

Mobile Enablement mit dem Oracle Service Bus

Johannes Mangold
Trivadis AG
Basel

Schlüsselworte

Oracle Service Bus 11g, Mobile Applications, RESTful Services, HTML5 WebSockets, JMS Messaging, Mobile Enablement, Protokoll Transformation, Integration

Einleitung

Der Zugriff auf Geschäftsanwendungen und Prozesse über mobile Endgeräte bringt Unternehmen reizvolle Vorteile, stellt sie jedoch auch vor eine Reihe von Herausforderungen. Eine dieser Herausforderungen ist die "Mobile Integration" respektive das "Mobile Enablement" vorhandener Geschäftsanwendungen. Wie kann eine Mobile App effizient und sicher mit meinen Geschäftsanwendungen und anderen Quellen in und außerhalb des Unternehmens kommunizieren?

Der Oracle Service Bus 11g ist ein ausgereiftes und mächtiges Werkzeug zur Integration im Unternehmensumfeld. Er bringt bereits viele Technologien und Ansätze mit, die für das „Mobile Enablement“ vorhandener Geschäftsanwendungen und Services benötigt werden. Der Vortrag soll aufzeigen, mit welchen Technologien und Patterns der Oracle Service Bus als „Mobile Enabler“ effektiv eingesetzt werden kann.

Problemstellung

Bei der Anbindung von Datenquellen und Unternehmensanwendungen an mobile Apps sind Unternehmen zunächst wie auch bei klassischen Integrationsaufgaben mit vielen Lösungsmöglichkeiten konfrontiert. Die perfekt passende Vorgehensweise zu finden ist hierbei nicht trivial. Mobile Clients bringen über die ohnehin schon vorhandene Komplexität solcher Integrationslösungen hinaus noch weitere spezifische Herausforderungen mit sich.

Vielfältige Plattformen

Häufig müssen unterschiedliche Betriebssysteme bei nativen und hybriden mobilen Apps bzw. verschiedene Internetbrowser bei mobilen Web Apps berücksichtigt werden. Um Insellösungen und Schnittstellenchaos zu vermeiden, sollten für den Informationsaustausch zwischen mobilen Clients und Backendsystemen Protokolle und APIs verwendet werden, die alle Plattformen gleichwohl beherrschen können. Ist das möglich?

Unzuverlässige Netzumgebungen

Mobile Clients bewegen sich üblicherweise in Netzen mit unterschiedlichster Bandbreite und unzuverlässiger Netzverbindung – in- und außerhalb der Unternehmensgrenzen. Wie könnte für solche Bedingungen eine performante und fehlertolerante Integrationsarchitektur aussehen?

Skalierung

Bei der Anbindung von mobilen Endgeräten muss meist mit einem deutlichen Zuwachs der Anzahl gleichzeitiger Verbindungen und der Gesamtlast für die Quellsysteme gerechnet werden. Wie können die Backendsysteme damit umgehen?

Security

Mobile Clients unterliegen nicht zwangsweise der Kontrolle des Unternehmens. Nach deren Integration werden diese jedoch teilweise mit unbekannter Integrität aus unbekanntem Netzumgebungen mit schützenswerten Backendsystemen kommunizieren. Sind die Backendsysteme darauf vorbereitet?



Abb. 1: Ungeordnete Punkt-zu-Punkt Integration mobiler Anwendungen (Quelle: Convertigo)

Der Vortrag soll aufzeigen, wie diese Herausforderungen bei der Anbindung mobiler Anwendungen an Unternehmen-Backendsysteme mit dem Oracle Service Bus als Integrationsinfrastruktur mit geeigneter Architektur und passenden Patterns gemeistert werden können.

Ideale Bedingungen für die Anbindung mobiler Anwendungen

Für den synchronen Zugriff mobiler Anwendungen auf Ressourcen ist REST ein sehr gut geeigneter Architekturstil. RESTful Services haben sich mittlerweile zu einem de facto Standard für die Bereitstellung von Ressourcen auf Basis von Web Standards entwickelt. REST besticht hier im Vergleich beispielsweise zu SOAP durch den schlanken Overhead. Die Kommunikation basiert auf dem HTTP Standard mit den bekannten HTTP Request Methoden, Response Status Codes und der URI Adressierung. Daten werden dabei in Form von Repräsentationen (z.B. als JSON oder XML Dokument) ausgetauscht. So kann eine lose Kopplung und damit eine breite Unterstützung erzielt werden, da Clients und Backendsystem (bzw. die Integrationsinfrastruktur) lediglich mit HTTP und JSON oder XML umgehen können müssen. Das ist bei allen relevanten Plattformen client- und serverseitig der Fall.

