

Mission Accomplished: Fybers Integration der Oracle ERP Cloud

Johannes Michler
PROMATIS SOFTWARE GMBH
Ettlingen (TechnologieRegion Karlsruhe)

Keywords

Oracle ERP Cloud, CRM System, Informatica Cloud Real Time, Salesforce, Business Intelligence (BI), BI Publisher, SOAP, File Based Data Import (FBDI), Integration Platform as a Service (iPaaS)

Abstract

Im Rahmen der Implementierung der Oracle ERP Cloud bei Fyber ergab sich die Herausforderung, dass mehrere bestehende Systeme integriert werden mussten. Besonders die Integrierung mit einem CRM-System (Salesforce) musste realisiert werden. So sollten Kunden und Lieferanten aus dem CRM-System in die ERP Cloud synchronisiert werden; dabei waren relevante Kundendaten (z. B. eine Auflistung von bezahlten und insbesondere noch unbezahlten Rechnungen) von der ERP Cloud wieder zurück ins CRM-System zu überführen. Darüber hinaus musste ein automatischer Import, die Verarbeitung von Kontoauszügen sowie die Synchronisierung der Benutzerbasis mit einem Microsoft Active Directory Verzeichnis implementiert werden.

Das Manuskript geht auf die von der Oracle ERP Cloud bereitgestellten Schnittstellen-Technologien (SOAP und REST Web Services, CSV-Uploads, ...) ein und zeigt auf dieser Basis, wie eine Integration mit Hilfe der Informatica Cloud Real Time Middleware implementiert werden konnte. Hierbei wird eindrücklich auf Erfahrungen bei der Nutzung der Oracle ERP Cloud Schnittstellen sowie beim Einsatz der Informatica Middleware eingegangen.

Einführung

Als führendes Werbetechnologieunternehmen mit Sitz in Deutschland Büros in San Francisco, New York, Tel Aviv, Beijing u.a. bietet Fyber App-Entwicklern smarte Monetarisierungsstrategien über eine einzige, einheitliche und offene Plattform, die Angebot (Publisher und App-Entwickler) und Nachfrage (Werbende) zusammenbringt. Diese Plattform ermöglicht es tausenden – oft weltweit führenden – App-Entwicklern und -Herausgebern, geschäftskritische Einnahmequellen zu generieren, und erreicht monatlich mehr als eine halben Milliarde aktive Benutzer.

Als Fyber rasant zu einem globalen Unternehmen wuchs, gelangte man an einen Punkt, an dem der Bedarf an schlanken, aber robusten Geschäftsprozessen entstand, die zudem von Online-Tools besser unterstützt werden mussten. Es war an der Zeit, ein neues, integriertes ERP-System für die Unterstützung von Finanz-Geschäftsprozessen zu finden. Ein Auslöser für diese Entscheidung stellte beispielsweise der erhebliche manuelle Aufwand dar, der bei der Rechnungsstellung sowie bei Monats- oder Jahresabschluss einschließlich der Konsolidierung erbracht werden musste. Um dies umzusetzen, suchte Fyber nach einem kompetenten und verlässlichen Partner, der die Prozesse verstand und wusste, wie man mit der erforderlichen Technik umging. Die große Herausforderung hierbei war die Umsetzung einer kompletten Finanz- und Beschaffungslösung für nicht weniger als 14 Mandanten in fünf Ländern binnen sechs Monaten. Zusätzliche Voraussetzungen waren die komplexe Migration der Altdatenbank und die Integration bestehender Plattformen, wie zum Beispiel Salesforce.com, über Webservices. In diesem Vortragsmanuskript wird ausgeführt, wie diese Integration unter Verwendung der Informatica Cloud Real Time (ICRT) Integrationsplattform realisiert wurde.

Informatica Cloud Real Time

Informatica Cloud Real Time ermöglicht seinen Kunden, schnell ereignisgesteuerte Anwendungsintegrationen, API-basierte Services und Serviceorchestrierungen zu konstruieren – sowohl für die Verwendung hinter der Firewall des Kunden als auch in der Cloud oder in einer Kombination der beiden Ansätze. Als Oracle Platinum Partner hätten wir

normalerweise die zugehörigen Oracle Produkte „Oracle SOA Suite“ (lokal oder in der Cloud) oder „Oracle Integration Cloud Services“ (ICS) bevorzugt; in diesem Projekt jedoch war die iPaaS-Lösung von Informatica bereits eingeführt und wurde daher zur Durchführung unserer Integrationsprozesse verwendet. Trotzdem kann die Integration, die im Verlauf dieses Manuskripts aufgeführt wird, auf sehr ähnliche Weise mithilfe der oben erwähnten Oracle Integrationsprodukte durchgeführt werden.

