Buzzword-Bingo:
Cloud, Mobile & HTML5 – lass mer mei Ruh'!?

Christian Weyer
christian.weyer@thinktecture.com

thinktecture

Christian Weyer

- **Solution architect and principal consultant at thinktecture**
- **Focus on**
  - distributed applications
  - service orientation, workflows
  - cloud computing
  - interoperability
  - pragmatic end-to-end solutions
  - Windows Server, WCF, WF, MSMQ, Windows Azure platform

- **Microsoft MVP for Windows Azure (Architecture)**
- **Independent Microsoft Regional Director for Germany**

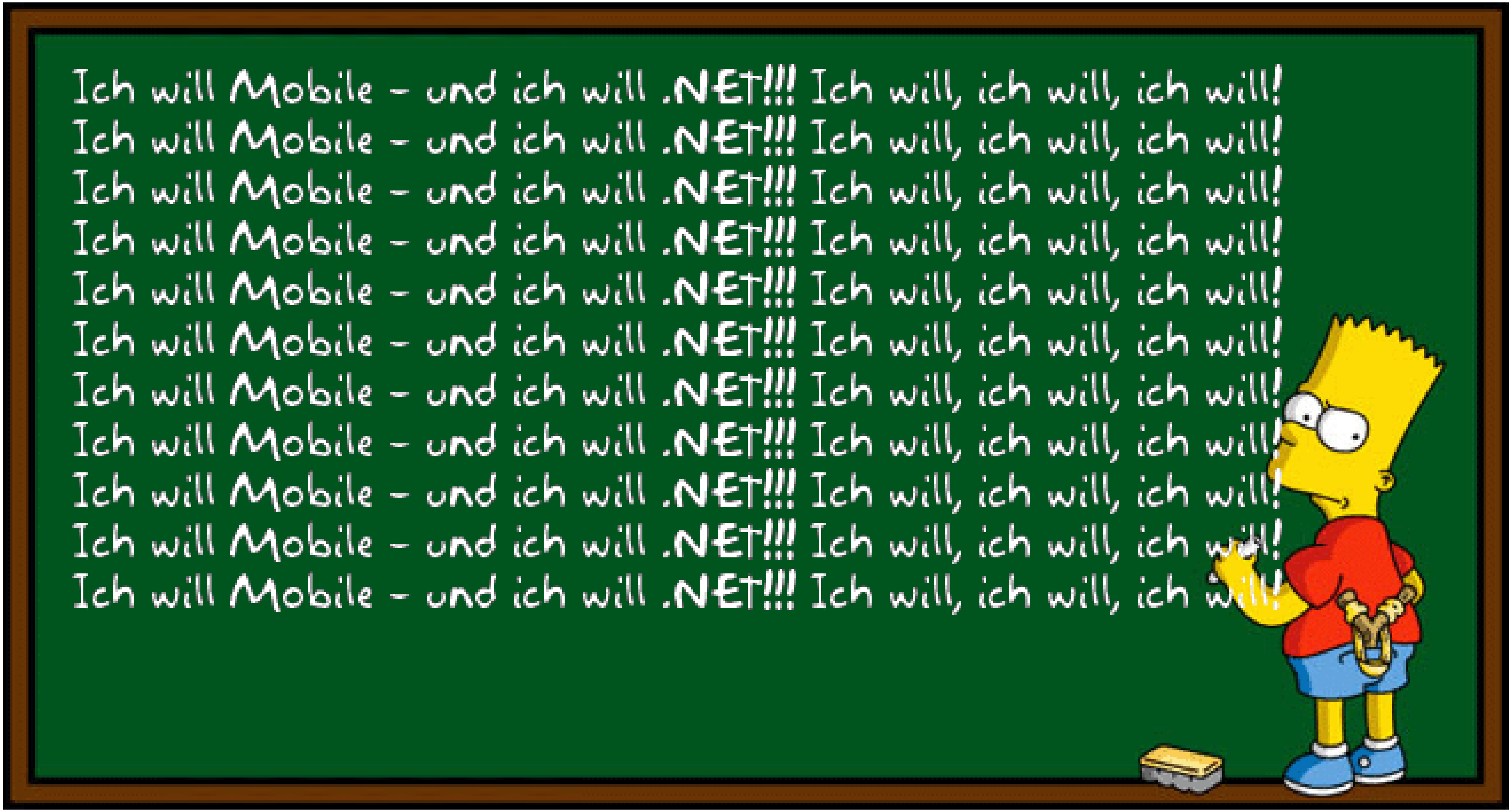
- **http://blogs.thinktecture.com/cweyer**
- **christian.weyer@thinktecture.com**

Agenda

- **Mobile**
  - why?
  - who?
  - what?
  - how?
- **HTML5 & Co.**
  - why?
  - who?
  - what?
  - how?
- **Cloud**
  - why?
  - who?
  - what?
  - how?

Mobile

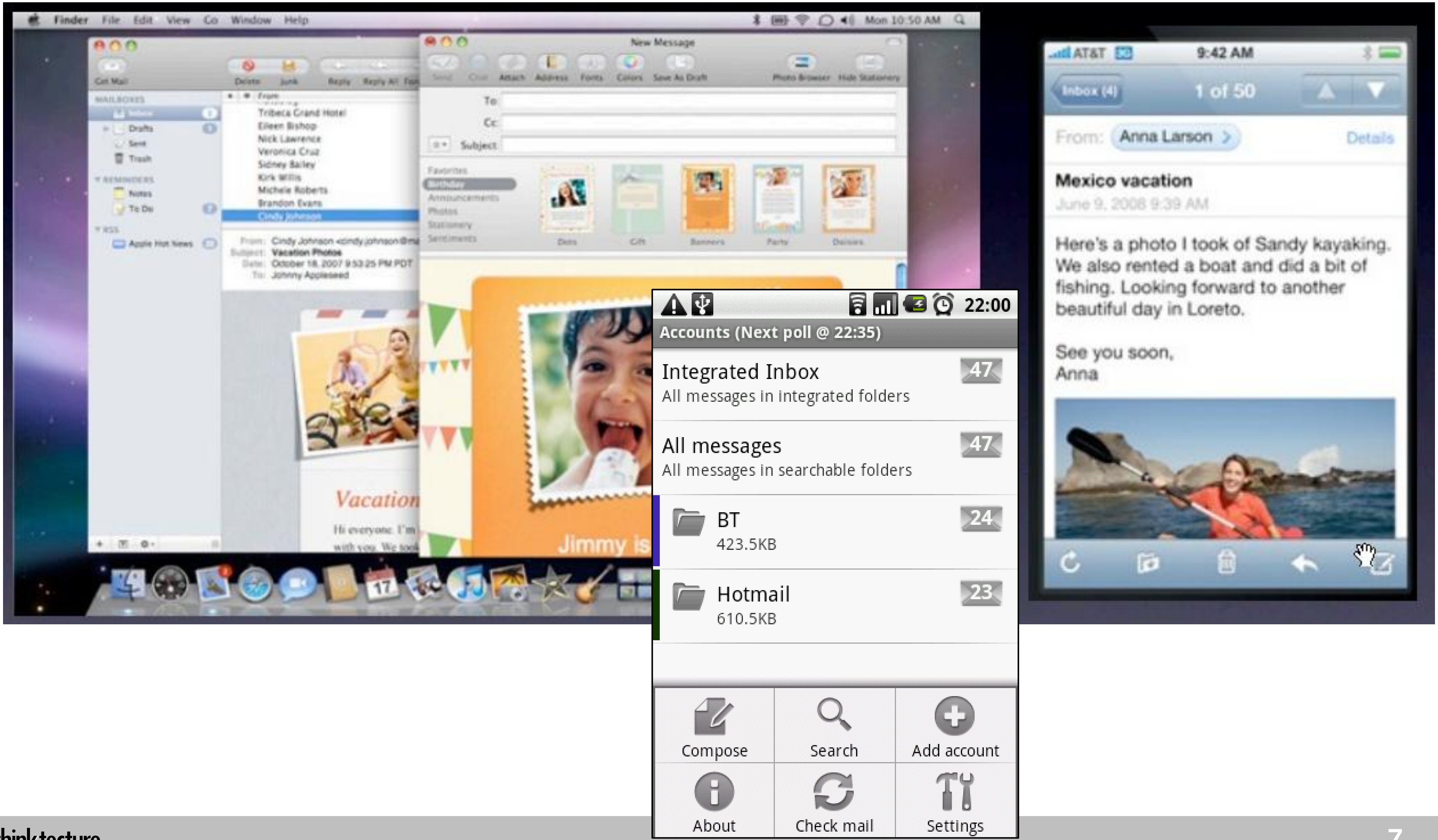
What would Bart do?


Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!
Ich will Mobile - und ich will .NET!!! Ich will, ich will, ich will!

Strategies to get on mobile devices

- **Web application**
- **Sencha Touch, jQuery Mobile**
- **PhoneGap**
- **Appcelerator**
- **Native (e.g. Java, Objective-C)**
- **Mono**
- **Some others…**

UI paradigma: there are differences, really



Accounts (Next poll @ 22:35)

Integrated Inbox 47
All messages in integrated folders

All messages 47
All messages in searchable folders

BT 24
423.5KB

Hotmail 23
610.5KB

From: Anna Larson

Mexico vacation
June 9, 2008 9:39 AM

Here's a photo I took of Sandy kayaking. We also rented a boat and did a bit of fishing. Looking forward to another beautiful day in Loreto.

See you soon,
Anna

Writing software for iOS – the Apple way

- Need to be part of the Apple developer program

- Using Objective-C as the programming language
- Using CocoaTouch as the application framework
- Using XCode, Interface Builder et. al. as the programming tools and IDE

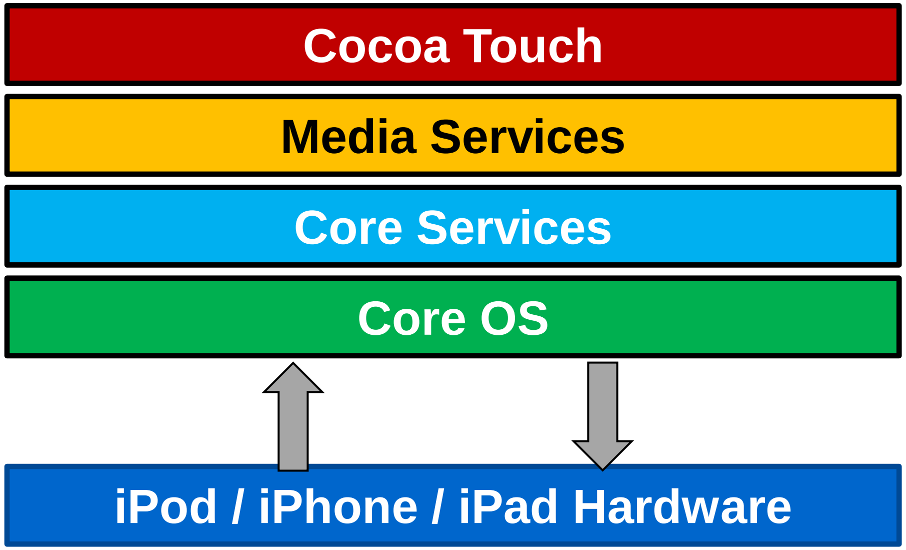
- Deal with devices, certificates, app IDs, profiles and other fun stuff

Writing software for Android – the Google way

- **Open source**
- **Need to be part of the Google developer program**

- **Using Java as the programming language**
- **Using Android SDK as the application framework**
- **Using e.g. Eclipse IDE**

iOS Architecture



CocoaTouch

- **Audio and Video**
  - Core Audio
  - OpenAL
  - Media Library
  - AV Foundation
- **Data Management**
  - Core Data
  - SQLite
- **Graphics and Animation**
  - Core Animation
  - OpenGL ES
  - Quartz 2D
- **Networking and Internet**
  - Bonjour
  - WebKit
  - BSD Sockets
- **User Applications**
  - Address Book
  - Core Location
  - Map Kit
  - Store Kit

Developing with XCode & Co.

- **Free, with your purchase of a Mac**
  - hey, you wanted a Mac anyways, eh?
- **iOS developer program enrollment**
  - US$ 99: single user
  - US$ 299: enterprise development
- **Basic testing of apps with the iOS simulator**

Objective-C

- **Object-oriented extension of C**
  - Smalltalk-like syntax
  - “full” C compatibility
- **Feels a bit like the late 80s/early 90s**
- **No gargabe collection**
  - reference counting
  - Mac's Objective-C has GC, though

```
ObjC:
[str stringByTrimmingCharactersInSet:
       [NSCharacterSet whitespaceCharacterSet]];

C#:
str.Trim();
```

Objective-C

- **Verbose?**

```
ObjC:
+ (NSDate *) stripTime:(NSDate *) date {
NSCalendar *gregorian = [[NSCalendar alloc]
initWithCalendarIdentifier:NSGregorianCalendar];
NSDateComponents *components = [gregorian
components:(NSYearCalendarUnit | NSMonthCalendarUnit |
NSDayCalendarUnit) fromDate:date];
date = [gregorian dateFromComponents:components];
[gregorian release];

return date;
}


C#:
date = otherDate.Date;
```

- **You need to be able to read Obj-C code, for documentation, at least**

What is MonoTouch?

- **Mono**
  - open source implementation of .NET
- **MonoTouch**
  - .NET/C# layer over iOS programming
- **MonoDevelop**
  - IDE
- **Uses Apple’s iOS SDK**
  - provides missing functionality through .NET APIs
- **Integrates with Interface Builder**
  - design surface builder
- **Provides CocoaTouch wrapper**
- **Provides AOT Compilation**

What MonoTouch is Not

- **Not Windows Forms/WPF/Silverlight on the iPhone**
- **Not a plugin to Visual Studio**
- **Free**

