

Apex mobile vs. Universal Theme

Steven Grzbielok, MT AG

Die größte Neuerung und Erleichterung durch Apex 5 ist für jeden Apex-Entwickler wohl das Universal Theme: Es ermöglicht die Entwicklung einer Responsive-Design-fähigen Anwendung, die sowohl auf dem Desktop als auch auf mobilen Endgeräten eine komfortable Nutzung erlauben soll. Doch was ist aus Apex mobile geworden? Wo stößt das Universal Theme an seine Grenzen und wo sollte man weiterhin auf Apex mobile zurückgreifen? Dieser Artikel gibt einen Überblick über die Unterschiede und jeweiligen Vorteile, um die Entscheidung zwischen den beiden Themes zu erleichtern.

Für die in *Abbildung 1* aufgeführte Gegenüberstellung wurde die Sample Database Application verwendet. Wer beide Themes testen möchte, kann auf „apex.oracle.com“ einen kostenfreien Workspace erstellen und die Packaged Application „Sample Database Application“ installieren. Sie wird direkt mit beiden Themes installiert. Ein Button schaltet zwischen Apex mobile und dem Universal Theme um.

Universal Theme – one size fits all?

Während frühere Apex-Themes das Responsive Design lediglich unterstützten, wurde dies durch die Einführung des Universal Theme in Apex 5 grundlegend erweitert: Bisher hatte der Entwickler bei der Erstellung einer Apex-Anwendung die Möglichkeit, aus vielen verschiedenen Themes zu wählen. Diese Entscheidung brachte dementsprechend unterschiedliche Features mit sich, besonders im Hinblick auf die mobile Nutzung. Manche Themes waren geeigneter als andere und je nach Anforderung an die Templates war hier einiges bei der Auswahl zu bedenken.

Oracle hat nun für Apex das Universal Theme eingeführt, wobei der Name schon die Intention dahinter vermittelt: Ein Theme, das allen gängigen Anwendungsfällen und Geräten gerecht werden soll. Oracle hat gezielt hierfür die ID nicht fortlaufend weitergeführt, sondern die ID 42 (die Antwort auf

die Frage nach dem Leben, dem Universum und allem Übrigen aus „Per Anhalter durch die Galaxis“) gewählt.

Das Universal Theme ist komplett auf die Nutzung auf sowohl mobilen als auch Desktop-Geräten ausgerichtet. Es werden die neuesten Standards hinsichtlich Webentwicklung wie HTML5 und CSS3 genutzt. Während früher die Navigation als Tabs verwendet wurde, wurde sie nun links in einer Baumstruktur implementiert (siehe *Abbildung 2*). Diese Navigationsleiste lässt sich mit dem weitverbreiteten „Hamburger“-Icon (dargestellt als drei

waagerechte Balken) ein- und ausklappen. Auf dem Desktop wird diese Navigation im Standard komplett mit Icons und Menüeinträgen eingeblendet, auf dem Tablet sind nur Icons für die Navigation zu sehen und auf dem Smartphone ist die Navigation im Standard komplett ausgeblendet.

Dieses Verhalten variiert jedoch immer in Abhängigkeit von der Bildschirmgröße, sodass etwa bei einem sehr großen Tablet die komplette Navigation angezeigt oder bei einem kleinen Netbook die Navigation auf die Icons beschränkt wird. Bei Bedarf kann der Nutzer jederzeit die Navigation