Daten aus der ERP Cloud extrahieren

Die dringendste Integrationsanforderung war bei Fiber die Darstellung von Kundenrechnungen und ihrer zugehörigen Zahlungsstände (Fälligkeitsdatum, bereits bezahlter (Teil-)Betrag) in Salesforce. Das war erforderlich, da die zugehörigen Account Manager keinen (vollständigen) Zugriff auf die Oracle ERP Cloud und damit keinen Zugang zu diesen Informationen hatten – die Informationen sollten weiter direkt im Kontext der CRM-Applikation dargestellt werden. Da die Oracle ERP Cloud aber glücklicherweise Oracle BI (Business Intelligence) und insbesondere den Oracle BI Publisher enthält, können beliebige Daten aus der Oracle ERP Cloud direkt (für fremde Systeme) extrahiert werden. Der erste Schritt, der für die Durchführung solch einer Integration erfolgen muss, besteht in der Anlage eines Oracle BI Publisher Reports, welcher die relevanten Daten aus der unter der Oracle ERP Cloud liegenden Datenbank abrufen. Die Daten werden dabei nicht in Text- oder Tabellen-Form, sondern im XML-Format extrahiert. Die folgende Abbildung zeigt das BI Publisher Datenmodell, in dem Rechnungsdaten und die relevanten Zahlungs- und Kundeninformationen zusammengetragen werden:

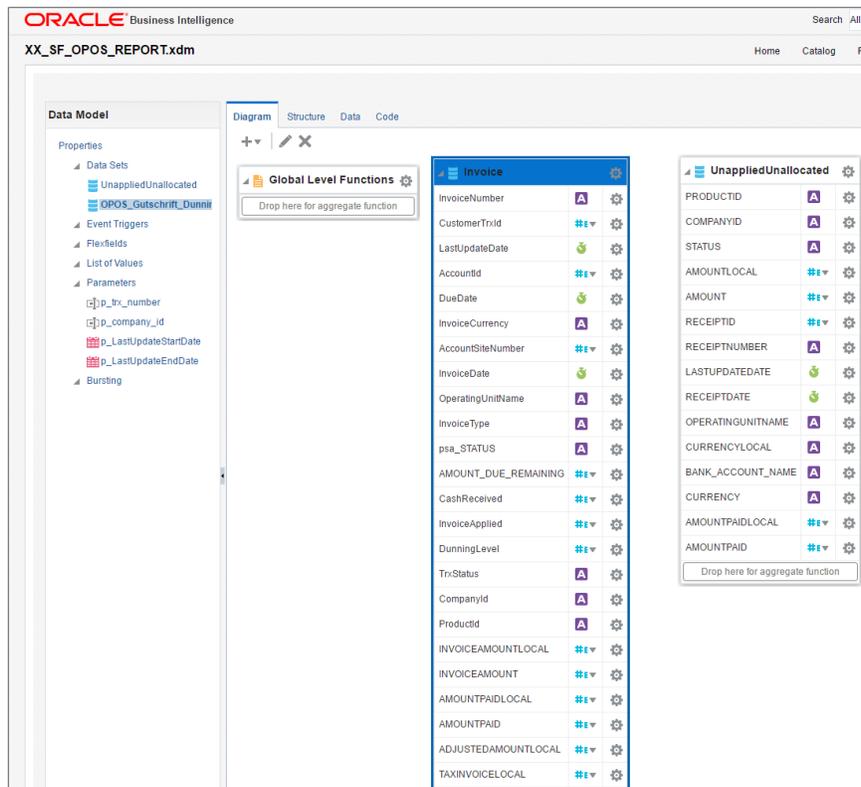


Abbildung 1: BI Publisher Report: Datenmodell

Die Bereitstellung im XML-Format erfolgt dadurch, dass als Ausgabeformat „Data (XML)“ anstelle eines benutzerdefinierten Report-Layouts verwendet wird (das z.B. eine Tabellen-Ansicht der Daten anzeigen würde):

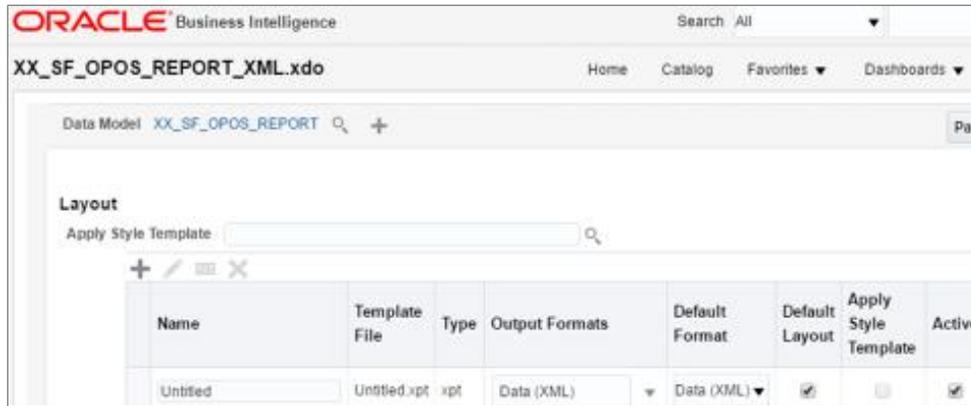


Abbildung 2: BI Publisher Layout-Definition

Oracle BI Publisher (und damit auch die Oracle ERP Cloud) beinhalten SOAP-Webservices, die es ermöglichen, einen beliebigen BI Publisher-Report „online“ abzurufen. Dabei können optional auch Parameter an den auszuführenden Report übergeben werden. Dies kann demonstriert werden, indem ein neues SOAP UI-Projekt für die folgende WSDL erstellt wird:

```
https://XYZ-test.bi.em4.oraclecloud.com:443/xmlpserver/services/ExternalReportWSSService?wsdl
```

