

SAP PRESS

SAP HANA – Die neue Einführung

In-Memory-Technologie, Werkzeuge, Datenbeschaffung und Datenmodellierung

Bearbeitet von
Penny Silvia, Rob Frye, Bjarne Berg

3., aktualisierte und erweiterte Auflage 2017. Buch. 583 S. Hardcover

ISBN 978 3 8362 4507 4

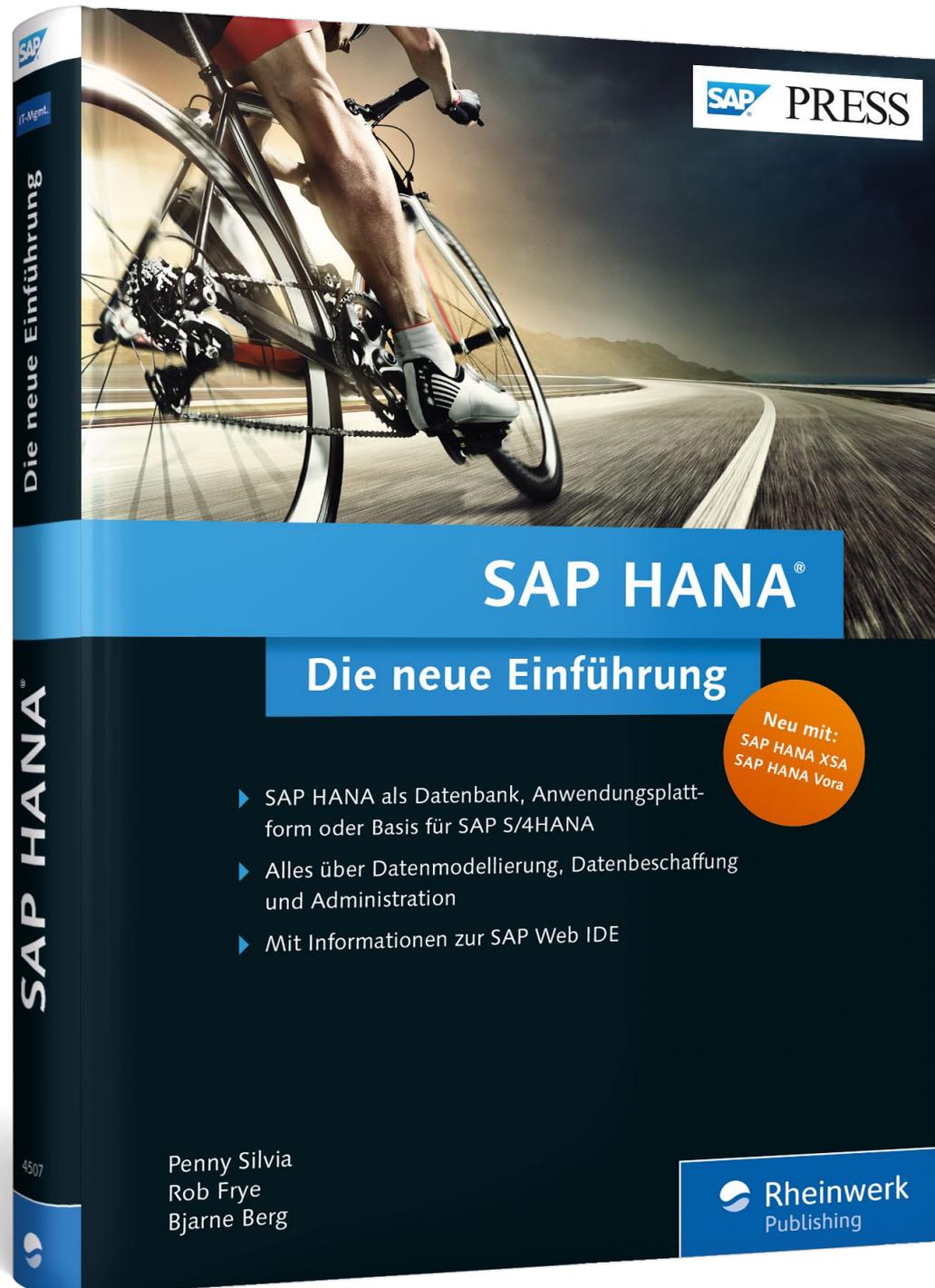
Format (B x L): 16,8 x 24 cm

[Weitere Fachgebiete > EDV, Informatik > Datenbanken, Informationssicherheit,
Geschäftssoftware > SAP](#)

schnell und portofrei erhältlich bei


DIE FACHBUCHHANDLUNG

Die Online-Fachbuchhandlung beck-shop.de ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.



Leseprobe

Dieses Buch ist Ihr Begleiter bei Ihrem Einstieg in die In-Memory-Technologie von SAP. Von den Grundlagen bis zur Administration machen die Autoren Sie fit für die HANA-Welt. In dieser Leseprobe erläutern sie, wie Sie SAP HANA als Anwendungsplattform nutzen können. Außerdem zeigen die Autoren Ihnen die neuen Funktionen von SAP HANA Extended Application Services, Advanced Model (SAP HANA XSA) in SAP HANA SPS 11 und SPS 12.



»SAP HANA als Anwendungsplattform«
»Einführung«



Inhaltsverzeichnis



Index



Die Autoren



Leseprobe weiterempfehlen

Penny Silvia, Rob Frye, Bjarne Berg

SAP HANA – Die neue Einführung

583 Seiten, gebunden, 3. Auflage 2017
69,90 Euro, ISBN 978-3-8362-4507-4



www.sap-press.de/4359

In diesem Kapitel erläutern wir, wie Sie SAP HANA als Anwendungsplattform nutzen können, und beschreiben die innovativen neuen Funktionen von SAP HANA Extended Application Services, Advanced Model (SAP HANA XSA) in SAP HANA SPS 11 und SPS 12.

3 SAP HANA als Anwendungsplattform

Wir reden von SAP HANA als Plattform, aber manchmal vergessen wir, dass diese Plattform weitaus mehr ist als eine schnelle Datenbank für SAP-ERP- oder SAP-BW-Implementierungen (SAP Business Warehouse). Im Grunde genommen ist SAP HANA eine Datenbank, die Unterstützung für alle Arten von Anwendungen bietet und nicht nur für Anwendungen, die SAP bereitstellt. SAP arbeitet auf Hochtouren an der Entwicklung einer flexiblen und mächtigen Anwendungsplattform, um die sich ändernden Anforderungen einer Geschäftslandschaft zu erfüllen, die sich stetig weiterentwickelt.

Mit der Einführung der *SAP HANA Extended Application Services* in SAP HANA SPS 5 hat SAP einen Schritt in die richtige Richtung gemacht, um das volle Potenzial von SAP HANA als Anwendungsplattform auszuschöpfen. Zum Zeitpunkt der Veröffentlichung wurde diese Technologie als *XS Engine* oder *SAP HANA XS* bezeichnet. Wir beziehen uns bei dieser Version der Anwendungsservices aber auf das *SAP HANA Extended Application Services, Classic Model* (SAP HANA XS Classic), da SAP die Anwendungsplattform mit SAP HANA SPS 11 wesentlich verbessert und erweitert hat. Die aktuelle Version der Anwendungsplattform wird *SAP HANA Extended Application Services, Advanced Model* bzw. *SAP HANA XSA* genannt.

Vor der Veröffentlichung von SAP HANA XS Classic brauchten Sie, um eine Webseite oder Anwendung mit den Daten in Ihrer SAP-HANA-Datenbank zu verknüpfen, einen weiteren Anwendungsserver, z. B. einen ABAP- oder SAP-Java-Anwendungs-Stack oder eine ODBC- (Open Database Connectivity) oder JDBC-Verbindung (Java Database Connectivity). Mit SAP HANA XS Classic wurde dieser Prozess vereinfacht, indem im SAP-HANA-System selbst ein leichtgewichtiger Anwendungsserver bereitgestellt wird. Mit SAP HANA XS Classic können Sie Anwendungen entwickeln, die in einem Browser oder auf einem mobilen Gerät angezeigt werden und über die XS Engine in SAP HANA direkt mit der Datenbankschicht verbunden sind.

SAP bietet über SAP HANA XSA Unterstützung für Node.js, Java, HTML und weitere APIs (Application Programming Interfaces) wie C++. Um die Entwicklung zwischen den SAP-HANA-On-Premise- und SAP-HANA-Cloud-Deployments zu vereinfachen und zu vereinheitlichen, basiert SAP HANA XSA auf der *Cloud Foundry*, weshalb nun auf einem einzigen Server mehrere Sprachen und Laufzeitinstanzen unterstützt werden können. Daher verfügt jedes Deployment einer Anwendung oder eines Service über eine Kopie der Java- oder Node.js-Laufzeit und wird in einer separaten Instanz in der SAP-HANA-XSA-Architektur ausgeführt.

In diesem Kapitel erhalten Sie weitere Informationen zu SAP HANA XS Classic und SAP HANA XSA. Läuft Ihre SAP-HANA-Plattform auf SPS 10 oder einer früheren Version, müssen Sie Ihre Anwendungen mit SAP HANA XS Classic erstellen. Läuft Ihre Plattform auf SPS 11 oder höher, können Sie SAP HANA XSA verwenden. Daher werden wir beide Plattform näher betrachten. Sie lernen außerdem, wie Sie Anwendungen mit SAP HANA Studio und der SAP Web IDE entwickeln.

Hinweis

Wenn Sie mit SAP HANA XS Classic beginnen, weil Ihr System auf SPS 10 oder einer früheren Version läuft, und später Ihr System auf SPS 11 oder höher aktualisieren, können Sie weiterhin die Inhalte verwenden, die Sie für SAP HANA XS Classic entwickelt haben.

3.1 SAP HANA XS Classic

Um SAP HANA unabhängiger einsetzen zu können, hat SAP das Werkzeug SAP HANA XS Classic entwickelt, mit dem native SAP-HANA-Anwendungen entwickelt werden können. Läuft Ihr SAP-HANA-System auf SPS 5 bis SPS 10, müssen Sie Ihre nativen SAP-HANA-Anwendungen mit SAP HANA XS Classic erstellen.

Hinweis

SAP HANA XS Classic wurde das erste Mal mit SAP HANA SPS 5 bereitgestellt. Sie benötigen daher mindestens SPS 5, um native SAP-HANA-Anwendungen mit SAP HANA XS Classic zu entwickeln. Wenn Sie bereits mit SPS 11 oder einer höheren Version arbeiten, sollten Sie SAP HANA XSA verwenden. In Abschnitt 3.2 und Abschnitt 3.3 erhalten Sie weitere Informationen zu SAP HANA XSA und zur Entwicklung von SAP-HANA-XSA-Inhalten.

3.1.1 Architektur

Wie bei den meisten Neuheiten im Bereich SAP HANA stellt die Veröffentlichung von SAP HANA XS Classic einen Paradigmenwechsel beim Modell der Anwendungsprogrammierung dar. Vor der Einführung von SAP HANA XS Classic erfolgte die Anwendungsprogrammierung in der Regel mit ABAP im SAP GUI (SAP Graphical User Interface). Die Anzeige, Anwendungslogik und Datenbankinteraktionen erfolgten im ABAP-Stack, und SAP HANA diente nur als Anwendungsserver für den ABAP-Stack und als schnelle Datenbank. Abbildung 3.1 zeigt die Systemlandschaft vor der Einführung von SAP HANA XS Classic. SAP HANA XS war bereits eine erhebliche Verbesserung verglichen mit den Deployment-Szenarien vor SAP HANA. Trotzdem gab es weiterhin Verbesserungspotenzial.

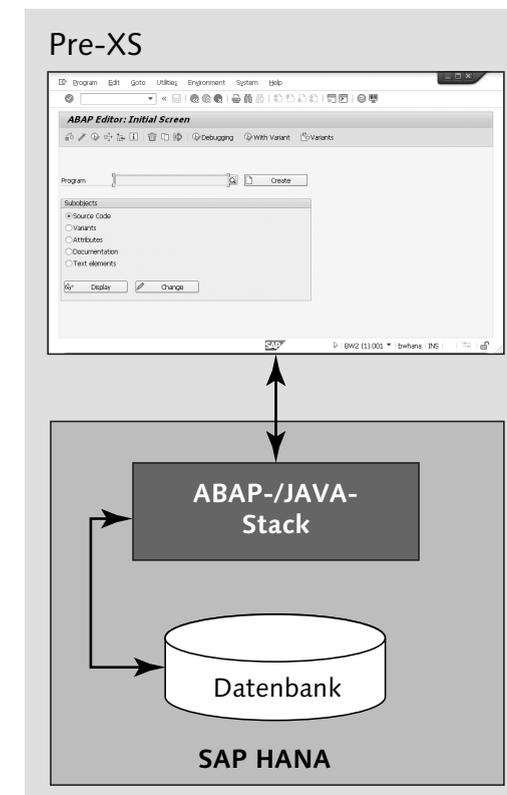


Abbildung 3.1 Systemlandschaft vor SAP HANA XS Classic

Die Veröffentlichung von SAP HANA XS Classic veränderte dann alles. Abbildung 3.2 zeigt die Systemarchitektur von SAP HANA XS Classic. Wie Sie sehen können, verbesserte die SAP-HANA-XS-Classic-Landschaft die Archi-

tektur und die verfügbaren Entwurfsoptionen für die Verknüpfung von Anwendungen mit Daten in SAP HANA. Mit SAP HANA XS Classic ist kein separater Anwendungsserver mehr erforderlich. Stattdessen wird die XS Engine als SAP-HANA-Standardkomponente deployt.

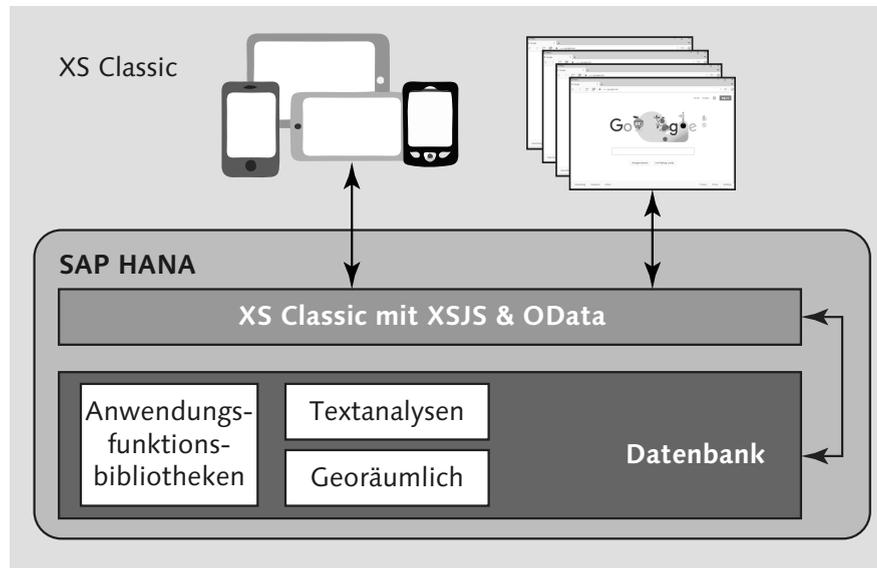


Abbildung 3.2 SAP-HANA-Anwendungslandschaft mit SAP HANA Extended Application Services (SAP HANA XS Classic)

SAP HANA XS Classic ermöglicht es Entwicklern, Anwendungen zu erstellen, deren Deployment einfach über einen Webbrowser oder ein mobiles Gerät erfolgen kann. Indem Sie HTML5 mit XSJS (XS JavaScript) kombinieren, können Sie Anwendungen entwickeln, die vollständig in SAP HANA liegen. Die Kosten und der Overhead für die Entwicklung separater Anwendungsserver können reduziert werden, und die Anwendungslogik kann in vielen Fällen in die Datenbankschicht für eine Verarbeitung in der In-Memory Computing Engine (IMCE) von SAP HANA verschoben werden. Des Weiteren ermöglicht SAP HANA XS Classic eine Authentifizierung und Sicherheitsmaßnahmen durch SAP HANA, sodass Sie keine komplizierten Authentifizierungsprotokolle zwischen dem Anwendungsserver und der Datenbank konfigurieren müssen.

3.1.2 SAP-HANA-XS-Classic-Anwendungen entwickeln

Mit der Veröffentlichung von SAP HANA XS Classic wurde auch eine neue Perspektive in SAP HANA Studio eingeführt: die Perspektive SAP HANA

DEVELOPMENT. Sie war zu dem Zeitpunkt eine der bevorzugten Lösungen für die Entwicklung von Anwendungen und Inhalten für Browser und mobile Geräte. (Bei der anderen Lösung handelt es sich um die SAP HANA Web-Based Development Workbench, die wir in Kapitel 8, »Datenmodellierung mit SAP HANA«, näher beschreiben werden.) Die Perspektive SAP HANA DEVELOPMENT ist für das serverseitige Scripting und die serverseitige Objektentwicklung konzipiert. (Damit stellt es im Gegensatz zu SAPUI5, das zur Laufzeit weitgehend auf Clientseite ausgeführt wird. Sie sollten jedoch beachten, dass SAPUI5 eine Komponente von SAP HANA XS ist.)

Mit SAP HANA XS Classic erhalten Sie Zugang zur serverseitigen Programmierung. Dies umfasst XMLA, OData und das komplexere JavaScript mit XSJS. Außerdem können Sie direkt auf die Funktionen des SAP-HANA-Webservers zugreifen. SAP verfügt mit den SAP-HANA-XS-Classic-Funktionen über eine offene Plattform für Unternehmen, die SAP HANA als Entwicklungsplattform für solche Zielanwendungen einsetzen möchten, die über herkömmliche SAP-ERP- und Data-Warehousing-Anwendungsbereiche hinausgehen.

Einer der wichtigsten Vorteile von SAP HANA XS Classic ist, dass kein weiterer Anwendungsserver erforderlich ist. Die Anwendung wird schlichtweg als native SAP-HANA-Komponente bereitgestellt, sodass keine zusätzliche Hardware oder Anwendungsserversoftware benötigt wird. Ihre Anwendung wird also in der SAP-HANA-Plattform aufgebaut und ausgeführt. Die Datenbank ist die SAP-HANA-Datenbank mit spalten- und zeilenbasierten Speichern. Zur Anwendungslaufzeit wird die Anwendung auf SAP HANA XS Classic ausgeführt. Bei dieser Art des Deployments fungiert SAP HANA XS Classic als interner SAP-HANA-Anwendungsserver mit geringem Speicherbedarf, der einen Webserver und eine zentrale Anwendungsentwicklungsplattform in SAP HANA bietet. Um Ihre Anwendung zu gestalten, können Sie die SAP HANA Web-Based Development Workbench (siehe Abbildung 3.3) oder die SAP Web IDE nutzen (siehe Abschnitt 3.3).

Im Gegensatz zur Entwicklung im Eclipse-basierten SAPUI5 und SAP HANA Studio können Sie entweder mit der SAP HANA Web-Based Development Workbench oder der SAP Web IDE eine komplette Anwendung erstellen, ohne etwas auf Ihrem Desktop installieren zu müssen. Die gesamte Entwicklungsarbeit für Ihre SAP-HANA-XS-Classic-Anwendung erfolgt direkt im Webbrowser.

Bei der Entwicklung von Anwendungen können Sie auch das Backend in SAP HANA Studio entwickeln. Anders als bei der Web-Interface-Option muss SAP HANA Studio auf Ihrem lokalen PC installiert sein, und Sie benötigen

eine Verbindung, einen Benutzernamen und ein Kennwort für das SAP-HANA-Datenbanksystem. Außerdem muss Ihnen vom Systemadministrator die Rolle des SAP-HANA-Entwicklers zugewiesen werden, damit Sie diese Art der nativen SAP-HANA-XS-Classic-Entwicklung durchführen dürfen.

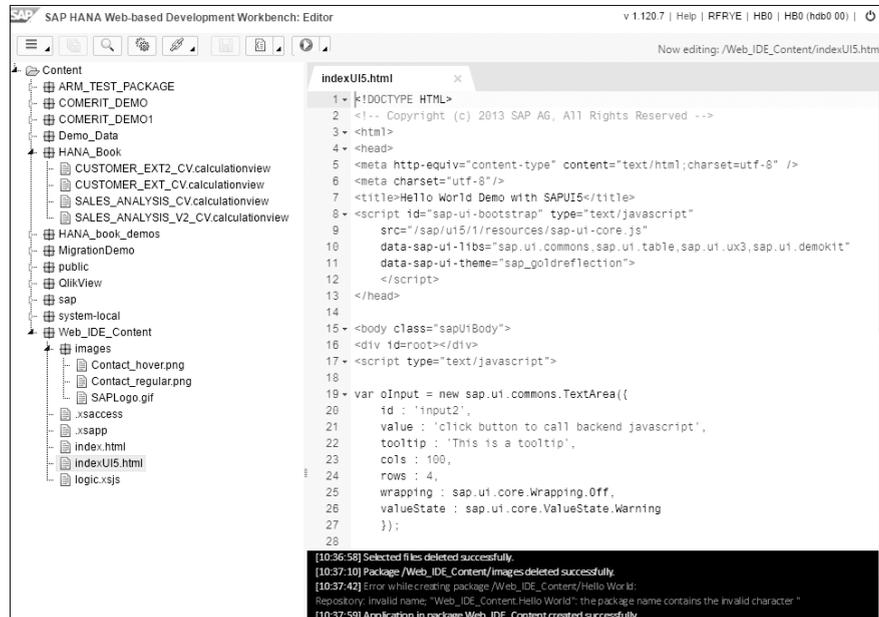


Abbildung 3.3 Der »SAP HANA Web-Based Development Workbench Editor« mit SAP HANA XS Classic

Im Allgemeinen können SAP-HANA-XS-Classic-Mini-Anwendungsserver in SAP HANA die in SAPUI5 und SAP HANA XS Classic entwickelten Frontend-Anwendungen unterstützen, indem die Daten aus vielen Tabellen in einfachen Access Views zusammengeführt werden.