Eine noch junge und sehr interessante Möglichkeit zur asynchronen und nachrichtenbasierten Anbindung von mobilen Clients an Backendsysteme bringen HTML5 WebSockets. Die HTML5 WebSockets Spezifikation definiert erstmals einen Ausbruch aus dem gewöhnlichen Request-Response-Paradigma des Webs. Das neue Web-Protokoll ermöglicht eine Full-Duplex Socket-Kommunikationsverbindung zwischen (Browser) Clients und Backendsystemen, basierend auf etablierten Web Standards – TCP Sockets für das Web. Nachrichten können in beide Richtungen nahezu Latenzfrei und mit minimalen Overhead über eine einzige Verbindung ausgetauscht werden. Ist eine WebSocket Connection aufgebaut, können beide Kommunikationspartner mit beliebigen Protokollen kommunizieren. Ein ideales asynchrones Protokoll für die Kommunikation mit Unternehmen-Backendsystemen ist hierbei JMS. JMS ermöglicht einen losen gekoppelten und dennoch verlässlichen, asynchronen Nachrichtenaustausch zwischen Systemkomponenten, also beispielsweise einem mobilen Client und einem Message Broker oder einem Enterprise Service Bus. Dieser Nachrichtenaustausch kann dabei 1:1 über Queues oder auch nach dem Publish-Subscribe

Pattern über Topics bei Bedarf auch transaktional abgebildet werden. Für diesen Integrations-Stack ist ebenfalls schon eine sehr breite client- und serverseitige Unterstützung vorzufinden.

Die Architektur einer mobilen Anwendung sieht idealerweise vor, dass Zustände der Applikation clientseitig verwaltet werden und Ressourcen und Daten über zustandslose Backendservices konsumiert werden. Wann immer das Auslösen von Aktionen auf Backendsystemen idempotent gestaltet werden kann, sollte das so realisiert werden. Asynchroner Nachrichtenaustausch wird sich bei großer Anzahl konkurrierender Zugriffe positiv auf Performance und Skalierbarkeit der mobilen Anwendung auswirken und kann für die speziellen Anforderungen an User-Experience bei mobilen Apps hilfreich sein.

Der Oracle Service Bus 11g als „mobile facade“ für mobile Anwendungen

Betrachtet man die oben genannten Idealbedingungen für die Integration mobiler Anwendungen vor dem Hintergrund der Herausforderungen in dem Umfeld und projiziert das Ganze auf die Konzepte und Funktionalitäten des Oracle Service Bus, kann man feststellen, dass der OSB auch für solche Integrationsaufgaben eine sehr hilfreiche und professionelle Integrationsinfrastruktur bereitstellt.

Der Oracle Service Bus 11g dient in erster Linie zur Anbindung, Mediation, Überwachung und Verwaltung der Interaktionen zwischen heterogenen Services und Anwendungen oder auch anderen Enterprise Service Bus Instanzen und bildet so eine zentrale Komponente einer SOA Infrastruktur. Der Oracle Service Bus setzt dabei auf etablierte Standards und eignet sich für geschäftskritische SOA Infrastrukturen, in denen hohe Stabilität, Performance und Skalierbarkeit wichtige Anforderungen sind.

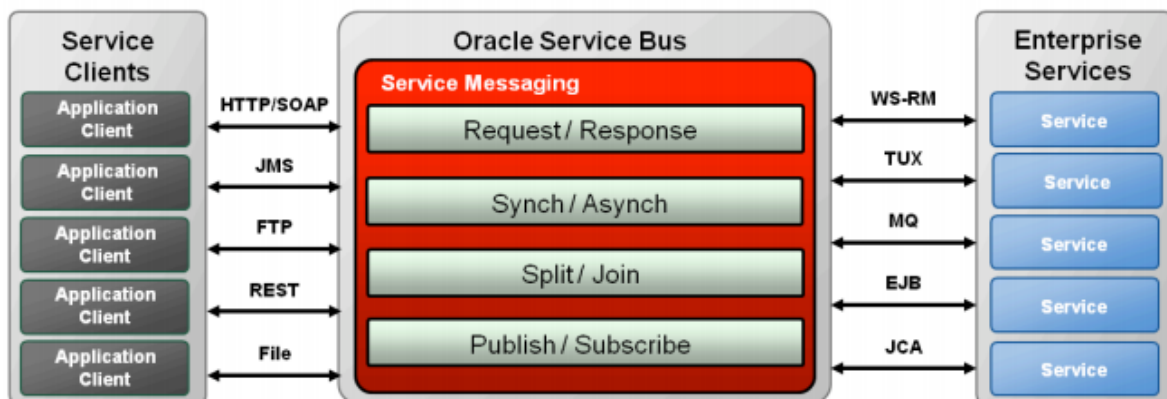


Abb. 2: Service Mediation mit dem OSB (Quelle: Oracle)

Über vielfältige Adapter und Transportprotokolle können Unternehmensanwendungen und Services über alle denkbaren Kommunikationspatterns deklarativ an den Bus angebunden und virtualisiert werden.