MonoTouch Editions

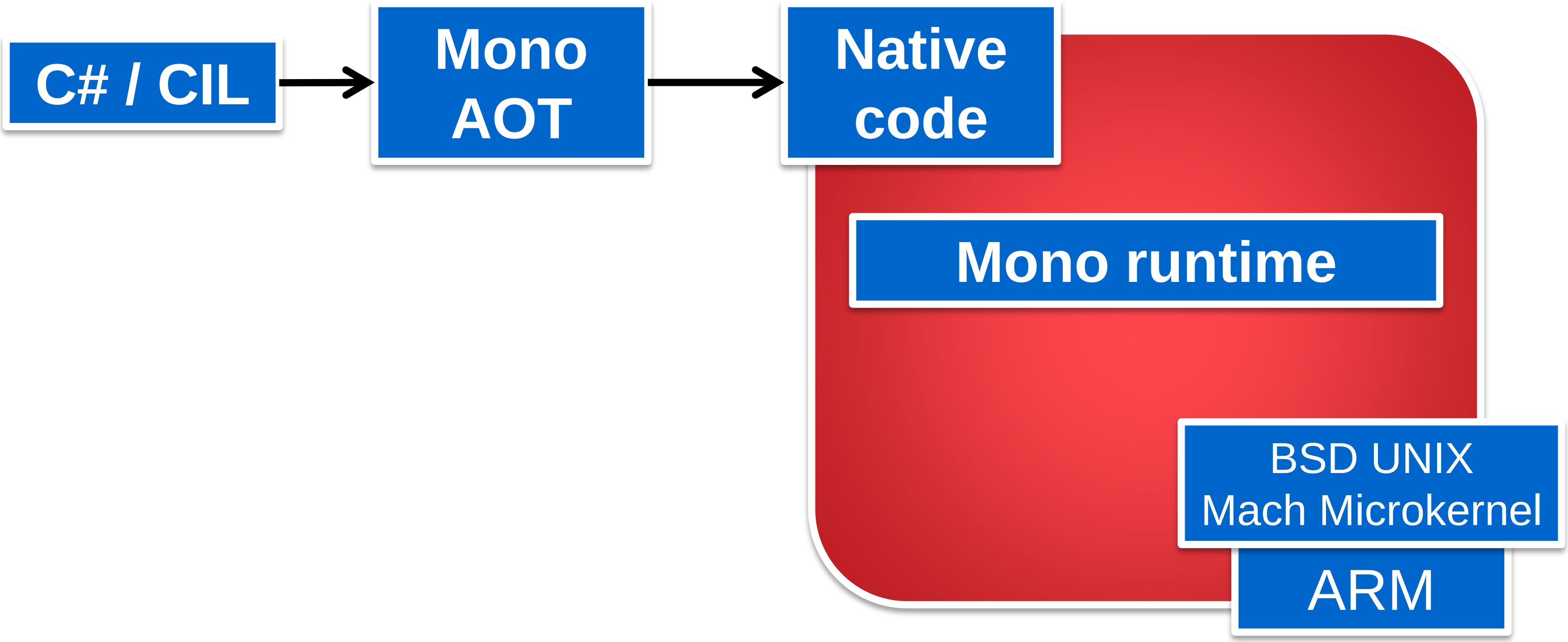
- **Community**
  - free
  - only simulator
  - can’t deploy to device
- **Single User**
- **Enterprise**

MonoTouch Features

- **mtouch**
- **Full static AOT compiler**
- **MonoDevelop iPhone Add-In**
- **CocoaTouch .NET wrappers**
- **Support for (almost) all your existing code**
  - reflection
  - generics
  - LINQ
  - anonymous methods
  - lambda’s etc...
- **...but some important parts of .NET are missing/ not complete ...**

MonoTouch

- **Apple does not like shared libraries on the devices**
- **JIT-ting not allowed on the iOS devices by Apple**
- **Monotouch uses Ahead-Of-Time compiling (AOT)**



MonoTouch’s APIs

.NET APIs

- mscorlib
- System
- System.Core (LINQ)
- System.Data
- Mono.Data.Sqlite
- System.ServiceModel
  - WCF
- System.Json
- System.Web.Services
- System.Xml
- System.Xml.Linq

MonoTouch

- AddressBook/ AddressBookUI
- AudioToolbox/ AVFoundation
- CoreAnimation
- Coregraphics
- CoreLocation
- GameKit
- MediaPlayer
- MessageUI
- StoreKit
- SystemConfiguration
- UIKit

Third Party

- OpenTK
  - OpenGL
  - OpenAL
- Sqlite-CS
- XnaTouch
- CocosNet
- ServiceStack

MVC in Apple’s world

- **Views, View Controllers, and Models**
  - inconsistent between different controls on how it is actually implemented
- **CocoaTouch has controller classes**
  - typical: navigation controller used to load different views `UINavigationController`
- **iPhone Apps commonly have multiple views**
  - push and pop the controllers for each view
  - views are usually in the XIB (NIB) files
- **“The MVC that is not”**
- **MonoTouch completely embraces CocoaTouch’s MVC**

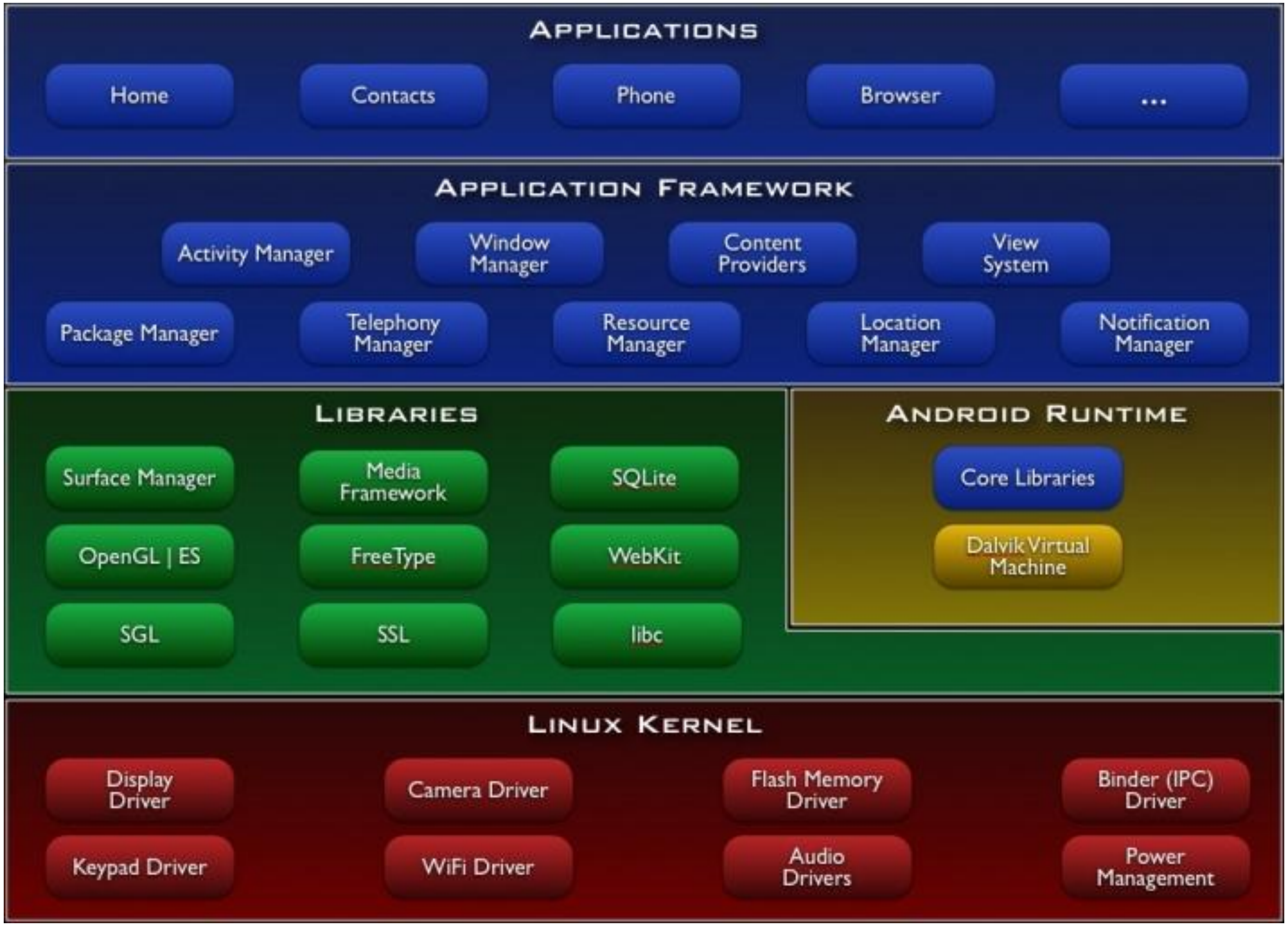
MonoTouch.Dialog

- **API toolkit to simplify creating typical data-driven apps**
  - create dialogs
  - show table-based information
  - without dozens of delegates and controllers
- **Currently supported:**
  - string informational rendering
  - text entry, password entry
  - jump to HTML page
  - radio elements
  - dates, times
  - on/off controls
  - slider (floats)
  - activity indicators
  - arbitrary `UIViews`

What is Android?

