

Oracle Business Rules 11g unleashed

Sören Halter, Steffen Miller
ORACLE Deutschland B.V. & Co. KG
Geschäftsstelle Frankfurt

Schlüsselworte:

Business Rules, Geschäftsregeln, Rule Engine, BRMS, Geschäftslogik, Inferenz

Einleitung

Mit dem 11g Release von Oracle Business Rules (OBR) wurde ein großer Schritt nach vorne gemacht. OBR ist ein Muss in allen Projekten mit Geschäftslogik, und welches Projekt hat keine. Dabei ist es egal, ob die Logik als Excel Datei, oder nur in Prosa vorliegt. Egal ob die Integration per Java, in einer SCA oder als Web Service realisiert wird. Egal ob die Entwickler Regeln umsetzen, oder Fachnutzer mitarbeiten werden. Egal ob die Regeln nur einmal eingegeben werden, oder stündlich verändert werden. Egal ob 5000 Nachrichten pro Sekunde verarbeitet werden müssen, oder nur wenige Anfragen im Rahmen eines Geschäftsprozesses. Regeln sind überall. Insbesondere die während eines PoCs gesammelten Erkenntnisse, geben einen tiefen Einblick in die Technik und Möglichkeiten von Oracle Business Rules.

Geschäftsregeln, eine Annäherung

Im Zentrum des Business Rules Ansatzes steht der Begriff „Business Rules“, was übersetzt „Geschäftsregel“ bedeutet. Man versteht darunter nicht notwendigerweise eine geschäftliche Angelegenheit, sondern Business-Rules können auch ganz elementare technische Sachverhalte beschreiben. Im engeren Sinne (geschäftlich) sind dies Regeln, Direktiven oder Richtlinien, die das Geschäftsverhalten beeinflussen und steuern. Beispiele von Geschäftsregeln sind:

- Ein Kunde ist ein guter Kunde, wenn er in den letzten 6 Monaten einen Umsatz von mehr als 10.000 € hatte.
- Aufträge mit einem Volumen von mehr als 50.000 € müssen vom Vertriebsleiter genehmigt werden.
- Bestellt ein Kunde mehr als 10 Artikel der gleichen Sorte erhält er 10% Rabatt.

Vielfach versteht man unter Business-Rules aber auch etwas viel einfacheres, nämlich Vorgaben, die beschreiben, wie sich ein Computerprogramm oder ein Geschäftsprozess, der durch ein Computerprogramm gesteuert wird, verhalten soll. Beispielsweise beschreibt eine Plausibilitätsregel, meist in Form einer Formel, wann eingegebene Daten gültig sind. Diese Daten werden nur dann zugelassen, wenn alle zutreffenden Plausibilitätsregeln erfüllt sind. Ebenso gibt es Berechnungs-, Sichtbarkeits-, Berechtigungs- und viele weitere Arten von Regeln.

