

# Energie-Unternehmen nutzt ADF Mobile für Wartungseinsätze

Stephan La Rocca, PITSS GmbH

*ADF Mobile auf iPads konnte die Qualität der Inspektionen des Energie-Unternehmens Power South in Alabama USA revolutionieren. In einem Zeitraum von weniger als drei Monaten wurden Teile der bestehenden Forms-Applikation unter der Verwendung mobiler Zugriffe restrukturiert und eine innovative, intuitive und performante Applikation auf Basis von Oracle ADF Mobile zur Verfügung gestellt.*

Power South Energy Cooperative mit Sitz in Andalusia, Alabama ist ein Energie-Erzeuger und -Lieferant für 20 kommunale und genossenschaftliche Tochterfirmen in Alabama und dem Nordwesten Floridas. Die Inspektoren haben die Aufgabe, Leitungen und Anlagen in regelmäßigen Abständen und akut nach Unwetterschäden zu warten. Im Backoffice existiert dazu eine Forms-basierte Applikation, in der die Ergebnisse aller Inspektoren zusammengefasst werden. Daraus können Maßnahmen abgeleitet, Arbeitsaufträge erstellt und nachfolgend relevante Controlling-Informationen gewonnen werden.

Ein Versuch von Power South, Teile dieser Applikation weiterhin basierend auf Oracle Forms mit Windows-mobile Geräten zur Verfügung zu stellen, scheiterte. Schlussendlich erfolgte die Datenerfassung auf Papier, zumal einige Bereiche, in denen sich die Anlagen und Leitungen befanden, keine ausreichende Netzversorgung aufwiesen. Neben der lästigen Aufgabe, die erfassten Berichte dann erneut im Office auf dem Rechner nachzuarbeiten, waren die Berichte in Bezug auf die Führung des Inspektors in keiner Weise geeignet. Unterschiedliche Anlagen benötigen unterschiedliche Wartungsarbeiten, die Daten in verschiedenen Werten und Prüfungen erfassen lassen. In Abhängigkeit von den erfassten Informationen sind vielleicht weitere Kenngrößen zu ermitteln. Dieses Wissen erforderte einen hohen Schulungsaufwand und neue Anlagen führten immer wieder zur Aktualisierung aller Berichte.

## Lösungsansatz mit ADF Mobile

Nach der Veröffentlichung der neuen Version 1.1 von ADF Mobile im Juni des vergangenen Jahres äußerte Power South den Wunsch, die Datenerfassung durch die Inspektoren komplett zu überarbeiten. Neben einer besseren Führung des Anwenders durch den Wartungskatalog sollte die Möglichkeit geschaffen werden, in einem Gerät auch gleichzeitig die GPS-Daten des Schadensortes sowie Fotos des Schadens aufnehmen zu können.

Die Auflösung des Gerätes und die verfügbaren Features sprachen schnell für die Verwendung eines iPads. Die Geräte wurden mit einem Hardcover und einer speziellen Anti-Blend-Folie ausgestattet, sodass sie für den Außeneinsatz gewappnet waren. Da die Arbeiten in der Regel in hellem Sonnenlicht erfolgen, galt es darüber hinaus, eine leicht bedienbare und kontrastreiche Oberfläche zu erstellen. Ebenfalls musste dafür Sorge getragen werden, dass sowohl eine Online- als auch eine Offline-Nutzung der Applikation möglich ist. Mit der Erfüllung dieser Anforderungen wäre es möglich, dass die Inspektoren ihre Wartungsarbeiten ohne weitere Hilfsmittel für die Datenerfassung erledigen könnten.

PITSS startete damit, die Datenstrukturen und Oberflächen-Elemente der bestehenden Back-Office-Applikation zu analysieren, und schlug daraufhin in verschiedenen Mockups mögliche Szenarien für eine Benutzerführung vor. Schritt für Schritt wurden diese Wireframes mit dem Kunden abgestimmt, bis schließlich für die Inspektoren

eine optimale Führung und Nutzung durch den Wartungskatalog zur Verfügung stand.

Sicherlich ist die User-Experience bei Tablets gezeichnet von den vielen Apps, die die Anwender im privaten Umfeld bereits auf mobilen Endgeräten nutzen, aber auch der effiziente Umgang mit lesenden und schreibenden Daten und die Ausnutzung der zur Verfügung stehenden Auflösung und der Device-Funktionalitäten führt zu dem, was Oracle in der Projektvorgehensweise der Software-Entwicklung neuerdings gerne mit dem Schlagwort „Mobile First“ bezeichnet. Die Architektur und die Prozessabläufe sind optimiert für mobile Verfügbarkeit. Das führte, wie Sie später lesen können, dazu, dass auch andere Bereiche der Forms-Applikation nach dem Muster der mobilen Applikation „reengineert“ wurden.