(wobei selbstverständlich XYZ mit dem Namen der konkreten Umgebung ersetzt werden muss). Die Ausführung eines Reports wird dabei beispielhaft mit folgendem SOAP-Payload angesteuert:

```
<soap:Body>
  <pub:runReport>
    <pub:reportRequest>
      <pub:parameterNameValues>
        <pub:item>
          <pub:name>p_LastUpdateStartDate</pub:name>
          <pub:values>
            <pub:item>05-09-2016</pub:item>
          </pub:values>
        </pub:item>
        <pub:item>
          <pub:name>p_LastUpdateEndDate</pub:name>
          <pub:values>
            <pub:item>22-09-2016</pub:item>
          </pub:values>
        </pub:item>
      </pub:parameterNameValues>
      <pub:reportAbsolutePath>/Custom/Financials/XX_GoBD/XX_SF_OPOS_REPORT_XML.xdo
      </pub:reportAbsolutePath>
      <pub:sizeofDataChunkDownload>-1</pub:sizeofDataChunkDownload>
    </pub:reportRequest>
  </pub:runReport>
</soap:Body>
```

Listing 1: SOAP-Payload zum Aufrufen eines BI Publisher-Reports

Die Authentifizierung dabei erfolgt beispielsweise (es gibt weitere Methoden zur Authentifizierung), indem ein gültiger Benutzername und ein gültiges Passwort mithilfe der HTTP-Basic Authentifizierung übergeben werden. Der Webservice liefert auf diese Anfrage den BASE64-kodierte Inhalt des Reports. In unserem Fall ist dieser Inhalt ein XML-Dokument, welches alle notwendigen Daten beinhaltet:

```

<?xml version="1.0" encoding="UTF-8"?>
<!--Generated by Oracle BI Publisher 11.1.1.7.0BI-FAREL11-BP -Dataengine, datamodel: Custom_Financials_XX_GoBD_Data_Models_XX_SF_OPOS_REPORT_xdm -->
<DATA_DS>
  <P_TRX_NUMBER/>
  <P_COMPANY_ID/>
  <P_LASTUPDATESTARTDATE>2016-09-05T00:00:00.000+00:00</P_LASTUPDATESTARTDATE>
  <P_LASTUPDATEENDDATE>2016-09-22T00:00:00.000+00:00</P_LASTUPDATEENDDATE>
</INVOICE>
<INVOICENUMBER>150630-52506-1</INVOICENUMBER>
  <CUSTOMERTRXID>1180</CUSTOMERTRXID>
  <LASTUPDATERDATE>2016-09-05T12:37:12.000+00:00</LASTUPDATERDATE>
  <ACCOUNTID>100000000309091</ACCOUNTID>
  <DUEDATE>2015-06-30T00:00:00.000+00:00</DUEDATE>
  <INVOICECURRENCY>USD</INVOICECURRENCY>
  <ACCOUNTSITENUMBER>100000000309535</ACCOUNTSITENUMBER>
  <INVOICEDATE>2015-06-30T00:00:00.000+00:00</INVOICEDATE>
  <OPERATINGUNITNAME>xxxxx</OPERATINGUNITNAME>
  <INVOICETYPE>Credit Memo</INVOICETYPE>
  <PSA_STATUS>OP</PSA_STATUS>
  <AMOUNT_DUE_REMAINING>-136.6</AMOUNT_DUE_REMAINING>
  <CASHRECEIVED>-63.7</CASHRECEIVED>
  <INVOICEAPPLIED>119201</INVOICEAPPLIED>
  <DUNNINGLEVEL>0</DUNNINGLEVEL>
  <TRXSTATUS>Open</TRXSTATUS>
  <COMPANYID>.....</COMPANYID>
  <PRODUCTID>52506</PRODUCTID>
  <INVOICEAMOUNTLOCAL>-207.65</INVOICEAMOUNTLOCAL>
  <INVOICEAMOUNT>-186.17</INVOICEAMOUNT>
  <AMOUNTPAIDLOCAL>-71.05</AMOUNTPAIDLOCAL>
  <AMOUNTPAID>-63.7</AMOUNTPAID>
  <ADJUSTEDAMOUNTLOCAL>0</ADJUSTEDAMOUNTLOCAL>
  <TAXINVOICELOCAL>0</TAXINVOICELOCAL>
</INVOICE>
<INVOICE>
<INVOICENUMBER>2015-219</INVOICENUMBER>
  <CUSTOMERTRXID>2101</CUSTOMERTRXID>
  <LASTUPDATERDATE>2016-09-13T09:20:25.000+00:00</LASTUPDATERDATE>
  <ACCOUNTID>100000000385279</ACCOUNTID>
  <DUEDATE>2015-11-30T00:00:00.000+00:00</DUEDATE>
  <INVOICECURRENCY>EUR</INVOICECURRENCY>
  <ACCOUNTSITENUMBER>100000000385312</ACCOUNTSITENUMBER>
  <INVOICEDATE>2015-11-30T00:00:00.000+00:00</INVOICEDATE>
  <OPERATINGUNITNAME>yyyyyyyy</OPERATINGUNITNAME>
  <INVOICETYPE>Invoice</INVOICETYPE>
  <PSA_STATUS>CL</PSA_STATUS>

