

Vorbereiten der Installation von Adobe Experience Manager Forms on JEE auf Serverclustern

Rechtliche Hinweise

Weitere Informationen zu rechtlichen Hinweisen finden Sie unter http://help.adobe.com/de_DE/legalnotices/index.html.

Inhalt

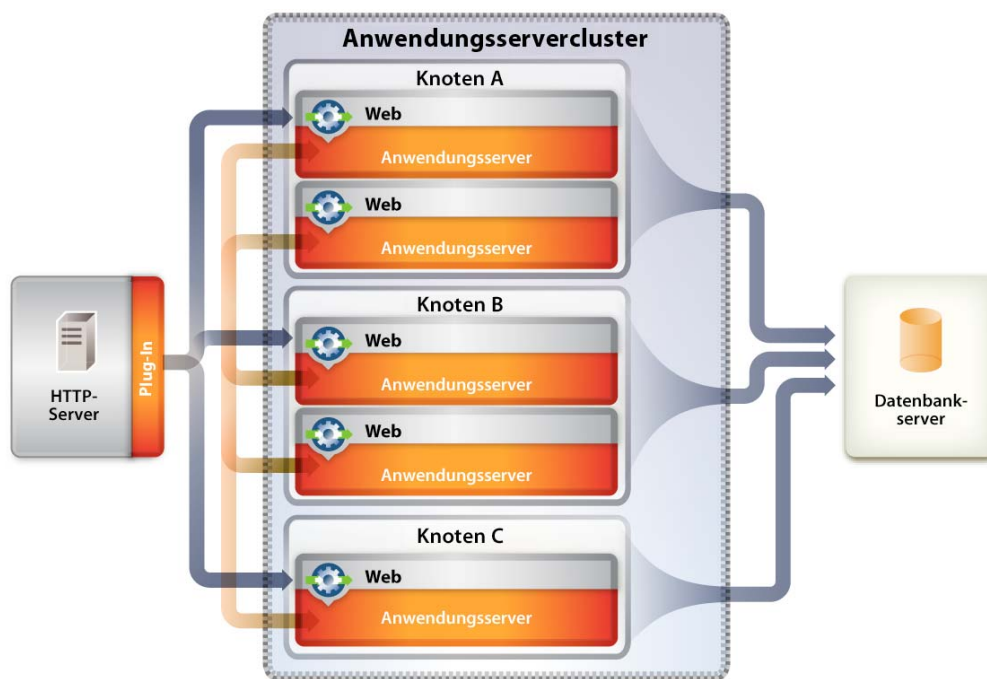
Einführung in Anwendungsservercluster

1.1 Infos zum Clustering von Anwendungsservern	1
1.2 Terminologie	3
1.3 AEM forms Clustering	3
1.4 Unterstützte Topologien	5
1.5 Nicht unterstützte Topologien	7

Einführung in Anwendungsservercluster

1.1 Infos zum Clustering von Anwendungsservern

Ein *Cluster* ist eine Gruppe von Anwendungsserverinstanzen, die gleichzeitig ausgeführt werden und wie ein Einzelsystem funktionieren, wodurch hohe Verfügbarkeit und Lastenausgleich ermöglicht werden. In einem Cluster können mehrere Serverinstanzen auf demselben Computer ausgeführt werden (auch als *vertikaler Cluster* bezeichnet), sich auf verschiedenen Computern befinden (auch als *horizontaler Cluster* bezeichnet) oder eine Kombination aus sowohl horizontalen als auch vertikalen Clustern bilden. Durch Clustering kann die Auslastung durch Clients auf verschiedene Knoten verteilt werden, anstatt von einem einzigen Anwendungsserver verarbeitet werden zu müssen.



