



Red Hat JBoss Enterprise Application Platform 7.1

Einführung in JBoss EAP

Zur Verwendung mit der Red Hat JBoss Enterprise Application Platform 7.1

Red Hat JBoss Enterprise Application Platform 7.1 Einführung in JBoss EAP

Zur Verwendung mit der Red Hat JBoss Enterprise Application Platform 7.1

Rechtlicher Hinweis

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.

Zusammenfassung

Dieses Dokument bietet einen Überblick über die Red Hat JBoss Enterprise Application Platform ebenso wie einige Themen, auf denen JBoss EAP basiert.

Inhaltsverzeichnis

KAPITEL 1. ÜBERBLICK ÜBER ALLGEMEINE KONZEPTE	3
1.1. JAVA	3
1.2. ANWENDUNGSSERVER	3
1.3. JAVA EE 7	3
KAPITEL 2. ÜBERBLICK ÜBER JBOSS EAP	4
2.1. ÜBER JBOSS EAP 7	4
2.2. SUBSYSTEME	5
2.3. HOHE VERFÜGBARKEIT	5
2.4. BETRIEBSMODI	5
KAPITEL 3. BEISPIELE	6
3.1. EINFACHES BEISPIEL	6
3.2. AUSFÜHRLICHES BEISPIEL	6

KAPITEL 1. ÜBERBLICK ÜBER ALLGEMEINE KONZEPTE

Sie müssen einige wichtige Konzepte kennen lernen, um zu verstehen, wie die Red Hat JBoss Enterprise Application Platform konfiguriert und bereitgestellt werden kann.

1.1. JAVA

Java ist eine Programmiersprache und Computing-Plattform, die Konzepte wie Objektorientierung, Klassen und Nebenläufigkeit umfasst. Java-Anwendungen werden als Bytecode kompiliert und innerhalb einer Java Virtual Machine (JVM) ausgeführt.

1.2. ANWENDUNGSSERVER

Bei einem Anwendungsserver handelt es sich um Software, die eine Umgebung zur Ausführung von Web-Anwendungen bietet. Die meisten Anwendungsserver bieten zudem über eine Reihe APIs Funktionen für Web-Anwendungen, die in ihrer Umgebung ausgeführt werden. Ein Anwendungsserver kann beispielsweise eine API zur Verbindung mit einer Datenbank bieten.

1.3. JAVA EE 7

Java EE (Java Platform, Enterprise Edition) ist eine Standards-basierte Enterprise Plattform, die sowohl eine API als auch eine Runtime-Umgebung zur Ausführung und Entwicklung von Java-Anwendungen bietet. Das Ziel besteht darin, die Produktivität von Entwicklern zu verbessern, indem umfangreiche Enterprise-Fähigkeiten in einfach verwendbaren Frameworks geboten werden, die Boilerplate eliminieren und technische Hürden verringern. Die Frameworks, aus denen Java EE besteht, werden in Kombination ausführlich getestet. Java EE 7, basierend auf [JSR 342](#), ist ein Nachfolger von Java EE 6, bei dem der hauptsächliche Schwerpunkt auf der weiteren Vereinfachung von APIs zum Zugriff auf Container-Dienste liegt, während zugleich die Unterstützung auf aufkommende Web-Technologien wie HTML5 verbessert wird.

Java EE7 bietet Support für mehrere Profile oder Subsets von APIs. Die Java EE7-Spezifikation definiert das *volle* Profil sowie das *Web*-Profil.

Java EE 7 Web Profile

Das Java EE 7 Web Profile ist auf die Entwicklung von Web-Anwendungen ausgelegt und unterstützt ein Subset von APIs, definiert gemäß Java EE 7-zugehörigen webbasierten Technologien.

Java EE 7 volles Profil

Java EE 7 volles Profil umfasst alle APIs, die von Java EE 7 definiert sind (einschließlich aller Elemente im Web-Profil). Verwenden Sie bei der Entwicklung von EJBs, Nachrichtenanwendungen und Web-Diensten (gegenüber Web-Anwendungen) das volle Profil.