Sie können auch die Anwendungslogik (z. B. Berechnungen) in die Analytical und Calculation Views in SAP HANA verschieben. Diese Funktion vereinfacht die Ausführung des Codes und reduziert die Rechenleistungsanforderungen auf kleineren mobilen Geräten, da ein Großteil der Logik im Speicher auf Datenbankebene ausgeführt wird und nicht auf den kleineren Geräten oder den herkömmlichen Anwendungsservern – die im Vergleich zu SAP HANA immer noch extrem langsam sind. Sie können einen Großteil der routinemäßigen Anwendungslogik auf die Ebene des SAP-HANA-Datenbankservers verschieben, anstatt dedizierte externe Anwendungsserver zu installieren. Somit bleiben die SAP-HANA-Umgebungen einfach und übersichtlich, was Ihr Basis-Supportteam freuen wird.

Öffnen Sie zunächst SAP HANA Studio, und klicken Sie auf die Menüoption WINDOW. Klicken Sie anschließend auf PERSPECTIVE • OPEN PERSPECTIVE • OTHER, und wählen Sie SAP HANA DEVELOPMENT aus (siehe Abbildung 3.4).

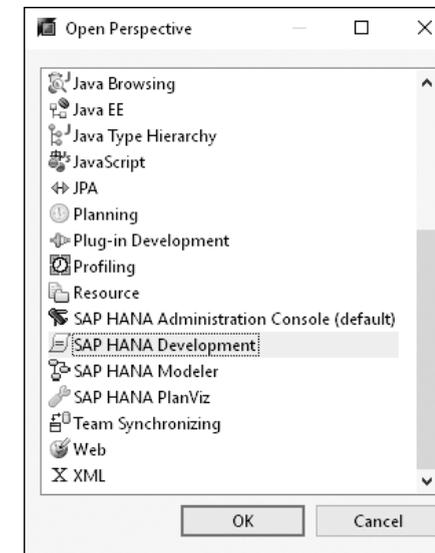


Abbildung 3.4 Erste Schritte in der SAP-HANA-Entwicklung in SAP HANA Studio

In der Perspektive SAP HANA DEVELOPMENT können Sie ein Projekt für Ihre Entwicklung erstellen. Wählen Sie im Menü FILE die Option NEW • XS PROJECT (siehe Abbildung 3.5) aus, um ein neues SAP-HANA-XS-Classic-Entwicklungsprojekt anzulegen. Durch das Erstellen eines neuen Projekts können Sie unabhängig von anderen Entwicklern arbeiten, die in anderen Bereichen des SAP-HANA-Systems tätig sind.

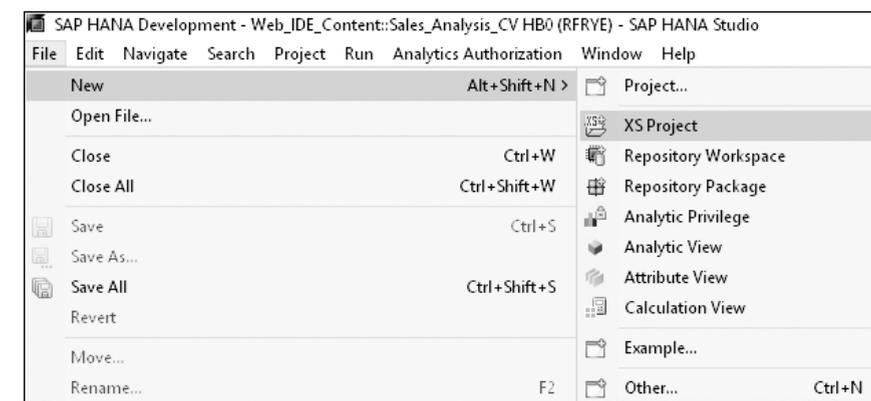


Abbildung 3.5 Ein neues Projekt für die Entwicklung in SAP HANA Studio erstellen

Hinweis

Wenn Sie mit mehreren Entwicklern in einer Arbeitsumgebung arbeiten, müssen Sie vor Beginn eines Projekts die Entwicklungsstandards und Namenskonventionen festlegen, um Verwirrung und unbeabsichtigte Konflikte mit der Entwicklungsarbeit anderer zu vermeiden.

Der Assistent NEW XS PROJECT wird geöffnet, und Sie werden aufgefordert, bei PROJECT NAME einen Namen für das Projekt einzugeben. Sie können das Kontrollkästchen SHARE PROJECT IN SAP REPOSITORY aktivieren, damit andere Entwickler mit Ihnen gemeinsam an diesem Projekt können. Wählen Sie anschließend die Option USE DEFAULT LOCATION aus, um die Dateien in Ihr Arbeitsverzeichnis für Ihre SAP-HANA-Studio-Installation zu speichern. Sie können auch die Option BROWSE TO THE LOCATION auswählen, um die Projektdateien in einem anderen Verzeichnis zu speichern.

Wenn Sie den Speicherort selbst festlegen möchten, müssen Sie auch das Dateisystem für die Dateien auswählen. Folgende Optionen stehen hier zur Verfügung:

- ▶ **DEFAULT**
Hierbei handelt es sich um das Standarddateisystem auf Ihrer Entwicklungsmaschine.
- ▶ **SEMANTIC FILE SYSTEM**
Hierüber können Sie einen Speicherort in der Landschaft PROJECT EXPLORER festlegen.

Wenn Sie das Projekt lokal speichern, klicken Sie auf FINISH, um das Projekt im PROJECT EXPLORER anzulegen.

Wenn Sie das Projekt im Repository freigeben möchten, müssen Sie auf NEXT klicken, um eine Option für REPOSITORY WORKSPACE für das Projekt auszuwählen (siehe Abbildung 3.6). Sie können entweder einen vorhandenen Workspace auswählen oder auf ADD WORKSPACE klicken, um einen neuen Workspace im Repository anzulegen.

Klicken Sie auf NEXT, um im nächsten Fenster mit dem SAP-HANA-XS-Projektassistenten (siehe Abbildung 3.7) fortzufahren. Hier konfigurieren Sie die allgemeinen Objektnamen für das Projekt-Repository. Sie müssen einen Namen für das SCHEMA des neuen Projekts, einen Namen für DDL und einen Namen für XS JAVASCRIPT für die Objekte eingeben, die Sie zum Repository hinzufügen möchten. Klicken Sie auf FINISH, um ein neues SAP-HANA-XS-Classic-Projekt anzulegen.

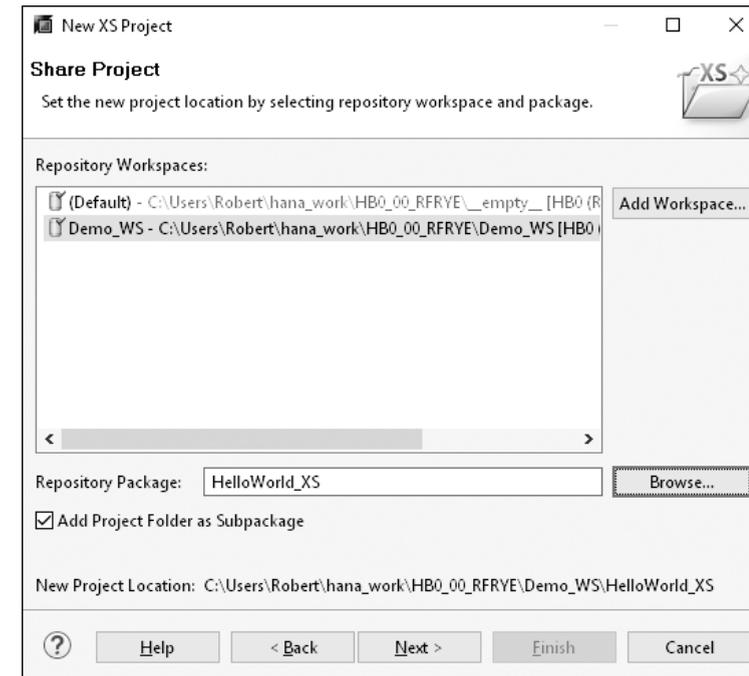


Abbildung 3.6 Einen Workspace für das neue SAP-HANA-XS-Classic-Projekt auswählen

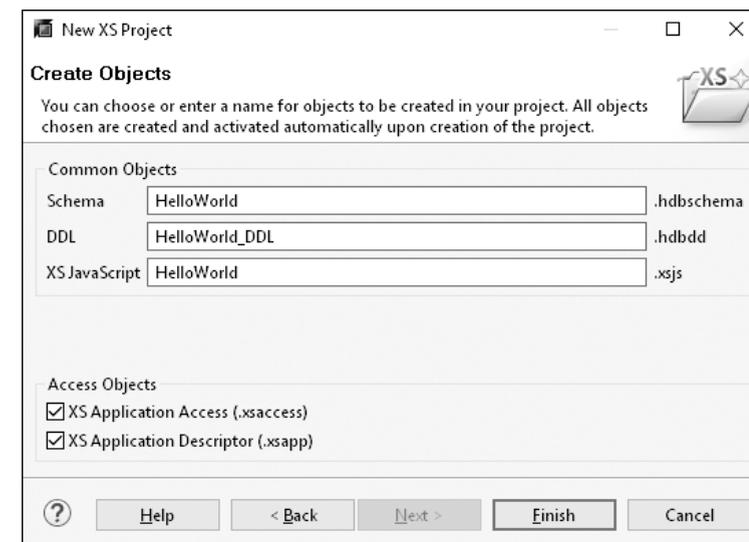


Abbildung 3.7 Ein Repository-Objekt für das SAP-HANA-XS-Classic-Projekt anlegen

Sobald das Projekt in SAP HANA Studio erstellt wurde, können Sie mit dem Aufbau der Anwendung beginnen. Beachten Sie, dass die Entwicklung von Anwendungen in SAP HANA XS Classic und SAP HANA XSA grundlegende Kenntnisse zum Thema Codierung erfordert. Sie sollten daher die erforderlichen Kompetenzen in diesem Bereich aufbauen oder erweitern, da es hier um das Erstellen von Anwendungen auf SAP HANA geht. Das Coding für SAP HANA XS Classic ist JavaScript sehr ähnlich, und SAP HANA Studio bietet verschiedene Entwicklungs-Wizards für den Einstieg in typische Aufgaben. Mit den Wizards können Sie zudem Ihre eigenen JavaScript-Bibliotheksdateien erstellen, in denen Sie die Steuerelemente, die Sie am häufigsten nutzen, oder benutzerspezifische Codeerweiterungen ablegen können. Die Wizards erleichtern Ihnen die Arbeit wesentlich.

Wenn Ihr Projekt angelegt wurde, werden Ihnen im PROJECT EXPLORER die Projektinhalte im Navigationsbereich angezeigt und Sie können den Code für Ihre Anwendungen im Editor eingeben (siehe unser einfaches HELLO_WORLD_XS-Beispiel in Abbildung 3.8).

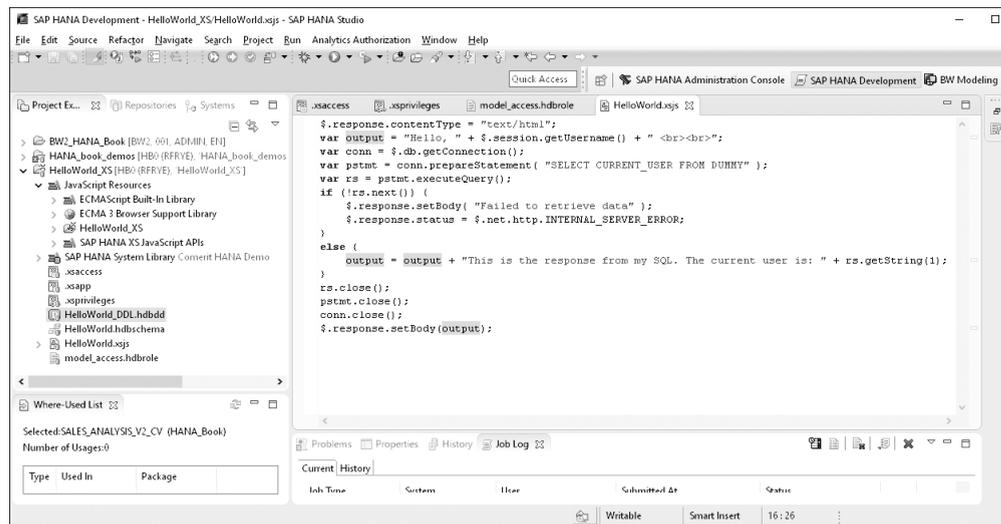


Abbildung 3.8 HelloWorld in SAP HANA XS Classic mit SAP HANA Studio

Hinweis

In den unten genannten Schritten gehen wir davon aus, dass Sie die Paket- und Objektnamen aus den oben beschriebenen Beispielen verwendet haben. Wenn Sie andere Namen angegeben haben, müssen Sie die Objektnamen in den nachstehenden Code-Listings anpassen.

Um die SAP-HANA-XS-Classic-Version des HelloWorld-Programms zu duplizieren, müssen Sie folgende Schritte ausführen:

1. Legen Sie das SAP-HANA-XS-Classic-Projekt wie zuvor beschrieben an.
2. Öffnen Sie die Datei `.xsaccess`, und fügen Sie folgenden Code zur Datei hinzu:
3. Speichern Sie die Datei `.xsaccess`.
4. Fügen Sie zum Projekt eine Datei mit dem Namen `.xsprivileges` hinzu.
5. Fügen Sie folgenden Code hinzu:

```
"default_file": "HelloWorld.xsjs",

{"privileges" :
 [
  { "name" : "Basic", "description" : "Basic usage privilege" } ]
}
```

Hinweis

Wenn Sie Dateien hinzufügen, können Sie möglicherweise den XS-Dateityp aus dem Kontextmenü auswählen, der zu dem Objekt passt, das Sie hinzufügen, oder indem Sie zu FILE • OTHER navigieren.

Wenn der gesuchte Dateityp nicht vorhanden ist, klicken Sie mit der rechten Maustaste auf PROJECT EXPLORER, um das Kontextmenü zu öffnen, und wählen NEW FILE aus.

6. Speichern Sie die Datei `.xsprivileges`.
7. Fügen Sie eine Datei mit dem Namen `model_access.hdbrole` zu Ihrem Projekt hinzu, und ergänzen Sie sie mit folgendem Code:
8. Öffnen Sie die Datei `HelloWorld.xsjs`, und fügen Sie den Code aus Listing 3.1 zur Datei hinzu:

```
role HelloWorld_XS::model_access {
  application privilege: HelloWorld_XS::Basic;}

```

```
$.response.contentType = "text/html";
var output = "Hello, " + $.session.getUsername() + " <br><br>";
var conn = $.db.getConnection();
var pstmt = conn.prepareStatement( "SELECT CURRENT_
  USER FROM DUMMY" );
var rs = pstmt.executeQuery();
if (!rs.next()) {
  $.response.setBody( "Failed to retrieve data" );
  $.response.status = $.net.http.INTERNAL_SERVER_ERROR;
}
```

```

else {
    output = output + "This is the response from my SQL.
    The current user is: " + rs.getString(1);
}
rs.close();
pstmt.close();
conn.close();
$.response.setBody(output);

```

Listing 3.1 Code für das SAP-HANA-XS-Classic-HelloWorld-Programm

9. Speichern Sie die Datei *HelloWorld.xsjs*.
10. Aktivieren Sie alle Objekte aus Ihrem Projekt über die Schaltfläche ACTIVATE ALL SAP HANA DEVELOPMENT OBJECTS (siehe Abbildung 3.9) oder mit der Tastenkombination `Strg`+`⇧`+`F3`.



Abbildung 3.9 So aktivieren Sie alle inaktiven Objekte in einem SAP-HANA-XS-Classic-Projekt.

11. Weisen Sie Ihrem Benutzer die Rolle `HelloWorld_XS::model_access` zu.

Wenn Sie alle Objekte im Projekt aktiviert haben, sollte Ihnen Ihre einfache HelloWorld-Begrüßung angezeigt werden, sobald Sie zu `http://<hana_server>:80<Instanz>/helloworld_xs/` navigieren. Melden Sie sich mit Ihrem Benutzer mit der Rolle `HelloWorld_XS::model_access` am Server an. Sie sollten dann mit Ihrem Benutzernamen begrüßt werden (siehe Abbildung 3.10).

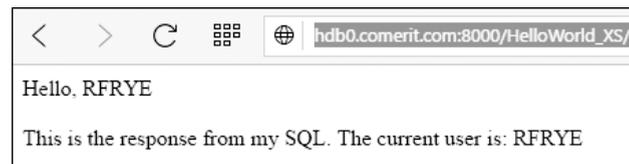


Abbildung 3.10 In SAP HANA Studio mit SAP HANA XS Classic entwickeltes HelloWorld-Programm

Anhand dieses Beispiels zum Erstellen einer Anwendung in SAP HANA XS Classic sollten Sie ausreichend Hintergrundwissen für den Einstieg gesammelt haben. Weitere Informationen zum Coding von Anwendungen in SAP

HANA XS Classic und SAP HANA XSA erhalten Sie in folgenden Referenzanleitungen von SAP, die zum Download zur Verfügung stehen:

- ▶ *SAP HANA XS JavaScript Reference*
- ▶ *SAP HANA JavaScript API Reference*
- ▶ *SAP HANA XSUnit JavaScript API Reference*
- ▶ *SAP HANA XS DB Utilities JavaScript API Reference*
- ▶ *SAP HANA REST API Reference*

Diese Anleitungen sowie weitere Informationen können Sie unter http://help.sap.com/hana_plattform herunterladen.

SAP HANA XS und SAPUI5

Obwohl SAP HANA XS und SAPUI5 separate Komponenten sind, ist es wichtig, ihr Zusammenspiel zu verstehen. Sie könnten eine SAPUI5-Anwendung ohne direkte Verwendung von SAP HANA XS erstellen. Wenn Sie aber eine SAP-HANA-XS-basierte Anwendung aufbauen, werden Sie SAPUI5 höchstwahrscheinlich als Ihr speziell entwickeltes Frontend-Werkzeug einsetzen. Alternativ können Sie auch die SAP Web IDE oder die SAP HANA Web-Based Development Workbench verwenden.

Diese Beziehung können Sie leicht erkennen, wenn Sie den Inhaltsordner Ihres SAP-HANA-XS-Projekts sorgfältig prüfen. Wie in Abbildung 3.11 dargestellt, entsprechen die Unterabschnitte den verschiedenen Komponenten.

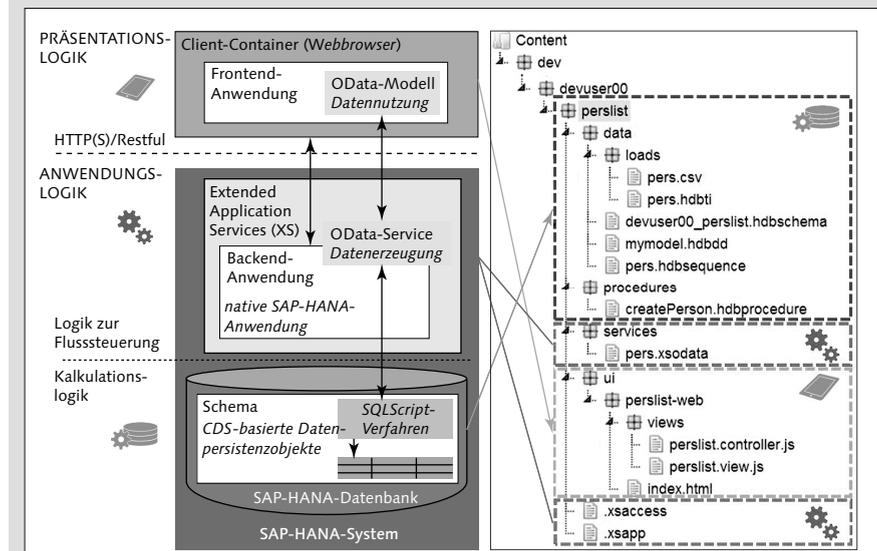


Abbildung 3.11 Entwicklungsarchitektur für SAP HANA XS und SAPUI5 auf SAP HANA

Bitte beachten Sie, dass diese Art von Entwicklung nicht für Benutzer mit begrenzten Programmierkenntnissen geeignet ist: Grundlegende Kenntnisse in den Programmiersprachen C++, Java, JavaScript und VBScript sind für die korrekte Nutzung unentbehrlich.

3.2 SAP HANA XSA

Mit SAP HANA SPS 11 hat SAP die SAP HANA Extended Application Services, Advanced Model (SAP HANA XSA) eingeführt. SAP HANA XSA basiert auf SAP HANA XS Classic und bietet Unterstützung für JavaScript, Node.js und Java sowie für GitHub und Maven. So können Sie auf dem gleichen SAP-HANA-Server Anwendungen mit verschiedenen Komponenten und in unterschiedlichen Programmiersprachen erstellen.

In SAP HANA XSA wird der Microservice-Ansatz angewandt. Bei den Anwendungen und ihren entsprechenden programmiersprachenspezifischen Laufzeitumgebungen handelt es sich um modulare, entkoppelte Services, die Informationen über RESTful APIs (Representational State Transfer für Application Programming Interfaces) austauschen. Dank dieser Microservice-Architektur können Sie einen Service pflegen, ohne dass dies Auswirkungen auf die anderen Services hat. Sie können zum Beispiel eine Komponente der Anwendung patchen, ohne dass die anderen Komponenten der Anwendung abstürzen. So können Sie Ausfallzeiten vermeiden, da neue Versionen eines vorhandenen Microservice auf der gleichen SAP-HANA-Instanz deploy und getestet werden können, ohne dass das ganze System ausfällt.