Die Wireframes wurden mit dem Tool WireframePro des Anbieters MockFlow (siehe „<http://mockflow.com/apps/wireframepro>“) erstellt. Damit konnten die Szenarien sehr leicht als HTML5-Dateien exportiert und so offline besprochen werden. Spezielle Templates für eine Tablet-Implementierung erleichtern den Start und erlauben, sich auf den eigentlichen Entwurf der Anwendung zu konzentrieren (siehe *Abbildung 1*).

## Wahl der Architektur

Parallel zur Entwicklung des UI galt es, eine performante und tragfähige Architektur für die Mobile-Applikation im Zusammenspiel mit der Oracle-Forms-Anwendung zu



Abbildung 1: Mockup der Applikation mit WireframePro

kreieren. Da das Einsatzgebiet der Anwendung weitere Bereiche des Bundesstaats Alabama umschloss, war nicht immer sichergestellt, dass eine ausreichende Netzverbindung zur Verfügung stand. Die Architektur musste so gewählt werden, dass die Anwendung auch offline genutzt werden konnte. Dazu bietet das ADF-Mobile-Framework die Verwendung einer Embedded SQLite-Datenbank an.

Abbildung 2 zeigt den Aufbau der als optimal eingestuftes Architektur.

Zunächst wurden alle notwendigen Tabellen des bereits für die Forms-Applikation bestehenden Datenmodells in ein ADF-Model-Projekt überführt. Analog zu einer normalen ADF-Web-Applikation beinhaltete das Projekt alle relevanten Entity- und View-Objekte. Statt eines View-Controller-Projekts wurden jedoch anschließend darauf SOAP Services exponiert. Da der WebLogic-Server bereits vorhanden war, wurden die SOAP Services auf dem WLS zur Verfügung gestellt.

Aus Sicht des iPads war der WebLogic-Server im VPN-Netzwerk des Kunden zu erreichen. Dazu lässt sich ein gängiger VPN-Client aus dem App-Store nutzen. Der Zugriff von ADF Mobile auf die Device-Services des iPads konnte innerhalb der Applikationen zu jedem Zeitpunkt feststel-

len, ob ein Online-Zugriff auf den WebLogic-Server möglich war. In diesem Fall war ein Hintergrundprozess dazu eingerichtet, alle aktuellen Daten über die Webservices in die SQLite-Datenbank zu überführen.

Innerhalb des Projekts bot es sich allerdings an, zwei verschiedene Synchronisationsmechanismen zu ermöglichen. Für den aktuellen Arbeitsauftrag des Tages war es hilfreich, die SQLite-Datenbank nur mit den Daten zu befüllen, die für die bevorstehenden Aufgaben notwendig waren. Es fand ein kompletter Synchronisationsprozess statt, der die lokale Datenbank entleerte und mit den aktuellen Daten befüllte. Der Inspekteur konnte somit zurückliegende Ergebnisse der Wartungsarbeiten von Anlagen, die er an diesem Tag warten wollte, vor Ort einsehen und mit den aktuellen Daten vergleichen.

Sollte zwischenzeitlich die Notwendigkeit bestehen, geänderte Daten auf den Server zu übertragen oder aktuelle Daten auf das iPad zu überführen, gab es einen „leichten“ Synchronisationsprozess, der nur die geänderten Daten überführt. Die Auswahl der Synchronisation wurde dem Anwender in der Applikation angeboten.

Der Use Case, der gekennzeichnet war durch eine strikte Aufteilung der Anlagen auf die verschiedenen Inspektoren an einem Tag und die Tatsache, dass durch die Wartungsarbeiten in der Regel nur neue