- **Full mobile stack**
  - purchased by Google in 2005
  - v1.0 released in October 2008
- **Open source**
- **SDK provides tools and Java API**
  - applications written in Java
  - Dalvik Virtual Machine
- **Customized Linux 2.6 kernel**

Android Architecture

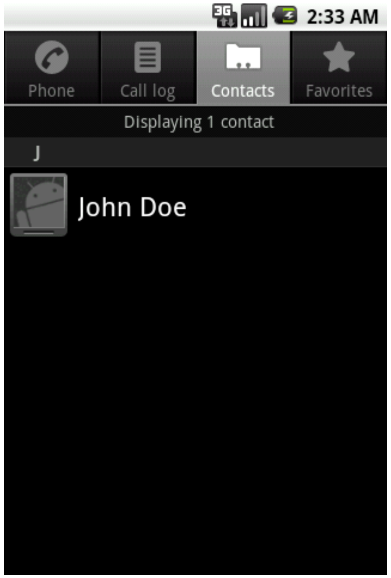


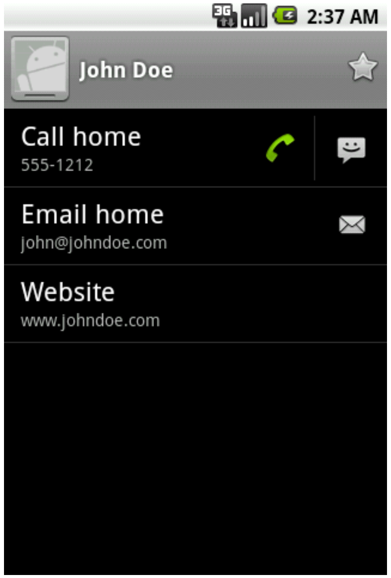
Developing with Eclipse & Co.

- **Free**
- **Android developer program enrollment**
  - US$ 25
- **Basic testing of apps with the Android emulator**

Activities

- Provides UI for one screen
- Can start other activities
- Hierarchy of views





Call home
555-1212

Email home
john@johndoe.com

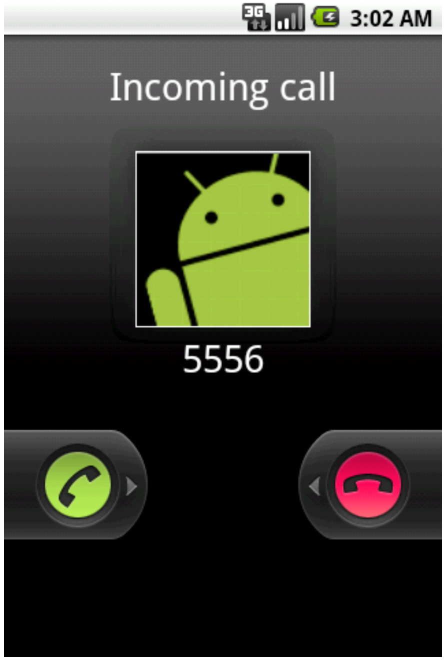
Website
www.johndoe.com

Activity Lifecycle (Simplified)

- **Activity stack**
- **Activity has 3 basic states**
  - running
  - paused
  - stopped
- **Configuration changes cause activity restart**

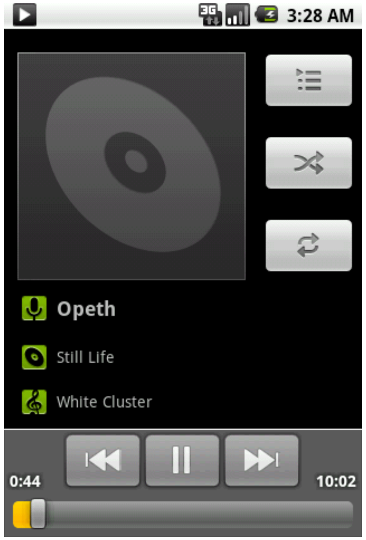
Broadcast Receivers

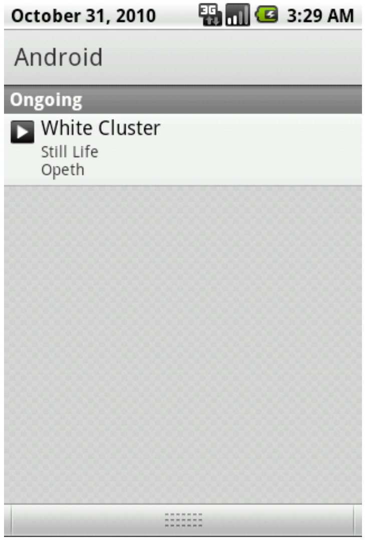
- No UI
- Can start an activity
- Receive/react to announcements
- Examples: low battery, phone call



Services

- No UI, runs in background
- Stays running when application loses focus
- Can be accessed by many applications
- Example: music player





What is MonoDroid?

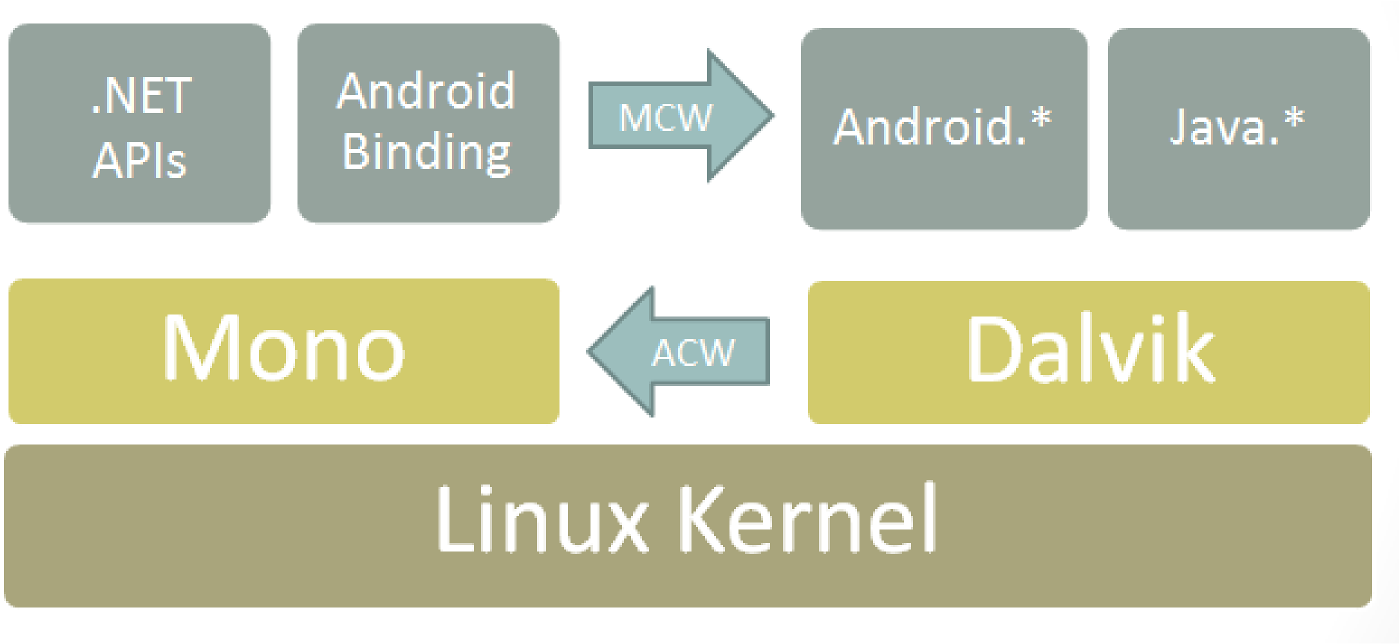
- **Mono**
  - open source implementation of .NET
- **MonoDroid**
  - .NET/C# layer over Android programming
- **Support for MonoDevelop & Visual Studio**
- **Uses Android SDK**
  - wraps Java/Android API bindings
  - provides missing functionality through .NET APIs
- **Runs side by side with Dalvik**