Hinweis

Bei REST handelt es sich um eine Architekturform, die für den Entwurf von vernetzten Anwendungen verwendet wird. REST basiert auf cachebaren, zustandslosen Client-Server-Kommunikationslogs, in den meisten Fällen auf HTTP. Bei Anwendungen mit einer REST-Architektur können Benutzer normalerweise durch den Programmablauf navigieren, indem sie Links auswählen, über die die nächste Seite der Anwendung geladen und angezeigt wird.

Mit SAP HANA XSA wurde zusätzlich ein neuer Laufzeit-Controller auf der Basis von Cloud Foundry eingeführt, der die Einführung von Build-Paketen in die SAP-HANA-Entwicklung ermöglicht. Build-Pakete umfassen eine Reihe an Umwandlungs- und Hilfsprogrammen für das Anwendungsdesign, und die neue SAP-HANA-XSA-Laufzeit bietet Unterstützung für Build-Pakete mit folgenden Sprachen:

- ▶ XSJS (XS JavaScript)
- ▶ Node.js
- ▶ Java on Apache TomEE 1.7.3
- ▶ Java on Apache Tomcat 8.0.32

Die Laufzeiten in SAP HANA XSA sind von der Datenbank in SAP HANA entkoppelt, wodurch eine flexible Skalierung während Lastspitzen möglich ist. Sie können neue Server oder Knoten nach Bedarf dynamisch hinzufügen oder entfernen.

Hinweis

Ihnen ist vielleicht aufgefallen, dass wir in unserem Landschaftsdiagramm aus Abbildung 3.12 C++ als Sprachlaufzeitumgebung aufgeführt haben, die mit SAP HANA XSA verfügbar ist. Auch wenn C++- und FastCGI-Laufzeiten intern zur Verfügung stehen, wird im allgemeinen Release von SAP HANA XSA für SAP HANA SPS 12 keine Unterstützung für C++ und FastCGI angeboten. Da die SAP-HANA-XSA-Architektur auf Cloud Foundry basiert, können Sie wahrscheinlich in Zukunft andere Cloud-Foundry-Build-Pakete wie Go, Ruby oder PHP als Sprache oder Laufzeit für SAP HANA XSA verwenden.

In Kürze wird mit der Veröffentlichung von SAP HANA XSA die Transformation von SAP HANA von einem spezialisierten Anwendungsserver und einer schnellen Datenbank-Engine in eine Plattform der nächsten Generation für Ihre spezifischen Geschäftsanforderungen abgeschlossen sein. Beachten Sie auch, dass SAP ab SPS 11 empfiehlt, SAP HANA XSA zur Entwicklung neuer Anwendungen für die SAP-HANA-Plattform zu verwenden. Im Folgenden erläutern wir die Rolle von SAP HANA XSA in der SAP-HANA-Architektur, und Sie erhalten Schritt-für-Schritt-Anleitungen zur Installation von SAP HANA XSA und der SAP Web IDE, damit Sie mit der Entwicklung Ihrer eigenen Anwendungen beginnen können.

3.2.1 Die SAP-HANA-XSA-Architektur

In diesem Abschnitt befassen wir uns mit der neuen SAP-HANA-XSA-Landschaft (siehe Abbildung 3.12) und vergleichen sie mit der Landschaft von SAP HANA XS Classic (siehe Abbildung 3.2). Die neue Landschaft bietet Unterstützung für alle Ihre SAP-HANA-XS-Classic-Anwendungen, damit ein Upgrade Ihrer SAP-HANA-Installation auf SPS 11 oder höher Ihre SAP-HANA-XS-Classic-Anwendungen nicht zum Absturz bringt. Allerdings können nur neue Anwendungen das volle Potenzial der Verbesserungen in SAP HANA XSA ausschöpfen.

Der Anwendungs-Router nimmt Serviceanfragen von browserbasierten Anwendungen und mobilen Anwendungen entgegen und leitet diese Anfragen an den entsprechenden Microservice weiter. Für jede Anwendungskomponente wird ein separater Microservice-Container erstellt, und die Komponenten tauschen die Daten über RESTful APIs aus. Sie können mehrere Container für eine Laufzeit erstellen, sodass Sie mehrere Anwendungskomponenten mit verschiedenen Sprachen entwickeln können.

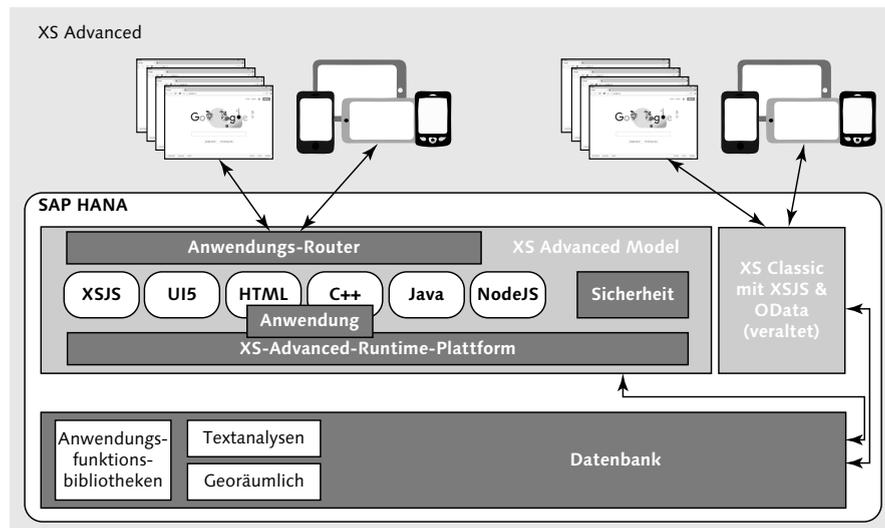


Abbildung 3.12 SAP-HANA-Anwendungslandschaft mit SAP HANA XSA

Jeder Microservice-Container verfügt über eine vollständige Kopie der gesamten Node.js-, Java- oder XSJS-Laufzeit, damit für jede Komponente die Version verwendet wird, mit der sie deployt wurde. Somit bleibt der Upgradepfad für zukünftige Erweiterungen gleich, da eine deployte Komponente weiterhin die ursprüngliche Laufzeitbibliothek verwendet, bis sie selbst mit einer aktualisierten Version in einer neuen Erweiterung deployt wird.

Der Microservice-Ansatz wird auch auf Betriebssystemebene angewandt, und es gibt für jede deployte Instanz einer Laufzeit einen eigenen dedizierten Betriebssystemprozess. Dank dieser entkoppelten Laufzeitarchitektur führt der Ausfall eines Service nicht zu weiteren Ausfällen von Anwendungskomponenten oder Services.

In den folgenden Abschnitten befassen wir uns mit einigen der wichtigsten Komponenten der SAP-HANA-XSA-Architektur, einschließlich der Rolle von Multi-Target-Anwendungen und wie sie in der Laufzeitplattform verarbeitet

werden. Des Weiteren erklären wir, wie Benutzerkonten und die Authentifizierung in der Landschaft verwaltet werden, und erläutern die Rolle von OData-Services. Abschließend erhalten Sie Informationen zu den Administrationswerkzeugen, die Ihnen in SAP HANA XSA zur Verfügung stehen.

Multi-Target-Anwendungen

Seit SAP HANA SPS 12 werden Multi-Target-Anwendungen (MTAs) unterstützt. Die Verwendung von MTAs wird auch *Blue-Green Deployment* genannt. Dieses Konzept basiert auf zwei identischen Zielumgebungen, die als Blue- und Green-Umgebung bezeichnet werden. In diesem Deployment-Szenario ist nur eine der beiden Umgebungen zu einem Zeitpunkt aktiv. Die Entwicklung und das Testen können daher in der anderen Umgebung stattfinden. Wenn Sie zwischen den Umgebungen wechseln möchten, müssen Sie einfach nur das Routing in Ihrem Netzwerk aktualisieren, um die Entwicklungsumgebung als Produktionsumgebung festzulegen.

Da Sie so einfach zwischen den Umgebungen wechseln können, können Sie schnell und problemlos zur vorherigen Umgebung umschalten, wenn ein unerwarteter Fehler in der neuen Version auftritt. Sobald die neue Umgebung durch die Risikoverfahrensverfahren in Ihrem Unternehmen als stabil eingestuft wurde, können Sie die alte Umgebung in einem Archiv speichern und die neue Umgebung kopieren, damit sie auch als Entwicklungsumgebung genutzt werden kann.

Eine MTA besteht aus einem oder mehreren Anwendungsmodulen, von denen jedes als Microanwendung betrachtet wird. Diese Microanwendungen können an die entsprechenden Services gebunden sein und auf eine Deployment-Plattform verschoben werden. In jedem Modul kann es Attribute geben, die in anderen Modulen verwendet werden, und jedes Modul kann von anderen Modulen abhängen.

In der MTA-Deployment-Deskriptordatei sind die Module und Beziehungen für eine Anwendung festgelegt, einschließlich der technischen Typen, der Abhängigkeiten und aller anderen Parameter, die für die Module erforderlich sind. Anhand der Deskriptordatei wird verifiziert, dass die erforderlichen Abhängigkeiten bestehen und dass die Module in der richtigen Reihenfolge deployt sind. Außerdem wird sie zur Einrichtung der entsprechenden Verbindungen zwischen den Modulen verwendet. Alle Anwendungsdateien für die MTA, einschließlich Manifesten, Bezeichnungen und Servicekonfigurationsinformationen, werden in einem Archivpaket gesammelt. Mit diesem

Paket kann dann die MTA deployt werden. Im MTA-Archiv gibt es in der Regel einen Ordner für jedes Modul der Anwendung. Die genaue Ordnerstruktur (wie im MTA-Manifest angegeben) hängt von der Struktur Ihrer Anwendung ab. Sie kann folgende Ordner enthalten:

- ▶ **java/**
Java-Anwendungsdateien
- ▶ **web/**
Zustands-Web-Content und Anwendungs-Routing-Konfiguration
- ▶ **db/**
Für SAP-HANA-Datenbanken übliche Views, Tabellen, Verfahren, Calculation Views etc.
- ▶ **js/**
Node.js-Anwendung und XSJS-Dateien

Wenn Sie Ihre Anwendungen über die Microservice-Architektur und mit MTAs erstellen, sollten Sie folgende Regeln beachten:

- ▶ **Isoliert und unabhängig**
Die Services müssen isoliert voneinander sein und dürfen keine Bibliotheken und Komponenten gemeinsam nutzen. Die allgemeinen Bibliotheken und Frameworks sind in jeden Service-Container integriert, wodurch eine Isolation der Services voneinander vereinfacht wird. Es bestehen Redundanzen, und die unterschiedlichen Services können mit verschiedenen Versionen der gleichen Bibliothek ausgeführt werden.
- ▶ **Sprachen und Laufzeiten**
Die Services können in allen Sprachen implementiert und in allen verfügbaren Laufzeitumgebungen ausgeführt werden, unabhängig von den in den anderen Services implementierten Sprachen und Laufzeitumgebungen.
- ▶ **Lebenszyklus**
Der Lebenszyklus der einzelnen Services ist unabhängig von den anderen Services, damit eine stabile Integration mit den anderen Services stattfinden kann und die Services problemlos aktualisiert oder ersetzt werden können.
- ▶ **Erweiterbarkeit**
Die Services können vorhandene Services einschließen, ersetzen oder erweitern und können mit neuen und bestehenden Services zusammengeführt werden, um einen Mashup-Service zu erhalten.

Weitere Informationen zur Entwicklung mit MTAs erhalten Sie im Bereich SAP HANA XS ADVANCED MULTI-TARGET APPLICATIONS auf der Seite INTRODUCTION TO APPLICATION DEVELOPMENT AND DEPLOYMENT (XS ADVANCED MODEL) unter <http://help.sap.com>.

Laufzeitplattform

Die Laufzeitplattform ist das Framework, auf dem die Microservice-Architektur von SAP HANA XSA basiert. Alle Microservice-Container für Ihre Multi-Target-Anwendungen nutzen die Laufzeitplattform, um unzählige Funktionen in der Landschaft zu verwalten – zum Beispiel:

- ▶ Routing
- ▶ Lastausgleich
- ▶ Identitäts- und Zugriffsverwaltung
- ▶ Lebenszyklusmanagement
- ▶ flexible Skalierung nach Bedarf
- ▶ API-Management für alle Container

Über die XS-Plattform werden die Microservice-Container nach Bedarf gestartet und angehalten und die einzelnen Containerinstanzen und Anwendungen sowie CPU, Arbeitsspeicher, Netzwerk und Dateisystemressourcen von jedem Container überwacht.

Ohne die von der SAP-HANA-XSA-Laufzeitplattform ausgeführte Koordinations- und Verwaltungsarbeit würden für die gesamte Landschaft mehrere Anwendungsserver (einer für jede Anwendung) benötigt werden, und die Vorteile des Microservice-Ansatzes würden fehlen, insbesondere im Bereich des Lebenszyklusmanagements. Ohne die Laufzeitplattform müsste bei der Aktualisierung einer Anwendungskomponente die gesamte Anwendung erneut deployt werden. Dank SAP HANA XSA können Sie von den entsprechenden Containern isolierte Änderungen vornehmen und deployen und benötigen keine separaten Anwendungsserver für jede Anwendung.

User-Account-and-Authentication-Service

Der UAA-Service (User Account and Authentication) kümmert sich um alle nicht authentifizierten Anforderungen an SAP-HANA-XSA-Anwendungen oder den XS Application Router. Der UAA-Service arbeitet mit externen Benutzerablagen wie SAP Cloud Identity, Lightweight Directory Access Protocol (LDAP) oder der SAP-HANA-Datenbank und kann unterschiedliche Authenti-

fizierungsmethoden wie SAML-Assertions (Security Assertion Markup Language), X.509-Zertifikate und Benutzername und Passwort verarbeiten.

Wenn über den UAA eine Anmeldung erfolgreich verarbeitet wurde, wird ein OAuth-Token übergeben. Dieser Token kann für alle weiteren Aufrufe von Anwendungsservices für den Benutzer verwendet und an weitere Services für benutzerbasierte Berechtigungen weitergeleitet werden. Der UAA-Service könnte z. B. ein OAuth-Token an die SAP-HANA-Datenbank übergeben, die Berechtigungen auf der Grundlage von SAP-HANA-Datenbankbenutzern verwaltet. Mit dem OAuth-Token steht ein einheitlicher Authentifizierungsansatz zur Verfügung, der auch den Umfang der Berechtigungen des Benutzers festlegt, damit je nach Umfang funktionale Berechtigungsprüfungen für jede Serviceschicht in einer Anwendung ausgeführt werden können.

OData-Services

Bei OData handelt es sich um das von SAP empfohlene Protokoll für einen RESTful-Datenzugriff. Um Unterstützung für OData-Services zu bieten, hat SAP OData-Serverbibliotheken für eine Verwendung mit SAP HANA XSA bereitgestellt, einschließlich Bibliotheken für Java und Node.js. Ihre SAP-HANA-XSA-Anwendungen können ihre eigenen Datenanbieter über die OData-Bibliotheken definieren oder `xsodata`-Metadatenartefakte oder CDS-Views (Core Data Services) als generische Datenanbieter definieren.

Hinweis

CDS-Views werden in diesem Buch nicht näher erläutert. Sie bieten Entwicklern die Möglichkeit, Entitäten und die Beziehungen zwischen diesen Entitäten über eine SQL-basierte Data Definition Language (DDL) zu definieren. Diese SQL-basierte DDL wurde erweitert, damit unter anderem Beziehungen zwischen CDS-Views, Annotationen hinsichtlich der spezifischen Verwendung von CDS-Objekten und Aggregationsausdrücke definiert werden können. Weitere Informationen zu CDS-Views erhalten Sie im Beitrag »Enhanced ABAP Development with Core Data Services (CDS) – How CDS Brings a Modern Data Modeling Approach to ABAP« von Karl Kessler, veröffentlicht am 8. Oktober 2015 in Volume 16, Issue 4 in *SAPinsider*.

Administrationswerkzeuge

Damit Sie Ihre SAP-HANA-XSA-Landschaft und alle von Ihnen entwickelten Anwendungen überwachen können, stellt SAP Administrationswerkzeuge für SAP HANA XSA bereit. Diese Werkzeuge können problemlos mit dem

SAP HANA Cockpit integriert werden (siehe Kapitel 10, »Administration von SAP HANA«, für weitere Informationen zum Hinzufügen von Werkzeugen zum Cockpit). Folgende Administrationswerkzeuge können mit dem SAP HANA Cockpit integriert werden:

- ▶ **Application Monitor** – überwacht SAP-HANA-XSA-Anwendungen.
- ▶ **Organization and Space Management** – dient zum Anlegen, Auflisten und Löschen von User Spaces und Organisationen in SAP HANA XSA.
- ▶ **Application Role Builder** – dient zum Verwalten und Pflegen von Benutzerrollen und Kollektionen in SAP HANA.
- ▶ **SAML Identity Providers Configuration** – ermöglicht die Konfiguration von SAML-Identitätsanbietern für SAP-HANA-XSA-Modellanwendungen, die SAML verwenden.
- ▶ **User Management** – dient zum Verwalten und Anlegen von Business-Benutzern für SAP HANA XSA.
- ▶ **SAP HANA Logical Database Configuration** – dient zum Verwalten von Datenbankinstanzen für SAP-HANA-XSA-Anwendungen.
- ▶ **SAP HANA Service Brokers** – dient zum Überwachen und Verwalten von SAP-HANA-XSA-Service-Brokern.
- ▶ **Job Scheduler Service Dashboard** – dient zum Terminieren, Anlegen und Verwalten von langwierigen SAP-HANA-XSA-Jobs.

3.2.2 Die SAP-HANA-XSA-Laufzeit installieren

Bevor Sie mit der Installation des SAP-HANA-XSA-Laufzeitserverns beginnen, müssen Sie ermitteln, ob es sich bei Ihrem SAP-HANA-Server um ein Single-Host- oder Multi-Host-System handelt. Die Single-Host-Installation ist am einfachsten vorzunehmen und zu konfigurieren und kann nach Bedarf vergrößert werden. Im Hostsystem werden die Hostrollen `worker` und `xs_worker` für die Datenbank für eine Single-Host-Installation benötigt. Abbildung 3.13 zeigt ein Beispiel für eine Systemservicelandschaft in einem Single-Host-System.

Hinweis

Die SAP-HANA-XSA-Laufzeit wird momentan angepasst und laufend aktualisiert. Daher sollten Sie vor einer Installation die SAP-Hinweise für SAP HANA XSA durchsehen. In Tabelle 3.1 finden Sie eine Liste aller relevanten Hinweise zu diesem Thema, die Sie vor der Installation lesen sollten.

Beachten Sie, dass eine Aktualisierung des SAP-HANA-XSA-Servers immer zu Kompatibilitätsproblemen mit dem SAP-HANA-XS-Client führen kann. Daher sollten Sie sich bei einem Upgrade Ihres Servers immer die aktuelle Version des Clients über die SAP-HANA-Installationsmedien besorgen, bevor Sie versuchen, eine Verbindung herzustellen.

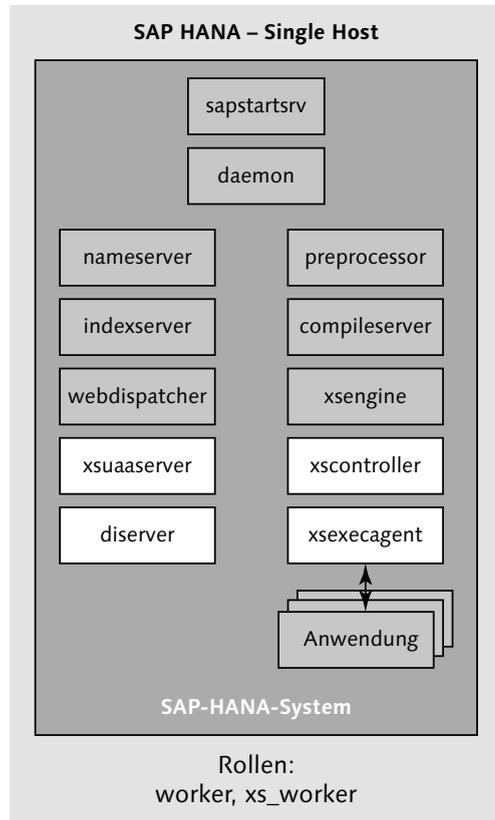


Abbildung 3.13 Single-Host-Deployment von SAP HANA XSA

SAP-Hinweis	Beschreibung
2242468	Einrichtung von SAP HANA Extended Application Services, Advanced Model über den Kommandozeilen-Client von der SAP-HANA-Installations-DVD
2303772	SAP HANA XS Advanced Model SPS 12, Release-Informationen
2324508	Deinstallation der Softwarekomponenten im SAP HANA XS Advanced Model

Tabelle 3.1 SAP-Hinweise mit Informationen zu SAP HANA XSA

SAP-Hinweis	Beschreibung
2245631	Konfiguration von Domänen und Routings für die SAP HANA Extended Application Services, Advanced Model
2304873	SAP Web IDE für SAP HANA SPS 12 – zentrale Release-Informationen
2300937	Sicherung und Wiederherstellung für SAP HANA Extended Application Services, Advanced Model
2300936	Failover & Hochverfügbarkeit mit SAP HANA Extended Application Services, Advanced Model
2298750	SAP-HANA-Plattform SPS 12 Release-Informationen
2243019	Bereitstellung von SSL-Zertifikaten für in SAP HANA Extended Application Services, Advanced Model definierte Domänen
2239095	SAP HANA XS ADVANCED DEMO MODEL – SHINE XSA Release-Informationen & allgemeine Informationen
2326004	So deinstallieren Sie SAP HANA XS Advanced manuell
2313789	Einschränkungen mit SAP HANA XS Advanced SPS 12
2243156	Sicherere Benutzereinrichtung für SAP HANA Extended Application Services, Advanced Model
2244998	Einschränkungen mit SAP HANA XS Advanced SPS 11

Tabelle 3.1 SAP-Hinweise mit Informationen zu SAP HANA XSA (Forts.)