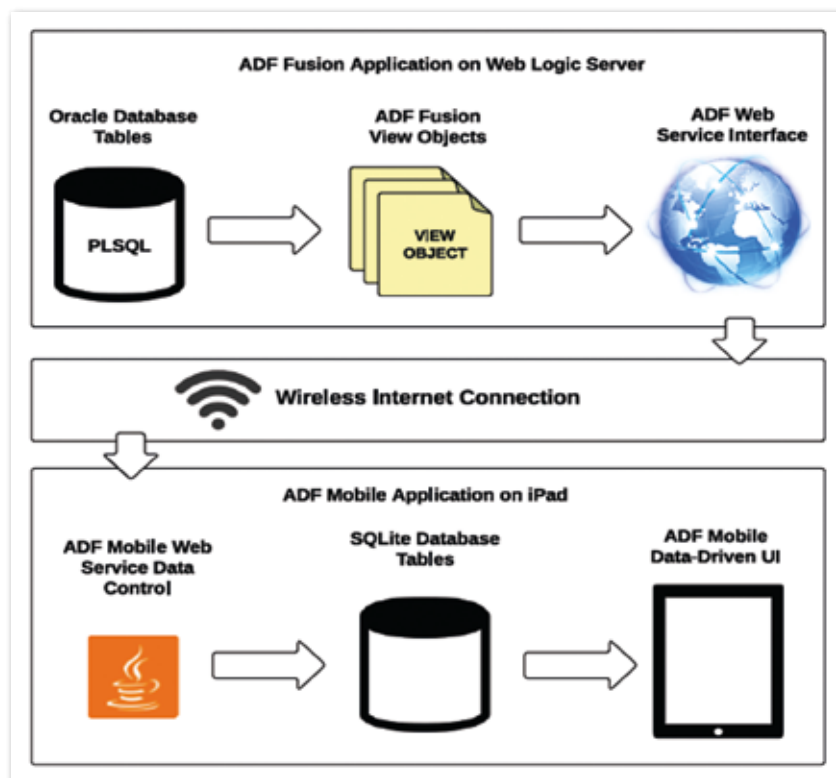


Abbildung 2: Die Architektur

Daten erfasst wurden, erlaubte weitestgehend, auf eine Konfliktbehandlung bei der Synchronisation zu verzichten.

Der Fall, dass offline an zwei unterschiedlichen mobilen Endgeräten der gleiche Datensatz geändert wurde oder dass diese Daten zwischenzeitlich auf dem Server verändert wurden, konnte durch die Modellierung des Prozesses ausgeschlossen werden. Dennoch wurden die Daten des mobilen Endgerätes zunächst ungesehen in eine „App2Sync“-Tabelle geschrieben, um die gesamte Logik der Synchronisation und somit der potenziellen Konfliktbehandlung in PL/SQL auf dem Server implementieren zu können. Aus Sicht des Endgerätes galt es, nur die Daten mit den neuesten Timestamps zurück auf den Server zu schreiben.

Für die Synchronisation vom Server auf das iPad galt zu berücksichtigen, dass neben den Tabellen, in denen die Berichtsdaten erfasst wurden, auch Tabellen für die Konfiguration der Applikation eingeschlossen werden mussten. Bei der Erfassung neuer Anlagen, der Definition weiterer Prüfungen oder bei Änderungen an der Einstufung der Schwellwerte waren diese

Daten betroffen. Für den Anwender war es aus der Sicht der Applikation ein Synchronisationsprozess, den er startete, allerdings waren insgesamt im Hintergrund fünf unterschiedliche Prozesse aktiv:

- Neue Wartungsdaten vom iPad auf den Server übertragen
- Neue Arbeitsaufträge vom Server an das iPad weitergeben
- Vorhandene Daten zu den Anlagen aus den Arbeitsaufträgen vom Server an das iPad senden
- App2Sync-Tabellen in die Produktion überführen
- Konfigurationsdaten auf das iPad überführen

Die Reihenfolge und die Abhängigkeiten innerhalb dieser Prozesse wurden auf dem Server orchestriert.

### Implementierungs-Entscheidungen

Neben den Entscheidungen, welche Architektur zum Einsatz kommen sollte und wie mit den Komponenten aus SOAP Webservices, ADF Mobile und SQLite-Datenbank umzugehen ist, boten sich in dem Projekt

weitere Implementierungsentscheidungen an. Die Zugriffe auf die Webservices und die Überführung der Daten in die lokale, verschlüsselte Datenbank erfolgten mit den zur Verfügung gestellten AdfmfJava-Utilities. Mit dem so erstellten Interface war es für den UI-Entwickler möglich, beim Design der AMX-Page-Views auf Java-Backing-Beans-Data-Controls zuzugreifen. Es wurden eigene Framework-Extensions etabliert, die nach und nach die Möglichkeit schafften, eigene Funktionalitäten in das Standard-Verhalten von ADF Mobile zu etablieren.

Der Aufbau des umzusetzenden Use Case erforderte, dass in Abhängigkeit von den Antworten der Inspektoren oder den zu wartenden Anlagenelementen sehr dynamisch verschiedene Page-Elemente auf dem iPad darzustellen sind. Diese Informationen werden in der Regel in Stammdaten zu den Anlagen und den zu prüfenden Werten erfasst.