KAPITEL 2. ÜBERBLICK ÜBER JBOSS EAP

2.1. ÜBER JBOSS EAP 7

Red Hat JBoss Enterprise Application Platform 7.1 (JBoss EAP 7) ist eine zertifizierte Implementation der vollen und Web-Profilespezifikationen von Java Enterprise Edition 7 (Java EE 7).

Größere Versionen von JBoss EAP werden aus dem [WildFly](#) Communityprojekt abgespalten, wenn dieses Projekt den jeweils gewünschten Funktionsumfang erreicht hat. Daraufhin werden ausführliche Tests sowie Produktisierung durchgeführt, bei denen JBoss EAP stabilisiert, zertifiziert und für den Serieneinsatz verbessert wird. Während dem Lebenszyklus einer größeren Version von JBoss EAP können ausgewählte Funktionen einzeln aus dem Communityprojekt über eine Reihe von kleineren Releases innerhalb derselben größeren Versionsfamilie portiert werden. JBoss EAP 7.1 wird beispielsweise aus einem funktionsfähigen Branch von WildFly 11 abgespalten.

JBoss EAP bietet vorkonfigurierte Optionen für Funktionen wie High-Availability-Clustering, Nachrichten und verteiltes Caching. Es ermöglicht Benutzern zudem, Anwendungen über die diversen APIs und Dienste, die JBoss EAP bietet, zu schreiben, bereitzustellen und auszuführen.

JBoss EAP umfasst eine modulare Struktur, die eine Aktivierung von Dienstleistungen nur ermöglicht, wenn dies erforderlich ist, wodurch die Startgeschwindigkeit verbessert wird. Die webbasierte Verwaltungskonsole und die Befehlszeilenschnittstelle machen eine Bearbeitung von XML-Konfigurationsdateien überflüssig und bieten die Möglichkeit, Aufgaben zu automatisieren und zu scripten. Zudem umfasst JBoss EAP APIs und Entwicklungs-Frameworks zur schnellen Entwicklung von sicheren und skalierbaren Java EE-Anwendungen.

Tabelle 2.1. Funktionen von JBoss EAP

Funktion	Beschreibung
Mit Java EE 7 konform	Zertifizierung gemäß Java Enterprise Edition 7 volles Profil und Web-Profil.
Managed Domain	Zentralisiertes Management mehrerer Serverinstanzen und physischer Hosts, während ein Standalone-Server eine einzige Serverinstanz erlaubt. Verwaltung der Konfiguration, Bereitstellung, Socket-Bindung, Modulen, Erweiterungen und Systemeigenschaften pro Servergruppe. Zentralisiertes und vereinfachtes Management der Anwendungssicherheit (einschließlich Sicherheitsdomänen).
Management-Konsole und Befehlszeilenschnittstelle	Neue Domänen- oder Standalone-Servermanagementschnittstellen. Die Befehlszeilenschnittstelle umfasst zudem einen Stapelmodus, der Managementaufgaben scripten und automatisierten kann. Eine direkte Bearbeitung der JBoss EAP XML-Konfigurationsdateien wird nicht empfohlen.

Funktion	Beschreibung
Vereinfachtes Verzeichnis-Layout	Das Modulverzeichnis umfasst alle Anwendungs-Servermodule. Die Domänen- und Standalone-Verzeichnisse umfassen die Artefakte und Konfigurationsdateien für die Domänen- bzw. Standalone-Bereitstellungen.
Modulares Klassenladesystem	Module werden nach Bedarf geladen oder entladen. Dies verbessert die Leistung, weist Sicherheitsvorteile auf und reduziert die Start- und Neustartzeit.
Verschlinktes Datenquellenmanagement	Datenbanktreiber werden wie andere Dienste bereitgestellt. Zudem werden Datenquellen über die Managementkonsole und die Befehlszeilenschnittstelle erstellt und verwaltet.

2.2. SUBSYSTEME

Viele APIs und Funktionen, die gegenüber in JBoss EAP bereitgestellten Anwendungen offenliegen, sind in Subsysteme organisiert. Diese Subsysteme können von Administratoren konfiguriert werden, um verschiedene Verhaltensweisen zu bieten, abhängig vom Ziel der Anwendung. Wenn eine Anwendung beispielsweise eine Datenbank benötigt, kann eine Datenquelle im Subsystem **Datenquellen** konfiguriert werden und die Anwendung nach Bereitstellung auf dem JBoss EAP-Server oder in der Domäne auf diese Datenquelle zugreifen.

