

Entwicklerleitfaden zur Cloud-Migration mit Lift-and-Shift

Inhaltsverzeichnis

| | | |
|----------|---|-----------|
| 1 | Einleitung | 2 |
| 2 | Migration mit Red Hat Application Migration Toolkit vereinfachen | 3 |
| 2.1 | Eine Java EE App analysieren | 3 |
| 2.2 | Bericht verstehen | 6 |
| 2.3 | Probleme beheben | 8 |
| 2.4 | Build erneut testen | 8 |
| 3 | Projekt in Red Hat JBoss Enterprise Application Platform ausführen | 10 |
| 3.1 | Red Hat JBoss Enterprise Application Platform installieren | 10 |
| 3.2 | Das WildFly Maven Plugin | 10 |
| 3.3 | Anwendung konfigurieren | 10 |
| 3.4 | Anwendung bereitstellen | 10 |
| 3.5 | Anwendung testen | 10 |
| 3.6 | Anwendung herunterfahren | 11 |
| 4 | Monolith auf Red Hat OpenShift Container Platform bereitstellen | 11 |
| 4.1 | Red Hat OpenShift Profil hinzufügen | 11 |
| 4.2 | Red Hat OpenShift Projekt erstellen | 12 |
| 4.3 | Monolith bereitstellen | 14 |
| 4.4 | Anwendung mit Binary Build bereitstellen | 15 |
| 5 | Zusammenfassung | 17 |



Einleitung

Angesichts der zunehmenden Akzeptanz von Cloud-Modellen tendieren immer mehr Organisationen zur cloudnativen Entwicklung. Neue Anwendungen können vollständig mithilfe von Cloud-Modellen und -Services entwickelt werden, die Microservices, autonome Entwicklungsteams, eine agile und kontinuierliche Entwicklung sowie containerisierte und orchestrierte Cloud-Bereitstellungen einsetzen. Leider macht eine Neuprogrammierung aller Legacy-Anwendungen aufgrund des hohen Zeit- und Kostenaufwands meist wenig Sinn.

Um agil und wettbewerbsfähig zu bleiben, müssen Unternehmen vorhandene Java™ Apps migrieren und modernisieren und zu cloudnativer Entwicklung übergehen. Bei den meisten Unternehmen besteht dieser Prozess darin, möglichst viele vorhandene Funktionen und Daten wiederzuverwenden, vorhandene Workloads auf eine moderne Bereitstellungsplattform zu verschieben und schließlich neue Prozesse, Produkte und Technologien anzuwenden.

Das Lift-and-Shift-Modernisierungsmodell ist ein attraktiver erster Schritt hin zur cloudnativen Entwicklung und beinhaltet Folgendes:

- Containerisierung bestehender monolithischer Workloads
- Bereitstellung von Workloads auf einer Platform-as-a-Service (PaaS)
- Beibehaltung externer Integrationen und Daten auf der Legacy-Plattform

Nach der Ausführung dieser Schritte können Entwickler damit beginnen, die monolithische Architektur schrittweise abzubauen und ausgewählte app-interne Funktionen durch Microservices zu ersetzen.

In diesem Leitfaden wird der Lift-and-Shift-Prozess mit Red Hat® Software erläutert, der in Abbildung 1 unten dargestellt ist. Dieser Prozess besteht aus folgenden Schritten:

1. Analysieren einer vorhandenen monolithischen Anwendung mit [Red Hat Application Migration Toolkit](#) und Migrieren zu offenen Standardschnittstellen.
2. Aktualisieren des Codes und der Konfiguration für die Ausführung auf [Red Hat JBoss® Enterprise Application Platform](#).
3. Bereitstellen des Monoliths als Container auf [Red Hat OpenShift® Container Platform](#) mit Features wie automatische Skalierung, integriertes Clustering und Failover zur Optimierung der Anwendung.

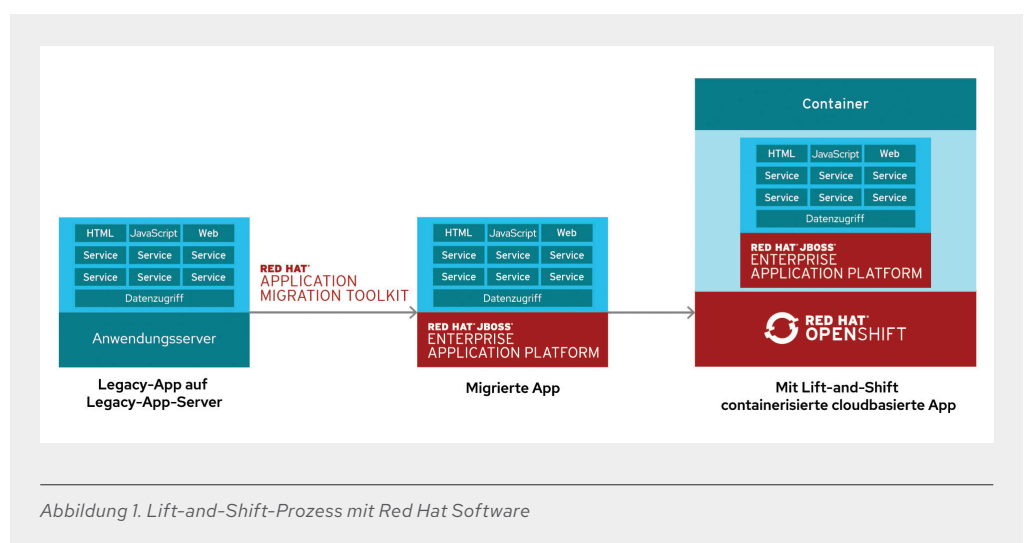


Abbildung 1. Lift-and-Shift-Prozess mit Red Hat Software

Nachdem die Anwendung auf Red Hat OpenShift Container Platform bereitgestellt ist, können bestimmte Funktionen mithilfe zusätzlicher Tools in Microservices konvertiert werden. Bei diesem Prozess kann die Funktionalität für mehr Parallelisierung und Autonomie in kleinere Einheiten aufgeteilt werden. Außerdem können Sie mithilfe der folgenden Tools eine schnellere Wertschöpfung erreichen:

- [Thorntail/MicroProfile](#)
- [Spring Boot](#)
- [Node.js](#)
- [Eclipse Vert.x](#)

Migration mit Red Hat Application Migration Toolkit vereinfachen

Red Hat Application Migration Toolkit ist ein erweiterbares und anpassbares regelbasiertes Tool, das die Migration von Java-Anwendungen vereinfacht. Es wird von Unternehmen in folgenden Bereichen eingesetzt:

- Planen und Schätzen des Arbeitsaufwands
- Identifizieren von Migrationsproblemen und Empfehlen von Lösungen
- Bewerten ausführlicher Berichte
- Verwenden von integrierten Regeln und Migrationspfaden
- Bereitstellen von Erweiterung und Anpassbarkeit von Regeln
- Analysieren von Quellcode oder Anwendungsarchiven

Beim Migrieren von Anwendungen sucht Red Hat Application Migration Toolkit nach häufig verwendeten Ressourcen und kennzeichnet Technologien sowie bekannte Problembereiche. Das Toolkit bietet eine allgemeine Übersicht über die von der Anwendung verwendeten Technologien sowie einen detaillierten Bericht, den Unternehmen beim Beurteilen, Dokumentieren und Migrieren von Java EE Anwendungen zu Red Hat JBoss Enterprise Application Platform verwenden können.