What MonoDroid is Not

- **Not Windows Forms/WPF/Silverlight on the Android**
- **Free (?)**

MonoDroid Architecture

- **Android/Managed Callable Wrappers**
  - JNI bridges to talk between Android and Mono
- **Shared runtime**

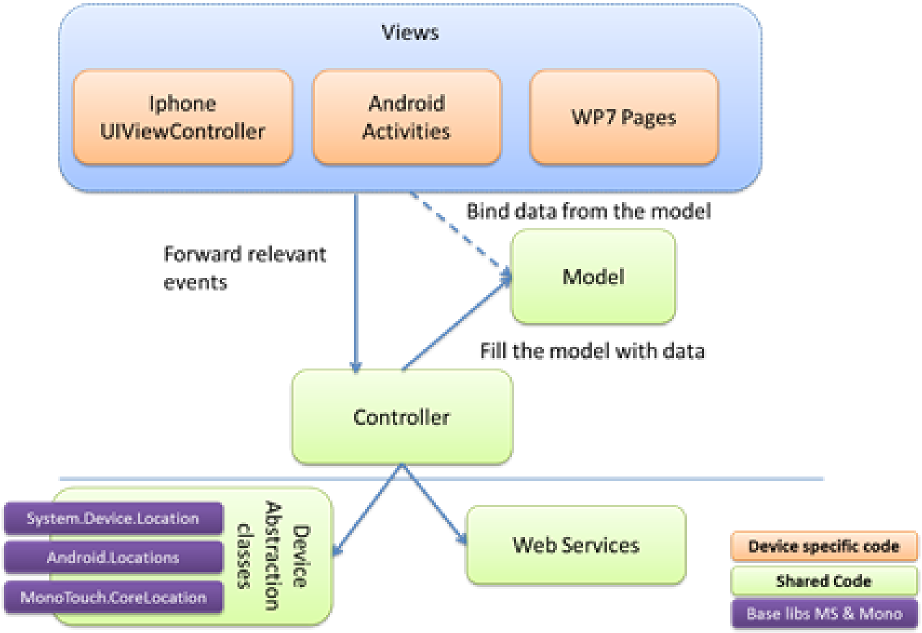


Integrating with a service-oriented world

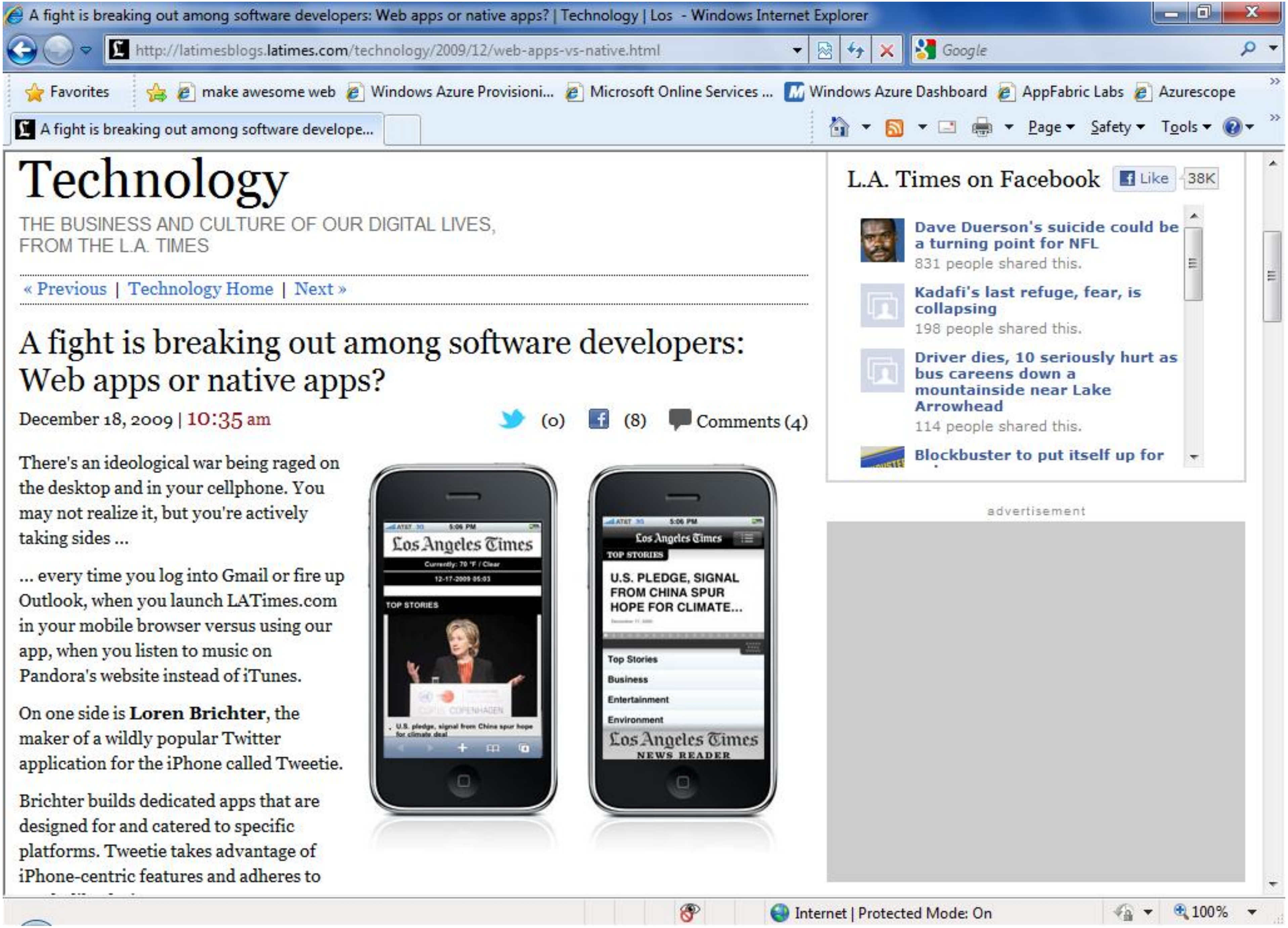
- **Mobile devices are modern „Services visualization UIs“**
  - async invocation of (web) services
- **Need to integrate with**
  - Internet services
  - enterprise-deployed services
- **Need to leverage both services styles**
  - SOAP (operation-driven)
  - REST (resource-driven)
- **MonoTouch supports**
  - ASMX web service clients
  - basic WCF client-side
  - REST client through `System.Net`
  - several serializers
  - OData through sample code

Write once... ?