```

Abbildung 3: Ergebnis des BI Publisher-Reports

Um die erneute Synchronisierung aller (historischen) Rechnungsdaten in jedem Durchlauf des Synchronisierungsprozesses zu vermeiden, wurden die Informationen zur Datenhistorie (insbesondere „letztes Update des Datensatzes“) aus den zugrundeliegenden Oracle ERP Cloud-Tabellen verwendet, um lediglich Informationen zu Rechnungen zurückzuliefern, die innerhalb eines definierten – als Parameterwert des Berichts übergebenen – Zeitraums geändert wurden.

Die so aus Oracle extrahierten Daten wurden schließlich mittels der Informatica Cloud in ein Objekt in Salesforce überführt, welches Informationen zur Rechnung inkl. Verbindung zum Kunden sowie dem Zahlungsstatus enthält. Hierauf wird hier oben er „Oracle-Ferne“ nicht weiter eingegangen.

Synchronisierung von Kunden in die ERP Cloud

Leider ist das Schreiben von Daten in die Oracle ERP Cloud komplizierter als das oben ausgeführte Extrahieren von Daten. In unserem Fall entschied sich Fyber dazu, alle Informationen über Kunden innerhalb des CRM Werkzeugs Salesforce zu pflegen. Alle abrechnungsrelevanten Kundeninformationen (von der Rechnungsadresse über E-Mail-Adressen zum Versand von Rechnungen und Mahnungen bis hin zu Zahlungsziel-Modalitäten) müssen daher von Salesforce nach Oracle synchronisiert werden. Das beinhaltet sowohl das erstmalige Erstellen als auch die späteren Aktualisierungen der Kundeninformationen. Da bei Fyber alle Änderungen der Kundenstammdaten ausschließlich in Salesforce durchgeführt werden, war es nicht notwendig, Änderungen innerhalb der Oracle ERP Cloud wieder zurück nach Salesforce zu synchronisieren. Die Oracle ERP Cloud bietet (besonders durch den eigenen „Oracle SalesCloud“-Kern) zahlreiche SOAP Webservices, um Kunden automatisiert zu erstellen und zu aktualisieren.

Das brachte uns zu dem grundlegenden Integrationsmodell, Kunden in Salesforce als „geändert/zu synchronisieren“ zu kennzeichnen, wann immer sie dort verändert werden, und dann anschließend alle Daten des Kunden unter der Verwendung von Informatica Cloud Real Time in die Oracle ERP Cloud zu übertragen. Die folgende Abbildung zeigt das vereinfachte und vereinheitlichte benutzerdefinierte Datenmodell, das für den Transfer verwendet wurde:

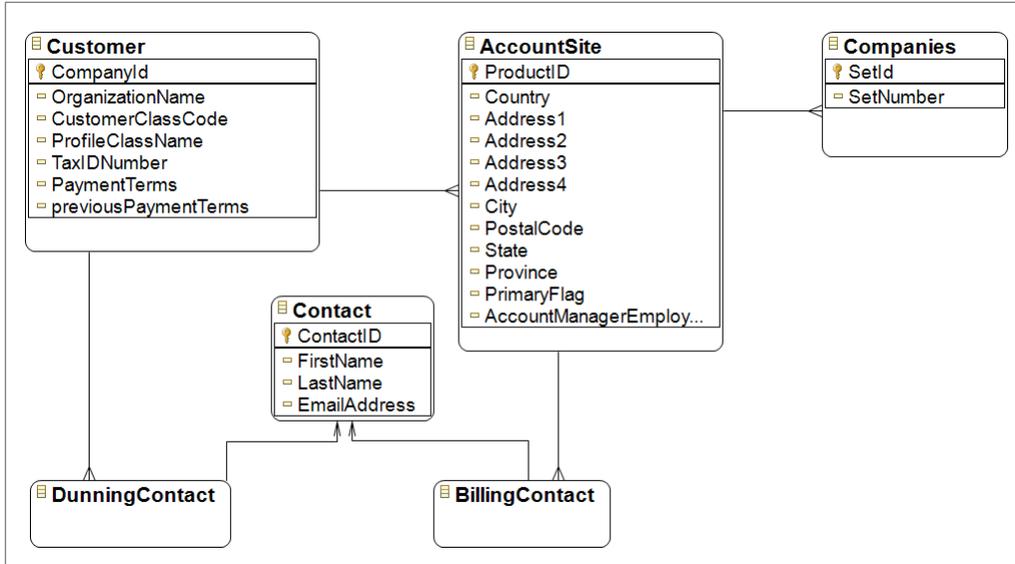


Abbildung 4: Datenmodell, das zwischen Salesforce und der ERP Cloud verwendet wurde