Hinweis: Das Red Hat Application Migration Toolkit kommt normalerweise nur im Rahmen eines umfassenderen Programms zur Anwendungsmigration und -modernisierung zum Einsatz. Die Programme umfassen in der Regel klar definierte und wiederholbare Phasen über mehrere Wochen oder Monate hinweg und können viele Mitarbeiter der jeweiligen Organisation einbeziehen.

Weitere Informationen zur Philosophie und zu den bewährten Methoden von Red Hat finden Sie in der [Dokumentation zu Red Hat Application Migration Toolkit](#) und auf der [Developer-Homepage](#).

Eine Java EE App analysieren

Red Hat Application Migration Toolkit untersucht Anwendungsartefakte, darunter Quellverzeichnisse von Projekten und Anwendungsarchive. Anschließend wird ein HTML-Bericht erzeugt, in dem Bereiche mit Änderungsbedarf hervorgehoben werden. Das Toolkit kann verwendet werden, um Java-Anwendungen von früheren Versionen von Red Hat JBoss Enterprise Application Platform (EAP) oder von anderen Anwendungsserver-Plattformen wie Oracle WebLogic Server oder IBM® WebSphere Application Server zu migrieren.

Für die Installation und Verwendung von Red Hat Application Migration Toolkit können Sie verschiedene Methoden verwenden:

- **Webkonsole.** Die Webkonsole für Red Hat Application Migration Toolkit ermöglicht einem Team von Entwicklern, ihren Migrations- und Modernisierungsaufwand für eine große Anzahl von Anwendungen zu beurteilen und zu priorisieren. Sie kann Anwendungen für Analysen in Projekte gruppieren. Die Ergebnisse werden in einer ganzen Reihe von Berichten zusammengefasst.

- **Befehlszeilenschnittstelle (CLI).** Die CLI ist ein Befehlszeilen-Tool, das Entwicklern die Beurteilung und Priorisierung von Schritten zur Migration und Modernisierung von Anwendungen ermöglicht. Es liefert zahlreiche Berichte, in denen die Ergebnisse der Analysen hervorgehoben werden. Aufgrund der Einfachheit der CLI ist sie das ideale Tool für das Analysieren einzelner Anwendungen.
- **Eclipse-Plugin.** Das Eclipse-Plugin für Red Hat Application Migration Toolkit bietet Unterstützung direkt in Eclipse und Red Hat JBoss Developer Studio für Entwickler, die Änderungen im Rahmen einer Migrations oder Modernisierung vornehmen. Es analysiert Ihre Projekte mithilfe des Red Hat Application Migration Toolkits, markiert Migrationsprobleme im Quellcode, erteilt Anweisungen zur Behebung der Probleme und ersetzt wenn möglich sofort automatisch den Code.

Alle diese Zugriffsmethoden können von der Webseite für das [Red Hat Application Migration Toolkit](#) heruntergeladen werden. Installationsanweisungen finden Sie in der Dokumentation für die jeweilige Zugriffsmethode. Für Szenarien mit mehreren Nutzern ist die Webkonsole eine gute Option. Dieses Dokument bezieht sich der Einfachheit halber auf die CLI. Die nachfolgenden Schritte beziehen sich auf die Ausführung von Red Hat Application Migration Toolkit.

1. Red Hat Application Migration Toolkit CLI prüfen

Nachdem Sie die Red Hat Application Migration Toolkit CLI installiert haben, prüfen Sie die ordnungsgemäße Installation des Tools, indem Sie folgenden Befehl ausführen:

```
$ ${HOME}/rhamt-cli-4.0.0.Beta4/bin/rhamt-cli --version
```

Die Ausgabe sollte in etwa wie folgt aussehen:

```
Using RHAMT at /root/rhamt-cli-4.0.0.Beta4
> Red Hat Application Migration Toolkit (RHAMT) CLI, version 4.0.0.Beta4.
```

2. Red Hat Application Migration Toolkit CLI für Ihr Projekt ausführen

Die Red Hat Application Migration Toolkit CLI verfügt über zahlreiche Optionen, mit denen Sie die Ausführung des Tools steuern können. Im folgenden Beispiel wird die CLI für ein Projekt namens „monolith“ ausgeführt. Außerdem wird ein gleichnamiger Bericht generiert. Ersetzen Sie `~/projects/monolith` im nachfolgenden Beispiel durch den Namen Ihres eigentlichen untersuchten Projekts.

```
$ ~/rhamt-cli-4.0.0.Beta4/bin/rhamt-cli \
  --sourcemode \
  --input ~/projects/monolith \
  --output ~/rhamt-reports/monolith \
  --overwrite \
  --source weblogic \
  --target eap:7 \
  --packages com.redhat weblogic
```

Beachten Sie die Verwendung der Optionen `--source` und `--target`. Mit diesen Optionen können Sie bestimmte von Red Hat Application Migration Toolkit unterstützte Migrationspfade als Ziele definieren. Folgende Migrationspfade sind verfügbar:

- Oracle WebLogic Server
- IBM WebSphere Application Server

- Red Hat JBoss EAP 5/6/7

Fahren Sie erst fort, wenn die Ausführung des Toolkits abgeschlossen ist. Wenn die Ausführung abgeschlossen ist, wird beispielsweise folgende Ausgabe angezeigt:

```
Report created: ~/rhamt-reports/monolith/index.html
```

3. Ergebnisse anzeigen

Um den Bericht anzuzeigen, navigieren Sie im lokalen Browser zur folgenden Datei:

```
~/rhamt-reports/monolith/index.html
```

Die Landing Page für den Bericht wird angezeigt, wie in Abbildung 2 dargestellt. Die Landing Page des Hauptberichts listet die verarbeiteten Anwendungen sowie die aufgetretenen Probleme auf. Jede Zeile auf der Seite enthält eine allgemeine Übersicht über die Story Points, die Anzahl der Vorfälle und die in der Anwendung verwendeten Technologien.