Geschäftsregeln und Geschäftsprozesse in einer SOA

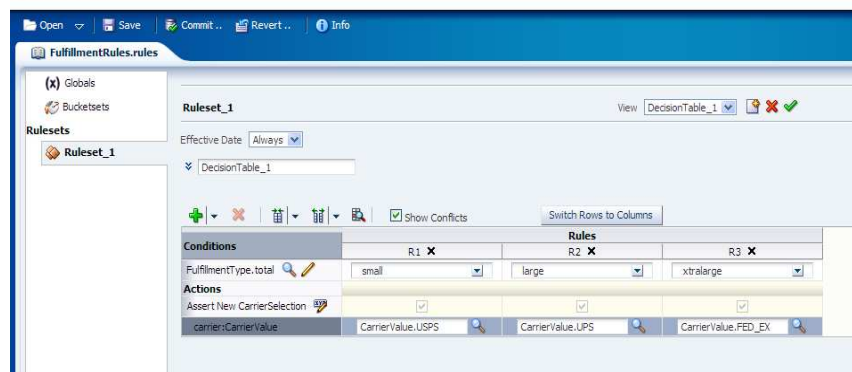
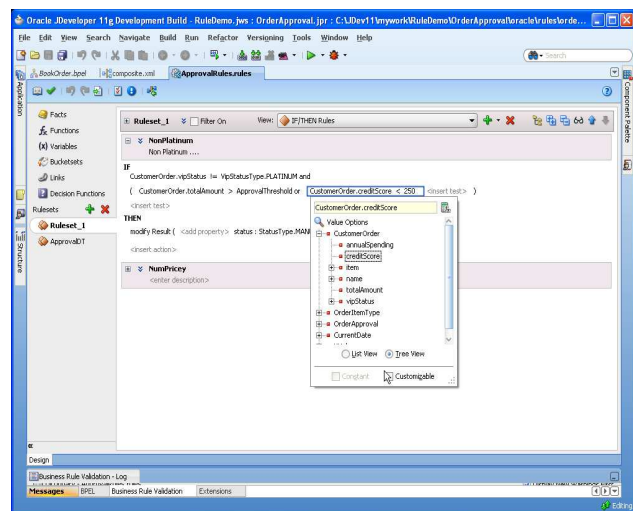
Eine Service Orientierte Architektur regelt das Zusammenspiel von Prozessen und Regeln neu. Bisher galt, dass Regeln und Prozesse voneinander zu trennen und zu managen sind, aber dass Prozessmaschinen für BPM und Regelmaschinen für BRM integriert sind. In einer SOA geht man

einen Schritt weiter. Statt einer Integration von BPM und BRM Technologien fasst man im Sinne einer Service-Orientierung die Regeln als Services auf, die von der Prozessmaschine orchestriert werden. So erhält man Rule-Services als eine Kategorie von Services. Ein Rule-Service kann als Kapselung komplexer Regeln verstanden werden. Mit anderen Worten, ein Rule-Service kann auch einen anderen Rule-Service als Sub-Service aufrufen. Das verhält sich analog dem Subprozess-Prinzip im BPM. Damit wird eine Regelmaschine zum Bestandteil einer SOA Infrastruktur. So haben die Regeln im Rahmen einer SOA ihre eigene Administration im Repository, in der Registrierung und in der Governance Infrastruktur.

Geschäftsregeln, mit Oracle Business Rules

Oracle Business Rules ist Bestandteil der Oracle SOA Suite 11g, kann aber auch in reinen Java oder JEE Anwendungen verwendet werden. Diejenigen die bereits die SOA Suite 10gR3 kennen, werden feststellen, dass im JDeveloper ein neuer, komfortabler Editor für die Erstellung und Bearbeitung von Regeln bereitgestellt wird, der viel intuitiver und einfacher zu benutzen ist, als der alte Browser-basierte Editor. Darüber hinaus führt die SOA Suite 11gR1 Entscheidungstabellen ein, die die Definition von Regeln in Tabellenform ermöglicht. Während der JDeveloper nach wie vor ein eher Entwickler-zentrisches Werkzeug ist, kommen jetzt auch fachlich ausgerichtete Anwender besser damit zurecht. Zusätzlich wurde mit dem SOA Composer auch ein, für Fachanwender benutzbares, webbasiertes Werkzeug zur Ansicht und Manipulation der Regeln eingeführt. Natürlich unter Berücksichtigung von Benutzerlogin und Zugriffsrechten.

ORACLE SOA Composer



Komponenten der Oracle Business Rule Engine

Folgende Komponenten bilden die Grundlagen für das Erstellen von Regeln:

- Facts: Fakten sind Objekte, die in Regeln verwendet, bzw. auf die die Regeln angewendet werden. Es können Java Facts, XML Facts, RL Facts, und ADF Facts verwendet werden.

- Rules: Eine Regel besteht aus einem Wenn-Teil, der einen oder mehrere Tests mit einem oder mehreren zu prüfenden Fakt(en) enthält und einem Aktionsteil (Dann Teil), der die Aktionen enthält, die beim Zutreffen des Wenn-Teils ausgeführt werden soll.
- Rule Set: Ist eine Zusammenfassung von einer oder mehreren Regeln zu einem Set, die zusammen betrachtet werden.
- Dictionary: Ist der Container für alle Komponenten der Business Rules. Dazu gehören die Facts, Rules Sets, die Regeln, Funktionen und einiges mehr.