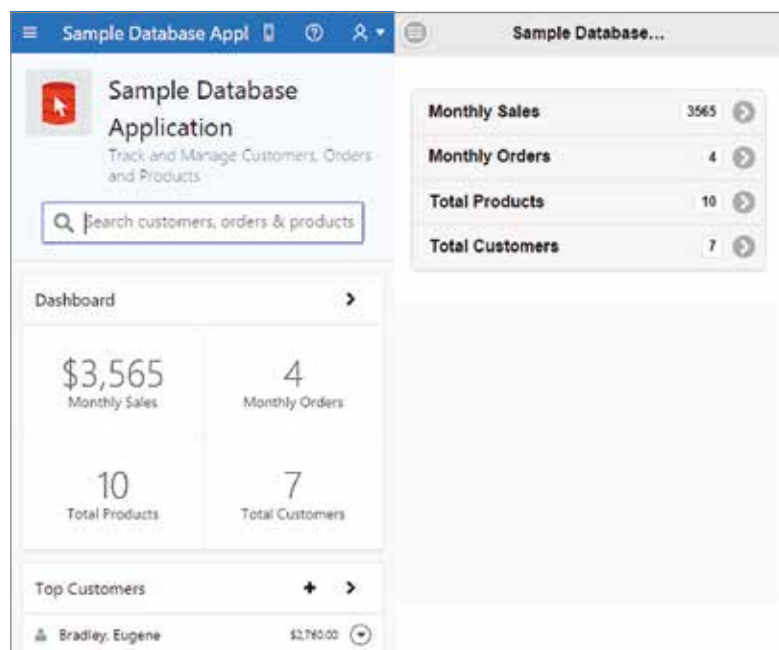


Abbildung 1: Sample Database Application im Universal Theme (links) und Apex mobile (rechts)

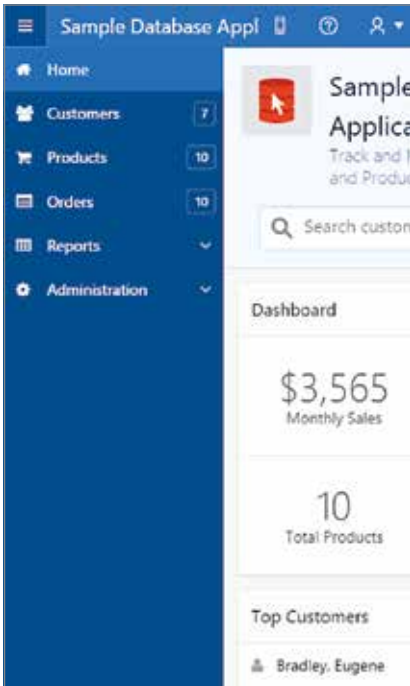


Abbildung 2: Die neue Navigation in einer Baumstruktur mit „FontAwesome“-Icons

ausklappen, um bequem durch die Seiten zu navigieren.

Außerdem fällt auf, dass die Schriftgröße der Texte sich ebenfalls an die Bildschirmgröße anpasst: Um nicht weniger Informationen auf kleinen Bildschirmen darzustellen, werden die Überschriften und Textkörper bei einem kleineren Bildschirm automatisch verkleinert.

Die Komponenten wurden in Apex 5 ebenfalls so überarbeitet, dass auch Reports so komfortabel wie möglich auf allen Bildschirmgrößen nutzbar sind. Zu beachten ist jedoch, dass vor allem Interactive Reports und Tabular Forms mit besonders vielen Spalten problematisch auf Smartphones werden und zu horizontalem Scrollen führen können oder lediglich im Querformat eine komfortable Nutzung ermöglichen (siehe Abbildung 3).

Regionen auf einer Seite werden nun automatisch in einem überarbeiteten Grid-Layout angeordnet: Eine Seite besteht aus zwölf Spalten. Man kann wählen, wie viele Spalten eine Region umfassen soll, oder Apex diese Entscheidung überlassen. Bei den unterschiedlichen Bildschirmgrößen vergibt Apex nun für die jeweiligen Regionen eine Breite in Abhängigkeit von den gewählten Spalten. Sobald der Bildschirm jedoch eine gewisse Größe unterschreitet, sodass diese

Spalten zu schmal werden und somit eine Region ebenfalls nicht mehr komfortabel nutzbar ist, werden die Regionen untereinander dargestellt. So kann auf einem Smartphone dieselbe Anwendung wie auf dem Desktop genutzt werden, trotz der unterschiedlichen Bildschirmverhältnisse.

Die meisten der Komponenten im Universal Theme wurden ebenfalls so entwickelt, dass Responsive Design ermöglicht wird. Es werden in den meisten Fällen keine manuellen CSS-Anpassungen benötigt: Listen-Regionen wie Cards werden zunächst nebeneinander dargestellt, wobei die Größe der Karten sowie deren Anordnung komplett abhängig von der Bildschirmgröße dynamisch ermittelt werden.