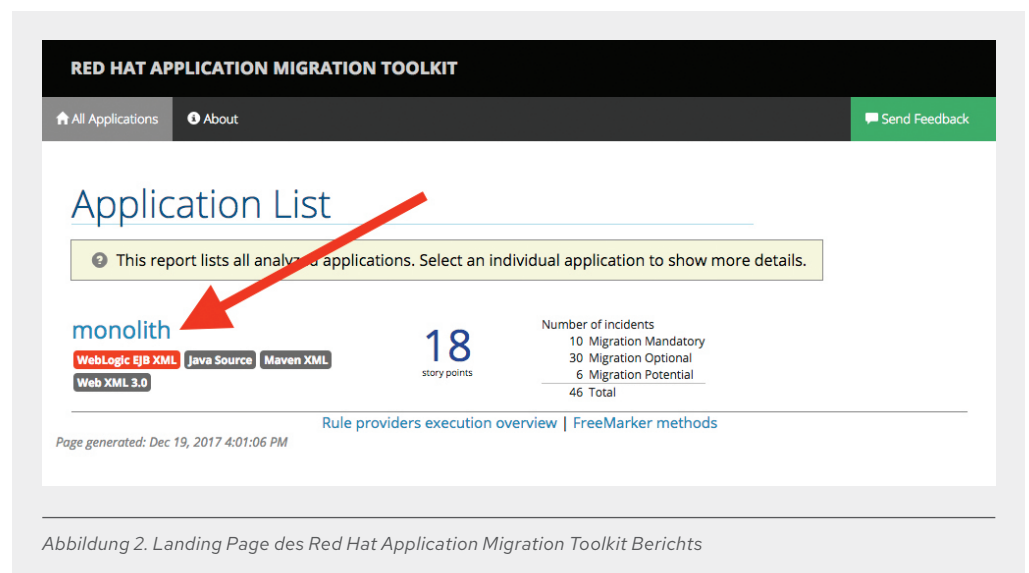


Abbildung 2. Landing Page des Red Hat Application Migration Toolkit Berichts

Wenn Sie in diesem Beispiel auf den Link „monolith“ klicken, wird ein Dashboard für das Projekt aufgerufen (siehe Abbildung 3).

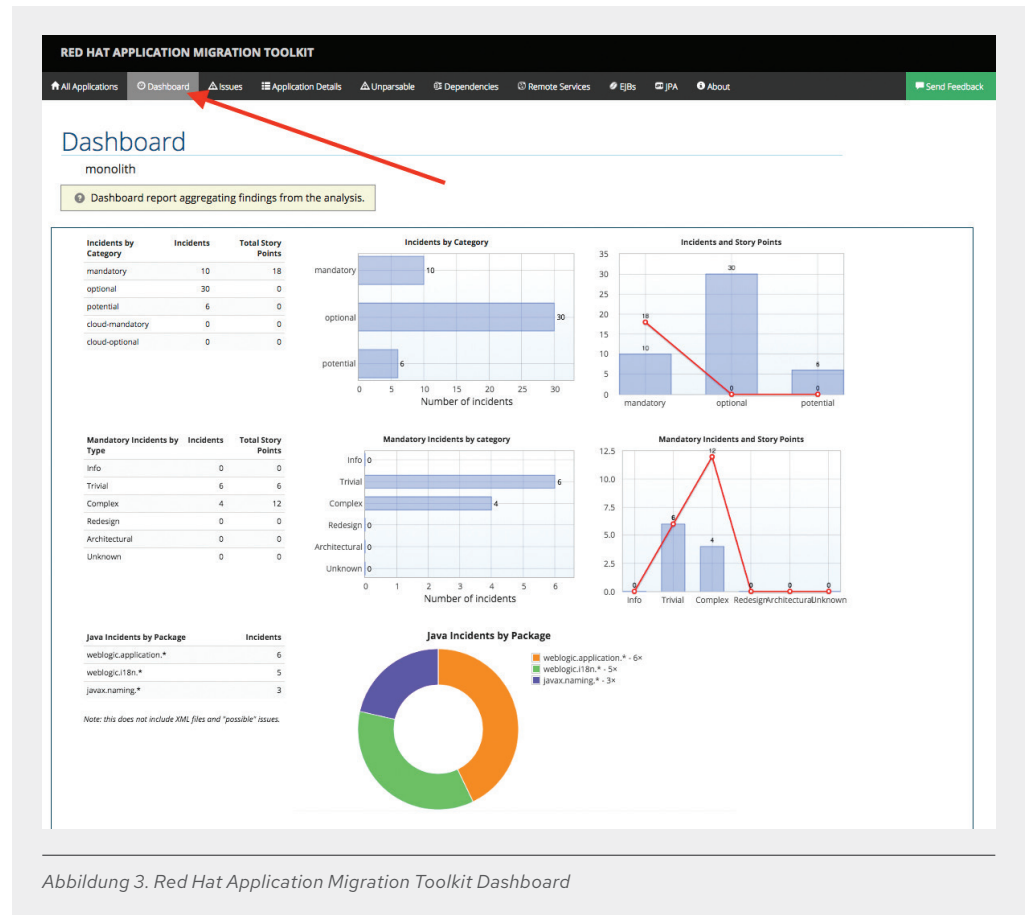


Abbildung 3. Red Hat Application Migration Toolkit Dashboard

Den Bericht verstehen

Das Dashboard bietet eine Übersicht über den gesamten Migrationsaufwand für die Anwendung. Folgendes wird zusammengefasst:

- Die Vorfälle und Story Points nach Kategorie
- Die Vorfälle und Story Points nach Arbeitsaufwand für die empfohlenen Änderungen
- Die Vorfälle nach Paket

Hinweis: Story Points sind eine abstrakte Messgröße, die häufig in der agilen Softwareentwicklung verwendet werden, um den relativen Arbeitsaufwand für die Implementierung eines Features oder einer Änderung zu schätzen. Red Hat Application Migration Toolkit verwendet Story Points, um den Arbeitsaufwand auszudrücken, der für die Migration bestimmter Anwendungsstrukturen - und der Anwendung als Ganzes - erforderlich ist. Der Arbeitsaufwand kann je nach Größe und Komplexität der zu migrierenden Anwendung(en) stark variieren.

Über die Menüpunkte oben können Sie auf mehrere Unterseiten zugreifen.

- **All Applications** enthält eine Liste aller gescannten Anwendungen.
- **Dashboard** bietet eine Übersicht für eine bestimmte Anwendung.
- **Issues** fasst alle Probleme zusammen, die Ihre Aufmerksamkeit erfordern.
- **Application Details** bietet eine ausführliche Übersicht über alle Ressourcen innerhalb der Anwendung, die bei der Migration berücksichtigt werden müssen.
- **Unparsable** zeigt alle Dateien an, die Red Hat Application Migration Toolkit nicht im erwarteten Format parsen konnte. Beispiel: Eine Datei mit dem Suffix „.xml“ oder „.wsdl“ wird als XML-Datei angesehen. Wenn der XML-Parser nicht erfolgreich ausgeführt werden kann, wird das Problem in diesem Abschnitt gemeldet. Außerdem wird angegeben, wo die jeweilige Datei aufgeführt wird.
- **Dependencies** zeigt alle in Java-Paketen vorhandenen Abhängigkeiten innerhalb der Anwendung an.
- **Remote Services** gibt alle Referenzen zu Remote-Services an, die innerhalb der Anwendung gefunden wurden.
- **EJBs** enthält eine Liste der Enterprise Java Beans, die innerhalb der Anwendung gefunden wurden.
- **JBPM** listet alle Ressourcen zu Java Business Process Management auf, die bei der Analyse ermittelt wurden.
- **JPA** beinhaltet Details zu allen Ressourcen zur Java Persistence API, die in der Anwendung gefunden wurden.
- **About** gibt die aktuelle Version von Red Hat Application Migration Toolkit an und bietet hilfreiche Links zur weiteren Unterstützung.