Im Folgenden wird betrachtet, wie mobile Anwendungen unter Berücksichtigung der oben genannten Idealbedingungen an den OSB angebunden werden können um mit den weiteren angebotenen Unternehmensanwendungen- und Services interagieren zu können. Abschließend wird geprüft, inwiefern der Oracle Service Bus Unterstützung bei der Lösung der eingangs genannten Herausforderungen bei der Anbindung mobiler Anwendungen liefert.

Anbindung mit REST Stil

Das Bereitstellen eines Interface im REST Stil ist mit dem OSB sehr einfach. Für die Anbindung im REST Stil kommt der HTTP Transport des Oracle Service Bus zum Einsatz. Hier können in einem HTTP Methoden Branch die jeweiligen Request-Methoden GET, POST, PUT, DELETE, HEAD und OPTIONS in verschiedene Pipeline Pairs im Flow aufgeteilt werden. Hier ein Beispiel:

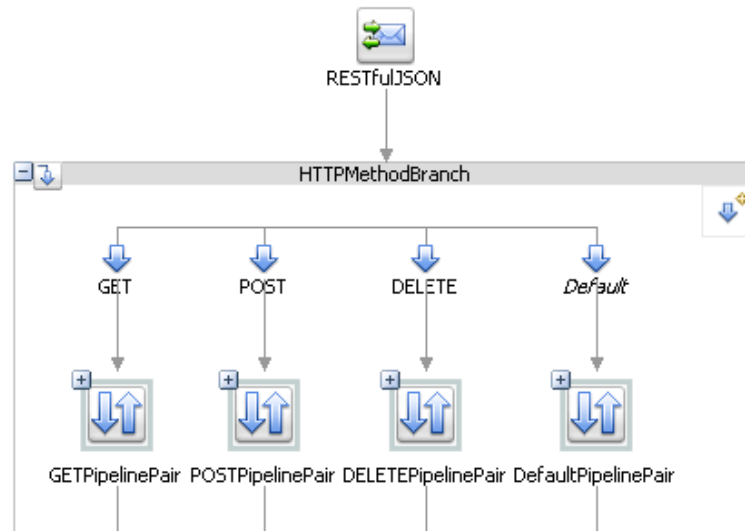


Abb. 3: RESTful Proxy Service Flow

Der HTTP Methoden Branch wertet mit folgendem XPath Ausdruck den Namen der HTTP Request Methode aus: `$inbound/ctx:transport/ctx:request/http:http-method/text ()`

Jeder Branch vergleicht dann gegen die jeweilige Request-Methode und leitet den Flow in separate Pipeline Pairs. Aufrufparameter können mit Assigns und entsprechenden XPath Ausdrücken aus der relative-URI oder query-parameters extrahieren.

Sehr interessant ist in diesem Zusammenhang die Möglichkeit der Protokoll- oder Transporttransformation im Oracle Service Bus, so dass nun im weiteren Flow beispielsweise zu einem asynchronen Service via JMS geroutet werden kann. Auch eine „RESTifizierung“ eines SOAP Services oder eines EJBs sind so einfach möglich. Das alles transparent für den aufrufenden Service-Konsumenten.

Für den Umgang mit JSON Dokumenten, deren Transformation oder Validierung muss aktuell noch auf Hilfsmittel zurückgegriffen werden. Bei nicht zu komplexen JSON Dokumenten kann hier eine XSLTransformation eingesetzt werden. Alternativ kann auch ein Callout auf einen effizienten JSON-Prozessor in den Flow integriert werden.

Alle relevanten mobilen Browser und Plattformen können mit REST und JSON oder XML umgehen.

Nachrichtenbasierte Anbindung über HTML5 WebSockets

Für die nachrichtenbasierte Anbindung mobiler Anwendungen über WebSockets bietet der OSB keine direkte Unterstützung. Aus diesem Grund muss hierfür noch ein Gateway vorgeschaltet werden, um Nachrichten von den mobilen Clients über WebSocket Verbindungen entgegenzunehmen und an den OSB weiterzuleiten. Ein optimal geeignetes Produkt hierfür ist der Kaazing WebSocket Gateway JMS Edition. Mit wenigen Konfigurationsschritten lässt sich dieser WebSocket Gateway so einrichten, dass er auf der einen Seite WebSocket Verbindungen von den mobilen Clients annimmt und sich auf der anderen Seite über JMS Transport mit dem OSB verbindet, wobei der OSB dann mehr oder weniger die Rolle des Message Brokers einnimmt.