Die Oracle Business Rule Engine

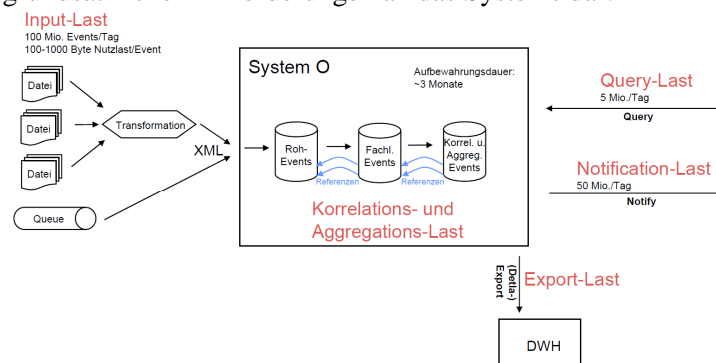
Die Oracle Business Rules Engine basiert auf dem Rete-Algorithmus. Eine Rete Engine besteht im Wesentlichen aus den folgenden Komponenten:

- Regelbasis: Enthält die Regeln als Wenn/Dann Regeln und/oder Entscheidungstabellen, wobei die Entscheidungstabellen intern auch wieder Wenn/Dann-Regeln sind.
- Working Memory: Enthält die Fakten, die in die Rule Engine übergeben wurden, bzw. die während der „Regelausführung“ veränderten oder neu erzeugten Fakten.
- Inference Engine: Die „Rules Engine“, welche die Regeln „abarbeitet“. Ermittelt mit Pattern-Matching welche Regeln durch die Fakten erfüllt sind, und führt die zugehörigen Aktionen aus.

Oracle Business Rules ist ein data-driven, forward-chaining System. Die Fakten bestimmen, welche Regeln „feuern“ können. Wenn eine Regel feuert, die einer Reihe von Fakten entspricht, kann die Regel als Ergebnis auch neue Fakten hinzufügen. Diese Fakten werden wieder gegen die Regeln ausgeführt. Dieser Vorgang wiederholt sich, bis ein Ergebnis erreicht ist oder der Zyklus beendet oder zurückgesetzt wird. So können in einem forward-chaining System, Fakten Regeln feuern lassen, deren verbundene Aktionen wiederum Fakten erzeugen, die ihrerseits weitere Regeln feuern lassen. Dieser Prozess wird auch Inferenz Prozess genannt.

Oracle Business Rules im Einsatz

Im Frühjahr 2010 sollte durch die Autoren im Rahmen eines PoCs ein System für einen großen Kunden realisiert werden, welches aus einer extrem großen Anzahl von Einzelereignissen anhand eines nicht minder aufwendigen Regelwerks komplexe, zusammengesetzte Ereignisse erzeugen sollte. Nachdem für dieses Projekt zunächst Oracle Complex Event Processing (Oracle CEP) das geeignete Produkt zu sein schien, stellte sich bei näherer Betrachtung der Anforderungen heraus, dass Oracle Business Rules in Kombination mit der Oracle Datenbank 11g die bessere Wahl war. Die folgende Abbildung stellt die grundsätzlichen Anforderungen an das Systems dar:



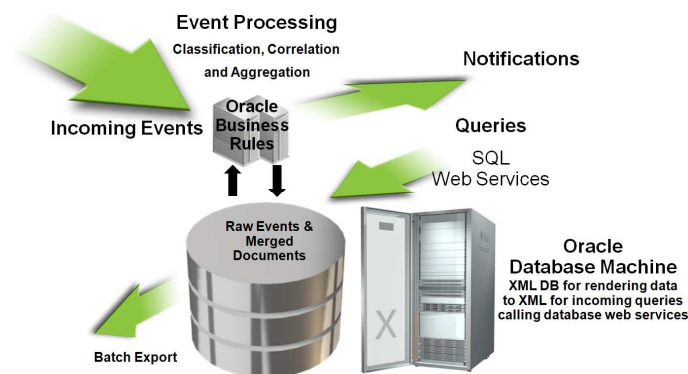
An jedem normalen Werktag werden im Mittel 100 Millionen geschäftsrelevante Ereignisse von anderen Systemen erzeugt. Ein Ereignis ist hierbei zwischen 100 und 1000 Byte groß. Für den eigentlichen Testlauf im PoC wurden aber 170 Millionen Ereignisse pro Tag betrachtet, um auch die,