Hinweis: Je nachdem, welche Probleme tatsächlich im Projekt erkannt wurden, werden einige der oben genannten Abschnitte möglicherweise nicht angezeigt.

Der Red Hat Application Migration Toolkit Bericht ist jetzt verfügbar, und Sie können mit der Migration der Anwendung beginnen.

Probleme beheben

Die Registerkarte mit dem Problebericht („Issues“) in Abbildung 4 enthält eine ausführliche Auflistung aller bekannten Probleme in der untersuchten Anwendung. Sie beinhaltet hilfreiche Links, damit Nutzer das Problem besser verstehen, und bietet Migrations- und Modernisierungsanleitungen.

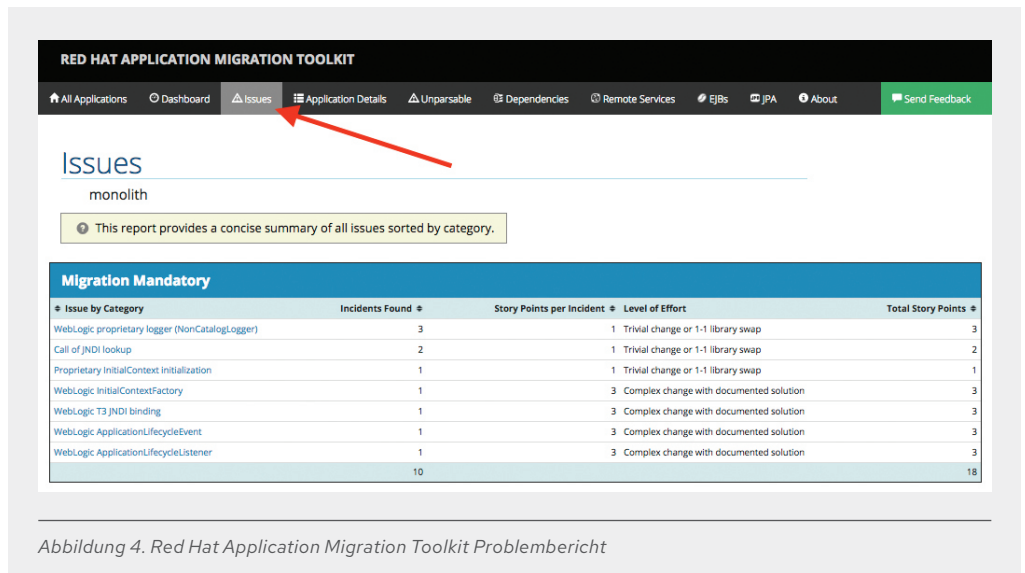


Abbildung 4. Red Hat Application Migration Toolkit Problebericht

Viele Anwendungen beinhalten plattformspezifischen Code, der für die Verwendung standardmäßiger Java EE Schnittstellen aktualisiert werden muss, darunter:

- **Startup-Code** zur Ausführung von Funktionen oder zum Planen von Jobs, nachdem die App gestartet oder gestoppt wurde.
- **Logging-Code** mit plattformspezifischen Protokollierungsfunktionen.
- **Messaging-Code** mit veralteten oder plattformspezifischen Mechanismen.

In jedem Fall enthält der Problebericht Empfehlungen zu möglichen Lösungen. Unter realistischen Bedingungen sind im Rahmen der Migration zusätzliche Überlegungen erforderlich, um diese Probleme zu beheben. Allerdings wird die Portierbarkeit des Codes durch Implementieren dieser Änderungen erhöht.

Nachdem die erforderlichen Änderungen vorgenommen wurden, können Sie die App mit Maven erstellen und paketieren, um sicherzustellen, dass der geänderte Code weiterhin kompiliert werden kann:

```
$ mvn clean package
```

Wenn der Code erfolgreich erstellt wurde (`BUILD SUCCESS` wird angezeigt), können Sie das nächste Problem im Problebericht angehen.

Build erneut testen

Wenn alle Probleme behoben wurden, wird Red Hat Application Migration Toolkit erneut ausgeführt, um zu prüfen, ob die Migration erfolgreich war.

1. Red Hat Application Migration Toolkit CLI für das Projekt ausführen

Führen Sie den unten angegebenen Befehl aus, um die alten Build-Artefakte zu bereinigen, die Red Hat Application Migration Toolkit CLI erneut auszuführen und das neue Projekt zu analysieren. Ersetzen Sie auch hier `~/projects/monolith` durch den Speicherort Ihres Projekts.

```
$ mvn clean && \  
~/rhamt-cli-4.0.0.Beta4/bin/rhamt-cli \  
  --sourcemode \  
  --input ~/projects/monolith \  
  --output ~/rhamt-reports/monolith \  
  --overwrite \  
  --source weblogic \  
  --target eap:7 \  
  --packages com.redhat weblogic
```

Nach Abschluss des Prozesses sollte in der CLI Folgendes angezeigt werden:

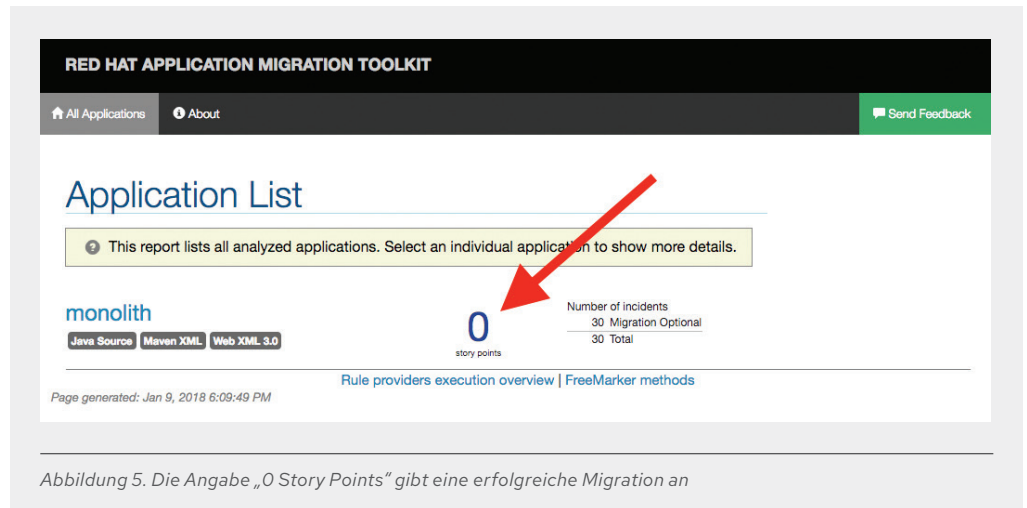
```
Report created: /root/rhamt-reports/monolith/index.html
```

2. Ergebnisse anzeigen

Aktualisieren Sie die lokale Berichtswebseite unter:

```
~/rhamt-reports/monolith/index.html
```

Nach der erfolgreichen Migration sollte die jeweilige Anwendung null (0) Story Points melden. Das heißt, dass die App erfolgreich modernisiert wurde und jetzt zu Red Hat JBoss EAP verschoben werden kann.