In einer Clusterkonfiguration sind Anwendungsserverinstanzen Mitglieder des Clusters, auf denen allen identische Anwendungskomponenten bereitgestellt sein müssen. Anders als bei den konfigurierten Anwendungen, müssen aber Clustermitglieder keine weiteren Konfigurationsparameter gemeinsam haben. So können beispielsweise mehrere Serverinstanzen auf einem Computer einen Cluster bilden, wobei eine einzelne Instanz sich auf einem anderen Computer befindet, vorausgesetzt, dass sie alle dieselbe Version des Anwendungsservers ausführen.

Durch Clusterbildung können Sie mindestens einen der folgenden Vorteile erzielen. Die Implementierungsart der Cluster bestimmt die Vorteile, die erzielt werden können:

- Failover
- Lastenausgleich
- Skalierbarkeit

1.1.1 Failover

Failover ermöglicht, dass mindestens eine Anwendungsserverinstanz als Sicherung für eine ausgefallene Anwendungsserverinstanz fungiert und die Verarbeitung der Aufgabe fortsetzt, wodurch ein anderer Anwendungsserver in die Lage versetzt wird, die Verarbeitung fortzusetzen. Fällt ein Anwendungsserver aber während einer Transaktion aus, stellt der Sicherungsanwendungsserver den Zustand der ausgefallenen Instanz nicht wieder her. Fällt ein Server beispielsweise aus, wenn der Benutzer Daten in ein Formular eingibt, müssen die Daten möglicherweise erneut eingegeben werden.

1.1.2 Lastenausgleich

Lastenausgleich ist eine Methode, die zum Verteilen von Arbeitslast auf eine Reihe von Systemen verwendet wird, damit kein einzelnes Gerät überlastet wird. Beginnt ein Server Überlastungserscheinungen zu zeigen, werden Anforderungen an einen anderen Server mit höheren Kapazitäten weitergeleitet.

1.1.2.1 Anwendungsserver-Lastenausgleich

Anwendungsserver-Lastenausgleich ist bei der Verwaltung der Auslastung zwischen Anwendungsservern nützlich. Anwendungsserver können so konfiguriert werden, dass sie eine gewichtete Round-Robin-Routingrichtlinie verwenden, die eine ausgeglichene Routingverteilung auf Grundlage des Satzes von Servergewichtungen sicherstellt, die den Mitgliedern eines Clusters zugewiesen wurden. Durch die Konfiguration aller Server in einem Cluster mit derselben Gewichtung wird eine Lastenverteilung erzeugt, bei der alle Server ungefähr die gleiche Anzahl von Anforderungen erhalten. Die stärkere Gewichtung einiger Server führt dazu, dass mehr Anforderungen an diese Server gesendet werden als an Server mit einer niedrigeren Gewichtung.

Bevorzugte Routingkonfigurationen können ebenso konfiguriert werden, um zum Beispiel sicherzustellen, dass nur Clustermitglieder auf diesem Knoten ausgewählt werden (unter Verwendung der Round-Robin-Gewichtungsmethode) und dass Clustermitglieder auf entfernten Knoten nur ausgewählt werden, wenn ein lokaler Server nicht verfügbar ist. Der Anwendungsserver-Lastenausgleich eignet sich am besten für den Ausgleich zwischen Stufen („Tiers“).

1.1.2.2 Webserver-Lastenausgleich

Webserver-Lastenausgleich eignet sich dazu, Anforderungen in Warteschlangen zu stellen bzw. einzuschränken. Beim HTTP-Server „Apache“ ist die gängigste Methode für den Lastenausgleich Round-Robin-DNS.

Round-Robin-DNS ist ein relativ einfaches Lastenausgleichsverfahren, bei dem ein DNS(Domain Name System)-Server eine Auflösung von Namen in Adressen bereitstellt und immer beteiligt ist, wenn ein Hostname in einer URL enthalten ist. Ein Round-Robin-DNS-Server kann einen einzelnen Hostnamen in mehrere IP-Adressen auflösen, sodass Anforderungen an eine einzelne URL (die einen Hostnamen enthält) tatsächlich auf verschiedene Webserver verweisen. Der Client fordert eine Namensauflösung für den Hostnamen an, erhält aber de facto unterschiedliche IP-Adressen, wodurch die Auslastung auf die Webserver verteilt wird. In einer einfachen Konfiguration durchläuft der Round-Robin-DNS-Server die Liste verfügbarer Server.