Leider sind die Webservice-APIs der Oracle ERP Cloud relativ komplex: in unserem Fall mussten insgesamt sieben Webservices aufgerufen werden, um einen einzigen Kunden zu erstellen. Die folgenden Prozessmodelle geben zunächst einen Überblick über den gesamten Integrationsprozess und zählen dann die Webservices auf, die für die Erstellung eines Kunden aufgerufen wurden:

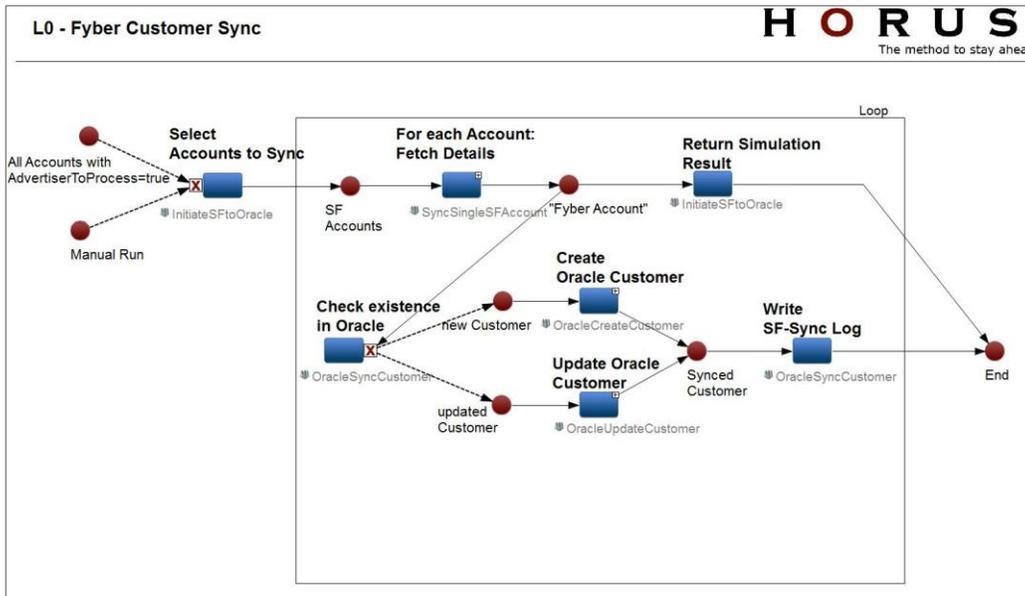


Abbildung 5: Gesamtprozess, um Kunden von Salesforce in die ERP Cloud zu synchronisieren

