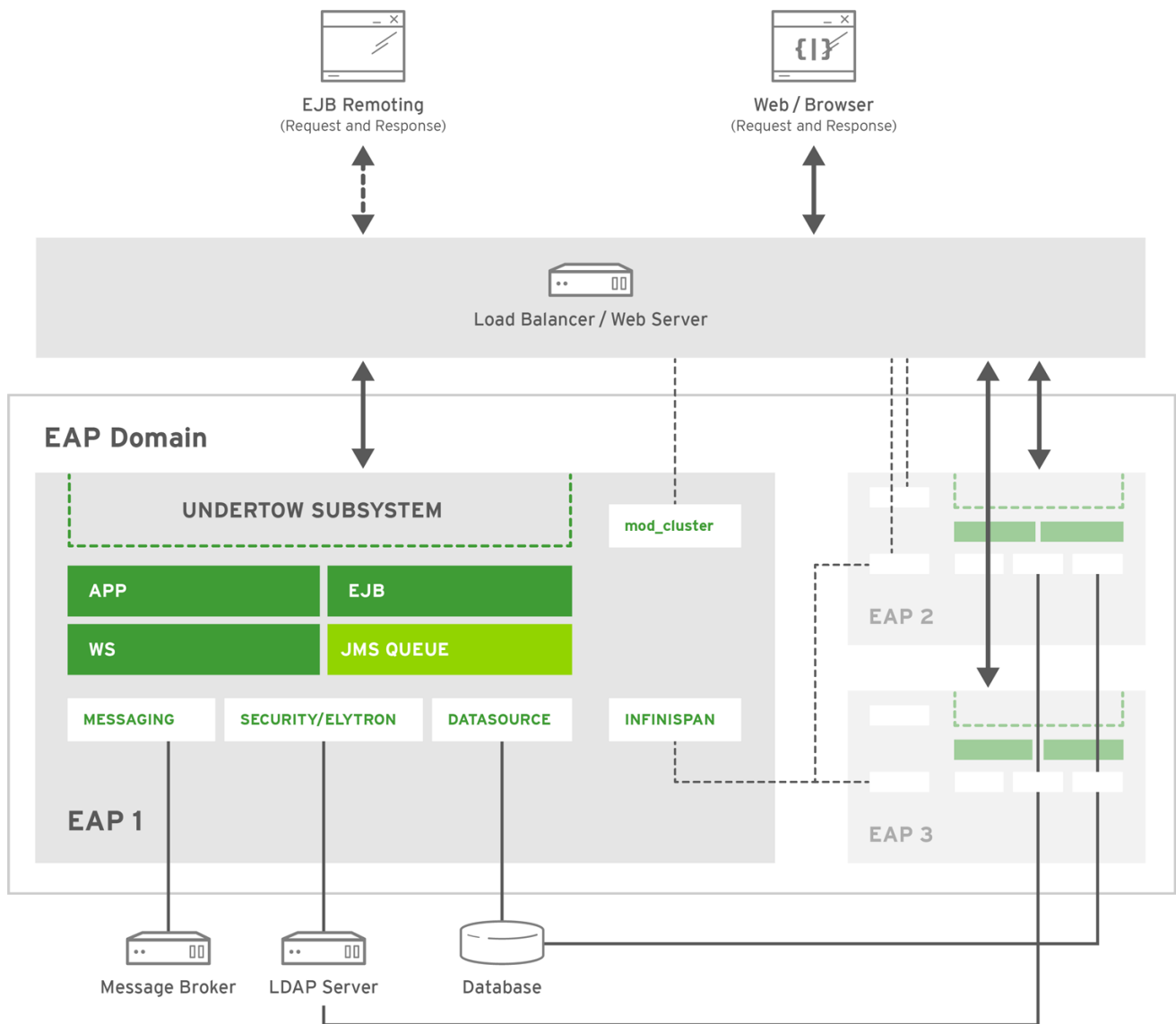
3.1. EXEMPLO SIMPLES



JBoss_430110_1216

Este exemplo mostra uma configuração simples do JBoss EAP. A instância do JBoss EAP tem dois aplicativos implantados nela. Também está configurada para se conectar a um banco de dados usando o subsistema **datasources** e um servidor Kerberos que pode usar o subsistema herdado **security** ou o **elytron**. Essas conexões são expostas aos aplicativos implantados. A instância do JBoss EAP manipula solicitações por meio do subsistema **undertow** e as direciona para o aplicativo apropriado. Os aplicativos usam as APIs expostas pelo JBoss EAP para se conectar ao banco de dados e ao servidor Kerberos e realizam sua lógica comercial implementada. Após a conclusão, os aplicativos enviam uma resposta de volta ao solicitante por meio do subsistema **undertow**.

3.2. EXEMPLO EXPANDIDO



JBoss_430110_1216

Este exemplo ilustra uma configuração mais complexa que envolve três instâncias do JBoss EAP organizadas em um domínio gerenciado com um balanceador de carga ou um servidor web. As três instâncias também são configuradas para dar suporte à alta disponibilidade por meio de balanceamento de carga com o uso de `mod_cluster` e replicação de sessão com Infinispan. Todas as três instâncias do JBoss EAP têm um aplicativo web, um serviço web e EJB implantados. Uma instância do JBoss EAP tem uma fila JMS configurada por meio do subsistema **messaging-activemq**. Todas as três instâncias do JBoss EAP têm conexões a um banco de dados por meio da fonte de dados. Elas também têm uma conexão ao servidor LDAP usando o subsistema herdado **security** ou o subsistema **elytron**. Além disso, uma instância do JBoss EAP é configurada para se conectar a um agente de mensagens externo por meio do subsistema **messaging-activemq**. Essas conexões configuradas são expostas aos aplicativos, serviços web, EJBs e filas JMS implantados na respectiva instância.

Todas as solicitações de entrada destinadas ao aplicativo, serviço web ou EJB são primeiramente recebidas pelo balanceador de carga ou servidor web. Com base no algoritmo de balanceamento de carga configurado e nas informações fornecidas por cada instância do JBoss EAP, o servidor web ou o balanceador de carga direciona a solicitação para a instância apropriada do JBoss EAP. A instância do JBoss EAP manipula solicitações por meio do subsistema **undertow** e as direciona ao aplicativo apropriado. Os aplicativos usam as APIs expostas pelo JBoss EAP para se conectar ao banco de dados e ao servidor Kerberos e realizam sua lógica comercial implementada. Após a conclusão, os aplicativos

enviam uma resposta de volta ao solicitante por meio do subsistema **undertow**. Todas as informações não persistentes, por exemplo, informações de sessão, são propagadas entre as instâncias do JBoss EAP por meio do subsistema **infinispan**.

Revised on 2018-01-11 05:30:17 EST