The screenshot shows the Red Hat Application Migration Toolkit web interface. The header includes 'RED HAT APPLICATION MIGRATION TOOLKIT', 'All Applications', 'About', and 'Send Feedback'. The main content area is titled 'Application List' and contains a message: 'This report lists all analyzed applications. Select an individual application to show more details.' Below this, the application 'monolith' is listed with links for 'Java Source', 'Maven XML', and 'Web XML 3.0'. A large '0' is displayed next to 'story points', with a red arrow pointing to it. To the right, a table shows 'Number of incidents' with '30 Migration Optional' and '30 Total'. At the bottom, it says 'Page generated: Jan 9, 2018 6:09:49 PM' and provides links for 'Rule providers execution overview' and 'FreeMarker methods'.

Abbildung 5. Die Angabe „0 Story Points“ gibt eine erfolgreiche Migration an

Projekt in Red Hat JBoss Enterprise Application Platform ausführen

Nachdem die App jetzt für die Verwendung von Java EE Standardschnittstellen aktualisiert wurde, können Sie sie bereitstellen, testen und einige der Features von Red Hat JBoss EAP ausprobieren.

Red Hat JBoss Enterprise Application Platform installieren

Laden Sie das neueste Release von [Red Hat JBoss EAP](#) herunter. Installieren Sie Red Hat JBoss EAP anschließend lokal, indem Sie den folgenden Befehl im Terminalfenster ausführen. Ändern Sie den Dateinamen so, dass er mit der heruntergeladenen Version übereinstimmt.

```
$ unzip -d $HOME $HOME/jboss-eap-7.2.0.zip
```

Legen Sie im nächsten Schritt die JBOSS_HOME-Umgebungsvariable fest:

```
$ export JBOSS_HOME=$HOME/jboss-eap-7.2.0
```

So leicht können Sie Red Hat JBoss EAP installieren.

Das WildFly Maven Plugin

Red Hat JBoss EAP enthält das Tool [wildfly-maven-plugin](#), mit dem Sie die Plattform direkt in Apache Maven starten, stoppen, bereitstellen und konfigurieren können. Nachdem das Maven-Plugin-Tool konfiguriert wurde, kann es verwendet werden, um:

- zwischen Java EE Full Profile und Java EE Web Profile auszuwählen.
- Datenbankressourcen zu konfigurieren.
- Services wie Java Message Service Topic zu konfigurieren.

Alle diese Funktionen werden in der Datei `pom.xml` konfiguriert.

Anwendung konfigurieren

Nachdem Sie Red Hat Application Migration Toolkit angewendet haben, basiert die Anwendung größtenteils auf Standards. Sie können die Anwendung wie folgt konfigurieren, dass Red Hat JBoss EAP gestartet wird, um die Anwendung zu starten, Ressourcen hinzuzufügen und wieder herunterzufahren:

```
$ export JBOSS_HOME=$HOME/jboss-eap-7.2.0 ; \ mvn wildfly:start wildfly:add-resource wildfly:shutdown
```

Am Ende dieses Schritts sollte die Meldung `BUILD SUCCESS` angezeigt werden.

Hinweis: Wir verwenden „wildfly:start“ und „wildfly:shutdown“, weil für den Befehl „add-resource“ ein ausgeführter Anwendungsserver erforderlich ist. Nachdem wir diese Ressource hinzugefügt haben, müssen wir diesen Befehl nicht wieder ausführen.

Die Anwendung bereitstellen

Jetzt können Sie die Anwendung bereitstellen:

```
$ export JBOSS_HOME=$HOME/jboss-eap-7.2.0 ; mvn wildfly:run
```

Wenn der Server gestartet ist, sollte Folgendes angezeigt werden:

```
Deployed "ROOT.war" (runtime-name: "ROOT.war")
```

Die Anwendung testen

Rufen Sie die Anwendung auf, indem Sie die folgende URL in Ihrem Browser laden:

```
http://localhost:8080
```

Eventuelle Warnungen werden in der Konsolenausgabe angezeigt.

Die Anwendung herunterfahren

Bevor Sie fortfahren, beenden Sie die Anwendung, indem Sie im Terminalfenster STRG+C eingeben.

Monolith auf Red Hat OpenShift Platform bereitstellen

Nachdem die Anwendung gestartet ist und in Red Hat JBoss EAP ausgeführt wird, können Sie den neu aktualisierten Monolith zu Red Hat OpenShift Container Platform verschieben. Für die folgenden Schritte wird vorausgesetzt, dass Sie Zugriff auf einen Red Hat OpenShift Container Platform Cluster haben. Wenn Sie keinen Zugriff auf einen Cluster haben, können Sie [Red Hat OpenShift Container Platform lokal installieren](#).

Red Hat OpenShift Profil hinzufügen

Öffnen Sie die Datei `pom.xml`, und fügen Sie ein Red Hat OpenShift Profil hinzu.

```
<profile>
  <id>openshift</id>
  <build>
    <plugins>
      <plugin>
        <artifactId>maven-war-plugin</artifactId>
        <version>2.6</version>
        <configuration>
          <webResources>
            <resource>
              <directory>${basedir}/src/main/webapp/WEB-INF</directory>
              <filtering>true</filtering>
              <targetPath>WEB-INF</targetPath>
            </resource>
          </webResources>
          <outputDirectory>deployments</outputDirectory>
          <warName>ROOT</warName>
        </configuration>
      </plugin>
    </plugins>
  </build>
</profile>
```

Abbildung 6. Red Hat OpenShift Profil (Datei „pom.xml“)

Ein Red Hat OpenShift Projekt erstellen

Öffnen Sie zunächst die Webkonsole von Red Hat OpenShift Container Platform.