1.1.3 Skalierbarkeit und Verfügbarkeit

Skalierbarkeit in einem Cluster bedeutet, dass ein Administrator die Kapazität der Anwendung dynamisch erhöhen kann, um die Anforderungen zu erfüllen, ohne den Dienst zu unterbrechen oder negativ zu beeinflussen. Mithilfe von Clustern können Administratoren Knoten aus einem Cluster entfernen, um Komponenten wie zum Beispiel Arbeitsspeicher zu aktualisieren, oder dem Cluster Knoten hinzufügen, ohne den Cluster selbst außer Betrieb setzen zu müssen.

1.2 Terminologie

Jeder Anwendungsserveranbieter verwendet spezifische Terminologie, die im Folgenden definiert wird, um Verwirrung zu vermeiden.

1.2.1 JBoss

Server: Eine Instanz einer Java™ Virtual Machine (JVM™).

Cluster: Eine logische Gruppierung mehrerer Anwendungsserver zu Zwecken der Administration, Anwendungsbereitstellung, des Lastenausgleichs und des Failover.

1.2.2 WebLogic

Server: Eine Instanz einer Java™ Virtual Machine (JVM™).

Machine: Ein physisches System, das mindestens eine Instanz von WebLogic Server ausführt.

Cluster: Eine logische Gruppierung mehrerer Anwendungsserver innerhalb einer Domäne zu Zwecken der Administration, Anwendungsbereitstellung, des Lastenausgleichs und des Failover.

1.2.3 WebSphere

Server: Eine Instanz einer Java™ Virtual Machine (JVM™).

Node: Ein physisches System, das mindestens eine Instanz von WebSphere Application Server ausführt.

Cell: Eine logische Gruppierung mehrerer Knoten zu Administrationszwecken.

Federation: Das Aufnehmen eines eigenständigen WebSphere-Node in eine WebSphere-Cell.

Cluster: Eine logische Gruppierung mehrerer Anwendungsserver innerhalb einer Cell zu Zwecken der Administration, Anwendungsbereitstellung, des Lastenausgleichs und des Failover.

1.3 AEM forms Clustering

Wenn Sie eine AEM forms-Komponente auf einem Anwendungsservercluster installieren, sollten Sie Folgendes wissen:

- Jeder Anwendungsserver im Cluster, auf dem AEM forms bereitgestellt wird, muss eine homogene Topologie besitzen (d. h. alle Knoten im Cluster müssen identisch konfiguriert sein). Sie können sicherstellen, dass alle Module identisch konfiguriert sind, indem Sie Laufzeiteigenschaften im Einzelinstallations-Testbereich konfigurieren.
- Die Konfiguration wird mithilfe des Einzelentitätsansatzes bereitgestellt, d. h. alle Knoten in einem Cluster werden so bereitgestellt wie auf einem Einzelknoten.

Das Einrichten einer Clusterumgebung für AEM forms besteht aus folgenden Aufgaben:

1.3.1 AEM forms Clustering auf JBoss

Sie müssen zum Bereitstellen von AEM forms auf einem JBoss Application Server-Cluster folgende Aufgaben ausführen:

- 1 Bereiten Sie die Hardware gemäß Ihren Anforderungen vor.
- 2 Installieren Sie das Betriebssystem und aktualisieren Sie es mit allen notwendigen Patches und Service Packs.
- 3 Installieren und konfigurieren Sie den Datenbankserver.
- 4 Installieren und konfigurieren Sie das Anwendungsservercluster.
- 5 Installieren Sie AEM forms-Module und konfigurieren Sie AEM forms für die Bereitstellung.
- 6 Stellen Sie AEM forms auf JBoss Application Server bereit.