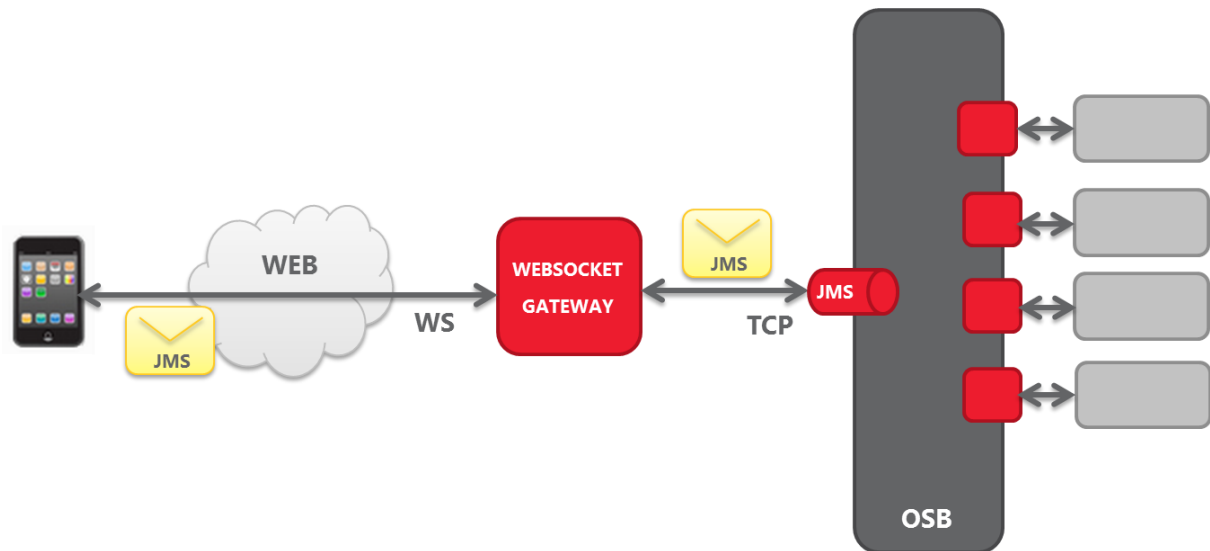


Abb. 4: Service Mediation mit dem OSB (Quelle: Oracle)

Hierbei eröffnet ebenso die mögliche Protokoll- oder Transporttransformation im OSB unter Umständen enorme Möglichkeiten bei der Interaktion mit verschiedenen Backendservices, die sonst nur sehr aufwändig realisierbar wären.

Die Browser-Unterstützung und Verfügbarkeit von WebSocket Client APIs ist bereits sehr breit gefächert. Für (Mobil) Browser ohne direkte WebSocket Unterstützung gibt es Fallbackoptionen die transparent greifen.

Optimal passt für diesen Einsatzzweck die von Kaazing erstellte Implementierung des JMS Protokolls für JavaScript, so dass mobile Web Clients End-To-End und Full-Duplex mit dem OSB JMS Nachrichten austauschen können – bei Bedarf auch transaktional.

Nützliche übergreifende OSB Funktionalitäten

Die folgenden OSB Features lassen sich bei beiden Integrationsarten per Konfiguration einbinden und bieten für die Anbindung mobiler Clients – unabhängig vom Integrationsstil - übergreifend wertvollen Zusatznutzen:

- Nachrichtenvalidierung
- Routing
- Monitoring
- Throttling
- Service Virtualisierung
- Policy basierte Transport und Message Security

Live-Demo Mobile Web Anwendung

In einer Live-Demonstration wird ein kleiner Showcase einer mobilen Web Anwendung gezeigt, die über die beiden Integrationsszenarien „RESTful“ und „Realtime asynchron“ an den OSB angebunden wurde.

Fazit

Abschließend werden im Rahmen des Vortrags die Funktionalitäten und Konzepte des Oracle Service Bus 11g auf die Herausforderungen der Integration mobiler Clients an Backendsysteme aus der Problemstellung zusammenfassend reflektiert und damit einhergehend die Eignung und Vorteile des OSB als „Mobile Enabler“ verifiziert.

Kontaktadresse:

Johannes Mangold

Senior Consultant
Application Development

Trivadis AG
Elisabethenanlage 9
CH-4051 Basel

Telefon: +41-61-279 97 55
Fax: +41-61-279 97 56
E-Mail: johannes.mangold@trivadis.com
Internet: www.trivadis.com