



## Ted Schadler's Blog

Ted serves Application Development & Delivery Professionals

# Mobile Needs A Four-Tier Engagement Platform

Posted by Ted Schadler on November 20, 2013

Michael Facemire, John McCarthy, and I recently published a clarion call to the technology industry: It's time for a new architecture! The aging Web isn't designed to handle mobile apps or sites. And it certainly can't handle the real-time demands of connected products.

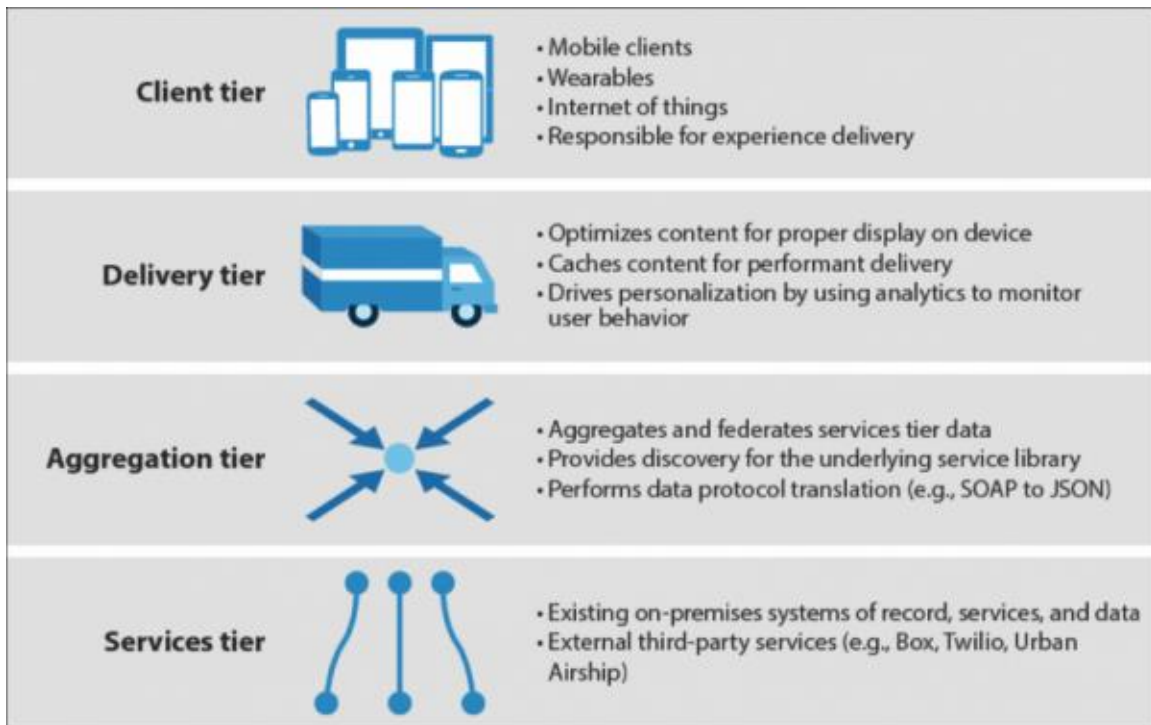
Here's how we summarize it:

Mobile is pushing aging web architectures to the brink. The three-tier architecture built for a browser-led PC world can't flex, scale, or respond to the needs of a good mobile experience or the emerging requirements for connected products. Mobile's volatility and velocity of change require a distributed four-tier architecture that we call an "engagement platform." The engagement platform separates technical capabilities into four parts: client, delivery, aggregation, and services. The new requirements of modern apps will force content distribution networks, application server vendors, mobile middleware vendors, platform-as-a-service suppliers, a myriad of startups, and enterprises to coalesce around this four-tier architecture. CIOs need to start planning immediately for the migration from three tiers to four.