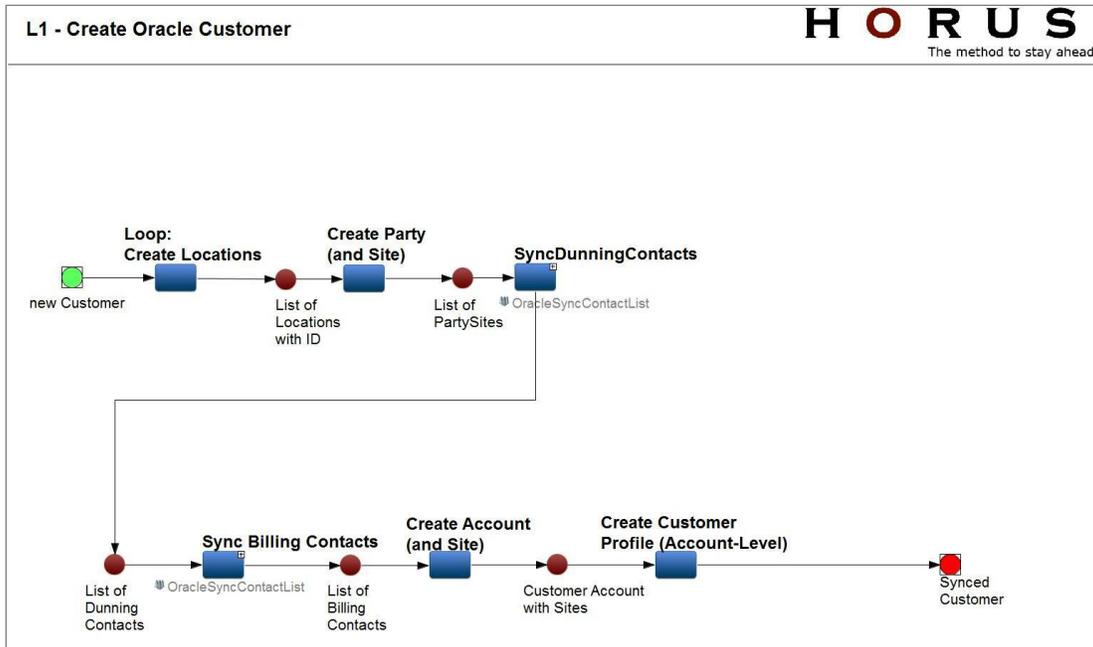


Abbildung 6: Webservices, die aufgerufen wurden, um einen Kunden in der ERP Cloud zu erstellen

Lessons Learned

Bei der Implementierung der obigen Integrationslogik wurde festgestellt, dass die „Qualität“ der Oracle Webservices für das Kundendatenmanagement sehr variiert: während die Webservices auf dem Party-, Party-Site- und Party-Relationship-Level stabil waren und ohne Probleme funktioniert haben, wurden die Dinge auf dem Kunden- und Kunden-Adress-Level schwieriger: die Dokumentation war lückenhaft und zunächst waren einige Felder durch die Webservices nicht änderbar. Als besonders problematisch erwies sich hierbei das Kundenprofil: Zu Beginn des Integrationsprojektes mit der ERP Cloud Version 10 war der Webservice praktisch funktionsunfähig. Durch zahlreiche Service- und (implementierte) Enhancement Requests und durch die verbesserte Funktionsfähigkeit in beinahe jedem monatlichen Patch-Bundle konnte zu guter Letzt die geforderte Funktionalität jedoch implementiert werden.

In diesem Prozess müssen zu jedem erstellenden oder aktualisierenden Kunden mehrere einzelne Webservices aufgerufen werden. Das führt zu einer Gesamtlaufzeit von etwa 10 Sekunden pro synchronisiertem Kunden. Optimierungen dieser Laufzeit sind prinzipiell möglich (z.B. gleichzeitige Verarbeitung mehrerer Kunden in einem einzigen Webservice-Call).

Im Zuge der Tests des Integrationsprozesses mit den Daten in Salesforce stellte sich heraus, dass eine möglichst stringente Validierung der Kundendaten in Salesforce essentiell ist (z.B. die Prüfung von vorgeschriebenen Adressfeldern je nach Land des Kunden). Auf diese Weise konnte sichergestellt werden, dass jeder Kunde, der für den „Transfer zu Oracle“ gekennzeichnet war, technisch dort auch erstellt werden konnte. Dies war wichtig, weil die Korrektur eines „halb-erstellten“ Kunden (z.B. der vierte von sieben Webservices schlägt fehl) unter Verwendung des Oracle ERP Cloud Interfaces eine komplexe Ausnahmebehandlungen innerhalb der Integrationsprozesse benötigen würden.

Synchronisierung von Lieferanten in die ERP Cloud

Während die Anforderungen und das Datenmodell in Salesforce für Kunden- und Lieferanten (hier: z.B. App-Entwickler, Webseiten- oder Video-Plattform-Betreiber, ...) bei Fyber sehr ähnlich waren, unterschieden sich die APIs in der Oracle ERP Cloud für Kunden und Lieferanten vollständig. Insbesondere bietet die Oracle ERP Cloud (bis einschließlich Release 12) keine „echten“ SOAP oder REST Webservice Schnittstellen für das Lieferanten-Management. Immerhin erlaubt die ERP Cloud eine sogenannte „dateibasierte Datenintegration“ (File Based Data Integration – FBDI): die Grundidee hierbei ist, die zu ladenden (zu erstellenden oder aktualisierenden) Daten in eine von Oracle bereitgestellte Excel-Tabelle einzutragen. Von dort werden die Daten mittels Excel-Makro-Funktionen in

eine CSV-Datei umgewandelt. Diese Datei kann in die Oracle ERP Cloud hochgeladen werden und erlaubt dann über zwei Hintergrund-Prozesse das Erstellen oder Aktualisieren von Lieferantendaten. Ähnlich wie im Falle der Kundendaten ist es dabei notwendig nicht nur eine, sondern mehrere Excel-Tabellen auszufüllen:

- Eine Tabelle, die die Lieferanten-Kopf-Eigenschaft enthält (Name, Lieferanten-Nummer, ...);
- eine, die die Adressinformationen des Lieferanten enthält;
- eine, die Lieferanten-Standort-Informationen anhand dieser Adressen erstellt;
- eine weitere, welche diese Standorte ERP Cloud „Business Units“ zuordnet
- und eine letzte, die Kontakte für die Lieferanten erstellt und zuweist.

Neben diesen grundlegenden (Procurement) Lieferanten-Upload-Sheets muss zusätzlich ein weiteres Sheet (bestehend aus drei Reitern) verwendet werden, um Bankkonteninformationen des Lieferanten zu erstellen und zuzuweisen.