PITSS entwickelte dabei eine sehr innovative Lösung, die es erlaubt, sehr dynamisch Komponenten auf der Seite anzuzeigen. Das betrifft sowohl die Nutzung der diversen List-Elemente als auch weitere Controls auf der AMX-Page. Kontextsensitiv können diese ein- oder ausgeblendet und mit verschiedenen Properties versehen werden. Damit ist der Kunde in der Lage, neue Anlagen mit neuen Prüfverfahren und Werten einzupflegen, ohne dass Anpassungen in der Oberfläche notwendig werden.

Außerdem galt es, sich Gedanken über den Umgang mit den Fotos zu machen.

Die Inspektoren hatten die Möglichkeit, bei den Wartungsarbeiten vor Ort im Falle eines Schadens direkt ein Foto von dem Schaden zu erfassen. Damit konnte eine völlig neue Qualität in die Erfassung der Wartungsarbeiten eingeführt werden. Die Aufgabenstellung bestand allerdings darin, bei der mobilen Applikation die Größe des Fotos adäquat berücksichtigen zu können. Mit der hochauflösenden Kamera des iPads entstanden so Bilder in der Größe mehrerer MBs. Werden sie als Dateien verwaltet, ist die Synchronisation mit den Serverdaten recht aufwändig, werden sie hingegen als Base64-String konvertiert, ist die Anzeige auf dem iPad sehr rechenintensiv.

Um dieses Problem zu umgehen, entwickelte die PITSS GmbH eine eigene Photo-Util-Klasse, die zunächst das normale ADF-



Abbildung 3: Screen für den Inspekteur aus der fertigen Applikation

Mobile-PhoneGap-Gateway nutzte, um auf die Foto-Services des Endgeräts zugreifen zu können. Die lokal erstellten Bild-Dateien wurden vor der Synchronisation mit dem Server in ein Base64-String transferiert und so über den Webservice übermittelt. Fotos, die im Synchronisationsprozess vom Server auf das mobile Endgerät übertragen wurden, konnten mit dieser Java-Klasse aus dem Base64-String zurück in eine lokale Datei konvertiert werden. Somit vereint die Implementierung die Vorteile beider Varianten – die performante Anzeige eines Bildes aus einer lokalen Datei und den Verzicht auf die Synchronisation eines Bild-Attachments. In einem Zeitraum von ca. acht Wochen für die Implementierung und den Architekturaufbau entstand so mit insgesamt sechs Entwicklern eine Applikation, die maßgebend für die weiteren Prozesse bei Power South als Vorlage diente (siehe Abbildung 3).

### Verbesserungen im Prozess

Neben der Möglichkeit, Bilder von den Schäden zu erfassen, gibt es durch die Ver-

wendung der Device-Services weitere Verbesserungen im Prozess, die aufgrund der zuvor eingesetzten Technologie gar nicht möglich waren. So ist der Inspekteur bei der Erfassung seines Schadens in der Lage, gleichzeitig die GPS-Koordinaten über das iPad zu erfassen. Diese Informationen sind in ADF Mobile durch das PhoneGap-Interface direkt aus dem Java-Code von ADF Mobile verfügbar. GPS-Koordinaten können ausgelesen und zum aktuellen Schadensdatensatz gespeichert werden.

Am Ende entstand eine Softwarelösung für den Kunden, die die Arbeit der Inspektoren bei den Wartungsarbeiten der Anlagen revolutionierte. In einem Zitat des IT-Leiters des Kunden heißt es von Kenneth Jones, Power South Business Applications Supervisor: „Leveraging Oracle ADF Mobile, we were able to transition our employees to use new mobile devices (iOS) leveraging the latest device capabilities and allowing for offline work.“ Ergänzend dazu PITSS-Projektleiter Ross Smith: „The mobile application interface is so intuiti-

ve that our users don't need any training. Thanks go to Oracle ADF Mobile's great components, which let us quickly turn our design plans into reality.“

Der Kunde ist bereits jetzt in der Planung, weitere Komponenten der Oracle-Forms- Applikation zu extrahieren und über Mobile Devices zur Verfügung zu stellen. Eine Präsentation der Applikation ist auf YouTube zu sehen.



Stephan La Rocca  
slarocca@pitss.de

**PROLICENSE**<sup>®</sup>  
OPTIMIZING SOFTWARE ASSETS  
Kompetent – Unabhängig – Erfolgsbasiert

# ORACLE-SOFTWARE IST JEDEN CENT WERT!

Unsere Mandanten zahlen trotzdem weniger.  
Sprechen Sie mit uns!

Wir sind nur unseren Mandanten verpflichtet.

- > **Compliance sichern**
- > **Audit vermeiden**
- > **Kosten senken**

**ProLicense GmbH**  
Friedrichstraße 191 | 10117 Berlin  
Tel: +49 (0)30 60 98 19 230 | [www.prolicense.com](http://www.prolicense.com)