1.3.2 AEM forms Clustering auf WebLogic

Sie müssen zum Bereitstellen von AEM forms auf einem WebLogic Server-Cluster folgende Aufgaben ausführen:

- 1 Bereiten Sie die Hardware gemäß Ihren Anforderungen vor.
- 2 Installieren Sie das Betriebssystem und aktualisieren Sie es mit allen notwendigen Patches und Service Packs.
- 3 Installieren und konfigurieren Sie den Datenbankserver.
- 4 Installieren und konfigurieren Sie das Anwendungsservercluster.
- 5 Installieren Sie AEM forms-Module und konfigurieren Sie AEM forms für die Bereitstellung.
- 6 Stellen Sie AEM forms auf dem WebLogic-Server bereit.

1.3.3 AEM forms Clustering auf WebSphere

Sie müssen zum Bereitstellen von AEM forms auf einem WebSphere Application Server-Cluster folgende Aufgaben ausführen:

- 1 Bereiten Sie die Hardware gemäß Ihren Anforderungen vor.
- 2 Installieren Sie die WebSphere Application Server Network Deployment-Software auf dem Server, der den Cluster verwaltet.

Hinweis: Wählen Sie die Standardumgebung (Zelle), wenn Sie Deployment Manager und ein Anwendungsserverprofil auf demselben Computer installieren möchten.

- 3 Installieren Sie WebSphere Application Server Network Deployment-Software auf einem anderen Server, der als Clusterknoten verwendet wird.

Hinweis: In den Schritten 2 und 3 können Sie auch „Keine“ auswählen und Deployment Manager- oder Anwendungsserverprofile später mit dem Befehlszeilen-Dienstprogramm erstellen.

Hinweis: Verwenden Sie für Cluster ausschließlich die Network Deployment-Software. Die Base Software-Version wird nur für eigenständige Anwendungsserver verwendet. Gemäß der Endbenutzerlizenzvereinbarung von IBM dürfen Benutzer Base Software nicht mit Network Deployment-Software verwenden.

- 4 Installieren Sie HTTP Server (IBM HTTP Server) und starten Sie den Webserver.
- 5 Installieren Sie WebSphere Update Installer auf jedem Server/Knoten und stellen Sie das erforderliche Fixpack bereit.

Vor der Bereitstellung von AEM forms müssen Sie auf die unterstützte Version von WebSphere aktualisieren. Weitere Informationen dazu finden Sie unter [Unterstützte Plattformen für AEM forms](#). Rufen Sie für Installationsanweisungen zu Fix-Paketen und den Zugriff auf WebSphere-Updates die [WebSphere-Support-Website](#) auf.

- 6 Installieren Sie Websphere Installation Manager auf jedem Server/Knoten und stellen Sie JPA 2.0 bereit (falls es nicht bereits auf WebSphere installiert ist).

Installieren Sie WebSphere Feature Pack for JPA 2.0. Informationen zum Herunterladen und Installieren finden Sie auf der [WebSphere-Support-Website](#) im Abschnitt zum Installieren und Deinstallieren des Feature Packs.

- 7 Richten Sie den Cluster ein:

- Starten Sie Deployment Manager.
- Starten Sie alle Server auf allen Knoten.
- Verbinden Sie die Knoten mit Deployment Manager.
- Erstellen Sie den Cluster mit vorhandenen Anwendungsserverprofilen oder erstellen Sie ein Mitglied mit der Anwendungsservervorlage.
- Starten Sie den Cluster.

- 8 Konfigurieren Sie die Clusterressourcen.

- 9 (Für manuelle Bereitstellung) Stellen Sie Anwendungen auf dem Cluster- sowie auf dem HTTP-Server bereit.

- 10 Generieren Sie das WebSphere HTTP-Plug-In und übertragen Sie es.