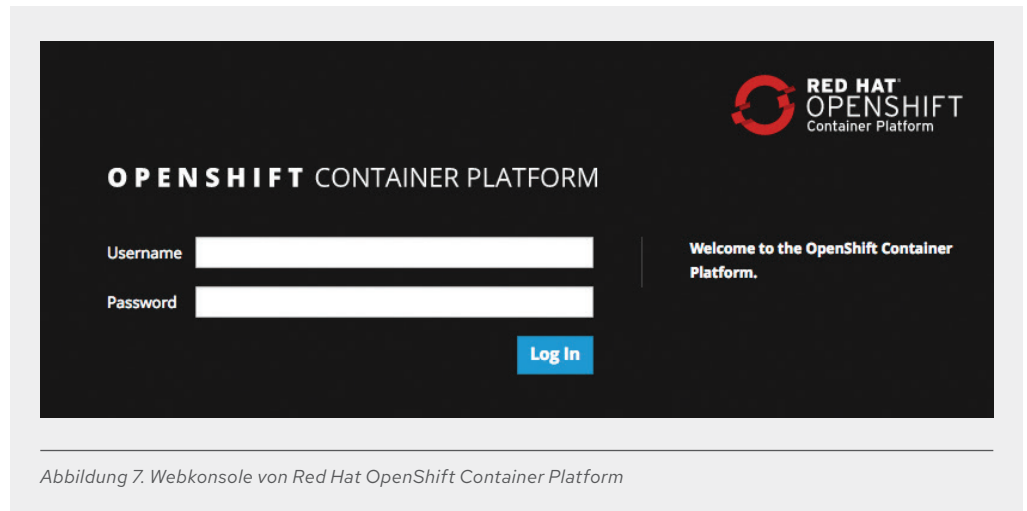


Abbildung 7. Webkonsole von Red Hat OpenShift Container Platform

Melden Sie sich mit Ihren Red Hat OpenShift Zugangsdaten an:

- Username: XXXXXX
- Password: XXXXXX

Die Landing Page von Red Hat OpenShift Container Platform wird angezeigt:

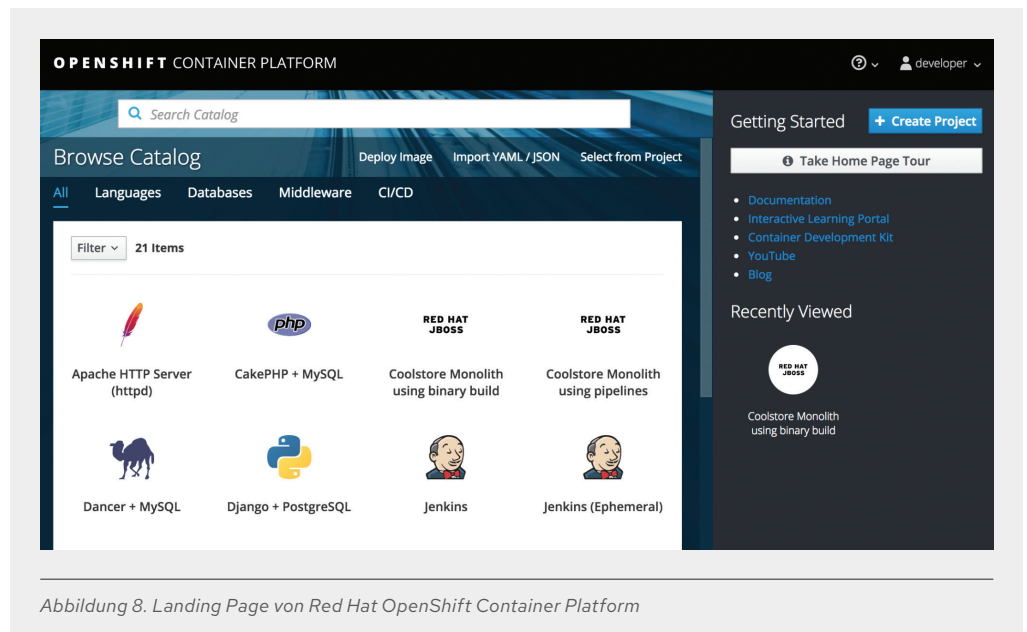


Abbildung 8. Landing Page von Red Hat OpenShift Container Platform

Klicken Sie auf „Create Project“, füllen Sie die Felder aus, und klicken Sie auf „Create“. In diesem Beispiel lautet der Name der Anwendung „coolstore“.

- Name: coolstore-dev
- Display Name: Coolstore Monolith - Dev
- Description: leer lassen

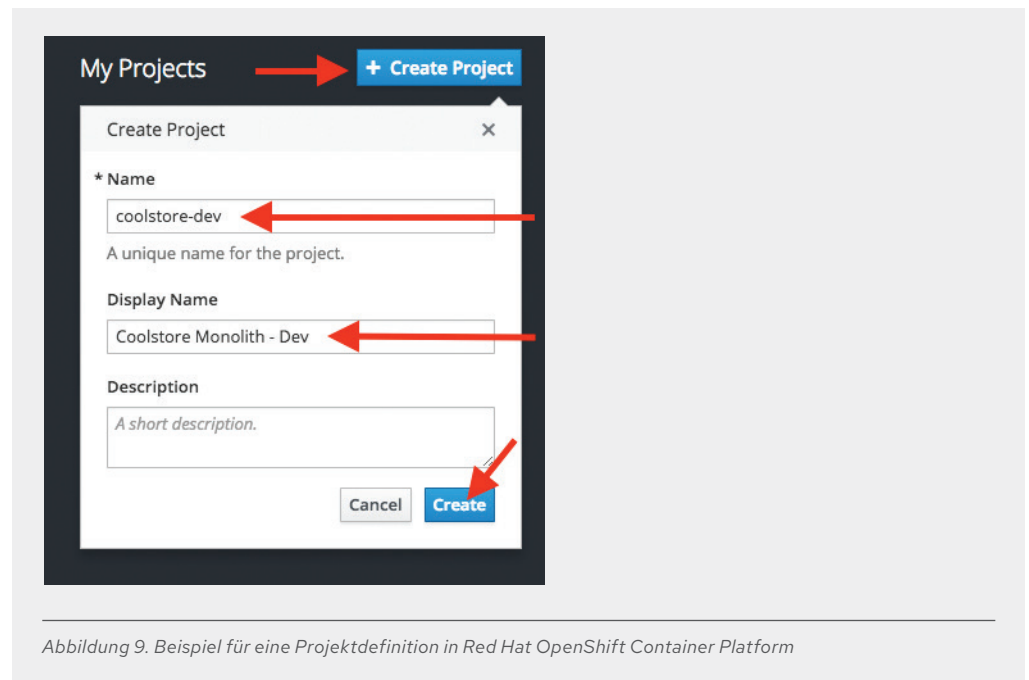


Abbildung 9. Beispiel für eine Projektdefinition in Red Hat OpenShift Container Platform

Klicken Sie auf den Namen des neu erstellten Projekts:

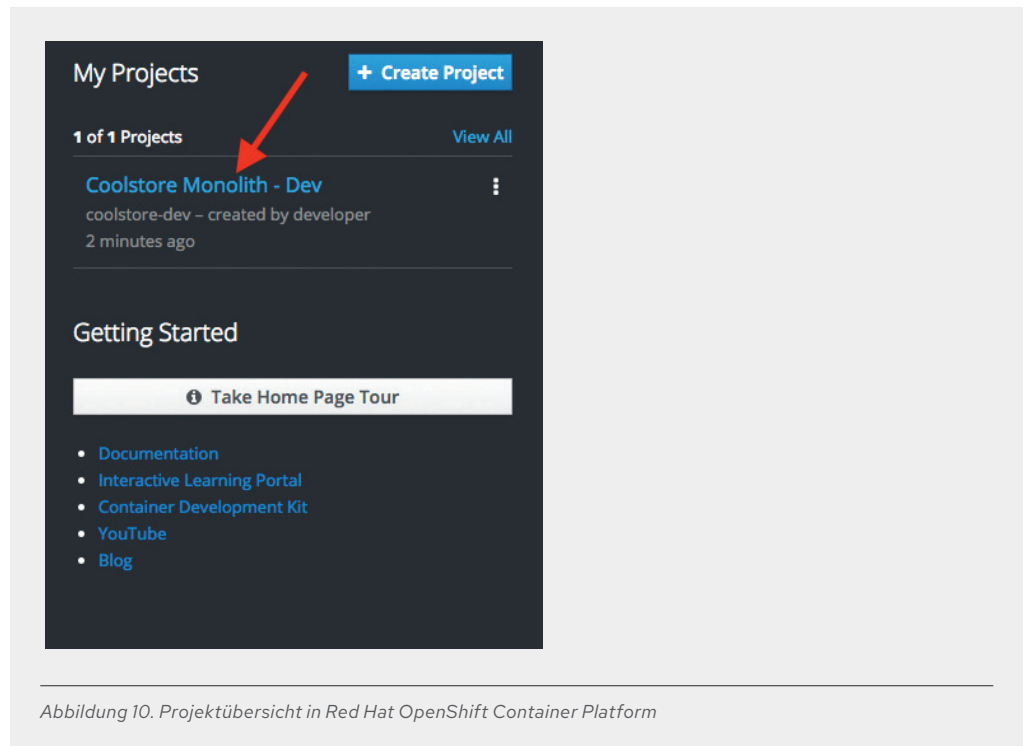


Abbildung 10. Projektübersicht in Red Hat OpenShift Container Platform

Die Projektübersicht wird aufgerufen. Im Moment wird hier noch kein Projekt angezeigt, aber das wird sich gleich ändern.

Monolith bereitstellen

Verwenden Sie die CLI, um die Komponenten des Monoliths bereitzustellen. Um die Vorlage des Monoliths über die CLI bereitzustellen, führen Sie die nachstehenden Befehle aus.

Wechseln Sie zunächst zu dem Entwicklerprojekt, das Sie zuvor erstellt haben:

```
$ oc project coolstore-dev
```

Stellen Sie abschließend die Vorlage bereit:

```
$ oc new-app coolstore-monolith-binary-build
```

In diesem Schritt werden alle von der Anwendung erforderlichen Datenbanken sowie Red Hat JBoss EAP bereitgestellt. Allerdings wird kein Build für die Anwendung gestartet.

Öffnen Sie anschließend die Seite mit der Übersicht zum Monolith unter:

`https://$OPENSHIFT_MASTER/console/project/coolstore-dev/`, und prüfen Sie, ob die Elemente der Monolith-Vorlage erstellt wurden.

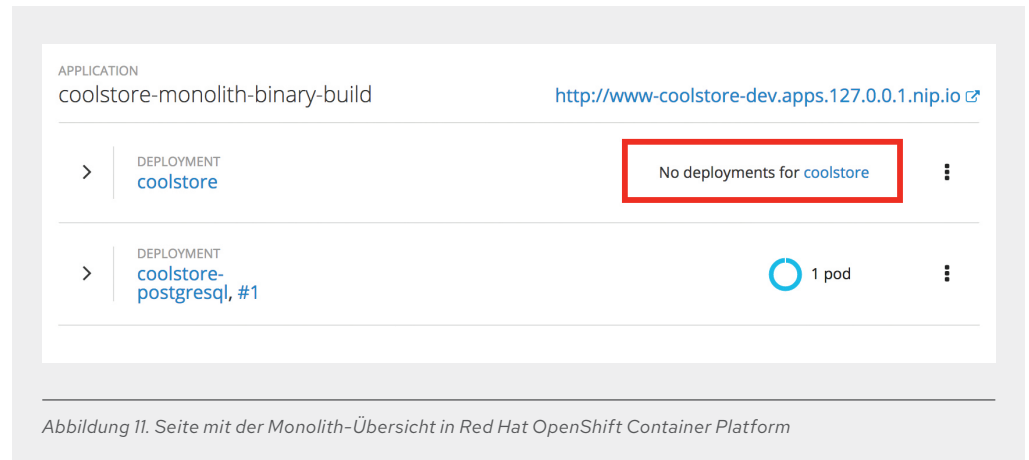


Abbildung 11. Seite mit der Monolith-Übersicht in Red Hat OpenShift Container Platform

Die Komponenten, die in der Projektübersicht bereitgestellt wurden, werden angezeigt. Allerdings deutet „No deployments for coolstore“ darauf hin, dass für „coolstore“ keine Bereitstellungen vorhanden sind. Sie haben das im vorherigen Schritt erstellte Container-Image noch nicht bereitgestellt. Dies erfolgt im nächsten Schritt.

Anwendung mit Binary Build bereitstellen

In diesem Entwicklungsprojekt verwenden wir einen Prozess namens „Binary Builds“. Dieser Prozess erstellt Builds lokal und lädt nur das Artefakt (z. B. die `.war`-Datei) hoch, anstatt auf ein öffentliches Git Repository zu verweisen und mit dem Build-Prozess „source-to-image“ (S2I) das Herunterladen und Erstellen eines Container-Images zu veranlassen. Die binäre Entwicklung beschleunigt den Build-Prozess beträchtlich.

Erstellen Sie zunächst erneut einen Build des Projekts mithilfe des OpenShift Maven-Profiles. Dadurch wird eine Binärdatei zur Verwendung mit Red Hat OpenShift Container Platform erstellt (dies ist noch kein Container-Image, sondern nur die `.war`-Datei). Wir erstellen das Projekt mit der Befehlszeile `oc`.