Eine Multi-Host-Installation von SAP HANA XSA sorgt für einen Lastausgleich zwischen den einzelnen Hosts. Bei dieser Installation wird die Serversoftware in einem gemeinsamen Dateisystem installiert und das Dateisystem muss von allen Hosts im verteilten System gemountet werden. Bei dieser Konfiguration können die Systemhosts aktiv sein oder sich im Leerlauf befinden und nur aktiviert werden, wenn die Systemlast steigt. Abbildung 3.14 zeigt ein Beispiel für die SAP-HANA-XSA-Systemlandschaft in einem einfachen Multi-Host-System.

Sie können für Ihren SAP-HANA-XSA-Laufzeitserver auch eine erweiterte Installation durchführen, bei der Ihre Rollen `worker`, `xs_worker`, `standby` und `xs_standby` über mehrere Systeme verteilt werden. Bei dieser Art der Installation muss die Rolle für jedes System während der Installation von SAP HANA XSA ermittelt und zugewiesen werden. Ein Beispiel für ein erweitertes Multi-Host-Deployment, einschließlich `worker`-System, `xs_worker`-System und `standby`-System, finden Sie in Abbildung 3.15.

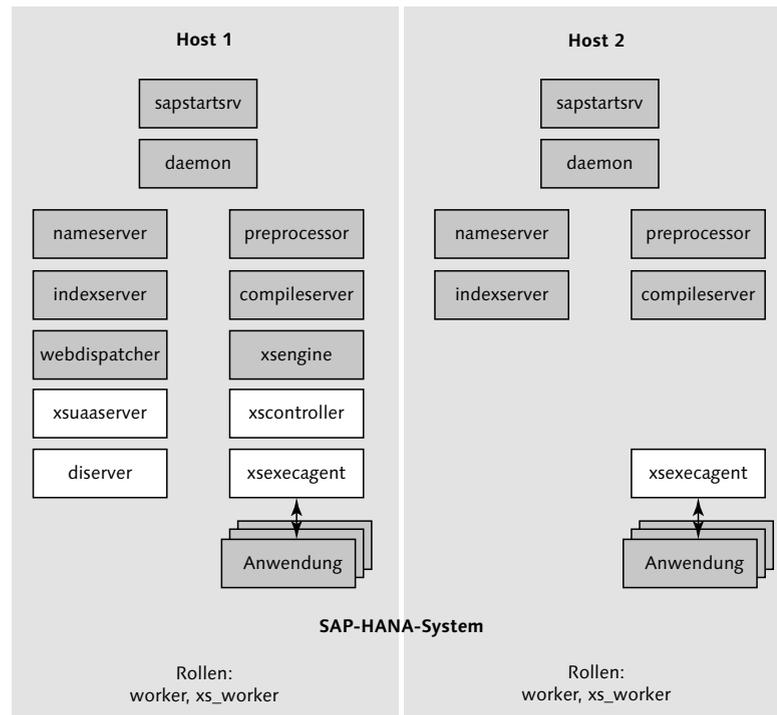


Abbildung 3.14 Einfaches Multi-Host-Deployment von SAP HANA XSA

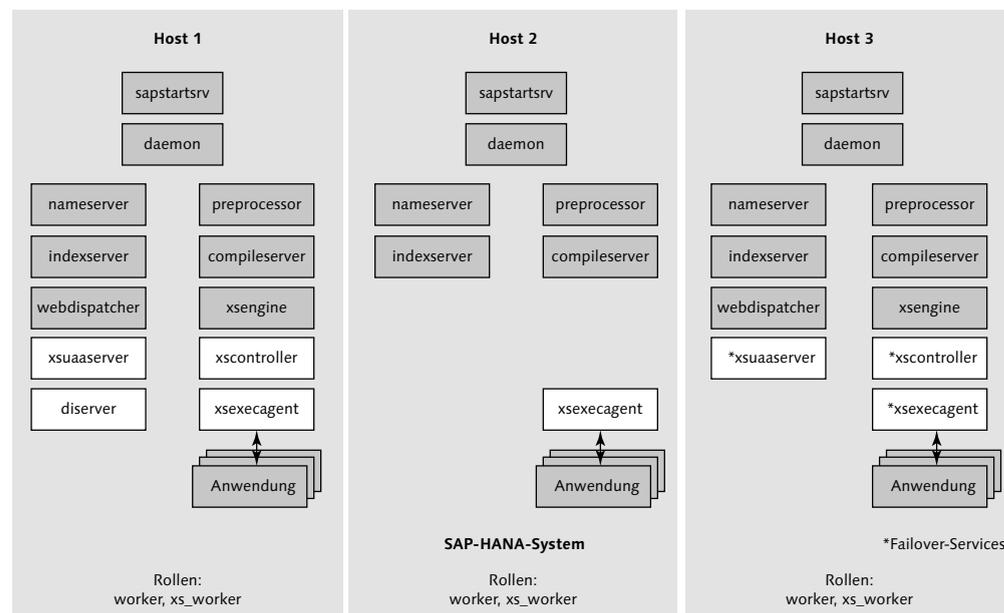


Abbildung 3.15 Erweitertes Multi-Host-Deployment von SAP HANA XSA mit Failover

Die Installation der SAP-HANA-XSA-Laufzeit ist relativ einfach, egal welche Version des SAP HANA Lifecycle Manager (SAP HANA LCM) Sie verwenden. Daher beschreiben wir in den folgenden Abschnitten die Schritte für die Installation der Laufzeit – sowohl mit der grafischen Benutzeroberfläche als auch über die Befehlszeile des SAP HANA LCM. Anschließend erläutern wir die Installation der SAP Web IDE.

Voraussetzung

Da Sie SAP-HANA-XSA-Anwendungen immer mit der SAP Web IDE erstellen sollten, sollten Sie die SAP-HANA-XSA-Komponenten, die Sie installieren möchten, herunterladen, bevor Sie mit der Installation beginnen. Die Komponenten können Sie vom SAP Marketplace unter <http://launchpad.support.sap.com/> herunterladen.

Klicken Sie auf die Kachel SOFTWARE DOWNLOADS, und wählen Sie anschließend BY ALPHABETICAL INDEX (A-Z) • H • SAP IN-MEMORY (SAP HANA) • HANA PLATFORM EDITION • SAP HANA PLATFORM EDITION • SAP HANA PLATFORM EDIT. 1.0 aus. Klicken Sie auf die Schaltfläche SUPPORT PACKAGES AND PATCHES oben rechts, und wählen Sie anschließend ENTRY BY COMPONENT aus, um die unten aufgeführten Komponenten zu suchen:

- ▶ XS ADVANCED RUNTIME
- ▶ HANA SHINE CONTENT FOR XSA (optional)
- ▶ XS ADVANCED SERVICES
- ▶ XS ADVANCED MONITORING
- ▶ SAP WEB IDE 4 SAP HANA
 - DI CORE 1
 - SAP WEB IDE 1
- ▶ SAP HANA RUNTIME TOOLS

Nachdem Sie alle Komponenten heruntergeladen haben, müssen Sie auf dem Linux-Host des SAP-HANA-Systems einen Ordner anlegen, in dem Sie die Installationsmedien speichern. Der Einfachheit halber können Sie auch ein separates Verzeichnis in diesem Verzeichnis erstellen, das die extrahierten SAP-HANA-XSA-Laufzeitkomponenten enthält.

Anschließend können Sie alle Installationsprogramme in dieses Installationsverzeichnis extrahieren und einfach im Lifecycle Manager auf diesen Speicherort mit der Option INCLUDE SUBDIRECTORIES verweisen, um weitere Komponenten hinzuzufügen.

Wählen Sie nun die Komponenten, die Sie installieren möchten, in der nachfolgend beschriebenen Reihenfolge aus. Die SAP-HANA-XSA-Laufzeit wird als Komponente ermittelt, die jedes Mal installiert werden soll. Abbildung 3.16 zeigt ein Beispiel eines Installationsverzeichnisses, das wir für die Installation unserer Komponenten verwendet haben.

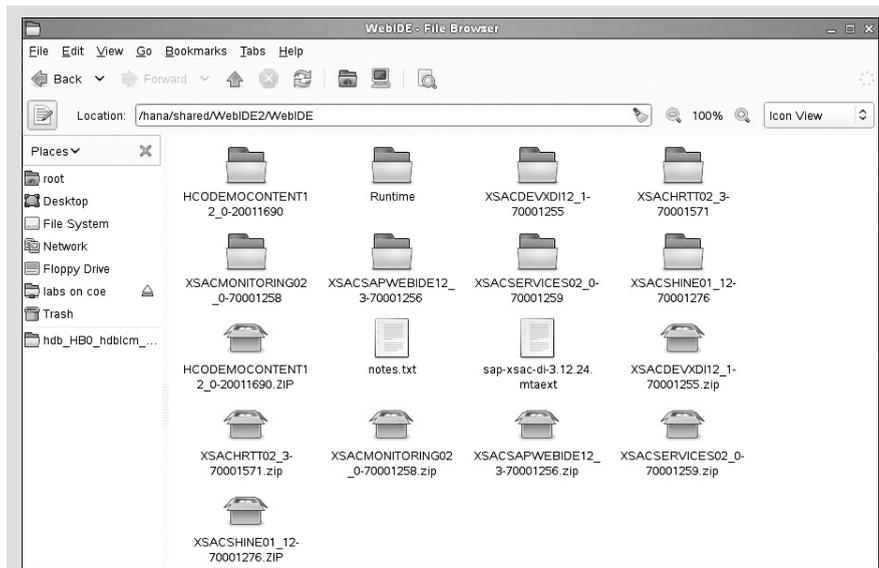


Abbildung 3.16 SAP-HANA-XSA-Komponenten mit einem Installationspfad

Warnung!

Wenn Sie die SAP-HANA-XSA-Laufzeit installieren, sollten Sie den vollqualifizierten Domainnamen für Ihr SAP-HANA-System angeben. Falls Sie dies nicht tun und stattdessen die lokale Domain auswählen, können Authentifizierungsprobleme auftreten, wenn Sie die SAP-HANA-XS-Befehlszeile (XS CLI) verwenden, um die URLs für den Zugang zur SAP Web IDE wie in Abschnitt 3.3 beschrieben abzurufen.

Installation mit der Webversion des SAP HANA Lifecycle Managers

Um die SAP-HANA-XSA-Laufzeit zu installieren, benötigen Sie den SAP HANA LCM. Sie können die Version entweder mit der grafischen Benutzeroberfläche (Graphical User Interface, GUI), der webbasierten Benutzeroberfläche oder der Befehlszeile des Lifecycle Manager verwenden. Die Installation mit der GUI-Version des Lifecycle Manager bzw. mit dem webbasierten Lifecycle Manager ist in der Regel einfacher.

Hinweis

Eine vollständige Beschreibung des Installationsprozesses finden Sie im *SAP HANA Server Installation and Update Guide*, den Sie im PDF-Format unter <http://help.sap.com/> herunterladen können. Die Schritte für die Installation sind bei der webbasierten und der GUI-Version des Lifecycle Manager nahezu analog, daher verwenden wir in diesem Abschnitt Beispiele aus der webbasierten Version.

Führen Sie folgende Schritte aus, um die SAP-HANA-XSA-Laufzeit zu installieren:

1. Melden Sie sich als Root-Benutzer am SAP-HANA-System an.
2. Navigieren Sie zum Verzeichnis mit dem Installationsmedium.
3. Geben Sie `<install_path>/<sid>/hdb1cmweb` in die Befehlszeile ein, um die webbasierte Version des SAP HANA Lifecycle Manager zu öffnen.
4. Wählen Sie **INSTALL NEW SYSTEM** aus, und klicken Sie anschließend auf **NEXT**.
5. Klicken Sie auf **ADD SOFTWARE LOCATIONS** (siehe Abbildung 3.17).

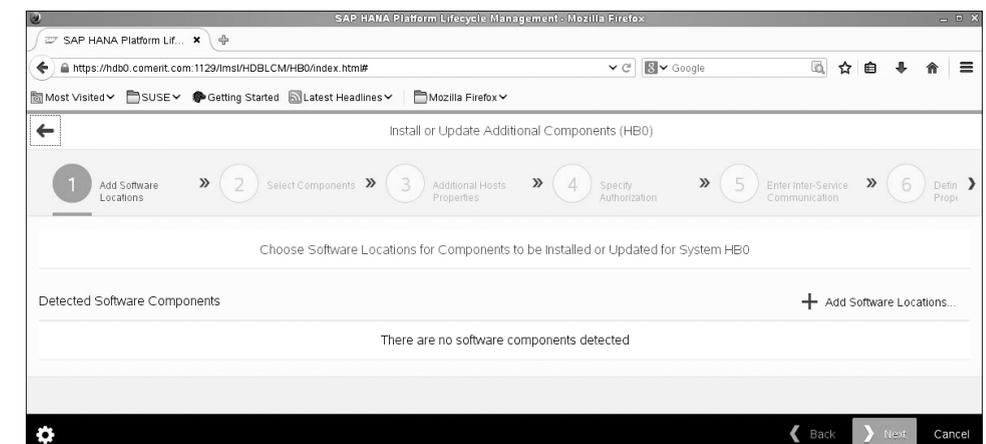


Abbildung 3.17 SAP HANA Lifecycle Manager: Webversion

6. Navigieren Sie zum Speicherort der Installationsmedien, und klicken Sie auf **OK**.
7. Bestätigen Sie die Auswahl **DETECTED SOFTWARE COMPONENTS**, indem Sie auf **NEXT** klicken.
8. Aktivieren Sie das Kontrollkästchen **INSTALL SAP HANA XS ADVANCED RUNTIME**, und klicken Sie auf **NEXT**.
9. Wählen Sie ein System aus **EXISTING HOSTS** in der SAP-HANA-Systemliste aus, und klicken Sie auf **NEXT**.
10. (Optional:) Wählen Sie die Option **ADD HOST** aus, um Parameter für zusätzliche Hosts in einer Multi-Host-Installation anzugeben. Folgende Angaben sind erforderlich:
 - **INSTALLATION PATH**: Pfad zum SAP-Mount-Verzeichnis (Wenden Sie sich für diese Information an Ihren Systemadministrator.)

- NON-STANDARD SHARED FILE SYSTEM: wird normalerweise verwendet, wenn die SID im Mount-Point enthalten ist.
- HOST NAME: vollqualifizierter Hostname der Maschine.
- ROLE: Zweck des Hosts, ausgewählt aus den Optionen von Tabelle 3.2.

Rolle	Rollenbezeichnung	Beschreibung
Database Worker	worker	Für die Datenbankverarbeitung verwendeter Work-Host
Database Standby	standby	Im Leerlauf und verfügbar für Failover bei Hochverfügbarkeit
Dynamic Tiering Worker	extended_storage_worker	Host für das Dynamic Tiering
Dynamic Tiering Standby	extended_storage_standby	Standby-Host für das Dynamic Tiering
Accelerator for SAP ASE Worker	ets_worker	Work-Host für den SAP ASE Accelerator
Remote Data Sync	rdsync	Host für den Remote-Daten-abgleich
Smart Data Streaming	streaming	Host für das Smart Data Streaming
XS advanced runtime worker	xs_worker	Host für SAP HANA XSA
XS advanced runtime standby	xs_standby	Verfügbar für Failover/dynamische Skalierung

Tabelle 3.2 SAP-HANA-Host-Systemrollen

- Wählen Sie für SYSTEM TYPE entweder die Option SINGLE-HOST SYSTEM oder die Option MULTIPLE-HOST SYSTEM aus. (Diese Aufforderung wird in Single-Host-Systemlandschaften nicht angezeigt.)
- Wählen Sie im Fenster AUTOMATIC ASSIGNMENT OF XS RUNTIME die Option YES oder NO für die Rollen aus. Mit dieser Option legen Sie fest, ob Sie die SAP-HANA-XS-Laufzeitrolle dem Host der ausgewählten Datenbank zuweisen möchten oder nicht. Falls Sie dies nicht tun möchten, müssen Sie einen anderen Host für die Laufzeitrolle angeben.
- Machen Sie Angaben für die Felder SYSTEM ADMINISTRATOR (<SID>ADM) PASSWORD, DATABASE USER NAME und DATABASE USER PASSWORD, und klicken Sie auf NEXT.

- Im Bildschirm CONFIGURE XS ADVANCED RUNTIME (siehe Abbildung 3.18) müssen Sie folgende Angaben machen:
 - BENUTZER: XS ORGANIZATION MANAGER (standardmäßig XSA_ADMIN)
 - XS ORGANIZATION MANAGER USER PASSWORD
 - CUSTOMER ORGANIZATION NAME
 - CUSTOMER SPACE NAME
 - ROUTING MODE
 - PORTS: Die SAP-HANA-XSA-Systemservices werden der Host-Domain mit unterschiedlichen Ports für jeden Service zugewiesen (*www.<Domain>.<Erweiterung>:3<Instanz>32* ist z. B. normalerweise der Link für den UAA-Security-Service).
 - HOSTNAMES: Die SAP-HANA-XSA-Systemservices werden der Host-Domain mit unterschiedlichen Domainpräfixen zugewiesen (z. B. *http://uaa-security.<Domain>.<Erweiterung>* und *http://webide.<Domain>.<Erweiterung>*).
 - DOMAIN NAME (wir empfehlen einen vollqualifizierten Domainnamen)

Abbildung 3.18 Konfigurieren der SAP-HANA-XSA-Laufzeiteinstellungen

- Klicken Sie auf NEXT, um die Einstellungen zu bestätigen.
- Vergeben Sie Benutzer-IDs für XS ADVANCED SAP SPACE OS USER ID und XS ADVANCED CUSTOMER SPACE OS USER ID.
- Klicken Sie auf NEXT, um die Benutzer-IDs zu bestätigen.

18. (Empfohlen:) Wählen Sie die Updates XS MONITORING 1 und XS SERVICES 1 für die Installation aus, sofern diese verfügbar sind, und klicken Sie auf NEXT.
19. Bestätigen Sie die von Ihnen eingegebenen Systemeinstellungen auf dem Bildschirm REVIEW AND CONFIRM ENTERED DATA (siehe Abbildung 3.19).

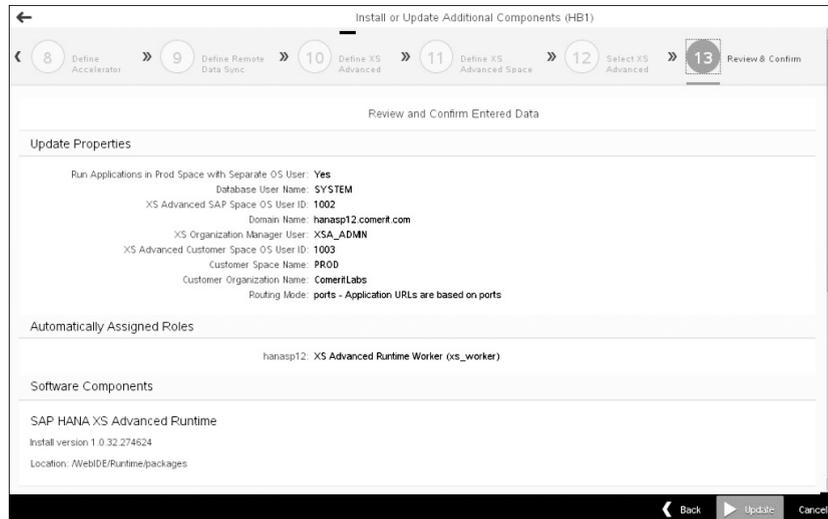


Abbildung 3.19 Bestätigen der Systemeinstellungen für die SAP-HANA-XSA-Laufzeit

20. Klicken Sie auf UPDATE, um die Laufzeit zu installieren.

Die Installation beginnt, und Sie erhalten eine Benachrichtigung über eine erfolgreiche Installation (siehe Abbildung 3.20).

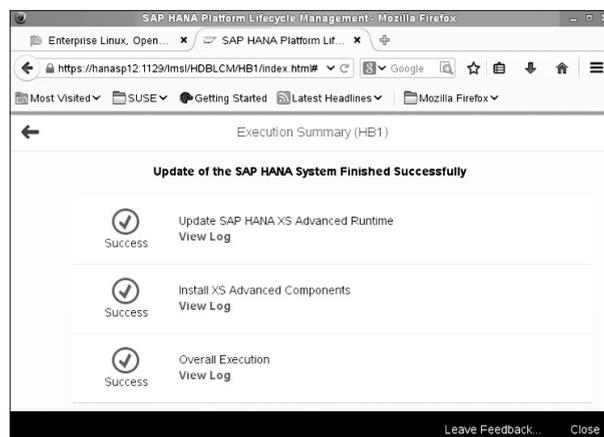


Abbildung 3.20 Erfolgreich installierte SAP-HANA-XSA-Laufzeit

Installation mit der Befehlszeile des SAP HANA Lifecycle Managers

Um SAP HANA XSA über die Befehlszeile zu installieren, müssen Sie folgende Schritte ausführen:

1. Melden Sie sich als Root-Benutzer am SAP-HANA-System an.
2. Gehen Sie zum Speicherort der Installationsmedien.
3. Führen Sie über die Befehlszeile den Befehl `<install_path>/<sid>/hdblcm` im Verzeichnis für die Installationsmedien aus.
4. Wählen Sie die Option UPDATE in der Befehlszeile aus. In unserem Beispiel (siehe Abbildung 3.21) lautet unsere SID HB1, weshalb wir 1 ausgewählt haben. Drücken Sie anschließend auf .