2.3. HOHE VERFÜGBARKEIT

Hochverfügbarkeit bezeichnet bei JBoss EAP die Zusammenarbeit mehrerer JBoss EAP-Instanzen, um Anwendungen bereitzustellen, die gegenüber Trafficschwankungen, Serverbelastung und Serverfehlern beständiger sind. Hochverfügbarkeit umfasst Konzepte wie Skalierbarkeit, Load Balancing und Fehlertoleranz.

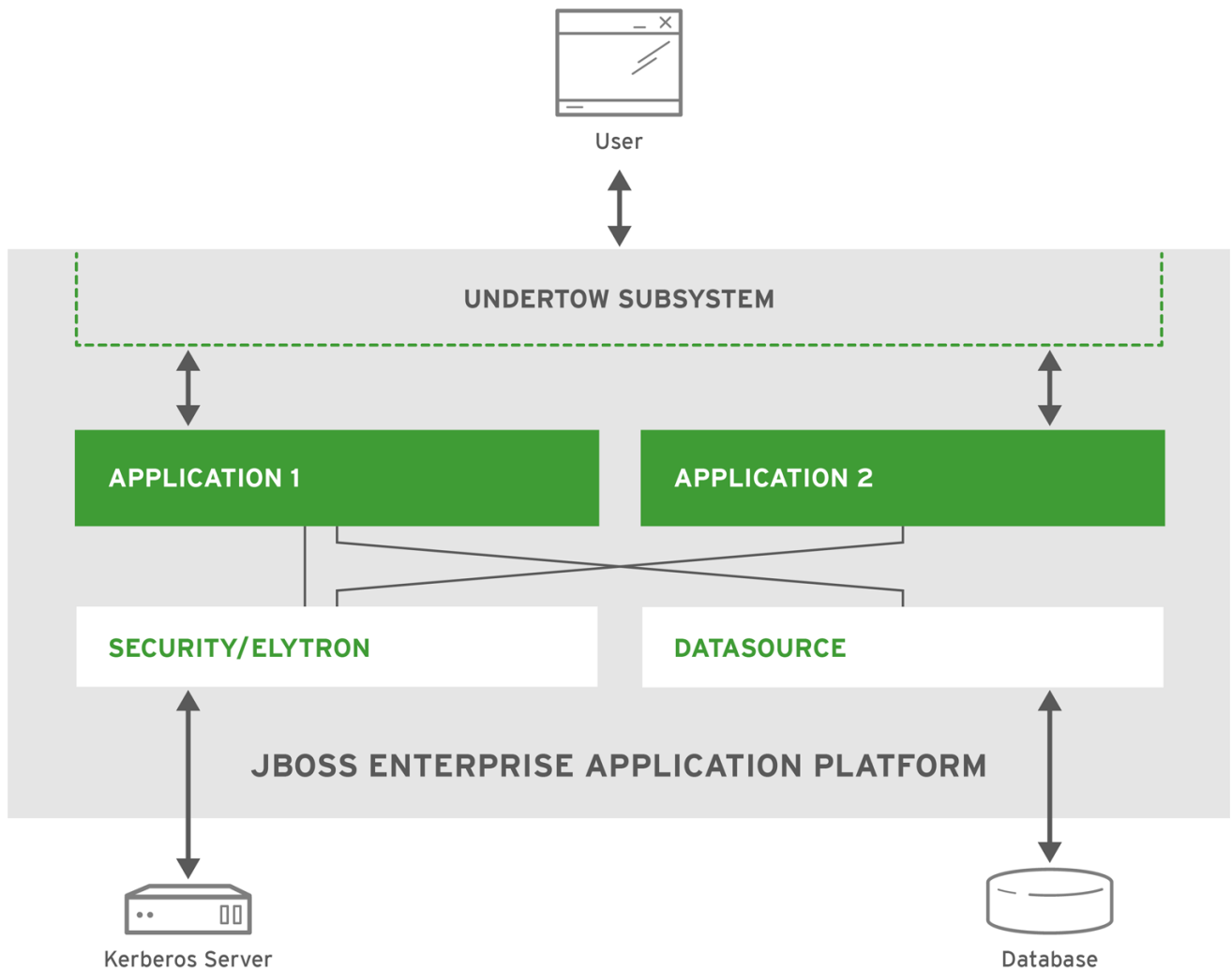
2.4. BETRIEBSMODI

Neben der Bereitstellung von Funktionen und APIs für seine jeweiligen Anwendungen hat JBoss EAP leistungsfähige Verwaltungsfunktionen. Diese Verwaltungsfunktionen unterscheiden sich je nach dem Betriebsmodi, der zum Start von JBoss EAP verwendet wird. JBoss EAP bietet einen Betriebsmodus *Standalone-Server* zum Management diskreter Instanzen und einen Betriebsmodus *Managed Domain* zum Management von mehreren Gruppen Instanzen über einen einzigen Kontrollpunkt.

KAPITEL 3. BEISPIELE

Im Folgenden finden Sie einige Beispiele, die die Funktionsweise von JBoss EAP und die Einpassung in verschiedene Umgebungen beschreiben.