Erstellen Sie einen Build des Projekts:

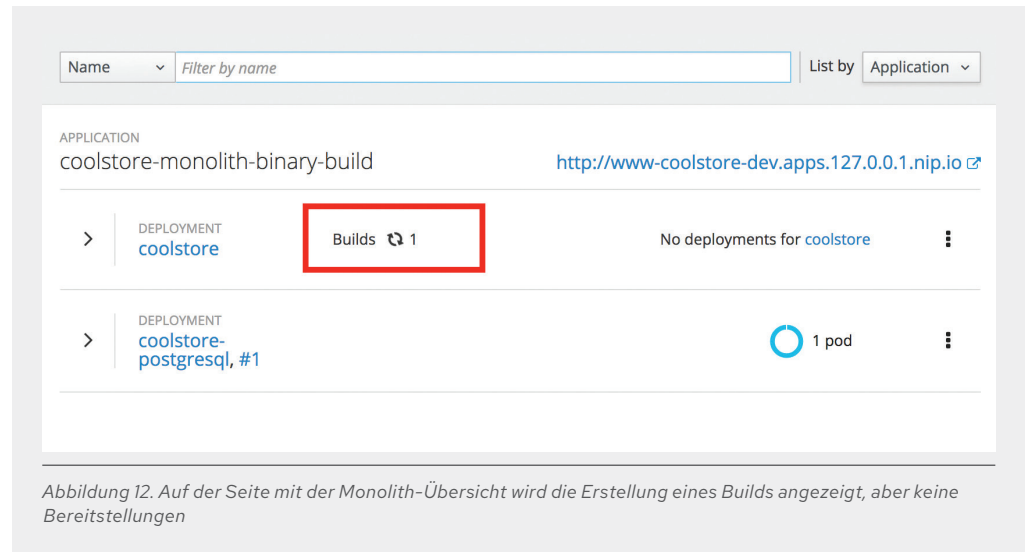
```
$ mvn clean package -Popenshift
```

Warten Sie, bis der Build abgeschlossen ist und die Meldung `BUILD SUCCESS` angezeigt wird.

Starten Sie abschließend den Build-Prozess, der aus der `.war`-Datei und Red Hat JBoss EAP ein Linux® Container-Image erzeugt. Dieses Image wird aufgrund des `DeploymentConfig`-Objekts, das aus der Vorlage erstellt wird, automatisch im Projekt bereitgestellt:

```
$ oc start-build coolstore --from-file=deployments/ROOT.war
```

In der Red Hat OpenShift Webkonsole sollte die erstellte Anwendung angezeigt werden. In diesem Fall wurde bereits eine PostgreSQL-Datenbank bereitgestellt.



Warten Sie, bis der Build und die Bereitstellung abgeschlossen sind:

```
$ oc rollout status -w dc/coolstore
```

Dieser Befehl wird oft verwendet, um auf den Abschluss von Bereitstellungen zu warten. Wenn Sie den Befehl verwenden, stellen Sie sicher, dass eine Erfolgsmeldung zurückgegeben wird. Folgendes sollte angezeigt werden:

```
replication controller "coolstore-1" successfully rolled out.
```

Hinweis: Wenn der oben genannte Befehl einen Fehler vom Server meldet („ServerTimeout“), führen Sie den Befehl einfach erneut aus, bis eine Erfolgsmeldung angezeigt wird.

Wenn der Prozess abgeschlossen ist, wird die erfolgreiche Bereitstellung der Anwendung mit blauen Kreisen für die Datenbank und den Monolith angezeigt:



Testen Sie die Anwendung, indem Sie auf den Routing-Link unter:

`http://www-coolstore-dev.$ROUTE_SUFFIX` klicken. Der Monolith „coolstore“ wird in Ihrem Browser geöffnet und dieses Mal auf Red Hat OpenShift Container Platform ausgeführt.

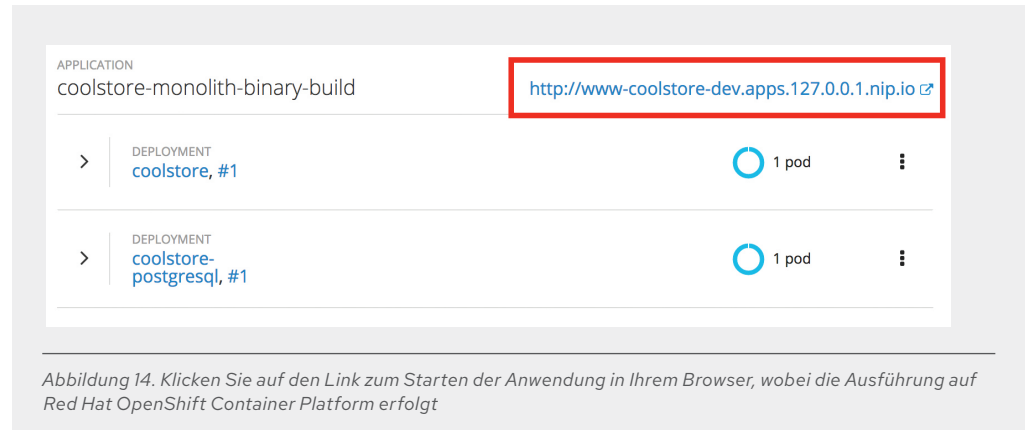
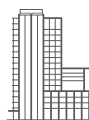


Abbildung 14. Klicken Sie auf den Link zum Starten der Anwendung in Ihrem Browser, wobei die Ausführung auf Red Hat OpenShift Container Platform erfolgt

Zusammenfassung

Dieser Prozess verdeutlicht, wie Sie eine vorhandene Java EE App mit Red Hat JBoss Enterprise Application Platform und Red Hat OpenShift Container Platform in die Cloud migrieren. Jetzt können Sie die Anwendung modernisieren, indem Sie den Monolith schrittweise in kleinere Microservices unterteilen und gleichzeitig moderne Techniken einsetzen, um die ordnungsgemäße Ausführung der Anwendung in einer verteilten und containerisierten Umgebung sicherzustellen.



ÜBER RED HAT

Red Hat, weltweit führender Anbieter von Open Source Software-Lösungen für Unternehmen, folgt einem community-basierten Ansatz, um verlässliche und leistungsstarke Technologien in den Bereichen Linux, Hybrid Cloud, Container und Kubernetes bereitzustellen. Wir unterstützen Kunden bei der Integration neuer und bestehender IT-Anwendungen, der Entwicklung cloudnativer Anwendungen, der Standardisierung auf unserem branchenführenden Betriebssystem sowie der Automatisierung, Sicherung und Verwaltung komplexer Umgebungen. Dank unserer vielfach ausgezeichneten Support-, Training- und Consulting-Services ist Red Hat ein bewährter Partner der Fortune 500 Unternehmen. Als strategischer Partner für Cloud-Anbieter, Systemintegratoren, Anwendungsanbieter, Kunden und Open Source Communities hilft Red Hat Organisationen auf ihrem Weg in die digitale Zukunft.



facebook.com/redhatinc
@RedHatDACH
linkedin.com/company/red-hat

**EUROPA, NAHOST,
UND AFRIKA (EMEA)**
00800 7334 2835
de.redhat.com
europe@redhat.com

TÜRKEI
00800 448820640

ISRAEL
1 809 449548

VAE
8000-4449549