Glücklicherweise kann der gesamte Vorgang – von der Erstellung des CSV, über das Laden des CSV in eine Staging-Tabelle bis hin zur eigentlichen Verarbeitung – mithilfe von SOAP Webservices automatisiert werden. Dies ist in der folgenden Abbildung ersichtlich:

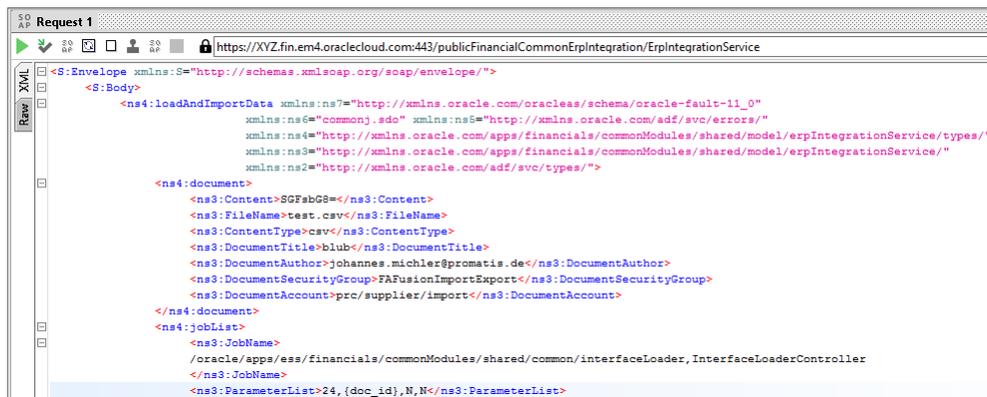


Abbildung 7: SOAP-Aufruf für den Upload des CSVs in die Staging-Tabelle

Innerhalb des „Content“-Elements muss die BASE64-kodierte Form der CSV-Datei (mit z.B. einem Lieferanten/einer Anbieteradresse/... pro Zeile) übergeben werden.

Lessons Learned

Eines der Probleme bei der Implementierung der Integrationslogik stellte das exakte Layout (besonders die Reihenfolge) der Spalten im CSV dar – diese wurde von Oracle bei größeren und kleineren Upgrades der Oracle ERP Cloud überraschend und inkompatibel geändert. Obwohl Oracle versprochen hatte, dass dieser Aspekt zukünftig besser kommuniziert und vermieden werden soll, bildet dies dennoch einen wesentlichen und inhärenten Nachteil bei der CSV-basierten Integration. Ein weiteres Problem war, dass beim Aktualisieren von Datensätzen anstelle von „Schlüsseln“ (z.B. die Lieferanten-Nummer) der bisherige (alte) „Name“ eines Lieferanten oder gar die Kombination des alten „Vornamen“, „Nachnamen“ und der „E-Mail-Adresse“ (für Lieferantenkontakte) angegeben werden musste. Um diese historischen Werte nicht in Salesforce speichern zu müssen, haben wir einen BI-Publisher-Report (analog wie weiter oben beschrieben) implementiert, um diese „alten Werte“ anhand der Schlüsselwerte (z.B. Lieferanten-Nummer) zu ermitteln. Leider gibt es in der ERP Cloud keinen „fachlichen Schlüssel“ für Lieferantenkontakte, wodurch wir gezwungen waren, einen eigenen Schlüssel mithilfe eines Flex-Felds auf Lieferantenkontaktebene zu erzeugen.

Aus Performance-Sicht dauert das Hochladen und Verarbeiten der in Summe neun CSV-Dateien ungefähr 10 Minuten (für einen Lieferanten). Aufgrund der geringfügigen Laufzeitdifferenzen zwischen dem Verarbeiten von 1 und 100 Datensätzen haben wir den Prozess so gestaltet, dass alle veränderten Datensätze auf einer bestimmten Ebene (z.B. Lieferanten-Kopfdaten) in einem CSV zusammengefasst wurden. Somit war es möglich, ungefähr 1.000 Lieferanten in 20 Minuten zu verarbeiten (Erstellung und Update).

Die Fehlerbehandlung stellt sich für die CSV-basierten Uploads einfacher als bei der Kundenschnittstelle dar: tritt ein Fehler auf, wird der weitere Synchronisationsprozess angehalten, woraufhin der User/Administrator die Daten in der Staging-Tabelle mittels weiterer Excel-Upload-Sheets korrigieren und den Prozess anschließend fortsetzen kann.

Durch die strengen Validierungen in Salesforce hat sich dies jedoch in der Praxis bisher nur sehr selten als notwendig erwiesen.