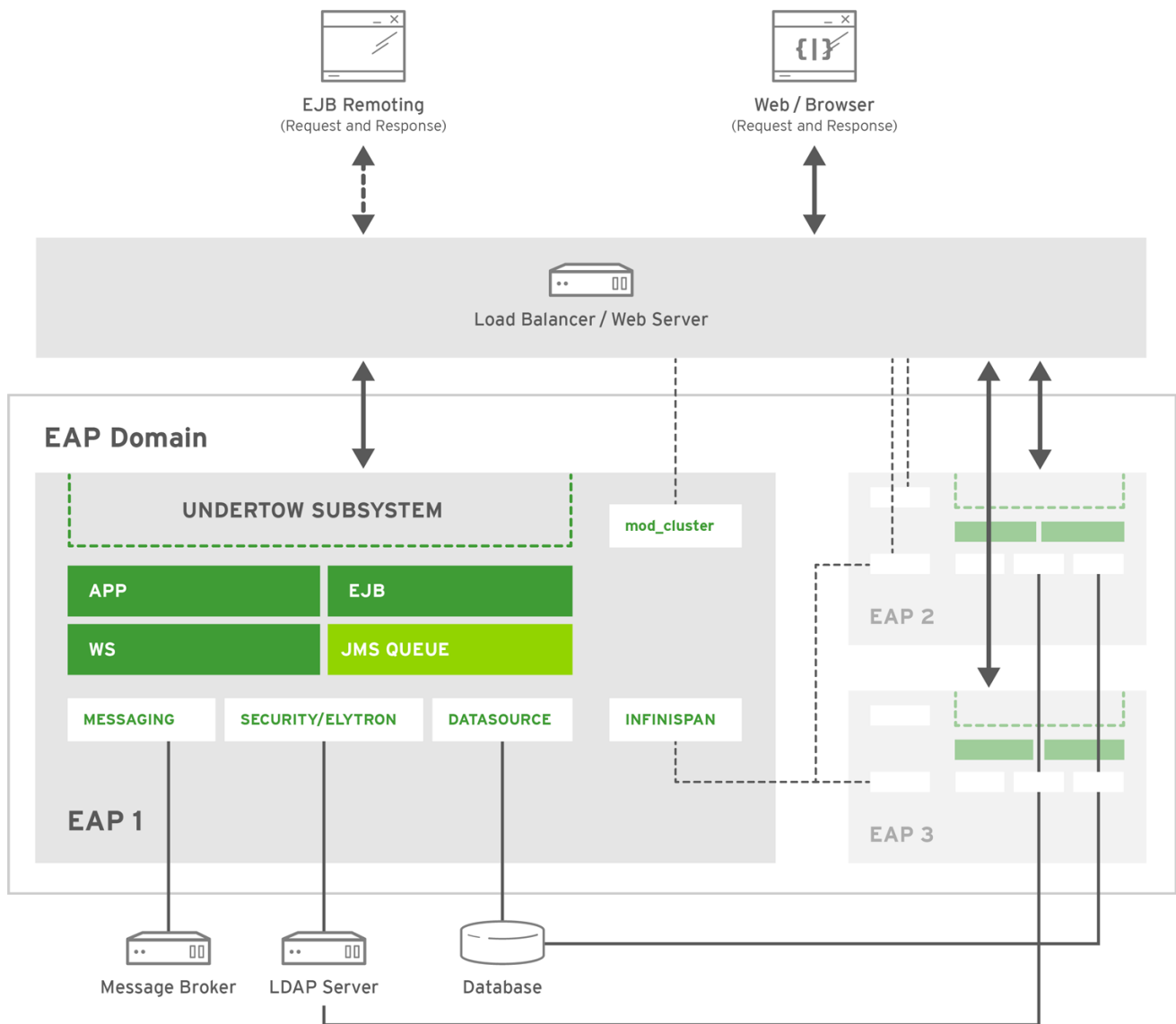
3.1. EINFACHES BEISPIEL



JBoss_430110_1216

Dieses Beispiel stellt ein einfaches JBoss EAP-Setup dar. In der JBoss EAP-Instanz sind zwei Anwendungen bereitgestellt. Sie ist zudem zur Verbindung mit einer Datenbank über das Subsystem **Datenquellen** und einen Kerberos-Server konfiguriert, der entweder das Legacy-Subsystem **Sicherheit** oder das Subsystem **Elytron** verwenden kann. Diese Verbindungen stehen den bereitgestellten Anwendungen offen. Die JBoss EAP-Instanz verarbeitet Anfragen über das Subsystem **Undertow** und leitet diese Anfragen an die entsprechende Anwendung weiter. Die Anwendungen nutzen das von JBoss EAP zugänglich gemachte API zur Verbindung zur Datenbank und dem Kerberos-Server und führen ihre implementierte Geschäftslogik aus. Nach Abschluss senden die Anwendungen über das System **Undertow** eine Antwort zurück an den Anfragenden.

3.2. AUSFÜHRLICHES BEISPIEL



JBoss_430110_1216

Dieses Beispiel beschreibt ein komplexeres Setup mit drei JBoss EAP-Instanzen in einer Managed Domain mit einem Load Balancer oder einem Webserver. Die drei Instanzen sind zudem auf die Unterstützung von Hochverfügbarkeit über Load Balancing mittels `mod_cluster` und Sitzungsreplikation über `Infinispan` ausgelegt. Bei allen drei JBoss EAP-Instanzen sind eine Web-Anwendung, ein Web-Dienst und EJB bereitgestellt. Bei einer JBoss EAP-Instanz ist eine JMS-Warteschlange über das Subsystem `messaging-activemq` konfiguriert. Alle