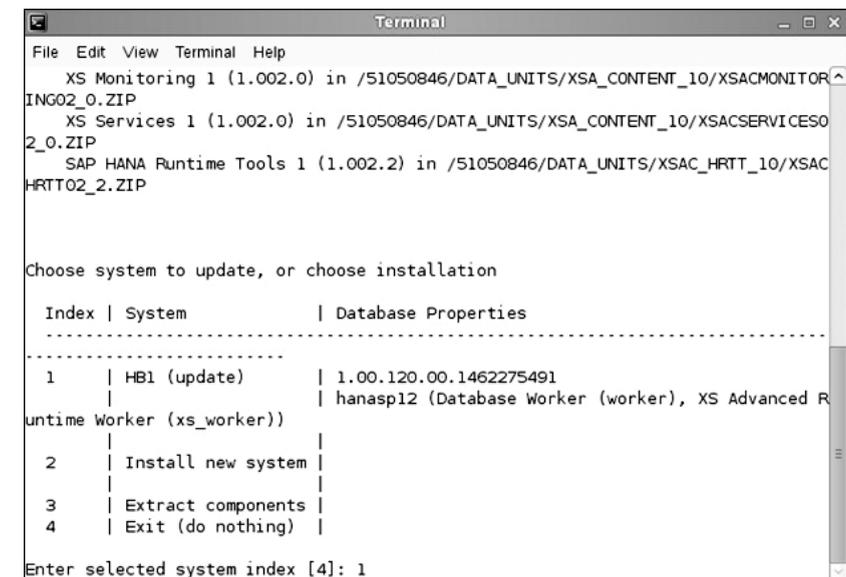


Abbildung 3.21 Die SAP-HANA-Installation über die Befehlszeile aktualisieren

5. Wählen Sie die Option UPDATE SAP HANA XS ADVANCED RUNTIME aus, und drücken Sie .
6. Geben Sie die Anmeldeinformationen für den Systemadministrator, Systembenutzer und SAP-HANA-XSA-Administrator ein (siehe Abbildung 3.22).
7. Wählen Sie die Komponenten aus, die Sie installieren möchten. Arbeiten Sie mit durch Komma getrennten Indizes für Mehrfachselektionen, oder geben Sie für alle »1« ein.
8. Geben Sie »y« über die Tastatur ein, und drücken Sie , um die Auswahl der Komponenten zu bestätigen.

```

Terminal
File Edit View Terminal Help

Index | Components | Description
-----|-----|-----
1 | all | All components
2 | server | Update SAP HANA Database from version 1.00.120.00.1462275
491 to version 1.00.120.00.1462275491
3 | client | Install SAP HANA Database Client version 1.00.120.024.146
1933179
4 | afl | Install SAP HANA AFL (incl.PAL,BFL,OFL,HIE) version 1.00.
120.00.1462291682
5 | smartda | Install SAP HANA Smart Data Access version 1.00.6.001.0
6 | studio | Install SAP HANA Studio version 2.3.5.000000
7 | trd | Install SAP TRD AFL FOR HANA version 1.00.120.00.14622916
82
8 | xs | Update SAP HANA XS Advanced Runtime from version 1.0.24.2
68783 to version 1.0.24.268783

Enter comma-separated list of the selected indices [2,3]: 8
Enter System Administrator (hbladm) Password:
Enter Database User Name [SYSTEM]:
Enter Database User (SYSTEM) Password:
Enter XS Advanced Admin User [XSA_ADMIN]:
Enter XS Advanced Admin User Password:

```

Abbildung 3.22 Die Systemanmeldeinformationen für die SAP-HANA-XSA-Installation eingeben

Die Installation von SAP HANA XSA sollte nun beginnen. Wenn die Installation abgeschlossen ist, können Sie zur Administrator Console im SAP HANA Studio navigieren und verifizieren, dass sich die SAP-HANA-XSA-Komponenten im Status ACTIVE befinden (siehe Abbildung 3.23).

Active	Host	Port	Service	Detail	Start Time	Process ID	CPU	Memory
<input checked="" type="checkbox"/>	hdb0	30010	compileserver		Jul 27, 2016 11:32:53 AM	72744		
<input checked="" type="checkbox"/>	hdb0	30000	daemon		Jul 27, 2016 11:32:34 AM	72524		
<input checked="" type="checkbox"/>	hdb0	30025	diserver		Jul 27, 2016 11:35:31 AM	74453		
<input checked="" type="checkbox"/>	hdb0	30011	dpserver	master	Jul 27, 2016 11:33:07 AM	72874		
<input checked="" type="checkbox"/>	hdb0	30003	indexserver	master	Jul 27, 2016 11:33:08 AM	72876		
<input checked="" type="checkbox"/>	hdb0	30001	nameserver	master	Jul 27, 2016 11:32:44 AM	72541		
<input checked="" type="checkbox"/>	hdb0	30002	preprocessor		Jul 27, 2016 11:32:56 AM	72746		
<input checked="" type="checkbox"/>	hdb0		sapstartsrv					
<input checked="" type="checkbox"/>	hdb0	30004	scriptserver	master	Jul 27, 2016 11:33:08 AM	72878		
<input checked="" type="checkbox"/>	hdb0	30006	webdispatcher		Jul 27, 2016 11:35:25 AM	74455		
<input checked="" type="checkbox"/>	hdb0	30029	xscontroller		Jul 27, 2016 11:37:23 AM	74457		
<input checked="" type="checkbox"/>	hdb0	30007	xsengine		Jul 27, 2016 11:33:08 AM	72880		
<input checked="" type="checkbox"/>	hdb0	59975	xexecagent		Jul 27, 2016 11:36:00 AM	74459		
<input checked="" type="checkbox"/>	hdb0	30031	xuaaserver		Jul 27, 2016 11:36:39 AM	74466		

Abbildung 3.23 Installierte und aktivierte SAP-HANA-XSA-Modellservices

Installieren der SAP Web IDE

Die SAP Web IDE ist eine Komponente der SAP-XSA-Plattform, wie auch teilweise in der Dokumentation für SAP HANA XSA beschrieben. Sie sollten sie trotzdem als separate Installation betrachten, auch wenn sie oft dieselben Komponenten wie die SAP-HANA-XSA-Laufzeit benötigt und Sie die SAP Web IDE nur installieren können, wenn die SAP-HANA-XSA-Laufzeit zu Beginn des Installationsprozesses als DETECTED SOFTWARE COMPONENT identifiziert wurde.

Warnung!

Navigieren Sie zu dem Speicherort, an dem das Laufzeitpaket extrahiert wurde, und fügen Sie es als Komponente zum Lifecycle-Manager-Prozess hinzu. Wenn Sie das nicht tun, erhalten Sie eine wenig hilfreiche Fehlermeldung, dass die anderen Komponenten für die Laufzeit installiert werden müssen, unabhängig davon, ob die Plattform bereits installiert wurde oder nicht.

Auch bei diesem Installationsprozess sind die Installationsdateien für die Laufzeitplattform als Teil der Komponenteninstallationen erforderlich. Stellen Sie also sicher, dass Sie immer die SAP-HANA-XSA-Plattform für alle Komponenteninstallationen ausgewählt haben.

Die Reihenfolge, in der Sie die SAP-HANA-XSA-Komponenten installieren, spielt eine große Rolle. Sie können nicht einfach alle Komponenten auswählen und sie alle auf einmal installieren. Jedes Mal, wenn wir dies versucht haben, traten Fehler auf, die sich nicht beheben ließen. (Wir mussten auf einen früheren Snapshot unserer virtuellen Maschine zurückgreifen.)

Unabhängig von der gewählten Methode sollten Sie die Komponenten in folgender Reihenfolge installieren:

1. Installieren Sie die Komponenten SAP HANA XS ADVANCED RUNTIME, XS SERVICES 1 und XS MONITORING 1.
2. Installieren Sie die SAP HANA RUNTIME TOOLS.
3. Installieren Sie die SAP WEB IDE DEVELOPMENT INFRASTRUCTURE.
4. Installieren Sie den SAP WEB IDE WEB CLIENT.
5. Installieren Sie SAP HANA DEMO MODEL FOR XS ADVANCED 1.0 (optionale Schulungsinhalte).

Jeder dieser Installationsschritte sollte als ein separater Installationszyklus im SAP HANA Lifecycle Manager abgeschlossen werden.

In Tabelle 3.3 erhalten Sie eine Beschreibung der SAP-HANA-XSA-Komponenten.

SAP-HANA-XSA-Komponente	Beschreibung
SAP-Web-IDE-Webclient	Umfassende integrierte browserbasierte Entwicklungsumgebung für komplexe Anwendungen mit webbasierten oder mobilen Benutzeroberflächen, Business-Logik und SAP-HANA-Datenmodellen
SAP-Web-IDE-Entwicklungsinfrastruktur	Wichtige Komponente der SAP Web IDE; wird benötigt, um die SAP HANA Web-Based Development Workbench in Ihrer Umgebung zu aktivieren.
XS Monitoring	Ermöglicht die Wartung der Anwendungsumgebung, einschließlich Sicherheit, Authentifizierung und Ressourcen-Monitoring.
XS Services	Ermöglicht die Verwendung von Katalogen mit vom SAP HANA Service Broker verwalteten Services, einschließlich OAuth-Clients, Benutzerkonten und Jobeinplanung.
SAP-HANA-Laufzeitwerkzeuge	Werkzeuge, die Unterstützung für die Design-Zeit-Entwicklung und Laufzeitverwaltung von MTAs mit SAP HANA XSA bieten
SAP HANA Demo Model for XS Advanced	SAP-HANA-Interactive-Education-Inhalte (SHINE) für SAP HANA XSA

Tabelle 3.3 SAP-HANA-XSA-Installationskomponenten

3.3 SAP-HANA-XSA-Anwendungen mit der SAP Web IDE entwickeln

Sie können SAP-HANA-XSA-Anwendungen entweder mit der XS CLI oder mit der SAP Web IDE entwickeln. In diesem Abschnitt befassen wir uns mit beiden Ansätzen für die Entwicklung von SAP-HANA-XSA-Anwendungen. Wir beginnen mit der XS CLI und erläutern anschließend die Entwicklung einer SAP-HANA-XSA-Beispielanwendung mit der SAP Web IDE.

Bevor wir mit der eigentlichen Entwicklungsarbeit mit der SAP Web IDE beginnen, möchten wir Ihnen einen Überblick über die Entwicklung von Anwendungen mit SAP HANA XSA verschaffen. Um SAP-HANA-XSA-Anwendungsprojekte erfolgreich abzuschließen, müssen Sie folgende Schritte ausführen:

1. Legen Sie die Ordnerinfrastruktur für Ihre Anwendungsdateien an.
2. Erstellen Sie die Deployment-Deskriptordateien.

3. Fügen Sie Datenbankartefakte und -inhalte nach Bedarf hinzu.
4. Fügen Sie die Business-Logik für die Datenbankartefakte hinzu.
5. Legen Sie OData-Servicedefinitionen nach Bedarf an.
6. Erstellen Sie die Benutzeroberfläche.
7. Sorgen Sie für Sicherheit im Projekt.
8. Definieren Sie Anwendungsrouten.
9. Legen Sie Serviceinstanzen für die Anwendung an.
10. Fügen Sie eingeplante Jobs nach Bedarf hinzu.
11. Deployen Sie die Anwendung.
12. Testen Sie die Anwendung.

In diesem Abschnitt beschreiben wir einige einfache Beispiele, um Ihnen den Einstieg in die Entwicklung Ihrer eigenen SAP-HANA-XSA-Anwendungen zu erleichtern. Viele der Informationen, einschließlich der Beschreibung, wie Sie die Tiny-World-Anwendung entwickeln, finden Sie im Entwicklerleitfaden *SAP HANA Developer Guide for SAP HANA XS Advanced Model*, den Sie unter <http://help.sap.com> herunterladen können.

Da man mit der SAP Web IDE das Potenzial von SAP HANA als Anwendungsplattform ideal ausschöpfen kann, erläutern wir in den nächsten Abschnitten folgende Schritte:

1. Abfragen der SAP-Web-IDE-Adresse
2. Einrichten eines Projekts
3. Entwickeln und Ausführen von Anwendungsmodulen

Hinweis

SAP Web IDE ist das Mittel der Wahl für die Entwicklung von SAP-HANA-XSA-Anwendungen. Daher liegt der Schwerpunkt auf diesem Entwicklungswerkzeug, auch wenn Sie den XS Advanced Command-Line Client (XS CLI) ebenfalls zur Entwicklung von SAP-HANA-XSA-Anwendungen verwenden können. Der XS CLI ist nicht sehr benutzerfreundlich und sollte nur von erfahrenen Entwicklern verwendet werden. Wir werden hier nicht näher auf den XS CLI eingehen. Eine Beschreibung der einzelnen Schritte für die XS-CLI-Entwicklung finden Sie unter <http://help.sap.com/>. Navigieren Sie zu TECHNOLOGY PLATFORM • SAP HANA PLATFORM • SAP HANA PLATFORM CORE SPS 12 • INTRODUCTION TO APPLICATION DEVELOPMENT AND DEPLOYMENT (XS ADVANCED MODEL) • GETTING STARTED WITH APPLICATION DEVELOPMENT IN XS ADVANCED • WORKING WITH THE XS ADVANCED COMMAND-LINE CLIENT.

Als Grundlage dient uns hier der Abschnitt über die TINY-WORLD-Anwendung aus WORKING WITH THE SAP WEB IDE FOR SAP HANA auf dieser Seite.

Danach sollten Sie ausreichend Hintergrundwissen gesammelt haben, um den Einstieg in SAP HANA XSA als Ihre Anwendungsplattform zu schaffen. Die Zielsetzung dieses Abschnitts ist nicht, eine vollständige Abhandlung zu liefern. Eine umfassende Darstellung der verwandten Themen zur Entwicklung von Anwendungen mit SAP HANA XSA (z. B. Debugging, Bündeln von Modulen und Deployment Ihrer Anwendung) finden Sie im Entwicklungsfaden *SAP HANA Developer Guide for SAP HANA XS Advanced Model*.

3.3.1 Die SAP-Web-IDE-Adresse abfragen

Als ersten Schritt Ihrer Entwicklungsarbeit mit SAP HANA XSA müssen Sie die SAP Web IDE öffnen. Die zuvor genannte XS CLI spielt hierbei eine Rolle, da Sie über die XS CLI die URL für Ihre SAP-Web-IDE-Installation ermitteln. SAP-Hinweis 2242468 enthält eine Anleitung zum Extrahieren des SAP-HANA-XS-Clients.

Zunächst einmal müssen Sie den SAP-HANA-XS-Client für Ihr System installieren. Dazu müssen Sie die Installationsmedien für die SAP-HANA-Plattform auf Ihrem Rechner extrahieren und zum Ordner `DATA_UNITS • XSA_CLIENT_10` navigieren (siehe Abbildung 3.24).

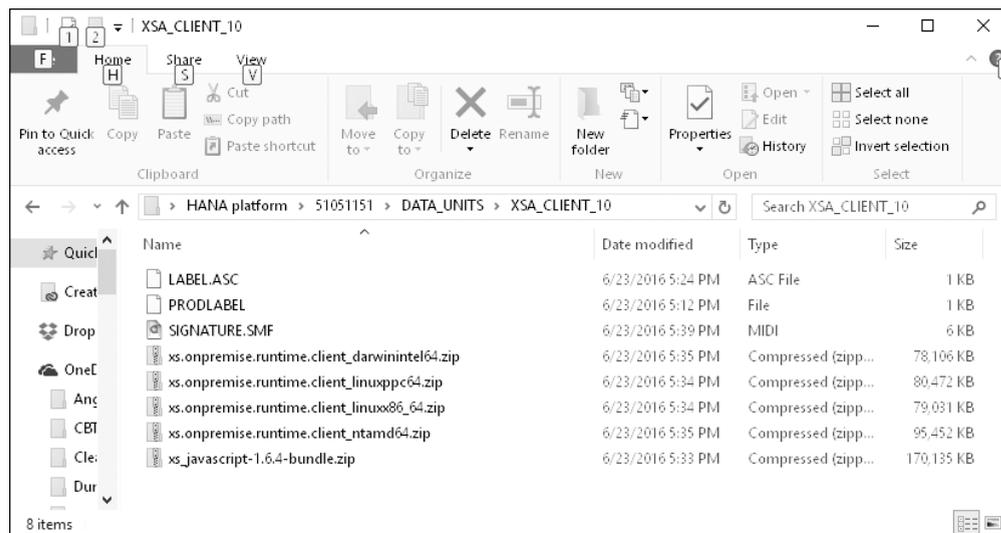


Abbildung 3.24 Der SAP-HANA-XS-Client in den SAP-HANA-Plattform-Installationsmedien

Wählen Sie die für Ihr System passende ZIP-Datei aus (der Windows-Client befindet sich z. B. in der Datei `xs.onpremise.runtime.client_ntamd64.zip`), und extrahieren Sie sie in einem Ordner Ihres Systems. Wenn Sie Windows als Sys-

tem einsetzen, navigieren Sie zu dem Ordner, in dem Sie die Dateien extrahiert haben, und halten `⌘` gedrückt, während Sie mit der rechten Maustaste auf den Ordner `BIN` klicken, um das alternative Kontextmenü für den Ordner zu öffnen. Wählen Sie aus dem Kontextmenü die Option `OPEN COMMAND WINDOW HERE` aus, um eine Eingabeaufforderung im Ordner `bin` zu öffnen.

Hinweis

Die Optionen für die Befehlszeile in den folgenden Abschnitten funktionieren sowohl in Windows- als auch in Linux-Systemen. Wenn Sie Linux einsetzen, müssen Sie das Fenster `TERMINAL` im Speicherort `BIN` öffnen und das Präfix `./«` zu allen Befehlen hinzufügen.

In der Eingabeaufforderung können Sie den XS-CLI-Client verwenden, um die URLs für die SAP-HANA-XSA-Plattformkomponenten abzufragen. Zunächst einmal aber müssen Sie die Clientumgebung und die Anmeldung einrichten. Um die Clientumgebung einzurichten und eine sichere SSL-Verbindung zwischen Ihrem System und der SAP-HANA-XSA-Plattform zu konfigurieren, müssen Sie SSL für Ihr System einrichten. In SAP-Hinweis 2242468 erhalten Sie Informationen zur Konfiguration Ihres Systems für SSL. Sie können sich auch an Ihren Systemadministrator wenden, um das SSL-Zertifikat für Ihren Server zu erhalten. Mit dem Zertifikat können Sie anschließend über folgenden Befehl Ihren Client mit dem Server verbinden:

```
xs api https://<FQDN>:3<instance>30 -cacert <path/certificate>
```

Als Nächstes müssen Sie sich an der Plattform anmelden. Geben Sie hierzu `xs login` in der Eingabeaufforderung ein. Sie werden aufgefordert, Ihren Benutzernamen und Ihr Kennwort einzugeben, und Sie sollten mit der Rückgabe der Systemverbindungsdaten benachrichtigt werden, dass Ihre Authentifizierung erfolgreich war (siehe Abbildung 3.25).

Um die URL für Ihre SAP-Web-IDE-Verbindung zu erhalten, geben Sie `xs app webide -urls` in der Eingabeaufforderung ein. Im Fenster `COMMAND PROMPT` wird Ihnen die Webadresse für Ihre SAP-Web-IDE-Verbindung angezeigt.

Hinweis

Über den Befehl `xs -v` in der XS CLI erhalten Sie die Adressen von weiteren wichtigen Komponenten in Ihrem SAP-HANA-XSA-System. Sie benötigen insbesondere die Adresse für das Portal `XS ADVANCED ADMINISTRATION AND MONITORING TOOLS`, in dem Sie Bereiche für Ihre Anwendungen anlegen, Entwicklerbenutzer hinzufügen und Benutzer Entwicklungsbereichen zuweisen müssen. Über den Service `xsa-admin` können Sie die Adresse für dieses Portal beziehen (siehe Abbildung 3.26).

```

C:\XSClient2\bin>xs api https://hanasp12.comerit.com:30030 -cacert default.root.crt.pem
API endpoint: https://hanasp12.comerit.com:30030 (API version: 1)
Not logged in. Use "xs login" to log in.
C:\XSClient2\bin>xs login
API URL: https://hanasp12.comerit.com:30030
USERNAME: XSA_ADMIN
PASSWORD:
Authenticating...
ORG: ComeritLabs
SPACE: SAP
API endpoint: https://hanasp12.comerit.com:30030 (API version: 1)
User: XSA_ADMIN
Org: ComeritLabs
Space: SAP
C:\XSClient2\bin>_

```

Abbildung 3.25 Erfolgreich hergestellte Verbindung zur SAP-HANA-XSA-Laufzeitplattform

```

C:\XSClient2\bin>xs -v
Client version: xs v1.0.28
Server version information:
  name           = XS Controller
  support        = http://service.sap.com/message
  build          = v1.0.32
  version        = 1
  user           = <not set>
  description    = SAP XS Runtime on premise
  controllerEndpoint = https://hanasp12.comerit.com:30030
  authorizationEndpoint = https://hanasp12.comerit.com:30032/uaa-security
  loggingEndpoint = <not set>
  allowDebug     = true
  limits         = memory: <not set>, apps: <not set>, app uris: <not set>, services: <not set>
  usage         = memory: <not set>, apps: 27, app uris: <not set>, services: 23
  databaseType  = <not set>
Registered service URLs:
  deploy-service      = https://hanasp12.comerit.com:51002
  xsa-admin           = https://hanasp12.comerit.com:51005
  job-scheduler-service-dashboard = https://hanasp12.comerit.com:51012
  hrtt-service        = https://hanasp12.comerit.com:51013
  hrtt-core           = https://hanasp12.comerit.com:51007
  product-installer   = https://hanasp12.comerit.com:51003
C:\XSClient2\bin>_

```

Abbildung 3.26 Adressen für SAP-HANA-XSA-Serviceportale

3.3.2 Projekteinrichtung

Mit der Adresse für die SAP Web IDE können Sie nun Ihr Projekt einrichten. Navigieren Sie so, wie im vorherigen Abschnitt beschrieben, zur gefundenen Webadresse, und melden Sie sich mit Ihrem SAP-HANA-XSA-Entwicklerbenutzer an der Plattform an. Die SAP Web IDE wird geöffnet (siehe Abbildung 3.27), und Sie können Ihr Projekt einrichten.