Synchronisierung von Mitarbeitern in die ERP Cloud

Da Fyber die Oracle ERP Cloud auch für die Abwicklung der Reisekostenabrechnung nutzt, müssen nicht nur Mitarbeiter der Finanzabteilung, sondern alle Mitarbeiter von Fyber in der Lage sein, sich an der Oracle ERP Cloud einzuloggen. Hierzu muss in der HCM-Komponente der Oracle ERP Cloud ein neuer Mitarbeiter erstellt werden. Dadurch wird automatisch die Erstellung eines neuen Benutzers ausgelöst.

Das Hauptsystem für Mitarbeiterdaten ist bei Fyber die Cloud HCM-Lösung „BambooHR“. Dies soll auch mit Einführung der Oracle ERP Cloud so bleiben. Daher muss jeder neue und geänderte Mitarbeiter von BambooHR in die Oracle ERP Cloud übertragen werden. Dies beinhaltet die Informationen zu Vor- und Nachnamen des Angestellten, seine Mailadresse, seinen Geschäftsbereich sowie seinen derzeitigen Vorgesetzten und die Kostenstelle (die einen wichtigen Teil Standard-Kostenstelle für Reisekosten-Buchungen darstellt). Oracle ERP Cloud bietet drei Arten von Schnittstellen im Bereich der HCM Komponente:

- SOAP Webservices: sind mittlerweile als veraltet eingestuft.
- REST Webservices: Verfügbar nach Freigabe durch Oracle (Controlled Availability).
- HCM Data Loader (HDL): Bietet einen textbasierten Upload.

In unserem Projekt haben wir die REST-API wegen ihrer Zukunftssicherheit und Benutzerfreundlichkeit gewählt. Die folgende Abbildung zeigt einen beispielhaften HTTP-Post-Aufruf, der verwendet werden kann, um einen neuen Mitarbeiter zu erstellen:

```
POST /hcmCoreApi/resources/11.1.11/emps HTTP/1.1
Host: XYZ-test.hcm.em4.oraclecloud.com:443
Authorization: Basic THIS-IS-SECRET-OF-COURSE
Content-Type: application/vnd.oracle.adf.resourceitem+json
Effective-Of: RangeStartDate=2017-02-10
{
  "FirstName" : "Johannes",
  "LastName" : "Michler",
  "AddressLine1" : "Pforzheimer Str. 160",
  "City" : "Berlin",
  "Country": "DE",
  "PostalCode" : "76275",
  "WorkEmail" : "johannes.michler+1014@promatis.de",
  "LegalEntityId" : 300000001059785,
  "UserName" : "johannes.michler+1014@promatis.de",
  "PersonNumber" : "JM1014",
  "assignments" : [ {
    "AssignmentName" : "EJM1014",
    "BusinessUnitId" : 300000001059832,
    "ActionCode" : "HIRE",
    "AssignmentStatus" : "ACTIVE",
    "WorkingAsManager" : "y",
    "LegalEntityId" : 300000001058053,
    "ManagerAssignmentId" : 300000001663982,
    "ManagerId" : 300000001663970,
    "ManagerType" : "LINE_MANAGER"
  } ]
}
```

Abbildung 8: REST Webservice Aufruf zur Erstellung eines Mitarbeiters

Lessons Learned

Der einzige Nachteil der REST-API, der aufgetreten ist, war das Fehlen des Standard-Kostenstellen-Feldes. Da dieses nicht so oft geändert wird und relativ einfach manuell zu pflegen ist, wurde fürs erste entschieden, das Feld manuell zu aktualisieren (indem eine E-Mail geschrieben wird, wann immer eine Änderung durchgeführt werden muss). Ein Enhancement Request dieses Felds in die REST-Schnittstelle aufzunehmen ist bei Oracle noch anhängig.

Sowohl die Performance als auch Fehlerbehandlung stellten keine großen Probleme während des Projektes dar, da nur ein REST-Webservice aufgerufen werden musste, um einen neuen Angestellten zu erstellen (oder diesen zu aktualisieren). Dieser Vorgang ist stets entweder erfolgreich oder scheiterte komplett, ermöglichte aber dann eine einfache Wiederherstellung (Modifizieren der Werte und erneutes Aufrufen des Dienstes).

Zusammenfassung

Eine große Herausforderung während des Projekts war die Integration mit den verschiedenen Bereichen der ERP Cloud – besonders CRM, HCM, Procurement – mithilfe von komplett unterschiedlichen Technologien. Darüber hinaus existier(t)en für einige relevanten Objekte, die sich über mehrere Bereiche ausdehnen keine (funktionierende) Schnittstellen – hier sei exemplarisch das Kundenprofil oder das Standard-Reisekosten-Konto am Mitarbeiter genannt. Durch gemeinsamen Einsatz mit der Oracle-Entwicklungsabteilung konnten jedoch alle blockierenden Probleme gelöst werden.

Zusammenfassend zeigte sich im Projekt, dass die geforderte Integration der Oracle ERP Cloud mit den existierenden Anwendungen möglich ist. Es ist somit gelungen, alle wichtigen Anforderungen zu erfüllen; die Integrationsprozesse laufen nun bereits seit einigen Monaten stabil und reibungslos.