Die neue Kalender-Region bietet zunächst auf dem Desktop eine Monatsansicht, die auf Smartphones (hochkant) viel zu breit wäre. Hier wechselt die Region automatisch in eine Listen-Ansicht, sodass alle Termine übersichtlich untereinander dargestellt und überflüssige Tage nicht angezeigt werden.

Zusätzlich wurde auch Tap Event Handling in dem Universal Theme eingeführt: Bei der Nutzung von Anwendungen auf einem Smartphone werden Berührungen mit einer Verzögerung von 300 ms interpretiert. Dies ist bei vielen nativen Apps eine Standard-Eigenschaft, da eine Berührung nicht zwangsläufig ein absichtlicher Klick ist. Die Verzögerung ermöglicht es der Anwendung, zwischen einem Klick durch Tippen oder dem Scrollen durch eine ziehende Berührung auf dem Bildschirm zu differenzieren.

Das Universal Theme wurde dahingehend optimiert, um derartige mobile Standards out of the box bei der Nutzung eines Touchscreens zu ermöglichen. Zusätzlich bietet das Universal Theme durch die Einführung von Template Options noch mehr Möglichkeiten, um ohne aufwändige CSS-Modifizierungen eine Anwendung „mobile friendly“ zu gestalten. Der Entwickler kann viele Einstellungen deklarativ definieren und so CSS-Klassen via Mausclick setzen. Sollten Tests auf mobilen Geräten nun Probleme in der Darstellung offenbaren, lassen sich durch die Template-Options verschiedene Einstellungsmöglichkeiten ausprobieren, um so die beste Option zu finden.

Die in Apex 5 eingeführten modalen Dialoge sind ebenfalls mobil verfügbar und bieten auch hier Vorteile: Zum Beispiel lässt sich ein Datensatz eines Reports bearbeiten, ohne dass die Seite gewechselt werden muss. Das Formular zur Bearbeitung wird einfach als modaler Dialog (vergleichbar mit einem Popup) eingeblendet und sobald die Bearbeitung beendet ist, lässt sich die Berichtsregion neu laden. Insgesamt hat das Universal Theme einen deutlichen Fortschritt hinsichtlich Responsive Design gemacht, sodass ein Apex-Entwickler ohne viel Aufwand eine Anwendung für verschiedene Endgeräte und Bildschirmgrößen entwickeln kann.

Mehraufwand für Apex mobile

Während das Universal Theme noch sehr neu ist, ist Apex mobile bereits seit der

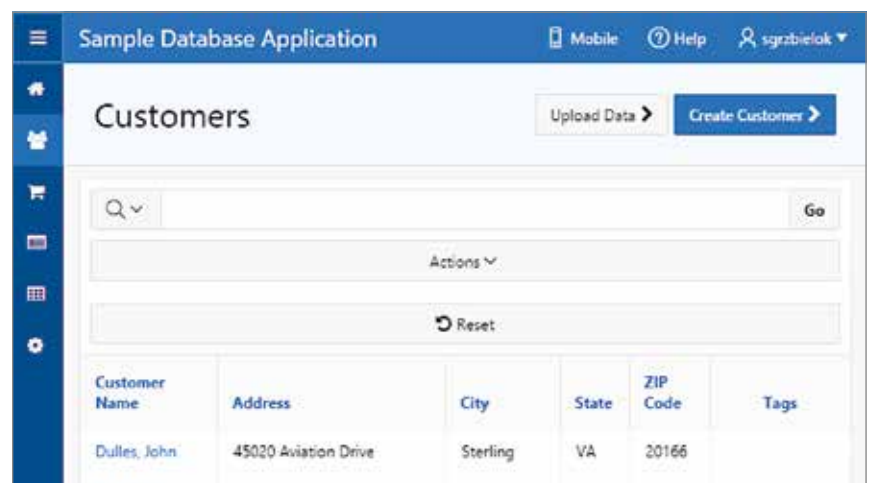


Abbildung 3: Im „Landscape“-Modus lassen sich auch Interactive Reports noch bequem auf dem Smartphone nutzen