It's time to throw out the old notion of a three-tier architecture -- presentation, application, data -- and replace it with a four-tier engagement platform that can handle the new demands:

*An engagement platform supports a distributed, four-tier architecture natively engineered to deliver compelling experiences, excellent performance, and modular integration on any device over any network at Internet scale.*

**Figure 1** The Four-Tier Engagement Platform Makes Delivery Its Own Tier



How the four tiers handle the load:

1. **A client tier accounts for the unique attributes of different devices.** This presentation layer insulates the unique capabilities of each app and device — desktop or mobile, browser or dedicated app — from the services that back-end applications deliver. This boundary allows developers to create back-end services like flight status and shipment notification independent of the mobile app that will consume them. Creating this clear boundary drives productivity for your developers without an onerous maintenance challenge; it will be critical for a fluid business partner network.
2. **A delivery tier handles special middle- and last-mile challenges.** This element uses intelligence from the client layer to determine the optimal way to deliver contextually appropriate content. The delivery tier accomplishes this by using over-the-wire content transformation — as opposed to protocol-based conversions at the next aggregation layer — and leveraging edge-of-network cache functionality for increasingly dynamic data. CDNs such as those provided by Akamai, along with delivery optimization solutions like Instart Logic, application delivery controllers like Riverbed Stingray, and on-premises in-memory database caches, fulfill this today.
3. **An aggregation tier integrates internal and external services and transforms data.** This API layer has two brokerage roles, providing discoverability between app requests and services and bidirectional translation

between client requests and back-end or third-party services. This makes composing the underlying data and services easier and enables relatively simple real-time translation to the appropriate data format. The service composition becomes more dynamic with the addition of business intelligence, analytics, and role-based access, which occurs in this tier.

4. **A services tier spans internally and externally provisioned data and functionality.** This final architectural element dynamically composes data and business processes through a set of continuously deployable services. This tier provides data to the layers above without concern for how that data is consumed; the other layers can exist behind the corporate firewall or externally — or both! This allows for the ultimate flexibility in the consumption and dynamic composition of services, whether leveraged by apps or by the evolving partner ecosystem.

## Plotting The Path Forward

Question: If engagement platforms are the future, then how do we get there? Answer: Disruptively at the architectural heart of the technology industry. The three-tier architecture long heralded by IBM, Microsoft, Oracle, and Akamai is already being hollowed out by an emerging four-tier approach used by Netflix, Kinvey, and Salesforce.com.

It's time for vendors, investors, enterprise architects, application developers, and CIOs to begin a thoughtful technical and business discussion about how to build and operate an engagement platform. Engagement platforms will handle the complex, contextual, real-time demands of mobile experiences and connected products and drive these disruptions:

- **Firms will rely on a ecosystem of providers to deliver engagement.** You're not going to buy an engagement platform in a box. And it won't all run under your direct control. Instead, you will assemble -- or rely on a provider to assemble -- a set of engagement capabilities loosely coupled and able to scale to the demand while remaining flexible and able to handle continuous delivery.
- **HTTP traffic on Nginx will surpass the traffic on Apache.** If you don't know what those things are, don't worry about it. If you don't but think you should, then see the last chart here [[news.netcraft.com/archives/2013/10/02/october-2013-web-server-survey.html](http://news.netcraft.com/archives/2013/10/02/october-2013-web-server-survey.html)] and learn about it here [[wiki.nginx.org/Main](http://wiki.nginx.org/Main)] and here [[nginx.com/](http://nginx.com/)].
- **IBM, Oracle, Microsoft, SAP, and Salesforce.com will rethink their middleware products and architectures.** Adding mobile app lipstick on an application server won't solve the delivery challenge by itself. These companies will slowly overcome their reluctance and begin to operate in a four-tier, ecosystem-dependent platform. But it will take five years for that to happen. Microsoft Azure is the farthest along, but still tentative in its strategy and execution.
- **The content delivery network industry will accumulate content and performance optimization services.** Akamai, Limelight, and Amazon CloudFront

have done good work for the Web. But new approaches from Instart Logic and rumblings from Akamai point to the delivery-tier future.

- **Platform-as-a-service providers like Amazon, Google, IBM, Microsoft, and Salesforce will grow many new platform services.** For example, services in the aggregation tier loosely coupled with an intelligent delivery tier will handle notification at scale, including personalized messages, open-message analytics, and device and network customization.

What say you?