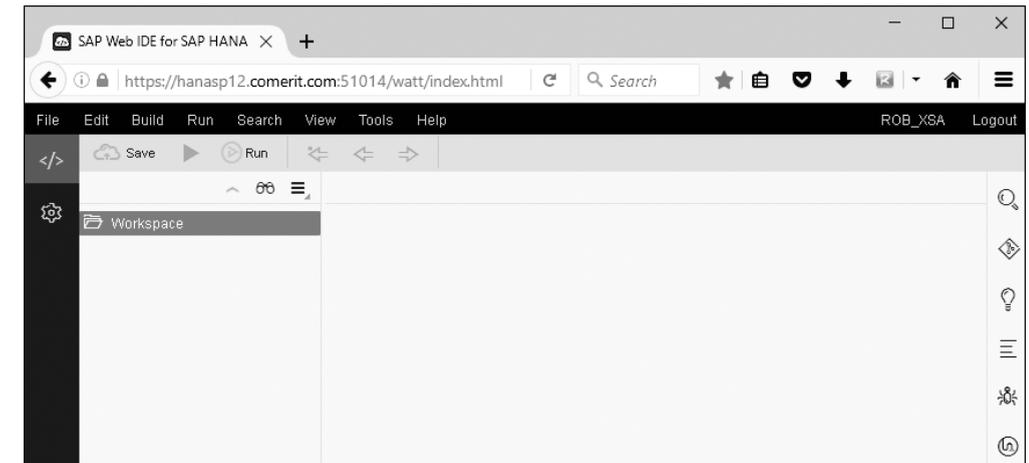


Abbildung 3.27 SAP Web IDE für die Entwicklung von SAP-HANA-XSA-Anwendungen

Wählen Sie in der SAP Web IDE die Option FILE • NEW • PROJECT FROM TEMPLATE aus. Das Fenster TEMPLATE SELECTION wird geöffnet, und Sie sollten eine Vorlage für ein MULTI-TARGET APPLICATION PROJECT auswählen können (siehe Abbildung 3.28). Wählen Sie die Vorlage aus, und klicken Sie auf NEXT.

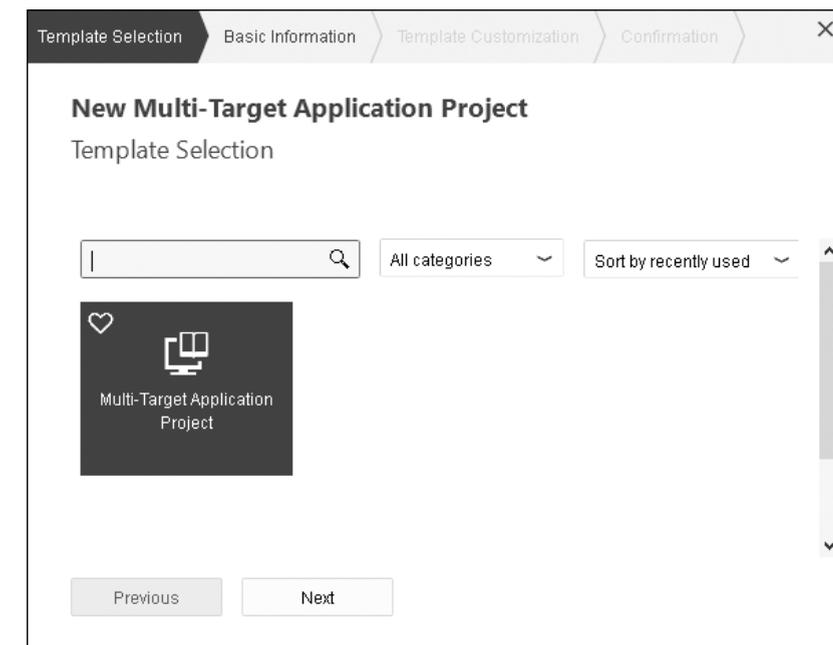


Abbildung 3.28 Eine SAP-HANA-XSA-Anwendung aus einer Vorlage erstellen

Vergeben Sie einen Namen für das Projekt, und klicken Sie auf NEXT. Da wir mit dem Tutorial aus dem Entwicklerleitfaden als Grundlage arbeiten, haben wir »TinyWorld« unter PROJECT NAME eingegeben (siehe Abbildung 3.29). Klicken Sie auf NEXT, um zur Registerkarte TEMPLATE CUSTOMIZATION zu gelangen.

Abbildung 3.29 Erstellen der TinyWorld-Anwendung

Auf der Registerkarte TEMPLATE CUSTOMIZATION können Sie für die neue Anwendung Angaben für APPLICATION ID, APPLICATION VERSION, DESCRIPTION und SPACE (für die Entwicklung) machen (siehe Abbildung 3.30). Geben Sie die erforderlichen Informationen ein, und klicken Sie auf NEXT, um zur Registerkarte CONFIRMATION zu gelangen. Klicken Sie dort auf FINISH, um das neue Projekt anzulegen.

Abbildung 3.30 Geben Sie aussagefähige Informationen für Ihr SAP-HANA-XSA-Projekt ein.

Die neue Anwendung wird im Ordner WORKSPACE in der SAP Web IDE angezeigt, und das neue Projekt enthält nur eine Datei mit dem Namen *mta.yaml*. Bei dieser Datei handelt es sich um die MTA-Deskriptordatei, die Abhängigkeiten oder Voraussetzungen enthält, die benötigt werden, wenn die Anwendung deployt werden kann. Wenn Sie Code und Module zu Ihrem Projekt hinzufügen, wird die Datei größer werden. Wenn das Projekt erfolgreich angelegt wurde, können Sie mit der Entwicklung des neuen Projekts beginnen.

3.3.3 Anwendungsmodule entwickeln und ausführen

Als Nächstes fügen wir ein Datenbankmodul zum Projekt hinzu. Klicken Sie dazu mit der rechten Maustaste auf den Projektordner TINYWORLD, und wählen Sie die Option NEW HDB MODULE aus dem Kontextmenü aus. Das Fenster NEW • HDB MODULE wird geöffnet, und Sie müssen unter MODULE NAME einen Modulnamen für das Projekt eingeben. In unserem Beispiel (siehe Abbildung 3.31) haben wir unser Modul »tinyWorldDB« genannt. Klicken Sie auf NEXT, um zur Registerkarte TEMPLATE CUSTOMIZATION zu gelangen.

Abbildung 3.31 Anlegen eines neuen SAP-HANA-Datenbankmoduls

Bestätigen Sie die Option INITIAL VIEW DETAILS für NAMESPACE. Wenn Sie alle Schritte für unser Beispiel ausgeführt haben, sollten Sie hier bereits `TinyWorld.tinyWorldDB` angegeben haben. Klicken Sie auf NEXT, um zur Registerkarte CONFIRMATION zu gelangen. Klicken Sie dort auf FINISH, um das neue Datenbankmodul anzulegen. Der Objektordner TINYWORLDDB wird im Projektordner TINYWORLD hinzugefügt, und der neue Objektordner enthält eine Datei mit dem Namen SRC, die zunächst leer erscheint.

Wenn Sie jedoch im SAP-Web-IDE-Menü die Option VIEW • SHOW HIDDEN FILES auswählen, werden Sie sehen, dass der Ordner SRC Dateien mit den Namen *.hdiconfig* und *.hdinamespace* enthält. Die Datei *.hdiconfig* wird für SAP-HANA-XSA-Projekte benötigt. In ihr werden die Plug-ins spezifiziert, die SAP HANA benötigt, um ein Katalogobjekt anzulegen. In der Datei *.hdinamespace* sind Regeln für Laufzeitnamensräume festgelegt. Diese Datei kann optional verwendet werden.

Als Nächstes müssen wir eine einfache Datenbanktabelle in unserem Projekt über CDS-Programmierung (Core Data Services) anlegen. Hierbei handelt es sich um eine verbesserte und erweiterte Version von SQL zum Definieren und Verwenden von Datenmodellen. Führen Sie folgende Schritte aus, um das neue Datenmodell anzulegen:

1. Klicken Sie mit der rechten Maustaste im Ordner TINYWORLDDDB auf den Ordner SRC, und wählen Sie im Kontextmenü die Option NEW CDS ARTIFACT aus.
2. Geben Sie »tinyTable« als Namen im Fenster CREATE NEW CDS FILE ein (siehe Abbildung 3.32).

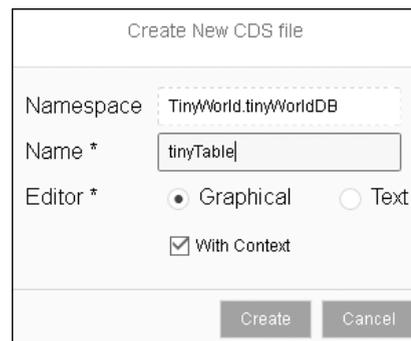


Abbildung 3.32 Ein neues CDS-Datenmodell anlegen

3. Wählen Sie für EDITOR die Option GRAPHICAL aus, und aktivieren Sie das Kontrollkästchen WITH CONTEXT.
4. Klicken Sie auf CREATE, um das neue Datenmodell anzulegen, und prüfen Sie, ob die Datei mit dem Namen TINYTABLE.HDBCDS im Ordner SRC angelegt wurde.
5. Klicken Sie doppelt auf die neue Datei, um den Editor zu öffnen.

6. Geben Sie den Code aus x im Editor-Bereich ein:

```
namespace TinyWorld.tinyWorldDB;
context tinyTable{
    entity world {
        key continent: String(100);
    };
};
```

Listing 3.2 Code für die Datei »tinyTable.hdbcds«

Hinweis

Der Code in Listing 3.2 definiert eine Tabelle mit dem Namen »world« mit einem Feld mit dem Namen »continent.« und einen Feldtyp mit bis zu 100 Zeichen in einem String.

7. Speichern Sie das HDB-Modul über das Symbol SAVE oder über die Tastenkombination **[Strg]+[S]**.
8. Wählen Sie im Dateimenü der SAP Web IDE die Option BUILD • BUILD aus.

Sie sollten eine Nachricht darüber erhalten, dass der Build erfolgreich im Konsolenfenster abgeschlossen wurde. Ihr einfaches Datenmodell wurde also in der SAP-HANA-Datenbank aktiviert und kann zum Speichern und Abrufen von Daten verwendet werden.

Als Nächstes legen wir ein Node.js-Modul an, das auf der XSJS-Bibliothek basiert. Führen Sie folgende Schritte aus, um ein einfaches Node.js-Modul zu erstellen:

1. Klicken Sie mit der rechten Maustaste auf den Ordner TINYWORLD, um das Kontextmenü zu öffnen.
2. Wählen Sie NEW NODE.JS MODULE aus dem Kontextmenü aus.
3. Nennen Sie das Modul »tinyWorldJS«.
4. Aktivieren Sie das Kontrollkästchen für ENABLE XSJS SUPPORT.
5. Klicken Sie auf FINISH.
6. Doppelklicken Sie auf die Datei *lib/index.xsjs*, um sie im JavaScript-Editor zu öffnen.
7. Ersetzen Sie im JavaScript-Editor den Standardcode durch folgenden Code:

```
$.response.contentType = "text/html";
$.response.setBody("Welcome to the tiny JS World!");
```

8. Speichern Sie die Datei mit der Tastenkombination `Strg+S`.
9. Klicken Sie mit der rechten Maustaste auf den Ordner TINYWORLDJS, um das Kontextmenü zu öffnen.
10. Wählen Sie die Option RUN • RUN AS • NODE.JS APPLICATION aus.

Eine neue Browserregisterkarte mit der Adresse für das neue XSJS-Modul wird geöffnet (oder, je nach Browsereinstellungen, ein neues Fenster). Wenn wir die Option zum Ausführen der Anwendung auswählen, befehlen wir dem Editor implizit, die Anwendung zu erstellen und anschließend auszuführen, damit wir zurück zur SAP-Web-IDE-Konsole navigieren können, um den Fortschritt zu überwachen. Der Status der Anwendung sollte schnell zu STATUS: RUNNING wechseln, und die Nachricht »Welcome to the tiny JS World!« sollte im Browserfenster der Anwendung angezeigt werden.

Das Hinzufügen eines HTML5-Moduls zu Ihrer neuen Anwendung erfolgt nahezu analog zum Hinzufügen eines Node.js-Moduls. Die einzige Ausnahme ist der Unterschied zwischen den Programmiersprachen von HTML5 und Node.js. Führen Sie folgende Schritte aus, um ein HTML5-Modul hinzuzufügen:

1. Klicken Sie mit der rechten Maustaste auf den Ordner TINYWORLD, um das Kontextmenü zu öffnen.
2. Wählen Sie NEW HTML5 MODULE aus dem Kontextmenü aus.
3. Nennen Sie das Modul »tinyWorldUI«.
4. Klicken Sie auf FINISH.
5. Doppelklicken Sie auf die Datei `resources/index.html`, um sie im Editor zu öffnen.
6. Ersetzen Sie den Standardcode durch folgenden Code:

```
<!DOCTYPE html>
<html> <body> Welcome to your Tiny HTML World! </html>
```

7. Speichern Sie die Datei mit der Tastenkombination `Strg+S`.
8. Klicken Sie mit der rechten Maustaste auf den Ordner TINYWORLDJS, um das Kontextmenü zu öffnen.
9. Wählen Sie die Option RUN • RUN AS • WEB APPLICATION aus.

Wie schon beim Node.js-Modul sollte sich eine neue Browserregisterkarte oder ein neues Browserfenster öffnen, und der Text »Welcome to your Tiny HTML World!« sollte angezeigt werden.

An dieser Stelle werden wir unsere Erläuterung zur SAP Web IDE abschließen, da Sie nun über ausreichend Hintergrundwissen für einen erfolgreichen Einstieg verfügen sollten. Um weitere, detailliertere Informationen zum Erstellen von SAP-HANA-XSA-Anwendungen zu erhalten, können Sie den Entwicklerleitfaden *SAP HANA Developer Guide for SAP HANA XS Advanced Model* herunterladen oder die hervorragenden Blogbeiträge von Chaim Bendelac zum Thema *Developing with XS Advanced: A TinyWorld Tutorial* lesen. In ihnen beschreibt Chaim Bendelac sowohl die grundlegende als auch die erweiterte Entwicklung mit SAP HANA XSA. Teil 1 des Blogs finden Sie im SCN unter <http://s-prs.de/v450700>.

3.4 Zusammenfassung

In diesem Kapitel haben wir die Funktionen vorgestellt, die Sie mit SAP HANA als Anwendungsplattform nutzen können. Sie können einfache Anwendungen mit der SAP HANA Web-Based Development Workbench entwickeln, aber zum Erstellen von angepassten, hochleistungsfähigen Webanwendungen ist die SAP Web IDE die ideale Entwicklungsoberfläche. Die Wahl Ihrer Entwicklungsoberfläche hängt von dem Support-Package-Stack Ihres SAP-HANA-Systems ab, und die SAP Web IDE wird regelmäßig aktualisiert, um ihre Effektivität und Benutzerfreundlichkeit zu optimieren. Die Kombination von SAP HANA als Anwendungsplattform und den neuen leistungsfähigen webbasierten Entwicklungswerkzeugen ermöglicht es Ihnen, alle möglichen Anwendungen ohne Einschränkungen auf einem der schnellsten Datenbanksysteme der Welt zu erstellen: SAP HANA.

Einführung

SAP HANA entwickelt sich mit jedem Release weiter – und wir freuen uns, dass sich dieses Buch parallel dazu weiterentwickeln kann. Willkommen zur dritten Auflage von »SAP HANA – Die neue Einführung« mit neuen Informationen zu SAP HANA in der Cloud, SAP S/4HANA, der SAP Web IDE und der SAP HANA Web-Based Development Workbench.

Wenn Sie dieses Buch zum ersten Mal lesen, finden Sie hier eine kurze Übersicht über die Themen der einzelnen Kapitel:

- ▶ In **Kapitel 1** erhalten Sie eine Einführung in das Thema mit allgemeinen Erläuterungen zum In-Memory-Computing und zum zeilen- und spaltenbasierten Speichern. Außerdem erfahren Sie, wie diese beiden Aspekte in SAP HANA aufeinandertreffen.
- ▶ In **Kapitel 2** gehen wir genauer auf die Lösung ein und bieten Informationen darüber, was mit SAP HANA möglich ist (und was nicht!). Wir skizzieren Big-Data-Lösungen und spezifische Implementierungsoptionen.
- ▶ **Kapitel 3** ist ganz neu und enthält Informationen über SAP als Anwendungsplattform. Außerdem werden SAP HANA XS Classic, SAP HANA XSA und die SAP Web IDE besprochen.
- ▶ **Kapitel 4** ist ebenfalls neu und stellt die SAP Cloud Platform, das PaaS-Angebot von SAP, vor.
- ▶ **Kapitel 5**, das letzte vollständig neue Kapitel, stellt Informationen über SAP S/4HANA bereit. Dabei konzentrieren wir uns auf SAP S/4HANA Finance und SAP S/4HANA Materials Management and Operations.
- ▶ In **Kapitel 6** erläutern wir die ersten Schritte zur Planung einer SAP-HANA-Implementierung mit ausführlichen Informationen zu allen vier Arten der On-Premise-Versionen.
- ▶ In **Kapitel 7** stellen wir Ihnen das Reporting mit SAP HANA vor. Dabei gehen wir auf die Business-Intelligence-Tools von SAP, einschließlich SAP BusinessObjects Lumira, ein, und wir verknüpfen diese Tools mit SAP HANA und SAP HANA Live.
- ▶ In **Kapitel 8** skizzieren wir die Optionen zur Datenmodellierung in SAP HANA und gehen dabei auf die Calculation Views (sowohl Dimension als auch Cube) und die SAP HANA Web-Based Development Workbench ein.

- ▶ In **Kapitel 9** erhalten Sie Informationen über die Datenbeschaffung mit SAP Data Services, SAP Landscape Transformation, dem SAP Replication Server und SAP HANA Smart Data Access (SDA).
- ▶ In **Kapitel 10** runden wir das Buch mit einer Erläuterung der SAP-HANA-Administration ab. Hier erhalten Sie die neuesten Informationen über das SAP HANA Cockpit, die mehrstufige Datenverwaltung usw.

Unser Ziel bei allen Auflagen war und ist es, Ihnen Informationen zur erstmaligen Umsetzung von SAP HANA an die Hand zu geben und Sie auf die nächsten Schritte vorzubereiten. Wir vertrauen darauf, dass diese neueste Ausgabe diese Tradition fortsetzt.