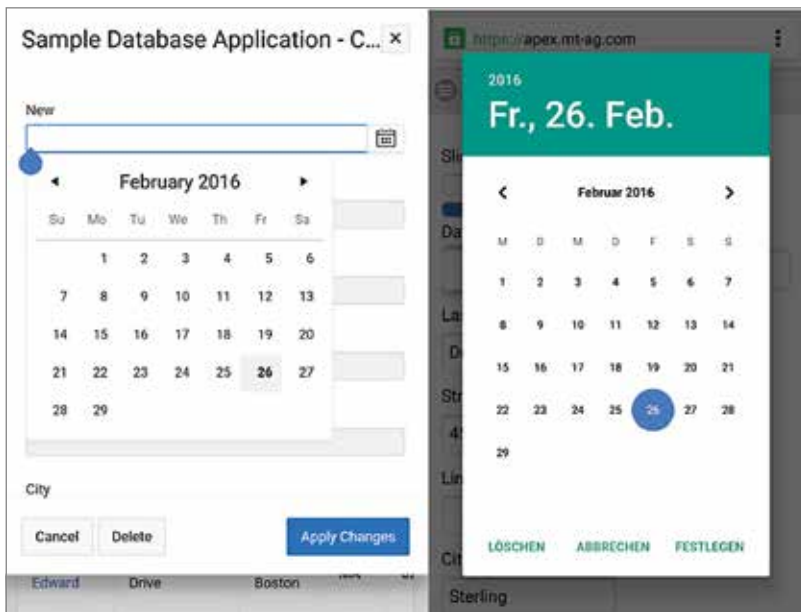


Abbildung 4: Apex mobile (rechts) unterstützt im Gegensatz zum Universal Theme (links) native Form-Elemente

Einführung von Apex 4.2 (im Jahr 2012) verfügbar. Apex mobile wurde ausschließlich für die Nutzung auf mobilen Endgeräten entwickelt und basiert auf dem jQuery-Mobile-Framework. Eine Nutzung der Anwendung ist zwar ebenfalls auf dem Desktop möglich, jedoch sollte ein separates User-Interface mit einem anderen Theme für den Desktop genutzt werden, da das Design von Apex mobile nicht für den Desktop-Monitor optimiert ist. Es wurde hier auf eine besonders hohe Kompatibilität mit verschiedenen mobilen Plattformen geachtet, damit möglichst unabhängig vom Endgerät die Funktionalität und der Komfort garantiert werden können.

Apex mobile ist im Gegensatz zum Universal Theme deutlich leichter konstruiert, es müssen also weniger HTML, CSS und JavaScript geladen werden, um eine Seite aufzubauen. Da auf mobilen Endgeräten oft die Netzwerk-Verbindung limitierend für Anwendungen wirken kann, wird die Seite auch bei schlechtem Empfang schnell aufgebaut und verbraucht weniger Datenvolumen. In Apex mobile werden dem Entwickler spezielle Form-Elemente zur Verfügung gestellt, die besonders komfortabel auf einem Touchscreen nutzbar und nach Möglichkeit auch mit den nativen Elementen des jeweiligen mobilen Betriebssystems verknüpft sind (siehe Abbildung 4). So wer-

den zum Beispiel Datums-Eingaben mit den Android- beziehungsweise Apple-Date-Picker verwendet, damit der Nutzer wie gewohnt sein Gerät nutzen kann. Slider und Kippschalter können in Formulare eingebunden werden, um die Bearbeitung der Daten auch mit Touch-Gesten zu ermöglichen.

In Apex mobile wurden ebenfalls einige Regionen integriert, die ausschließlich auf mobilen Geräten verfügbar sind: Die Reflow Table dreht die Ansicht, sobald der Bildschirm zu schmal für einen herkömmlichen Report wird. Es werden die Spalten horizontal dargestellt und dann untereinander die einzelnen Datenreihen angezeigt. So muss der Nutzer ausschließlich vertikal scrollen und behält auch auf einem kleineren Bildschirm den Überblick über komplexere Datensätze.

Der Column Toggle Report hingegen ist ein einfacher Report, der lediglich so viele Spalten darstellt, wie auf dem Bildschirm ausreichend Platz finden. Der Nutzer kann dann über Checkboxes auswählen, welche Spalten er angezeigt haben möchte.