- **Idea of having one codebase for all kinds of devices**
  - dream on...
  - device platform concepts too different
- **Encapsulate non-device & non-UI related code into class libraries**
- **Build device-specific UI on top with MVC approach**



HTML 5



A fight is breaking out among software developers: Web apps or native apps? | Technology | Los - Windows Internet Explorer

http://latimesblogs.latimes.com/technology/2009/12/web-apps-vs-native.html

Google
Favorites
make awesome web
Windows Azure Provisioni...
Microsoft Online Services ...
Windows Azure Dashboard
AppFabric Labs
Azurescope
A fight is breaking out among software develope...
Page
Safety
Tools

Technology

THE BUSINESS AND CULTURE OF OUR DIGITAL LIVES, FROM THE L.A. TIMES

« Previous | Technology Home | Next »

A fight is breaking out among software developers: Web apps or native apps?

December 18, 2009 | 10:35 am

There's an ideological war being raged on the desktop and in your cellphone. You may not realize it, but you're actively taking sides ...

... every time you log into Gmail or fire up Outlook, when you launch LATimes.com in your mobile browser versus using our app, when you listen to music on Pandora's website instead of iTunes.

On one side is **Loren Brichter**, the maker of a wildly popular Twitter application for the iPhone called Tweetie.

Brichter builds dedicated apps that are designed for and catered to specific platforms. Tweetie takes advantage of iPhone-centric features and adheres to

L.A. Times on Facebook

Dave Duerson's suicide could be a turning point for NFL
831 people shared this.

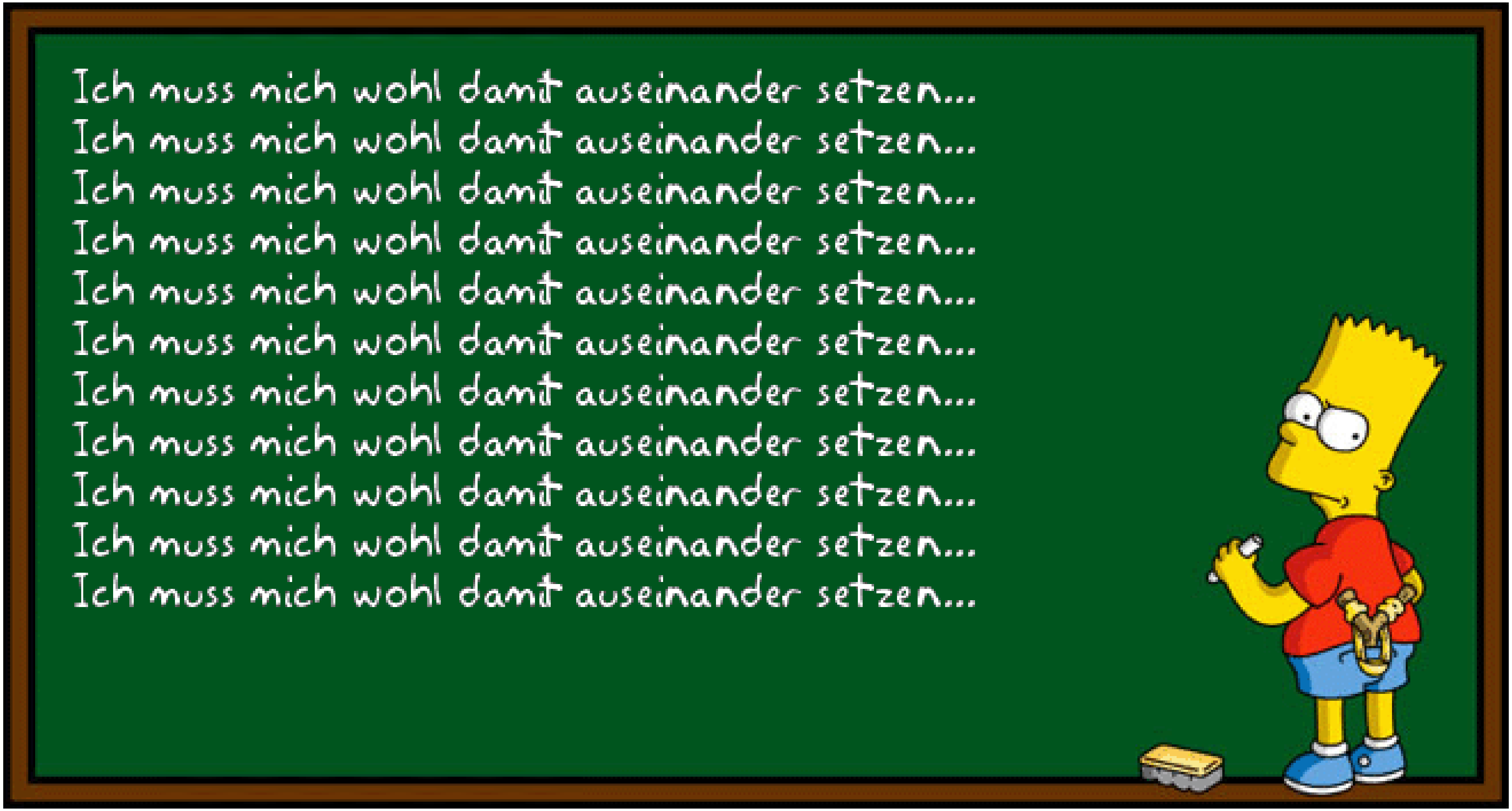
Kadafi's last refuge, fear, is collapsing
198 people shared this.

Driver dies, 10 seriously hurt as bus careens down a mountainside near Lake Arrowhead
114 people shared this.

Blockbuster to put itself up for

advertisement

What would Bart do?


Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...
Ich muss mich wohl damit auseinander setzen...

Motivation

- **Which platform will succeed?**
- **Which paradigm will succeed?**
- **... maybe a mixture of several approaches ...**

- **We as .NET developers may need to rethink certain things**
- **Let‘s re-grab our web knowledge**
  - if we ever had any...
- **HTML5 & Co. is early, but it is here and will mature**

HTML5 101

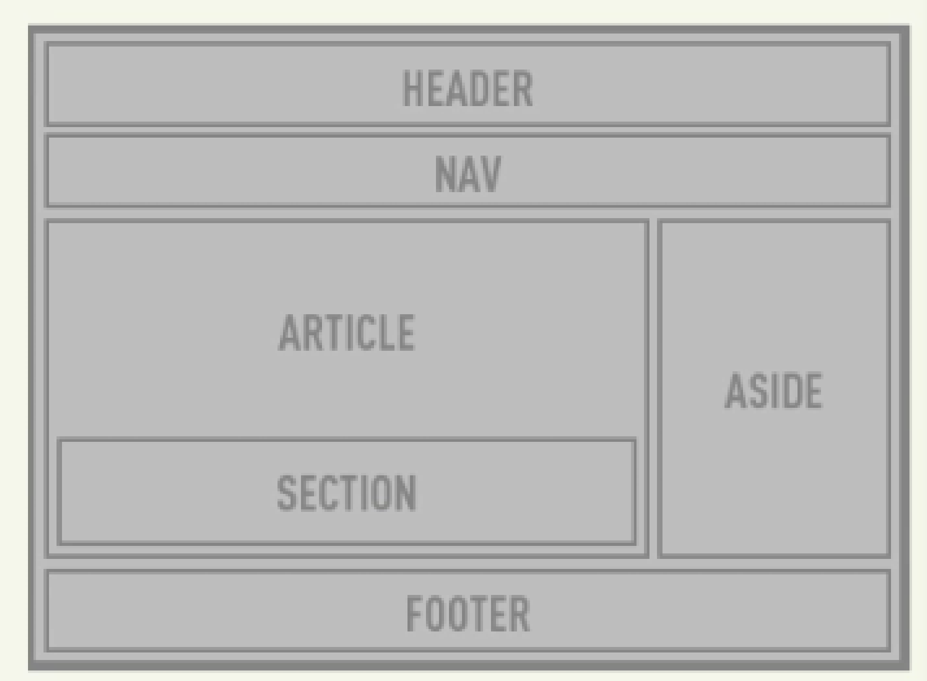
- **Cooperation between World Wide Web Consortium (W3C) and Web Hypertext Application Technology Working Group (WHATWG)**
  - WHATWG was working with web forms and applications
  - W3C was working with XHTML 2.0
  - In 2006, they decided to cooperate and create a new version of HTML
- **Some rules for HTML5 were established**
  - new features should be based on HTML, CSS, DOM, and JavaScript
  - reduce the need for external plugins (like Flash)
  - better error handling
  - more markup to replace scripting
  - HTML5 should be device independent: ‘detect features, not browsers’
  - development process should be visible to the public