Penny Silvia

Rob Frye

Dr. Berg

Auf einen Blick

1	In-Memory-Computing, Big Data und SAP HANA	19
2	SAP HANA als Datenbank	41
3	SAP HANA als Anwendungsplattform	107
4	SAP Cloud Platform	153
5	SAP S/4HANA	169
6	Planung einer SAP-HANA-Implementierung	189
7	Reporting mit SAP HANA	267
8	Datenmodellierung mit SAP HANA	315
9	Datenbeschaffung	407
10	Administration von SAP HANA	485

Inhalt

Danksagung	15
Einführung	17
1 In-Memory-Computing, Big Data und SAP HANA	19
1.1 Einführung in das In-Memory-Computing und Big Data	20
1.1.1 In-Memory-Computing und Analysen	21
1.1.2 Big Data	27
1.2 Spaltenbasierte Speicherung versus zeilenbasierte Speicherung	32
1.3 Zusammenfassung	40
2 SAP HANA als Datenbank	41
2.1 Die Grundlagen von SAP HANA	41
2.1.1 Was SAP HANA kann	44
2.1.2 Was SAP HANA nicht kann	45
2.2 Big-Data-Lösungen	46
2.2.1 SAP HANA	47
2.2.2 Dynamisches Tiering	50
2.2.3 SAP HANA Smart Data Access	53
2.2.4 SAP HANA Vora und Hadoop	57
2.3 SAP HANA für nicht SAP-bezogene Anwendungsfälle	60
2.3.1 SAP HANA für nicht SAP-bezogene Analysen	60
2.3.2 SAP HANA für Nicht-SAP-Anwendungen	74
2.4 SAP BW auf SAP HANA	76
2.4.1 Technische Voraussetzungen	84
2.4.2 Erforderliche Qualifikationen	87
2.4.3 Schritte der Projektplanung	90
2.4.4 Wann wählen Sie SAP BW auf SAP HANA?	92
2.5 SAP Business Suite powered by SAP HANA	94
2.6 SAP S/4HANA	97
2.7 Weitere Funktionen von SAP HANA	99
2.7.1 SAP HANA Smart Data Streaming	99
2.7.2 SAP HANA Predictive Analysis Library	101
2.7.3 SAP HANA Graph Engine	103
2.7.4 SAP HANA Sentiment Analysis	103
2.8 Zusammenfassung	104

3	SAP HANA als Anwendungsplattform	107
3.1	SAP HANA XS Classic	108
3.1.1	Architektur	109
3.1.2	SAP-HANA-XS-Classic-Anwendungen entwickeln	110
3.2	SAP HANA XSA	120
3.2.1	Die SAP-HANA-XSA-Architektur	121
3.2.2	Die SAP-HANA-XSA-Laufzeit installieren	127
3.3	SAP-HANA-XSA-Anwendungen mit der SAP Web IDE entwickeln	140
3.3.1	Die SAP-Web-IDE-Adresse abfragen	142
3.3.2	Projekteinrichtung	144
3.3.3	Anwendungsmodule entwickeln und ausführen	147
3.4	Zusammenfassung	151
4	SAP Cloud Platform	153
4.1	Überblick	154
4.2	Funktionen	156
4.2.1	Zugriff und Berechtigungen	156
4.2.2	Einrichtungsservices	157
4.2.3	Schemata	158
4.2.4	Content Management	159
4.2.5	Mobile Services	160
4.3	SAP Cloud Platform Cockpit	160
4.4	SAP Cloud Platform Portal	164
4.5	Erste Schritte	166
4.6	Zusammenfassung	168
5	SAP S/4HANA	169
5.1	Überblick	169
5.2	SAP S/4HANA Finance	174
5.2.1	Universal Journal	175
5.2.2	Periodenabschluss	176
5.2.3	Anlagenbuchhaltung	177
5.2.4	SAP Cash Management	177
5.2.5	SAP-Material-Ledger	177
5.2.6	SAP BPC for SAP S/4HANA Finance	178
5.3	SAP S/4HANA Materials Management and Operations	179
5.3.1	Fertigung	181
5.3.2	Logistikkette	181

5.4	Business Case für SAP S/4HANA	183
5.5	Migration von SAP ERP auf SAP S/4HANA	185
5.6	SAP Activate	187
5.7	Zusammenfassung	188
6	Planung einer SAP-HANA-Implementierung	189
6.1	Technische Grundlagen	189
6.1.1	In-Memory Computing Engine	189
6.1.2	Softwarespezifikationen	192
6.1.3	Softwareeditionen für bestimmte Zwecke	193
6.1.4	Netzwerkgeschwindigkeit zwischen den Servern	199
6.1.5	Softwareinstallation	200
6.2	Hardwarespezifikationen und -optionen	202
6.2.1	Vergrößerte SAP-HANA-Systeme	203
6.2.2	Erweiterte SAP-HANA-Systeme	203
6.2.3	Hardwareplattformen	206
6.3	SAP HANA als eigenständiges Data Warehouse	207
6.3.1	Datenmodellierung	208
6.3.2	Sizing	210
6.4	SAP Business Suite powered by SAP HANA	213
6.4.1	Neue Funktionen	214
6.4.2	Implementierungsoptionen	216
6.4.3	Sizing	222
6.5	SAP S/4HANA	227
6.5.1	Wechsel zur SAP S/4HANA Cloud	229
6.5.2	Wechsel zu SAP S/4HANA on Premise	229
6.6	SAP Business Warehouse auf SAP HANA	232
6.6.1	Sizing	234
6.6.2	Eine Migration vorbereiten	243
6.6.3	Die Migration durchführen	248
6.6.4	Die Migration optimieren	253
6.6.5	Neue Funktionen in SAP BW 7.4	261
6.6.6	Neue Funktionen in SAP BW 7.5	263
6.7	Zusammenfassung	266
7	Reporting mit SAP HANA	267
7.1	Die Werkzeuge im Überblick	267
7.1.1	SAP BusinessObjects Dashboards	268
7.1.2	SAP BusinessObjects Web Intelligence	269
7.1.3	SAP BusinessObjects Explorer	271

- 7.1.4 SAP BusinessObjects Analysis 276
- 7.1.5 SAP BusinessObjects Design Studio 277
- 7.1.6 SAP Crystal Reports 279
- 7.1.7 SAP BusinessObjects Lumira 280
- 7.2 SAP-BusinessObjects-BI-Werkzeuge mit SAP HANA verbinden 284
 - 7.2.1 Direkte Verbindung über die SAP-BusinessObjects-4.2-Plattform 285
 - 7.2.2 Universen mit Open-Database- und Java-Database-Verbindungen 288
 - 7.2.3 Verbindung zu Excel mit Open Database Objects und Multidimensional Expressions (MDX) herstellen 296
 - 7.2.4 Eine Microsoft-Query für SAP HANA erstellen 300
 - 7.2.5 BICS-Verbindungen 301
- 7.3 SAP HANA Live 301
 - 7.3.1 Installieren von SAP HANA Content Tools und SAP-HANA-Live-Content-Auslieferungseinheiten 303
 - 7.3.2 SAP HANA Live Browser 305
 - 7.3.3 Virtuelle Datenmodelle in SAP HANA Live 309
 - 7.3.4 SAP HANA Live Extension Assistant 311
- 7.4 Zusammenfassung 313

8 Datenmodellierung mit SAP HANA 315

- 8.1 Datenmodellierung mit dem SAP HANA Studio 316
 - 8.1.1 SAP-HANA-Development-Perspektive 317
 - 8.1.2 Tabellen und Daten anlegen und importieren 322
 - 8.1.3 Zeitdaten generieren 327
 - 8.1.4 Attribute Views und Analytic Views migrieren 329
 - 8.1.5 Auslieferungseinheiten und Pakete 335
- 8.2 Calculation Views: dimensionale Views 337
 - 8.2.1 Dimensionale Views 338
 - 8.2.2 Projektionsknoten (Projection Nodes) hinzufügen 342
 - 8.2.3 Filter anwenden 346
 - 8.2.4 Union-Knoten (Union Nodes) 348
 - 8.2.5 Steuerelemente im Bereich »Scenarios« 351
 - 8.2.6 Zeichenfolgenfunktionen 352
 - 8.2.7 Eingabeparameter 355
 - 8.2.8 Semantischer Knoten (Semantics Node) 360

- 8.3 Calculation Views: Cube-Views 367
 - 8.3.1 Join-Knoten (Join Nodes) 368
 - 8.3.2 Temporale und räumliche Joins 374
 - 8.3.3 Aggregationsknoten (Aggregation Nodes) 376
 - 8.3.4 Rangknoten (Rank Nodes) 378
 - 8.3.5 Währungsumrechnungen 379
- 8.4 SAP-BW-Modellierungswerkzeuge für SAP HANA Studio 385
 - 8.4.1 CompositeProvider 388
 - 8.4.2 Advanced-DataStore-Objekte 389
 - 8.4.3 Open-ODS-Views 391
- 8.5 SAP HANA Web-Based Development Workbench 392
 - 8.5.1 Catalog 395
 - 8.5.2 Security 396
 - 8.5.3 Traces 397
 - 8.5.4 Editor 398
- 8.6 Zusammenfassung 406

9 Datenbeschaffung 407

- 9.1 Auswählen einer Methode für die Datenbeschaffung 407
 - 9.1.1 Methoden der Datenbeschaffung im Überblick 408
 - 9.1.2 Strategische Überlegungen 411
 - 9.1.3 Technische Überlegungen 418
- 9.2 Triggerbasierte Datenreplikation: SAP Landscape Transformation 422
 - 9.2.1 Funktionsweise 422
 - 9.2.2 Installation 425
 - 9.2.3 Konfiguration 427
 - 9.2.4 Administration 429
 - 9.2.5 Erweiterte Funktionen 430
 - 9.2.6 Neuen Replikationskonfiguration in SAP HANA einrichten 436
 - 9.2.7 Tabellen zu einer vorhandenen Replikationskonfiguration hinzufügen 443
- 9.3 ETL-basierte Datenreplikation: SAP Data Services 448
 - 9.3.1 Voraussetzungen für die Konfiguration 449
 - 9.3.2 SAP HANA für den Empfang von Daten über SAP Data Services vorbereiten 450
 - 9.3.3 Methoden der Datenaktualisierung 453
 - 9.3.4 Laden von Daten 458

- 9.4 Log-basierte Datenreplikation: SAP Replication Server und SAP HANA Load Controller 474
 - 9.4.1 Installation 476
 - 9.4.2 SAP Replication Server ausführen 477
- 9.5 SAP HANA Smart Data Access für Datenverbund und -virtualisierung 478
 - 9.5.1 Datenbanktreiber einrichten 479
 - 9.5.2 Hadoop mit SAP HANA Smart Data Access verbinden 480
 - 9.5.3 Daten mit SAP HANA Smart Data Access konsumieren 481
- 9.6 Zusammenfassung 483

10 Administration von SAP HANA 485

- 10.1 Administration Console von SAP HANA verwenden 486
 - 10.1.1 Systeme hinzufügen 488
 - 10.1.2 Systeme exportieren und importieren 491
 - 10.1.3 Details zur Systeminstallation anzeigen 492
 - 10.1.4 Administration Editor und Diagnosemodus 494
 - 10.1.5 Speicherort für Dateien ändern 494
 - 10.1.6 Ändern von Konfigurationen 494
 - 10.1.7 Die Administration Console anpassen 495
- 10.2 SAP HANA Cockpit 497
 - 10.2.1 Customizing 499
 - 10.2.2 Rollen für Benutzer und Kataloge 502
 - 10.2.3 Offline-Administration 503
- 10.3 System-Monitoring 504
 - 10.3.1 Überwachen der Datenträgerverwendung 505
 - 10.3.2 Performance-Monitoring 507
 - 10.3.3 Überwachen mithilfe von Alerts 507
 - 10.3.4 Alerts konfigurieren 508
 - 10.3.5 Services und verteilte Systeme überwachen 509
 - 10.3.6 Tabellendaten und Definitionen exportieren und importieren 511
 - 10.3.7 Die Speichernutzung überwachen 513
 - 10.3.8 Große Tabellen durch Partitionierung verwalten 514
 - 10.3.9 Lastausgleich durch Verschieben von Dateien und Partitionen 517
 - 10.3.10 Einen Disk Full Event beheben 518
 - 10.3.11 Unterstützung für nicht reagierende Systeme 518

- 10.4 Updates 519
 - 10.4.1 SAP-HANA-Appliance aktualisieren 519
 - 10.4.2 SAP HANA Studio aktualisieren 521
- 10.5 Sicherheit 523
 - 10.5.1 Systemberechtigungen 523
 - 10.5.2 Sicherheit bei der Authentifizierung 525
 - 10.5.3 Sicherheit von Berechtigungen 525
 - 10.5.4 Definieren einer Kennwortrichtlinie 531
- 10.6 Lizenzschlüssel 532
 - 10.6.1 Temporäre Lizenzschlüssel 533
 - 10.6.2 Permanente Lizenzschlüssel 535
- 10.7 Sicherung und Hochverfügbarkeit 535
 - 10.7.1 Sicherung 536
 - 10.7.2 Hochverfügbarkeit 538
- 10.8 Virtualisierung 539
- 10.9 Mehrere Datenbanken und Komponenten auf derselben Hardware 543
 - 10.9.1 Multiple Components in One Database 543
 - 10.9.2 Multiple Components in One System 544
- 10.10 Mandantenfähigkeit (Multitenancy) und mandantenfähige Datenbank-Container 546
 - 10.10.1 Vergleich zwischen Mandantenfähigkeit und Virtualisierung 547
 - 10.10.2 Plattformübergreifende Datenverarbeitung 548
 - 10.10.3 Horizontale Skalierung mit Mandantenfähigkeit 549
- 10.11 Mehrschichtiges Datenmanagement (Multi-Tier Data Management) 550
 - 10.11.1 Dynamisches Tiering 551
 - 10.11.2 Data Lifecycle Manager 557
 - 10.11.3 SAP HANA Vora 561
- 10.12 SAP Solution Manager und SAP HANA 565
- 10.13 DBA Cockpit für SAP HANA 567
- 10.14 Zusammenfassung 570

- Die Autoren 571
- Index 573

Index

A

ABAP-Routine-Analyzer-Programm 254
Absatz- und Produktions-
grobplanung 182
Abstraktionsschicht 478
Access Control List (ACL) 159
Administration 67, 87
 Administration Editor 67, 494
 Aktualisierung 519, 521
 Alert 507
 Datenträgerverwendung
 überwachen 505
 Diagnosemodus 494
 Disk Full 518
 große Tabellen verwalten 514
 Hash-Partitionierung 516
 Installation 492
 Konfiguration ändern 494
 Lastausgleich 517
 mit Alert überwachen 507
 nicht reagierendes System 518
 Partitionierung 514
 Performance überwachen 507
 Range-Partitionierung 515
 Round-Robin-Partitionierung 516
 Service überwachen 509
 Services, Registerkarte 510
 Speichernutzung überwachen 513
 Speicherort für Dateien 494
 System exportieren und importieren 491
 System hinzufügen 488
 System Information, Registerkarte 485
 Tabellendaten 511
 Überwachung 504
 Werkzeuge 127
Administration Console 486
 anpassen 495
 Zugriff 487
Administrator Workbench 256
Advanced-DataStore-Objekt (aDSO) 234
Aggregations-Engine 80
Aggregationsknoten 376
aggregierte Daten 82, 86, 92
AKID 24, 103
Amazon Web Services 540
Ambari 480
Amount with Currency 380

Analytic View 288, 295
Änderungsmanagement 183
Anlagenbuchhaltung 175, 177
Anwendungs-Router 122
Anwendungs-Sandbox 155
Apache Spark 561
Atomarität, Konsistenz, Isolation und
 Dauerhaftigkeit 24, 103
Attribute View 288, 295

B

Bank Account Management 177
Bestands-Cube 259
BEx Query Designer 264
BEx Web Application Designer 268, 278
BI Launchpad 270, 417
BICS 269, 301
Big Data 20, 27, 29, 30, 31, 46, 49
Bitmap-Index 33
Blue-Green Deployment 123
Branchenlösung 214
Bull 202
Business Intelligence Consumer
 Services 269, 301
Business-Analyst 88
Business-Content-Repository 73
BW-Modeling-Perspektive 388

C

Calculation View 288, 295, 329, 337
 Cube 360, 367
 dimensionaler 338
 Eingabeparameter 355
 Filter anwenden 346
 grafisch 345
 Join 367
 neu anlegen 342
 Semantik extrahieren 364
 semantischer Knoten 360
 Spalte 345
 Union-Knoten 348
Cash Operations 177

CDC 453
quellbasierte 453
zeitstempelbasierte 453
zielbasierte 453, 456
 CDS 233
 Central Management Console 285
 Change and Transport System 566
 Change Data Capture → CDC
 Cisco 202, 205
 Cisco/EMC 227
 Cloud Foundry 120
 CMC 285
 CMIS 159, 164
 CMISQL 159
 CompositeProvider 234, 261, 264, 388
 Configuration and Monitoring Dashboard 427
 Content Management Interoperability Service 159, 164
 Controller-Module 423
 Controlling 175
 CO-PA 175
 Core Data Services 233
 CPU 85
 Customer Vendor Integration (CVI) 230

D

Data Aging 215
 Data Definition Language 126
 Data Governance 31, 48
 Data Lifecycle Manager 557
Administration 560
Installation 557
KPI 560
 Data Migration Server (DMIS) 426
 Database Migration Option 219, 246
 Database Structured Query Language 272
 DataStore 458, 459
 DataStore-Objekt 51, 244, 251, 254
 Dateireplikationsliste 306
 Daten
aggregierte 288
Aktualisierung 453
echtzeitnahe 412
externe 416
laden 458
operative 413
semi-strukturierte 163
strategische 414

Daten (Forts.)
unstrukturierte 416
 Datenänderungserfassung → CDC
 Datenbank-Container, mandantenfähiger 546
 Datenbankentwicklungsobjekt 320
 Datenbankmanagementsystem, relationales 251
 Datenbankmigration 87
 Datenbankmodul 147
 Datenbankschema 428
 datenbankübergreifende Abfrage 548
 Datenbeschaffung 407
Echtzeitdaten 413
externe Daten 416
Methode auswählen 407
strategische Daten 414
unstrukturierte Daten 416
 Datenfluss 449
 Datenintegrität 31, 48
 Datenkomprimierung 23
 Datenmanagement, mehrschichtiges 550
 Datenmenge 31, 47
 Datenmodellierung 72, 91, 208, 315
Advanced-DataStore-Objekt 389
Calculation View 337
dimensionale 260
SAP HANA Studio 316
SAP HANA Web-Based Development Workbench 392
SAP-BW-Modellierungswerkzeuge 385
 Datenredundanz 37
 Datenreplikation 200, 422
ETL-basierte 409, 448
log-basierte 407, 410, 474
Replikationskonfiguration 427
triggerbasierte 407, 408, 422
 Datenspeicher 85
 Datenträgerverwendung 505
 Datentransferprozess 244, 258, 423
 Datenumfang 31, 48
 Datenverarbeitung, plattformübergreifende 548
 Datenverbund 478
 Datenvirtualisierung 478
 DB02 87
 DB2 für Linux, UNIX und Windows 410
 DB6 → DB2 für Linux, UNIX und Windows
 DBA Cockpit 567, 569
DB Performance Monitor 568

DBA Cockpit (Forts.)
Transaktion 568
 DBSQL → Database Structured Query Language
 Dedicated Host Deployment 553
 Default Session Client 384
 Dell 202, 227
 Delta-Erkennung 454
 Deltapuffer 35
 Dictionary-Komprimierung 38
 DIM ID → Dimensionsschlüssel
 Dimensionsmodellierer 208
 Dimensionsschlüssel 256
 Disaster Recovery 26, 88
 DLM → Data Lifecycle Manager
 Dokumentenservice 158
 DSN-Verbindung 287
 DSO → DataStore-Objekt
 DTP → Datentransferprozess
 DWF → SAP HANA Data Warehousing Foundation
 dynamisches Tiering 50, 551, 557
Abfrageoptimierung 553
Administration und Monitoring 554
Host 553
mit gleichem Host bereitstellen 553

E

E- und F-Tabellen 243
 ECDA → Enterprise Connect Data Access
 Echtzeit 62, 65, 427
 Echtzeitdaten 412
 Echtzeitverarbeitung 173
 Eclipse IDE 167, 234, 311, 316
Mars 167
Modellierung für SAP BW 261
Neon 167
 ECP → CompositeProvider:erweiterter
 EDGE-Lizenz 193
 EIM → Enterprise Information Management
 Embedded Replication Agent System Database 475
 Embedded Replication Server System Database 475
 Engine, relationale 190
 Enterprise Connect Data Access 475
 Enterprise Data Warehouse 203
 Enterprise Information Management 69, 89

Entwicklungsstandards 114
 ERASD 475
 ERSSD 475
 ETL → Extraktion, Transformation und Laden
 Export Wizard 511
 Expression Editor 354
 Extended Table 53
 Extraktion, Transformation und Laden 46, 72, 73, 192, 407, 448
Werkzeuge 209

F

Facette 272
 Failover-System 539
 Feedbackservice 158
 Filterausdruck 347, 355
 Flatfile 458
 Frank, Christopher 30
 Fujitsu 202, 227
 Full Refresh 453

G

Geometrie 374
 Graphdatenbank 103

H

Hadoop 57, 410, 480, 561
 Hadoop Distributed File System 58, 480, 561
 Hauptspeicher 23, 25
 HDBSQL-Script 560
 HDFS → Hadoop Distributed File System
 »heiße« Daten 551
 Historientabelle 341
 Hitachi 202, 227
 Hochverfügbarkeit 538
 HP 202, 227
Converged System 500 205
Sizing-Werkzeug für die SAP Business Suite 227
 HTML 108
 HTML5-Modul 150
 HTTPS-Protokoll 155

Huawei 202, 227
hXS Engine 107

I

IBM 202, 227
Identity-Service 158
IDT → Information Design Tool
IMCE → In-Memory Computing Engine
Index-Server 538
InfoCube 244, 251, 256
 Faktentabelle 258, 260
 nach der Optimierung 256
 SAP-HANA-optimiert 256, 257
 vor der Optimierung 256
 Zukunft 260
InfoObjects 387
Information Design Tool 284, 288, 291
Information Space 271
Informationsmodell 288
In-Memory Computing Engine 42,
 110, 189
In-Memory-Computing 20, 21, 22,
 24, 25
In-Memory-Datenbank 21
Intel 202
Internet der Dinge 58, 100, 173
Investitionsschutz 205
ISO 27001 155

J

Java 108, 167
Java Database Connectivity 107, 157,
 280, 288
Java on Apache Tomcat 121
Java on Apache TomEE 121
Java Persistence API (JPA) 157
Java Runtime Environment (JRE) 154
Java Virtual Machine (JVM) 168
JavaScript 120
JavaScript-Editor 149
JDBC → Java Database Connectivity
Join
 dynamischer 370
 räumlicher 374
 temporaler 374
Join-Knoten 368

Join-Typ 372
 full 373
 inner 372
 referential 372
 Right Outer 373
 text 374

K

Kennwortrichtlinie 531
Kennzahl
 Bestand 258
 mit Währungen verknüpfen 384
 Typ 380
Kerberos 525
Komprimierung 37
Konnektivitätsservice 158
Kundenbindung 32

L

Lastausgleich auf mehreren Hosts 517
Laufzeitspeicher, dynamischer 240
LCM → SAP HANA Lifecycle Manager
Lenovo 202, 205, 207
Lenovo Labs 206
Lesemodul 423
Lesen, paralleles 449
Lightweight Directory Access Protocol
 (LDAP) 125
Linux 194, 410
Lizenzschlüssel 532
 permanenter 532, 535
 Prüfung 535
 temporärer 532, 533
Load Controller 475
Log-Speicher 85
Log-Tabelle 423
LSA++ 234, 244, 254

M

M_Service_Memory 513
Mandanten-Datenbank 546
Mandantenfähigkeit 546
 horizontale Skalierung 549
 Virtualisierung 547
MapReduce 480