in der Realität vorkommenden, Spitzenlasten verarbeiten zu können. Die Ereignisse werden dem System per JMS Queues im XML-Format übergeben. Das System wandelt die Rohereignisse in fachliche Ereignisse um, die anschließend anhand der Regeln durch Aggregations- und Korrelationsaktionen zu komplexen fachlichen Ereignissen verdichtet werden. Diese heißen im System MergedDocuments. Sowohl die Rohereignisse, als auch die MergedDocuments und die Verfahrensschritte, die zu diesen MergedDocuments führten, müssen für 3 Monate vorgehalten werden. D.h. die Anforderungen an das System sind nicht nur in Bezug auf die eigentliche Verarbeitung, sondern auch hinsichtlich der Speicherung und des Zugriffs auf bestehende MergedDocuments extrem.

Das System wird im Schnitt 5 Millionen Mal am Tag über eine Webservice Schnittstelle abgefragt. Dies umfasst die Abfrage von MergedDocuments zu einer Korrelations-ID in einem bestimmten Zeitraum und die Abfrage eines spezifischen MergedDocuments inklusive aller zugehörigen Rohereignisse basierend auf dem Primärschlüssel des MergedDocuments.

Ferner sendet das System Benachrichtigungen wiederum über eine JMS Queue an weitere Systeme, um diese über wichtige Vorkommnisse, wie etwa „Neues MergedDocument angelegt“ oder „MergedDocument geschlossen“, zu informieren. Dies geschieht bei etwa jedem zweiten Eingangsereignis.

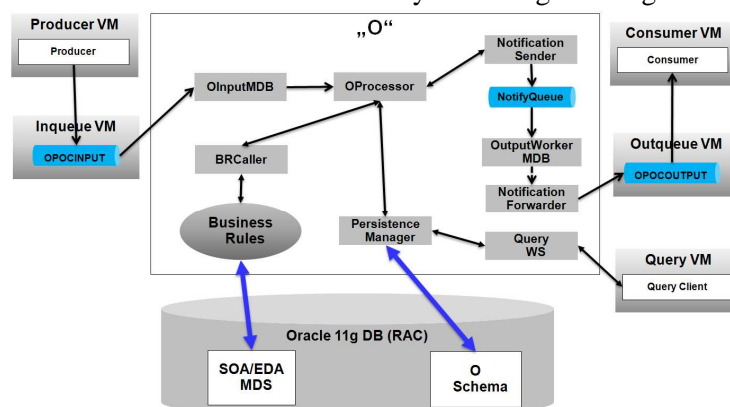
Wie in der nächsten Abbildung dargestellt, basierte die Umsetzung auf Oracle Business Rules, als Bestandteil der Oracle SOA Suite 11g, und der Oracle Exadata V2 Datenbankmaschine.



Für diesen PoC wurden sowohl die Datenbank 11g als auch die SOA Suite 11g auf einer Exadata V2 betrieben.

Der Datenbank/Middleware-Cluster bestand aus insgesamt 4 Servern mit je 2 Quad-Core CPU (Nehalem) und 72 GB RAM. Das Betriebssystem war Oracle Enterprise Linux 5.