Als Best Practice werden zudem häufig in Reports lediglich die ID und der Name (beziehungsweise eine fachliche Bezeichnung) eines Datensatzes in Listenform angezeigt. Der Nutzer kann dann über einen Klick auf den Datensatz zu einer Detailansicht gelangen.

Die intuitive Nutzung von Touch-Gesten ist mittlerweile ein Standard auf mobilen Endgeräten. In Apex mobile lassen sich diese Touch-Events auch für Dynamic Actions verwenden. Beispielsweise wird ein Wischen über den Bildschirm so von Apex erkannt und kann eine entsprechende Aktion auslösen. Mit Apex mobile wird versucht, dem Nutzer das Look & Feel einer nativen Anwendung im Browser zu vermitteln.

Der Header und der Footer sind im Standard fixiert und bilden den Rahmen der Anwendung. Der Seitenaufruf findet durch AJAX statt, es muss also nicht die komplette Seite neu aufgebaut, sondern per JavaScript lediglich der Body geladen werden. Dies führt zu einer schnelleren Reaktionszeit und verstärkt das App-Gefühl. Apex mobile bietet somit eine Vielzahl spezifischer mobiler Features, die aktuell im Universal Theme nicht deklarativ nutzbar sind. Weniger manuelle CSS- und JavaScript-Anpassungen sind hier notwendig.

Was ist besser?

Das Universal Theme bietet eine schnelle Möglichkeit, Anwendungen für Desktop und mobile Endgeräte zu entwickeln. Die Nutzung von Apex mobile hingegen bedeutet einen erheblichen Mehraufwand, da hier eine doppelte Entwicklung und Anwendungspflege nötig ist. Dafür ist Apex mobile jedoch deutlich optimierter für mobile Geräte und bringt einige Features ausschließlich für mobile Geräte mit.

Um zu entscheiden, welches Theme in einem speziellen Use Case am sinnvollsten ist, sollte man sich zunächst selbst einige Fragen stellen: Mit welcher Priorität wird die Anwendung mobil genutzt? Wird die Anwendung primär auf einem Desktop eingesetzt und sollen nur sekundär mobile Geräte unterstützt werden? Oder liegt der Fokus klar auf mobilen Geräten? Was ist wichtiger: Design oder Funktionalität?

Das Universal Theme hat deutlich mehr in Sachen Optik zu bieten als Apex mobile: Es wirkt wesentlich schicker, wohingegen Apex mobile eher einen nüchternen Eindruck macht.

Die integrierte Nutzung des Theme Roller sowie die ansprechende, an Googles Material Design orientierte Gestaltung des Theme machen einen sehr moder-

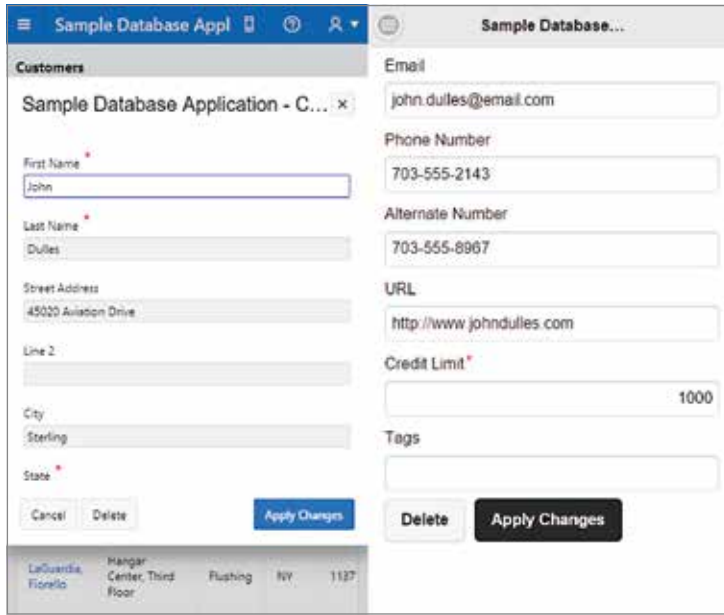


Abbildung 5: Die Nutzung von Formularen ist bei Apex mobile (rechts) deutlich bequemer, da sich die Elemente intuitiver gestalten und sich die Größe präziser am Bildschirm ausrichtet