drei JBoss EAP-Instanzen sind über die Datenquelle mit einer Datenbank verbunden. Zudem sind sie mit dem LDAP-Server entweder über das Legacy-Subsystem `Sicherheit` oder das Subsystem `elytron` verbunden. Zudem ist eine JBoss EAP-Instanz auf die Verbindung mit einem externen Message Broker über das Subsystem `messaging-activemq` konfiguriert. Diese konfigurierten Verbindungen stehen den Anwendungen, Webdiensten, EJBs und JMS-Warteschlangen zur Verfügung, die in der jeweiligen Instanz bereitgestellt sind.

Alle eingehenden Anfragen für die Anwendung, den Webdienst oder EJB werden zunächst vom Load Balancer oder Webserver empfangen. Auf Grundlage des konfigurierten Load-Balancing-Algorithmus und den von jeder JBoss EAP-Instanz zur Verfügung gestellten Informationen leitet der Webserver oder Load Balancer die Anfrage an die entsprechende JBoss EAP-Instanz weiter. Die JBoss EAP-Instanz verarbeitet Anfragen über das Subsystem `undertow` und leitet diese Anfragen an die entsprechende Anwendung weiter. Die Anwendungen nutzen die von JBoss EAP verfügbar gemachten APIs zur Verbindung mit der Datenbank und dem Kerberos-Server und führen ihre implementierte Geschäftslogik aus. Nach dem Abschluss senden die Anwendungen über das Subsystem `undertow`

eine Antwort zurück an den Anfragenden. Nicht persistente Informationen, beispielsweise Sitzungsinformationen, werden über die JBoss EAP-Instanzen über das Subsystem **infinispan** verbreitet.

Revised on 2018-01-11 05:28:14 EST