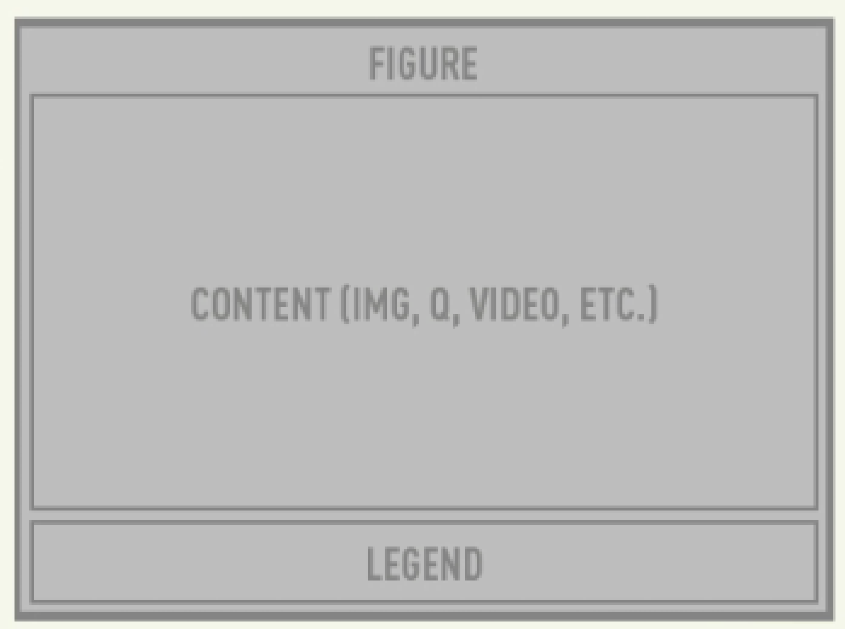
HTML5 101

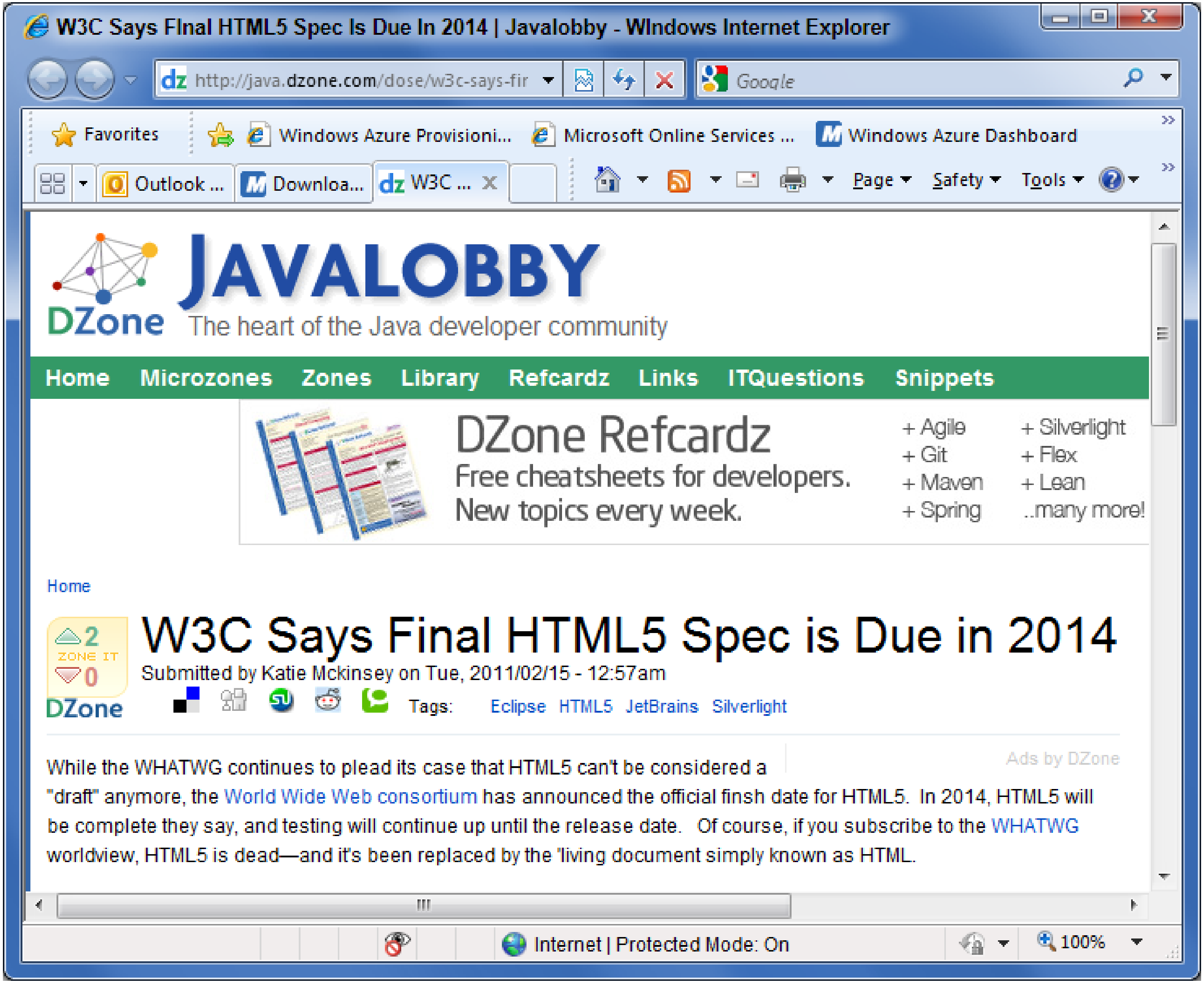
- **No time & space here for HTML5 intro**
- **HTML5 moniker stands for variety of technologies**
  - HTML new tag support
  - CSS3 styling
  - JavaScript API’s to support both
- **Simplification of things**
- **Seperation of content from design**
- **Interoperability**
- **Platform rather than just a markup language**

- **Browser support details: http://www.findmebyip.com/litmus/**

Semantic document structure






W3C Says Final HTML5 Spec Is Due In 2014 | Javalobby - Windows Internet Explorer
http://java.dzone.com/dose/w3c-says-fir
Google
Favorites
Windows Azure Provisioni...
Microsoft Online Services ...
Windows Azure Dashboard
Outlook ...
Downloa...
W3C ...
Page
Safety
Tools
JAVALOBBY
DZone
The heart of the Java developer community
Home
Microzones
Zones
Library
Refcardz
Links
ITQuestions
Snippets
DZone Refcardz
Free cheatsheets for developers.
New topics every week.
+ Agile
+ Git
+ Maven
+ Spring
+ Silverlight
+ Flex
+ Lean
..many more!
Internet | Protected Mode: On
100%

Home

W3C Says Final HTML5 Spec is Due in 2014

2 ZONE IT 0 DZone

Submitted by Katie Mckinsey on Tue, 2011/02/15 - 12:57am

Tags: Eclipse HTML5 JetBrains Silverlight

Ads by DZone

While the WHATWG continues to plead its case that HTML5 can't be considered a "draft" anymore, the World Wide Web consortium has announced the official finsh date for HTML5. In 2014, HTML5 will be complete they say, and testing will continue up until the release date. Of course, if you subscribe to the WHATWG worldview, HTML5 is dead—and it's been replaced by the 'living document simply known as HTML.

