

Optimierung von Website-Werbung mit Oracle Real Time Decisions

**Thomas Kaiser
Daniel Chitralla
Riverland Reply GmbH
München**

Schlüsselworte:

Oracle Real Time Decisions, Self Learning, Website Optimization

Einleitung

In diesem Proof-of-Concept wurde Oracle Real-Time Decisions verwendet, um den Kunden eines Webshops zielgerichteter Werbung präsentieren zu können. Dadurch wurde der Umsatz der begrenzten Werbeflächen vergrößert und gleichzeitig die User-Akzeptanz der Werbung erhöht. Als "Nebenprodukt" fielen anonymisierte Informationen über das Verhalten bestimmter Nutzer-(Gruppen) ab, die für eine bessere Kundenansprache im klassischen Kampagnen-Marketing verwendet werden können. Die Hauptvorteile der Lösung sind der niedrige initiale Aufwand und die einfache Handhabung im Betrieb, da das manuelle Erarbeiten und Erstellen von Regeln in diesem Fall durch die selbstlernende RTD-Engine in weiten Teilen obsolet wurde.

Grundlagen von Oracle Real Time Decisions

Oracle RTD ist eine Decision Engine, die in der Lage ist, nahezu in Echtzeit Entscheidungen auf der Basis sich im Zeitablauf optimierender Korrelationen zu treffen. Das unterscheidet sie zum einen von regelbasierten Systemen, deren Logik vorab aufwändig definiert werden muss. Zum anderen grenzt sich RTD durch sein sofortiges Antwortverhalten von den meisten BI-Systemen ab, die zur Analyse meist Minuten bis Stunden benötigen.

Durch die Verwendung offener Standards (wie z.B. XML, SOAP, Webservices, Java,...) erlaubt die Applikation gute Integrationsmöglichkeiten und einen Einsatz in vielfältigen Bereichen. Als Beispiel seien hier folgende genannt:

- Website Optimization
- Fraud Detection
- Call/SR Routing
- Customer Retention
- Agent-Support am POS
- Cross Sell/Up Sell
- Online Personalization

Da RTD nur sehr schwer manuell „trainiert“ werden kann, wird von Oracle empfohlen, die Evaluation und evtl. anschließende Einführung anhand eines sogenannten „Proof of Solution“ durchzuführen. Darunter ist eine voll funktionsfähige Lösung mit überschaubarem Funktionsumfang zu verstehen, die mit Produktivdaten beliefert wird. Auf dieser Basis können in folgenden Projektphasen sowohl weitere Funktionalitäten in die bestehende Lösung integriert als auch komplett neue Applikationen (Cross-Channel) angebunden werden.

Projektbeschreibung

Der Kunde betreibt einen Webshop, der Werbeflächen für Produkt enthält, die gut zu dem Kunden passen bzw. ihm gefallen könnten. Dieses Matching wird von Oracle RTD anhand verschiedener Attribute wie z.B. der Kaufhistorie, des Warenkorbes, etc. durchgeführt.

Zum besseren Verständnis wird erst die Architektur und anschließend ein Beispiel-Usecase beschrieben.

Architektur

Die Architektur der RTD-Lösung stellt im ersten Schritt einen minimalinvasiven Ansatz dar, der nur die Einbindung einiger Komponenten (primär Skripte und Grafiken) in die bestehende Webseite und einen Export der verfügbaren Angebote erfordert. Dadurch ist eine schnelle und fast release-unabhängige Umsetzung möglich.

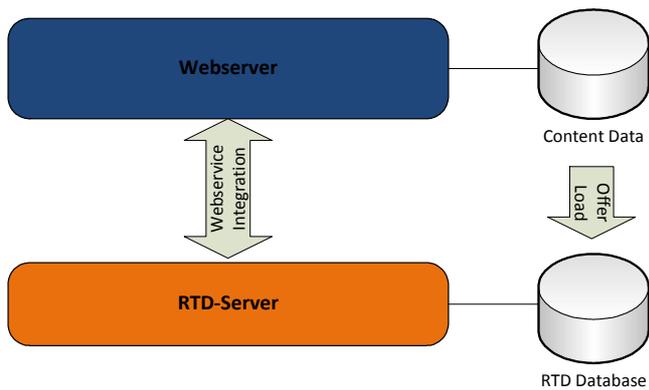


Abb. 1: Die Serverarchitektur

Aufgrund spezieller Kundenwünsche war der Einsatz einer Zwischenschicht notwendig:

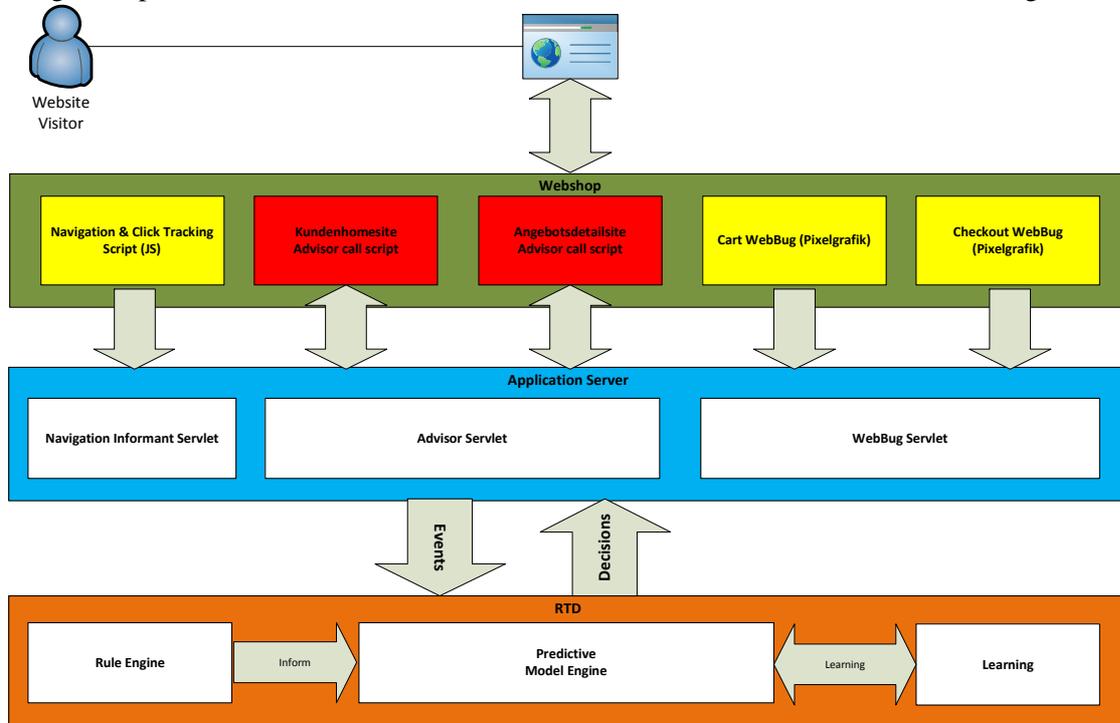


Abb. 2: Die Schichtenarchitektur

Beispiel-Usecase

Die folgende Abbildung zeigt einen möglichen Ablauf der Interaktionen zwischen dem Webshop und der RTD-Applikation, die durch die Aktivitäten des Nutzers auf der Seite ausgelöst werden:

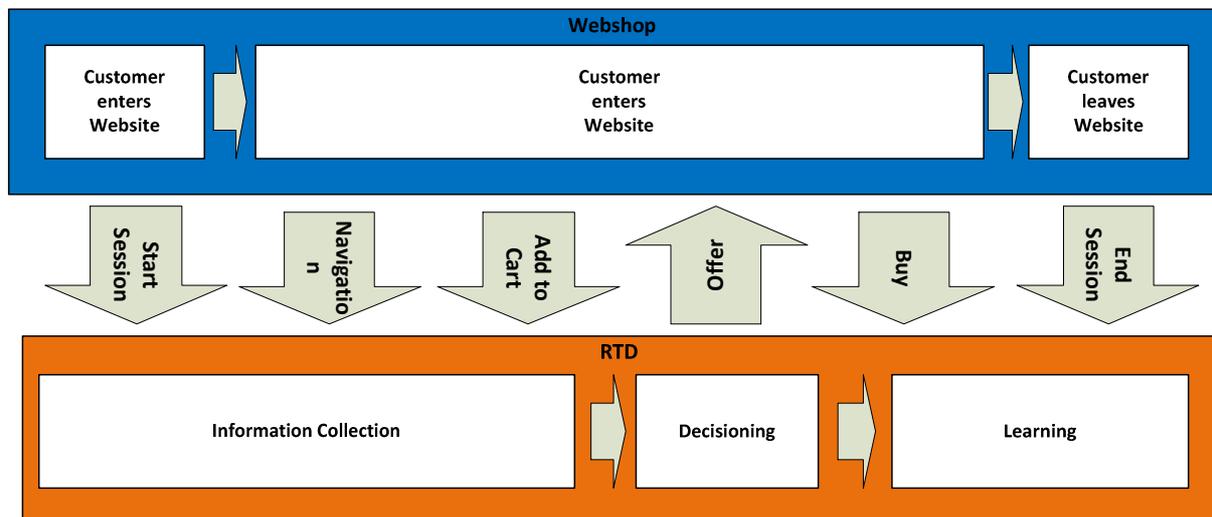


Abb. 3: Beispielhafter Ablauf der Interaktionen

Mit dem Betreten der Seite beginnt die Aufzeichnung von Daten für diese Kunden-Session. Während der Kunde auf der Webseite navigiert und ggfs. Produkte in den Warenkorb legt, werden bevorzugte Kategorien, Preisniveaus, etc. registriert. Sobald eine Seite mit personalisierter Werbung aufgerufen wird, setzt der Webshop im Hintergrund einen Aufruf an RTD ab, welches in Echtzeit die Angebote mit der größten Kaufwahrscheinlichkeit für diesen Kunden ermittelt und ihm präsentiert. Anschließend wird registriert, ob der Kunde das angebotene Produkt kauft und auf Basis dieser Information der interne Lernprozess angestoßen. Das Verlassen der Seite schließt den Prozess und damit die Session ab.

Zusammenfassung und Ausblick

Die Vorteile der gewählten Architektur liegen auf der Hand: Mit relativ geringem Initialaufwand kann eine funktionierende Lösung relativ zeitnah (1-2 Monate) anhand von Produktivdaten bzgl. ihres Mehrwertes evaluiert werden. Zur Bestimmung des Mehrwertes ist eine Kontrollgruppen-Funktionalität bereits in der Basis-Applikation integriert.

Darauf aufbauend kann sowohl der Funktionsumfang der bestehenden Lösung erweitert oder sogar neue Kanäle (z.B. Callcenter, Point of Sale, ...) angebunden werden. Weiterhin ist die Verwendung der Daten für eine Website-Personalisierung zur Laufzeit vorstellbar, die kundenindividuelle Navigations-Menüs erlaubt.

Darüber hinaus lassen sich die statistischen Korrelationen, die für den Entscheidungsprozess verwendet und ständig aktualisiert werden, zum einen für Analysen des Kaufverhaltens oder in einem weitergehenden Schritt auch für z.B. Emailkampagnen einsetzen.

Für den Kunden ergibt sich der Vorteil, speziell auf ihn zugeschnittene Angebote zu erhalten anstatt mit einer Flut unpassender Werbung überschüttet zu werden. Dies kann in der Folge zu mehr Umsatz und auch Loyalität führen, da der Kunde sich eher verstanden fühlt.

Das speziell in Deutschland oft angesprochene Thema Datenschutz muss natürlich in allen Phasen des Projektes beachtet werden, was aber aufgrund der Applikationsarchitektur kein Problem darstellen sollte. Denn zum einen speichert die RTD-Datenbank keine Inter- bzw. Transaktionsdatensätze, sondern arbeitet (schon aus Performancegründen) nur mit aggregierten Summendaten. Zum anderen

ist die eindeutige Identifikation eines Kunden gar nicht notwendig, da der verwendete Fuzzy-Ansatz primär mit ähnlichen Mustern und nicht mit eindeutigen Zuordnungen (z.B. Kundennummer o.ä.) arbeitet.

Kontaktadresse:

Thomas Kaiser
Riverland Reply GmbH
Holbeinstr. 22
D-80469 München

Telefon: +49 (0) 176 70018207
E-Mail t.kaiser@reply.de
Internet: www.reply.de