Den eigentlichen Testaufbau und die Architektur des Systems zeigt die folgende Abbildung:



Wie zu erkennen ist, wurden für die Lasterzeugung virtualisierte Produzenten und Konsumenten verwendet. Da die Anforderungen an die Performanz sehr hoch waren, wurde vorsichtshalber die, die

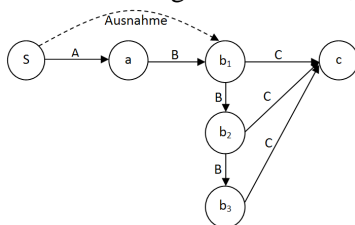
Business Rules Engine umgebende, Verarbeitungslogik mittels leichtgewichtiger JEE Komponenten und POJOs realisiert. Aus heutiger Sicht wäre eine Umsetzung als SCA unter Ausnutzung der in der SOA Suite vorhandenen Möglichkeiten hinsichtlich Persistenz, XML Transformationen etc. aber der deutlich effizientere Weg gewesen.

Die Hauptarbeit, nämlich die Verarbeitung der Ereignisse, übernahm die Business Rules Engine.

Umgesetzte Regelbasis im Überblick

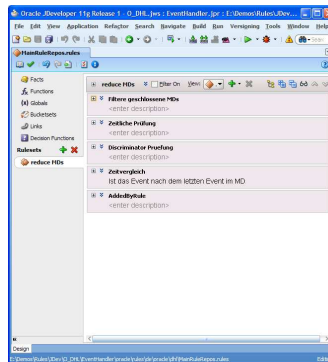
Für ein eingehendes Ereignis wird ein neues MergedDocument erzeugt, wenn nicht nach den Regeln a. – e. ein existierendes MergedDocument gefunden wurde. Wird ein Match gefunden, dann wird mit den weiteren Regeln fortgefahren.

- a. Stimmt die Korrelations-ID zwischen dem Ereignis und dem MergedDocument überein?
- b. Stimmen die vorgegebenen Diskriminatoren
 $discr1 \mid discr2 \mid discr3 \mid \dots$ des Ereignisses
 $discr1 \mid discr2 \mid discr3 \mid \dots$ des in a. gefundenen MergedDocuments
 überein?
- c. Prüfung der zeitlichen Korrelation: Das neue Ereignis darf nicht jünger als n Tage hinsichtlich der letzten Änderung am MergedDocument sein.
- d. Die Auswertung von weiteren, komplexen Regeln, die als Zustandsautomat notiert sind, z. B.

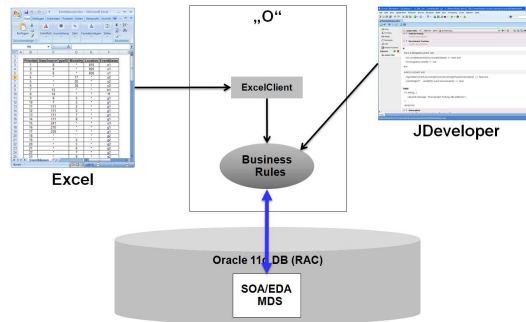


- e. Aufruf einer beliebigen, in Java implementierten Funktion

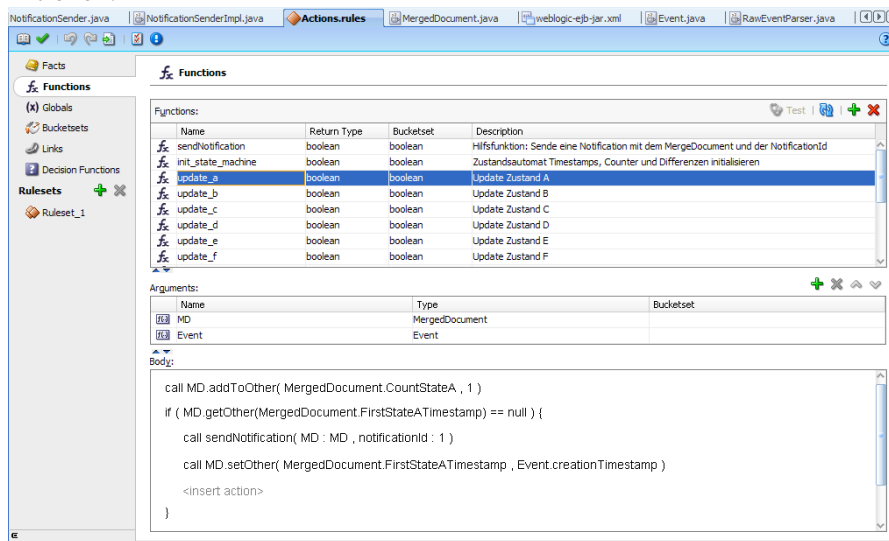
Die nachfolgende Abbildung zeigt einen Teil der zugehörigen Umsetzung dieser Anforderung in Oracle Business Rules:



Der in Regel d. erwähnte Zustandsautomat wurde uns vom Kunden in Form einer Microsoft Excel-Tabelle geliefert. Diese wurde durch einen, im Rahmen des Projekts geschriebenen, Importclient automatisch in Oracle Business Rules überführt. Die so automatisch erzeugten Regeln, konnten anschließend über JDeveloper weiterverarbeitet werden:



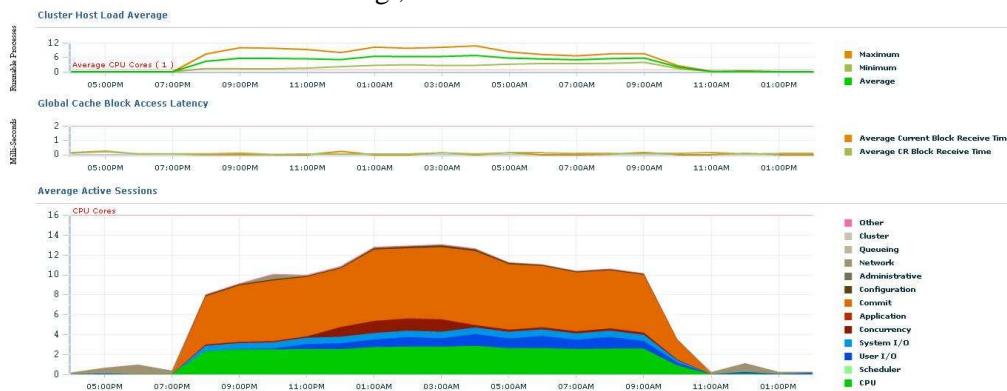
Der folgende Screenshot vermittelt einen Eindruck der, durch die Regeln des Zustandsautomaten ausgelösten, Aktionen:



Wie im Body der dargestellten Aktion update_a zu erkennen ist, wird hier sowohl der Zustand des übergebenen MergedDocuments MD manipuliert (MD.addToOther; MD.setOther) als auch eine entsprechende Nachricht verschickt (call sendNotification).

Ergebnisse der Lasttests

Wie in der nächsten Abbildung zu erkennen ist, waren während des PoC-Lasttests mit 170 Millionen Ereignissen im Datenbank/Middleware-Cluster maximal 13 CPU-Cores ausgelastet. Dabei wurden von der Datenbank 3 CPU-Cores benötigt, von der Middleware 10 CPU-Cores.



In Bezug auf die Verarbeitung der Ereignisse war das System in der Lage, im Peek über 3.700 Ereignisse/s zu verarbeiten. Im Durchschnitt des 16-stündigen Testlaufs wurden ~3.000 Ereignisse/s verarbeitet.

Neben der Verarbeitungsgeschwindigkeit und dem erreichten Durchsatz, zeichnet sich das System insbesondere durch die „sprechende“ Umsetzung der Geschäftsregeln und der Flexibilität bezüglich der Veränderbarkeit dieser Regeln aus.

Kontaktadresse:

Sören Halter Oracle Deutschland B.V. & Co.KG Geschäftsstelle Frankfurt Robert-Bosch Straße 5 D-63303 Dreieich	Steffen Miller Oracle Deutschland B.V. & Co.KG Geschäftsstelle Frankfurt Robert-Bosch Straße 5 D-63303 Dreieich
Telefon: +49 (0) 174 3443752	Telefon: +49 (0) 6103-397 778
E-Mail soeren.halter@oracle.com	E-Mail steffen.miller@oracle.com
Internet: www.oracle.com/de	Internet: www.oracle.com/de