1.4 Unterstützte Topologien

In den folgenden Abschnitten werden verschiedene Topologien diskutiert, sowohl mit als auch ohne Cluster, die zum Einsatz kommen können. Zusätzliche Informationen zum Konfigurieren des Anwendungsservers in einem Cluster finden Sie auf der Website für Ihren Anwendungsserver:

- (WebLogic Server) [Unterstützte Konfigurationen für WebLogic Platform](#)
- (WebSphere Application Server) [WebSphere Application Server \(Distributed platforms and Windows\)](#)
- (JBoss Application Server) [High Availability Enterprise Services via JBoss Clusters](#)

1.4.1 Kombinierte Web-, Anwendungs- und Datenbankserver

Diese Topologie besteht aus einem Webserver, einem Anwendungsserver und einem Datenbankserver auf demselben Knoten. Bei dieser Topologie handelt es sich um die einfachste, die nur für die Bereitstellung verwendet werden darf.

1.4.2 Kombinierte Web- und Anwendungsserver mit gesondertem Datenbankserver

Diese Topologie kommt für den Produktionseinsatz infrage, wenn die Belastung der Benutzerschnittstelle (einschließlich der Webstufe) minimal und die Anzahl der Benutzer gering ist.

Die Kombination der Web- mit den Anwendungsservern bedeutet, dass alle EJB(Enterprise JavaBeans™)-Suchen lokal ausgeführt werden, sodass der Aufwand durch die Ausführung einer Remotesuche verringert wird. Darüber hinaus verringert diese Topologie die Netzwerkbelastung eines Round-Trips zwischen der Webstufe und der Anwendungsstufe.

Wenn sich aber beide Server auf demselben Knoten befinden, sind beide Stufen bei einer Beschädigung der Webstufe nicht verfügbar. Kommt es bei der Webstufe zu hohen Lasten, wird die Anwendungsserververarbeitung beeinträchtigt und umgekehrt. Benutzerreaktionszeiten sind normalerweise in Situationen betroffen, wenn Benutzer spürbar darauf warten müssen, bis ein Paket zurückgegeben wird, weil alle Serverressourcen (d. h. CPU und/oder Arbeitsspeicher) vom Anwendungsserver belegt sind.

1.4.3 Einzelner Webserver mit kombiniertem Anwendungs- und Datenbankserver

Die einfachste Topologie, die für eine Produktionsumgebung in Betracht gezogen werden sollte, ist ein Webserver zusammen mit einer Kombination aus Anwendungs- und Datenbankserver. Verwenden Sie diese Topologie nur, wenn Sie sicher sind, dass Ihre Datenbankauslastung minimal sein wird. In einem solchen Szenario stellt der Webserver dann eine Umleitung an den Anwendungsserver zur Verfügung. Die Vorteile dieser Topologie sind niedrige Kosten, geringe Komplexität und keine Notwendigkeit von Lastenausgleich. Die Nachteile dieser Topologie sind geringe Redundanz, niedrige Skalierbarkeit, Unfähigkeit zur Durchführung von Aktualisierungen und Upgrades sowie möglicherweise niedrige Leistung aufgrund zu vieler CPU-Prozesse.

1.4.4 Separate Web-, Anwendungs- und Datenbankserver

Diese Topologie wird am häufigsten in Produktionssystemen verwendet, weil sie die Zuweisung gesonderter Ressourcen zu jeder der Stufen erlaubt. In diesem Fall fungiert der Webserver als Proxy zur Webstufe auf dem Anwendungsserver, der die Webkomponenten beinhaltet. Dieser Grad der Indirektheit bietet zusätzliche Sicherheit durch das Absichern des Anwendungsservers, auch wenn die Sicherheit des Webservers beeinträchtigt ist.

1.4.5 Zusätzliche Webserver hinzufügen

Sie können zur Erzielung von Skalierbarkeit und Failover zusätzliche Webserver hinzufügen. Bei Verwendung mehrerer Webserver muss die Konfigurationsdatei des WebLogic/WebSphere-HTTP-Plug-Ins auf alle Webserver angewendet werden. Wird dies nach Einführung einer neuen Anwendung versäumt, wird wahrscheinlich ein Fehler „404 - Datei nicht gefunden“ erzeugt, wenn ein Benutzer versucht, auf die Webanwendung zuzugreifen.