nen Eindruck. Dem Nutzer wird direkt der Eindruck vermittelt, dass er mit einer hochaktuellen Anwendung arbeitet. Jedoch hat eben diese Optik seinen Preis: In einem Test bei regulärer 3G-Verbindung dauert der Seitenaufbau mit dem Universal Theme mehr als eine Sekunde länger. Die benötigte Datenmenge, die vom Server auf das Endgerät übertragen wird, ist nahezu doppelt so groß. Dies mag zwar bei gelegentlicher Nutzung nicht besonders gravierend sein, jedoch macht es gerade bei schlechter Netzwerkverbindung und sehr intensiver Nutzung der Applikation auf Dauer einen Unterschied.

Durch den Seitenaufbau per AJAX vermittelt Apex mobile in den Tests einen noch zügigeren Eindruck beim Seitenwechsel. Der Nutzer verwendet zwar auf seinem mobilen Gerät einen Browser, jedoch wird durch die AJAX-Nutzung und den fixierten Header und Footer das Look & Feel einer nativen App vermittelt (siehe Abbildung 5).

Ein weiterer entscheidender Faktor ist die Auswahl der Regionen. Das Universal Theme bietet einige Standard-Regionen, die in Apex mobile nicht verfügbar sind: Classic Report, Interactive Report, Tabular Form, Map Chart, Region Display Selector und Tree. Besonders das Map Chart und der Interactive Report bieten Funktionalitäten, auf die man oft nicht verzichten kann. Diese Regionen wer-

den jedoch bewusst nicht in Apex mobile angeboten, da auf einem kleinen Bildschirm ein zu komplexes User-Interface nicht dargestellt werden kann. Hier muss entschieden werden, ob eine korrekte Darstellung oder die gebotenen Funktionalitäten der Regionen wichtiger sind.

Apex mobile bietet jedoch ebenfalls einige Regionen an, die wiederum nicht im Universal Theme nutzbar sind: Column Toggle Report, List View und Reflow Re-

port. Diese sind speziell für mobile Geräte entwickelt worden und bieten viele Vorteile hinsichtlich Usability, die von den Regionen des Universal Theme nicht erfüllt werden können. Entsprechend den Anforderungen der Nutzer sollte nun beurteilt werden, welche Regionen eher die benötigten Funktionen liefern können (siehe Abbildung 6).

Letztendlich kommt es bei der Entwicklung mit den verschiedenen Themes darauf an, wie wichtig in dem jeweiligen Anwendungsfall die Nutzung auf mobilen Endgeräten ist. Liegt der Fokus auf mobilen Geräten, sollte weiterhin Apex mobile zum Einsatz kommen, da durch Touch-Events eine bessere Performance und ein natives Look & Feel eine komfortablere und zuverlässigere Nutzung möglich ist. Auch die speziell für mobile Geräte entwickelten Regionen und Komponenten bieten hier viele Vorteile.

Wird die Anwendung jedoch primär für Desktops entwickelt und soll lediglich auch auf mobilen Geräten lauffähig sein, so ist dem Universal Theme der Vorzug zu geben. Apex nimmt dem Entwickler hier in den meisten Fällen schon die Arbeit ab und stellt die Anwendung im responsiven Design dar. Ebenfalls sind die gewohnten Regionen auf dem Desktop und auf dem mobilen Gerät verfügbar, wodurch an keiner Stelle Einbußen bezüglich der Funktionalitäten in Kauf genommen werden müssen.