Massive Parallel Processing 76
Masterjob 430
Master-Server 538
Materialbedarfsplanung 181
MCOB 543
MCOS 543, 544
MDX → Multidimensional Expression
Mehrknotensystem 85, 549
Microservice-Container 122, 125
Microsoft Excel 61, 66, 298
Microsoft-Query 300
Mobile Service 160
MPP 76
MTA → Multi-Target-Anwendung
Multidimensional Expression 43, 66,
 191, 284
Multiple Components in
 One Database 543
Multiple Components in
 One System 543, 544
MultiProvider 82
Multi-Target-Anwendung 123
 Deployment-Deskriptordatei 123
 Ordnerstruktur 124
 Regeln 124
Multitenancy → Mandantenfähigkeit

N

Name-Server 538
Nearline Storage 51, 231, 243, 569
NEC 202, 227
Netezza 410
NLS → Nearline Storage
Node.js 108
Node.js-Modul 149
NVRAM (nichtflüchtiger Speicher) 21

O

OASIS-Protokoll 164
OAuth-Token 126
Objekt, abhängiges 331
OData 111
OData-Service 126, 160
ODBC → Open Database Connectivity
ODP 78, 454
ODS 209, 261
OLAP → Online Analytical Processing

OLTP → Online Transaction Processing
Online Analytical Processing 104, 170,
 236, 277, 286, 561
Online Database Connectivity 479
Online Transaction Processing 104, 170
Open Database Connectivity 56, 107,
 280, 411, 475
Open Database Object 261, 284, 296
Open-ODS-View 54, 56, 391
Operational Data Provisioning 78, 454
Operational DataStore 209, 261
Operational DataStore-Objekt 51
operative Delta-Queue 78
operatives Reporting 74
Oracle 87
Oracle 11g 79
Order-to-Cash 173

P

PaaS 153
PAL 101
Partitionierung
 mehrstufige 515
 Round-Robin 516
Performance 507
Periodenabschluss 176
Persistent Staging Area 239
persistenter Speicher 25
Persistenz und Protokollierung 190
Persistenzservice 158
Plan-to-Produce 172
Platform as a Service 153
Plattenspeicher 190
PLM 95, 215
Private View 310
Procure-to-Pay 172
Product Availability Matrix (PAM) 418
Projektionsknoten 344, 369
Projektplanung
 SAP BW auf SAP HANA 90
 SAP HANA als Data Warehouse 71
PSA → Persistent Staging Area

Q

Quellsystem 422
Query Designer 264
Query View 310

R

R3load 475
RAM 85
Range-Partition 515
Rangknoten 378
Rapid Deployment Solution 192, 235
Rauhe, Hannes 38
RDBMS 33, 186
Regel
 ereignisbasierte 431
 parameterbasierte 432
Reisezeiten-Querys 341
relationales Datenbankmanagement-system 33, 186
Remote Function Call 68, 71, 88, 424
Replikation → Datenreplikation
Request, komprimierter 260
Request-to-Service 173
REST-Architektur 120
RESTful API 120
Reuse View 310
RFC → Remote Function Call
Rolle Administration 528
Runbook 250

S

SAML → Security Assertion Markup Language
Sandbox-Umgebung 541
SAP Activate 187
SAP Active Embedded 187
SAP BPC für S/4HANA Finance 178
SAP Business Client 430
SAP Business Explorer (BEx)
 Web Analyzer 236
SAP Business Suite powered by SAP HANA 94, 98, 175, 211, 213
 Anforderungen 95
 Architektur 95
 Greenfield-Implementierung 217
 Implementierungsoptionen 95, 216
 In-Place-Migration 219
 SAP-Hinweise 215
 Sizing 222
SAP Business Warehouse
 Accelerator 272
SAP BusinessObjects Analysis 236, 276, 277, 285

SAP BusinessObjects BI Launchpad 270, 417
SAP BusinessObjects Business Intelligence 19, 46, 63, 85, 267, 282, 284, 289
SAP BusinessObjects Dashboards 268
SAP BusinessObjects Design Studio 268, 277
SAP BusinessObjects Explorer 63, 271, 272
SAP BusinessObjects Live Office 277
SAP BusinessObjects Lumira 267, 280, 563
 an SAP HANA anbinden 282
 Visualisierung 282
SAP BusinessObjects Predictive Analytics 104
SAP BusinessObjects Web Intelligence 269
SAP BW 243
 7.3 Unicode 85
 7.4 79, 80, 232, 261
 7.5 232, 263
 ABAP Routine Analyzer 255
 Administrator Workbench 237
 CompositeProvider 388
 Cube 73, 76, 86
 Datenmodellierung 263
 Housekeeping 243
 Modellierungswerkzeuge 264, 315, 385
 Open-ODS-View 391
 SAP BI Analytical Engine 569
SAP BW Accelerator 50, 79, 257, 414
SAP BW auf SAP HANA 27, 46, 76, 87, 89, 92, 232, 414
 Anforderungen 84, 87
 Architektur 84
 Kopie migrieren 252
 Migration 243, 246, 248, 251, 252, 253
 SAP-Hinweise 249
 Sizing 234
 Standardmigration 249
 Vorteile 77
SAP BW Migration Cockpit für SAP HANA 234, 240, 245, 246, 255
SAP BW Workspace Query Designer 264
SAP Cash Management 177
SAP Cloud Identity 125
SAP Cloud Platform 153, 173, 546
 Adapter 154
 Developer-Paket 155

SAP Cloud Platform (Forts.)
 Enterprise-Paket 156
 lokale Komponenten installieren 167
 Medium-Business-Paket 155
 Schema 158
 Starter-Paket 155
 Überblick 154
 Zugriff 156, 163
SAP Cloud Platform Cockpit 157, 160
SAP Cloud Platform Integration 160
SAP Cloud Platform Portal 164
SAP CO-PA Accelerator 544
SAP CRM 95
SAP Crystal Reports 279
SAP Customer Relationship Management 95
SAP Data Services 43, 57, 63, 65, 70, 71, 192, 232, 407, 409, 448
 DataStore 459
 Daten aktualisieren 453
 Daten laden 458
 Datenempfang 450
 Datenflüsse 467
 Designer 458
 File Format Editor 465
 Hintergrundjob 452, 466, 471
 Konfiguration 449
 Metadaten definieren 464
 Metadaten importieren 462
 Metadatenreplikation 450
 Projekt 459
 Query 469
 Rapid Deployment Solutions 449
 Release 4.2 449
 Verbindung mit SAP HANA 451
SAP EarlyWatch Alert 236, 566
SAP ERP 213
SAP ERP Financials 222
SAP ERP Materials Management 225
SAP ERP Sales and Distribution 302
SAP Finance and Controlling
 Accelerator 544
SAP Fiori 170, 214, 230
SAP Fiori Launchpad 498
SAP GUI 109
SAP-GUI-Server 218
SAP HANA 19, 42
 aktualisieren vs. einfügen 35
 als Data Warehouse 61, 64, 66
 als Datenbank 66
 Anwendungsplattform 107
 Application Services 196

SAP HANA (Forts.)
 Arbeitsplatzbeschreibung 67
 Big Data 49
 Branchen 45
 Business-Analyst 68
 BW-Modeling-Perspektive 388
 Calculation und Analytic View 278
 Datenbank 41, 194
 Development-Perspektive 556
 Edition 192
 eigenständiges Data Warehouse 207
 Einsatzbereiche 45
 Endanwender-Clients 198
 erweiterte Systeme 203
 für Nicht-SAP-Anwendungen 74
 Hardwareplattform 206
 Hardwarespezifikation 202
 Implementierung 189
 Lifecycle Management 194
 Linux-Host 131
 Möglichkeiten 61
 native Anwendungen 196
 Netzwerkgeschwindigkeit 199
 Optimierung 255
 Pilotprogramm 26, 63
 Reporting 267
 Sidecar-Implementierung 60, 308, 391
 Sizing 210
 Softwarespezifikation 192
 vergrößertes System 203
 verteiltes System 514
 View 283, 285
SAP HANA Application Lifecycle Management 558
SAP HANA Cloud Platform → SAP Cloud Platform
SAP HANA Cockpit 101, 127, 198, 392, 497, 555, 558
 Administrationswerkzeuge 127
 Benutzerrolle 502
 Customizing 499
 Kachelkatalog 500
 Kacheln und Gruppen 502
 Offline-Administration 503
 Single Sign-on 499
SAP HANA Content Tools 304
SAP HANA Data Services 198
SAP HANA Data Warehousing
 Foundation 557
SAP HANA Direct Extractor
 Connection 192
SAP HANA Enterprise Cloud 157, 546

SAP HANA Graph Engine 103
 SAP HANA Lifecycle Manager 131, 132, 335, 521
 SAP HANA Live 215, 301
 Inhaltspaket 301
 View 306
 SAP HANA Live Browser 305
 SAP HANA Live Extension Assistant 310, 312
 SAP HANA Load Controller 43, 61, 191, 474
 SAP HANA Predictive
 Analysis Library 101
 SAP HANA Sentiment Analysis 104
 SAP HANA Smart Data Access 47, 54, 319, 388, 407, 410, 478, 548
 SAP HANA Smart Data Integration 57, 234, 319, 388
 SAP HANA Smart Data Streaming 99, 388
 SAP HANA Studio 46, 101, 111, 113, 197, 304, 315, 316, 403, 498
 Administrator Console 138
 aktualisieren 521
 Auslieferungseinheiten 335
 Best Practices für die Migration 333
 Development-Perspektive 317
 Expression Editor 354
 Kalendertyp 328
 Migrationswerkzeug 329
 Pakete 335
 Project-Explorer-View 320
 Projektionsknoten hinzufügen 342
 Repositories-View 319
 Schemazuordnung 327
 Session Client 339
 Systems-View 318
 Tabellen anlegen und importieren 322
 Workspace anlegen 319
 Zeitdaten generieren 327
 SAP HANA Unified Installer 200
 SAP HANA Vora 57, 481, 561
 Engine 59
 Installation 562
 SAP HANA Web-Based Development
 Workbench 111, 315, 329, 392, 560
 Calculation View anlegen 400
 Catalog 395
 Editor 398
 Inaktivität 401
 Knoten verknüpfen 404
 Security 396

SAP HANA Web-Based Development
 Workbench (Forts.)
 Trace 397
 SAP HANA XS Classic 107, 108, 109, 113, 119, 154
 .xsprivileges 117
 Administrationswerkzeuge 498
 Anwendung entwickeln 110
 Architektur 109
 HelloWorld-Beispiel 117
 inaktive Objekte aktivieren 118
 native Entwicklung 112
 Referenzanleitung 119
 Repository 114
 XS CLI 132
 SAP HANA XSA 107, 120, 154, 197, 228
 Administrationswerkzeuge 126
 Architektur 121
 C++ 121
 Installation über die Befehlszeile 137
 Installationsreihenfolge der Komponenten 139
 Laufzeitinstallation 133
 Laufzeitplattform 125
 Laufzeitserver 127, 129
 Multi-Host-Installation 129
 SAP Web IDE 140
 Schritte im Anwendungsprojekt 140
 Single-Host-Deployment 128
 UAA-Service (User Account and Authentication) 125
 SAP HCP → SAP Cloud Platform
 SAP HEC → SAP HANA Enterprise Cloud
 SAP Host Agent 61, 476, 503
 SAP Integrated Business Planning (IBP) 260
 SAP IQ 52, 57, 569
 SAP JVM 168
 SAP Landscape Transformation 61, 63, 65, 69, 71, 191, 229, 231, 308, 407, 408, 422
 Administration 429
 Filter 431
 Installation 425
 Konfiguration 427
 Log-Tabelle 423
 Release 2.0 425, 428
 Replication Server 408, 426
 Replikationsfrequenz 427
 Replikationskonfiguration 436, 443
 selektive Tabellenreplikation 431
 System 422

SAP Landscape Transformation (Forts.)
 Transformation 434
 SAP Liquidity Management 177
 SAP MaxAttention 187
 SAP NetWeaver 7.02 426
 SAP NetWeaver Application Server 230
 SAP Product Lifecycle Management 95, 215
 SAP QuickSizer 211, 222, 235
 SAP Replication Agent 474
 SAP Replication Server 43, 61, 63, 191, 407, 410, 474
 SAP S/4HANA 26, 97, 169, 227
 Business Case 183
 digitaler Kern 171
 Editionen 229
 einstufige Migration 186
 geschäftliche Sicht 183
 Landschaftstransformation 231
 Migration 185
 Neuimplementierung 232
 On-Premise-Implementierung 229
 Systemkonvertierung 230
 technische Sicht 184
 Überblick 169
 zweistufige Migration 186
 SAP S/4HANA Cloud 98, 173, 229
 SAP S/4HANA Enterprise Management 170, 172
 SAP S/4HANA Finance 26, 171, 174, 184, 230
 SAP S/4HANA Materials Management and Operations 179
 Fertigung 181
 Logistikkette 181
 SAP SCM 65, 95
 SAP Solution Manager 202, 236, 565
 SLT-Integration 426
 System Landscape Setup Guide 565
 SAP Solution Manager
 Diagnostics (SMD) 569
 SAP Supplier Relationship Management (SRM) 95
 SAP Supply Chain Management 65, 95
 SAP Test Data Migration Server 229
 SAP Visual Composer 268
 SAP Web IDE 111, 131, 139, 140, 316, 556
 Adresse abfragen 142
 Anmeldung 143
 Installation 139
 Konsole 150

SAP Web IDE (Forts.)
 Projekt aus Vorlage anlegen 145
 Projekteinrichtung 144
 src-Ordner 148
 SAP_BW_HOUSEKEEPING-
 Aufgabenliste 245
 SAPCAR 61
 SAP-Hauptbuch (G/L) 175
 SAP-ID-Service 157
 SAP-Material-Ledger 175, 177
 Ist-Kalkulation 178
 parallele Bewertungen 178
 Preisabweichung 178
 SAPS 213
 SAPUI5 119
 SAS-Speicher 85
 Savepoint 25, 26
 Schatten-Repository 220
 Schattensystem 250
 Schedule by Interval 427
 Schedule by Time 427
 Schema 323
 Schneeflockenmodellierungstechnik 208
 Schreibmodul 423
 SCM 65, 95
 SDA → SAP HANA Smart Data Access
 SDI → SAP HANA Smart Data Integration
 SDK 167
 SDS 99, 388
 Security Assertion Markup Language 126, 525
 Sentiment-Analyse 103
 Serial-Attached SCSI → SAS-Speicher
 Server 85
 Server, aktiver 538
 Service Level Agreement 507
 SHC_CONT 2011 426
 Sicherheit 523
 Administrationsrolle zuordnen 528
 Authentifizierung 525
 Benutzer hinzufügen oder deaktivieren 525
 Benutzer und Berechtigungen 88
 Berechtigung 525
 Kennwortrichtlinie 531
 Rolle hinzufügen und anpassen 527
 Systemberechtigung 523
 Sicherung 535, 536
 Silicon Graphics 202, 239
 Single Sign-on 157
 Sizing 210
 ABAP-Report 225

Sizing (Forts.)
Komprimierung 210
 SLES 85, 194
 SLT → SAP Landscape Transformation
 Software Development Kit 167
 Software Logistics Toolset 200
 Software Provisioning Manager 217
 Software Update Manager 200, 219,
 230, 249, 304, 520
 spaltenbasierte Speicherung 32, 34,
 36, 39
 Spark 480
 Speicher, physischer 505, 513
 Speichertabelle, erweiterte 555
 Speicherverbrauch 513
 SQL 43, 66, 191
 SQL-Konsole 318, 324
 SQLScript 191, 318
 SSL-Verbindung, sichere 143
 SSL-Verschlüsselung 490
 Standard-MDX für Excel 2010 296
 Star-Join 367, 400
 Statistik, Abfrageperformance 507
 Statistikserver 539
 Sternschema 208, 260
 Stored Procedure 556
 SUM → Software Update Manager
 SUSE Linux Enterprise Server 85, 194
 Sybase → SAP Replication Server
 System Landscape Directory 494
 System-Monitoring 504

T

Tabelle
denormalisierte 208
erweiterte 552
spaltenbasierte 288
virtuelle 478
 Tabellendaten exportieren und
 importieren 511
 Tabellenmetadaten 325
 Teradata 410
 Tivoli Storage Manager 538
 Trace-Datei 397
 Transaktion
DBACOCKPIT 567
IUUC_SYNC_MON 442, 443

Transaktion (Forts.)
LTR 427, 436, 440
LTRC 429, 430
LTRS 431
RSDDSTAT 243
RSZDELETE 244
STC01 245
 Transformation 69, 81, 434
 Trendanalyse 27
 triggerbasierte Daten-
 replikation 192, 422

U

Unicode 246
 Unified Installer 519
 Union-Knoten 348
 Unisys 202
 Universal Journal 175, 176
 Universe Design Tool 285, 288
 Universum 288
 UNIX 410

V

VBScript 120
 VDM 309
 View, personalisierter 272
 Virtualisierung 539
Anbieter 541
Vorteile 541
 virtuelle Tabelle 54
 virtuelles Datenmodell 309
 VMware Hypervisor 539

W

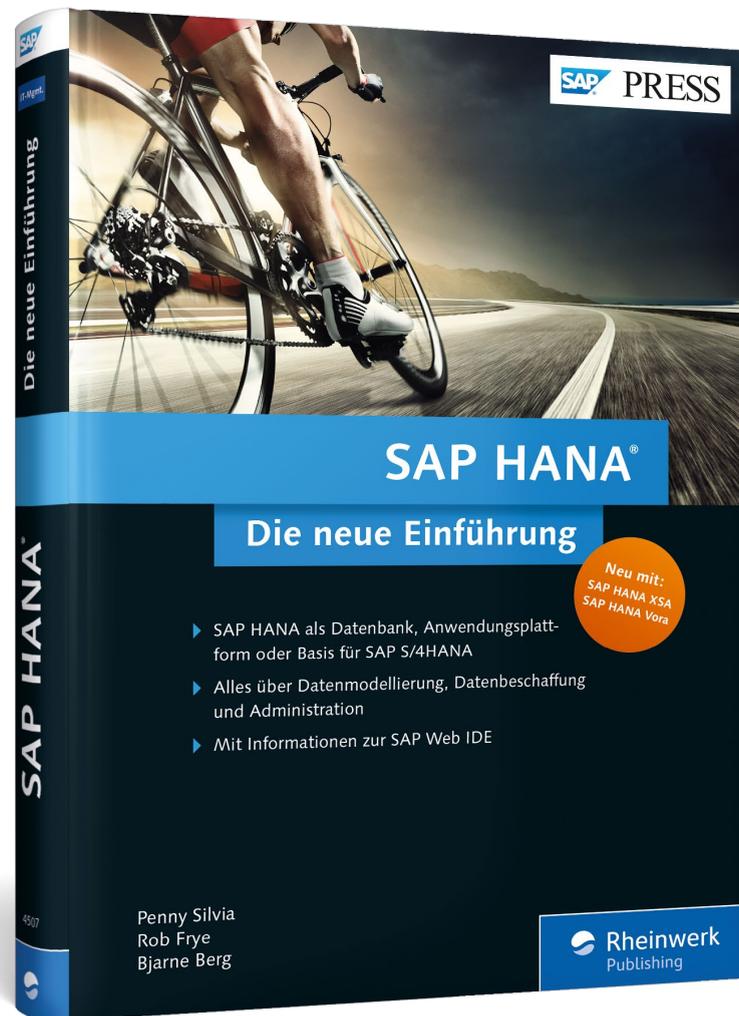
Währungsumrechnung 379
 »warme« Daten 552
 Windows 410
 Windows-Server-Firewall 217

X

x3850 206
 XMLA 111
 XS Advanced Command Line Client 140,
 141, 143
 XS Application Router 125
 XS CLI → XS Advanced Command Line
 Client
 XS JavaScript (XSJS) 110, 149
 XS Engine → SAP HANA XS Classic

Z

Zeichenfolgenfunktion 352
 zeilenbasierte Speicherung 32, 34, 39,
 191
 Zielschema 308



Penny Silvia, Rob Frye, Bjarne Berg

SAP HANA – Die neue Einführung

583 Seiten, gebunden, 3. Auflage 2017

69,90 Euro, ISBN 978-3-8362-4507-4

 www.sap-press.de/4359

Penny Silvia Als Mitglied des IBM Global Leadership Team for SAP Data and Analytics verfügt Penny Silvia über weitreichende Erfahrungen bei der Implementierung von fortschrittlichen Systemen und In-Memory-Analyselösungen für SAP-Kunden.

Rob Frye ist ein erfahrener SAP-HANA- und SAP-BusinessObjects-Entwickler, der Inhalte, Trainingsmaterial und Prozesse für SAP HANA gestaltet und in diesem Rahmen z. B. InfoObjects für SAP BW auf SAP HANA für Fortune-500-Firmen modelliert.

Dr. Bjarne Berg verfügt über umfangreiche Erfahrungen bei der Implementierung von SAP NetWeaver Business Warehouse Accelerator und SAP HANA in Europa und in den USA. Er hat u. a. BW-Projekte für sehr große Organisationen weltweit geleitet und war an über 50 SAP-Projekten zur Leistungsoptimierung maßgeblich beteiligt.

Wir hoffen sehr, dass Ihnen diese Leseprobe gefallen hat. Sie dürfen sie gerne empfehlen und weitergeben, allerdings nur vollständig mit allen Seiten. Bitte beachten Sie, dass der Funktionsumfang dieser Leseprobe sowie ihre Darstellung von der E-Book-Fassung des vorgestellten Buches abweichen können. Diese Leseprobe ist in all ihren Teilen urheberrechtlich geschützt. Alle Nutzungs- und Verwertungsrechte liegen bei den Autoren und beim Verlag.

Teilen Sie Ihre Leseerfahrung mit uns!