1.4.6 Zusätzliche Anwendungsserver hinzufügen

Diese Topologie kommt bei den meisten Großproduktionssystemen zum Einsatz, bei denen die Anwendungsserver in Clustern zusammengefasst sind, um auf Grundlage der Topologie sowie von Failover und Lastenausgleich hohe Verfügbarkeit zu erzielen.

Das Clustering mit Anwendungsservern bietet folgende Vorteile:

- Möglichkeit zum Einsatz billigerer Hardwarekonfigurationen bei Erzielung höherer Leistung
- Möglichkeit zur Aktualisierung von Software auf Servern ohne Ausfallzeiten
- Höhere Verfügbarkeit (d. h., bei Ausfall eines Servers übernehmen die anderen Knoten im Cluster die Verarbeitung)
- Möglichkeit zur Ausnutzung der Lastenausgleichsalgorithmen auf dem Webserver (durch Verwendung von Lastenausgleichsmodulen) sowie auf der EJB-Stufe zur Verarbeitung von Anforderungen

AEM forms-Komponenten sind normalerweise CPU-abhängig. Die Leistungssteigerung ist daher höher, wenn weitere Anwendungsserver hinzugefügt werden, anstatt einem vorhandenen Server zusätzlichen Arbeitsspeicher oder Datenträgerspeicherplatz hinzuzufügen.

1.4.7 Mehrere JVMs

Die vertikale Skalierung mehrerer JVMs bietet folgende Vorteile:

Erhöhte Verarbeitungsleistungseffizienz: Eine Instanz eines Anwendungsservers wird in einem einzelnen JVM-Vorgang ausgeführt. Die inhärenten Parallelitätseinschränkungen eines JVM-Prozesses verhindern aber, dass dieser den Arbeitsspeicher und die Verarbeitungsleistung von Systemen mit mehreren CPUs vollständig nutzt. Die Erstellung zusätzlicher JVM-Prozesse stellt Multithreadpools zur Verfügung, von denen jeder dem JVM-Prozess entspricht, der dem jeweiligen Anwendungsserverprozess zugeordnet ist. Diese Entsprechung verhindert Parallelitätseinschränkungen und ermöglicht dem Anwendungsserver die Ausnutzung der vollständigen Verarbeitungsleistung des Computers.

Lastenausgleich: Topologien mit vertikaler Skalierung können die Lastverwaltungsfunktionen von WebLogic-Server oder WebSphere Application Server nutzen.

Prozess-Failover: Eine Topologie mit vertikaler Skalierung bietet außerdem Failoverunterstützung zwischen Anwendungsservercluster-Mitgliedern. Wird eine Anwendungsserverinstanz offline geschaltet, setzen die anderen Instanzen auf dem Computer die Verarbeitung von Clientanforderungen weiter fort.

1.5 Nicht unterstützte Topologien

Die folgenden Topologien werden von AEM forms nicht unterstützt.

Trennen des Webcontainers vom EJB-Container

Das Aufteilen der AEM forms-Server in Stufen nach Präsentations-/Geschäftslogik sowie deren Ausführung auf verteilten Computern werden nicht unterstützt.

Geographisch verteilte Konfiguration

Viele Anwendungen ordnen ihre Systeme geographisch an, um die Auslastung besser verteilen zu können und um eine zusätzliche Stufe der Redundanz hinzuzufügen. AEM forms unterstützt diese Konfiguration nicht, weil AEM forms-Komponenten nicht auseinandergenommen werden können, um auf verschiedenen Hosts ausgeführt zu werden. AEM forms wird als monolithische Anwendung bereitgestellt.

Die meisten Informationen zu Ordern in diesem Dokument gelten plattformübergreifend (unter Linux und UNIX wird bei allen Dateinamen und Pfadangaben die Groß- und Kleinschreibung beachtet). Plattformspezifische Informationen werden bei Bedarf aufgeführt.