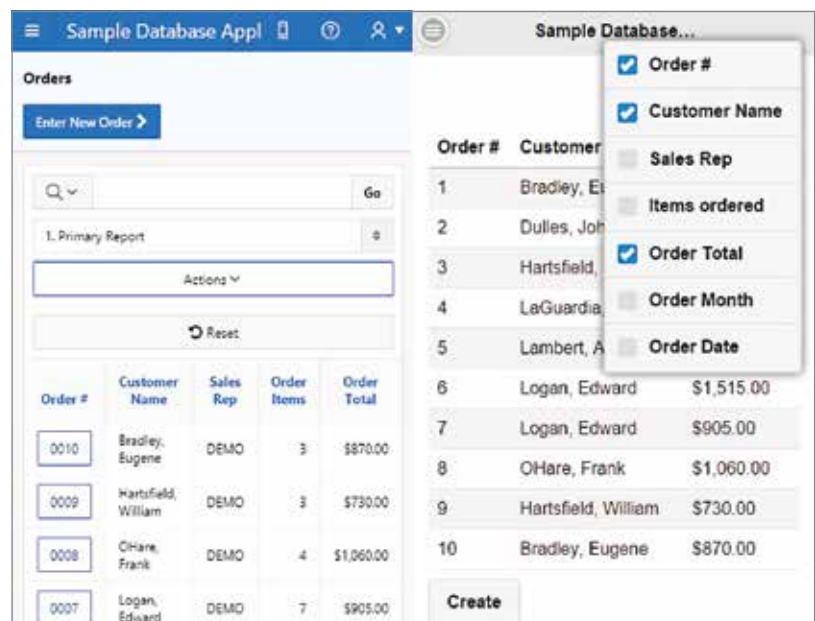


Abbildung 6: Interactive Reports (links) bieten mehr Funktionen zum Suchen und Filtern, Column Toggle Reports (rechts) sind jedoch übersichtlicher

Da komplexe Regionen jedoch auf mobilen Geräten unter Umständen nicht korrekt dargestellt werden, sollte die Anwendung besonders auch auf kleinen Bildschirmgrößen getestet werden. Zu komplexe Regionen oder Seiten sollte man gegebenenfalls bewusst ausblenden, damit überladene Seiten und zu viel Scrollen vermieden werden. Um eine komfortable Nutzung auf mobilen Endgeräten zu garantieren, ist das ausführliche Testen auf verschiedenen Geräten bei beiden Themes essenziell.

Was die Zukunft bringt

Das Universal Theme wird in den kommenden Updates viele Verbesserungen erhalten. Vorrangig für den Desktop interessant, werden die neue Region „Interactive Grid“ als Ablösung der Tabular Forms eingeführt und die Master-Detail-Ansicht optimiert. Zusätzlich wird eine neue Charting-Engine eingeführt: Oracle-JET-Charts werden künftig als Standard-Charting-Engine für Apex eingesetzt. Es werden neue

und verbesserte deklarativ erstellbare Diagramme verfügbar sein. Da Oracle JET jedoch noch nicht alle Diagramm-Typen unterstützt, die aktuell durch AnyCharts nutzbar sind, werden diese zunächst weiterhin deklarativ mit AnyCharts verfügbar sein. Außerdem soll Tap-Event-Handling ab Apex 5.1 ebenfalls deklarativ für Dynamic Actions im Universal Theme genutzt werden können.

Für Apex 5.1 sind aktuell keine speziellen neuen Features für Apex mobile geplant, jedoch wird Apex mobile ebenfalls Features wie Oracle-JET-Charts nutzen können. Der integrierte Theme Roller, der bereits für das Universal Theme eingesetzt werden kann, soll ebenfalls für Apex mobile verfügbar sein. Aktuell kann für Apex mobile nur der externe jQuery Mobile Theme Roller verwendet werden. Außerdem wird daran gearbeitet, dass auch neue Updates von jQuery Mobile in Apex integriert werden.

Nach einer langen Pause werden aktuell die Arbeiten an jQuery Mobile 1.5 weitergeführt und frühestens in Apex 5.2 können Apex-Entwickler ebenfalls mit

der Einbindung dieser Neuerungen rechnen. Damit Apex 5.1 vor dem Release getestet werden kann, arbeitet Oracle wieder an einem Early Adopter.

Eine erweiterte Gegenüberstellung von Apex mobile und dem Universal Theme in Verbindung mit Oracles Mobile Application Framework (MAF) wird auf der Apex connect 2016 von Davide Groppuso und Steven Grzbielok vorgestellt. Weitere Informationen hierzu auf „apex.doag.org“.



Steven Grzbielok
steven.grzbielok@mt-ag.com

Die aktuellen Trends der Logistik-Branche und deren IT-Herausforderungen



DOAG 2016 Logistik + IT

@ CeMAT Hannover | 2. Juni 2016