WebKit-based browsers

- HTML5 + CSS3
- Transforms, transitions, and animations
- Canvas
- SQLite
- Cache manifest
- Geo-Location

jQuery Mobile

- **Touch-optimized HTML5 framework for building mobile web sites and apps**
  - based on successful jQuery
- **Currently in Alpha**
  - not all features are implemented or stable

- **Features**
  - pages & dialogs
  - toolbars (header & footer bars)
  - buttons (including a set of stock icons)
  - form controls (sliders, toggles, enhanced radio, checkbox etc)
  - list view control

jQuery Mobile – first steps

```
<!doctype html>
<html>
<head>
    <title>jQuery Mobile Example</title>
    <link rel="stylesheet"
      href="jquery.mobile-1.0a2/jquery.mobile-1.0a2.css" />
    <script src="jquery-1.4.4.min.js"></script>
    <script
      src="jquery.mobile-1.0a2/jquery.mobile-1.0a2.js"></script>
</head>
<body>
</body>
</html>
```

- **HTML5 Doctype**
- **jQuery Mobile CSS**
- **jQuery Core JS**
- **jQuery Mobile JS**

jQuery Mobile data roles

- **jQuery Mobile uses HTML5 attribute data-role**
  - associate an element with a widget
- **For example:**
  - data-role=”page”
  - data-role=”header”
  - data-role=”footer”
  - data-role=”navbar”
  - data-role=”button”
  - data-role=”listview”
  - data-role=”controlgroup”
  - data-role=”fieldcontain”

jQuery Mobile pages

- **HTML document can consist of multiple pages**
  - linked together via a link to #pageElementID

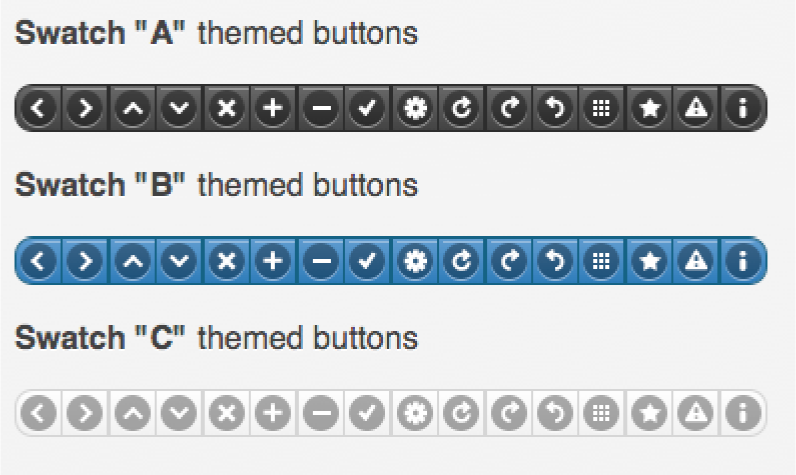
```
<div data-role="page" id="home">
    <div data-role="header">
        <h1>Home</h1>
        <a href="#settings" data-role="button">Settings</a>
    </div>
</div>
<div data-role="page" id="settings" data-theme="b">
    <div data-role="header">
        <h1>Settings</h1>
    </div>
    <div data-role="content">
        Some content...
    </div>
</div>
```

jQuery Mobile themes

- **Several color schemes which can be controlled using the data-theme attribute**
- **Can specify a data-theme on a page, buttons, toolbar’s, etc.**
  - try specifying data-theme=a,b,c,d,e

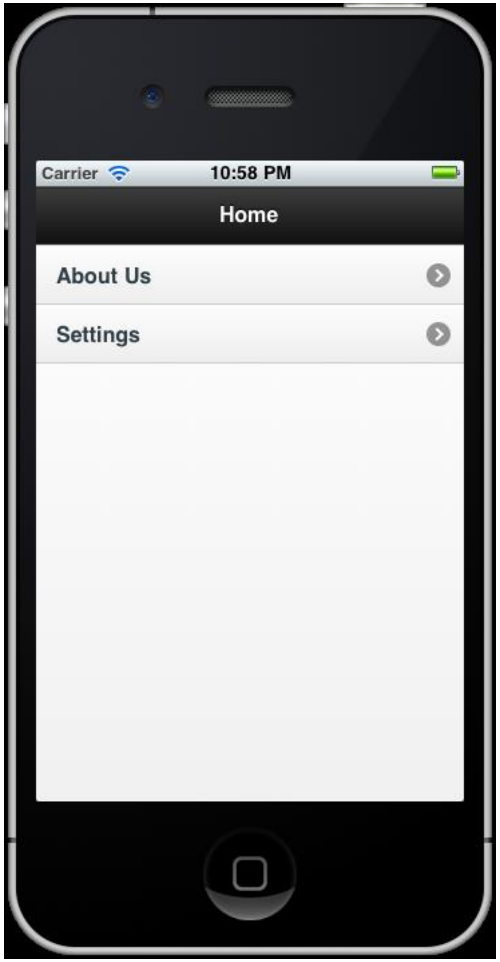
jQuery Mobile buttons

- **Can create a button by adding data-role=”button” to a *button* tag, an *a* tag, or *input type=submit|button|reset|image***
- **Add an icon to the button using data-icon=”icon-name”**
  - by default icon goes on the left
  - can put it on the right, top or bottom by specifying data-iconpos
  - just use the icon use data-iconpos=”notext”
- **Comes with several icons**
  - you can use your own custom icons easily


Swatch "A" themed buttons
Swatch "B" themed buttons
Swatch "C" themed buttons

jQuery Mobile list view

```
<ul data-role="listview" data-theme="c">
    <li><a href="#about">About Us</a></li>
    <li><a href="#settings">Settings</a></li>
</ul>
```



Sencha Touch

- **HTML/JavaScript framework for mobile apps**
  - built on web standards
  - currently targets WebKit
  - abstracted for performance/ease
- **Benefits**
  - cross-platform
  - faster, cheaper, easier (if you know your stuff)
  - styling with CSS3
  - flexible deployment
- **Device support**
  - iOS
  - Android
  - soon: Blackberry

Sencha Touch features

- **Touch abstraction**
- **Scroller**
- **Orientation events**
- **Data**
- **Layouts**
- **Animations**
- **Theming & icons**
- **Components**
- **MVC support**

Sencha Touch features

- **Touch Events**
  - built on native events
  - abstracted for performance
  - additional events

  - Tap
  - Double tap
  - Tap and hold
  - Swipe
  - Rotate
  - Drag & drop

- **Scrolling**
  - momentum/bounce physics
  - hardware accelerated
  - throughout components

  - Lists
  - Carousel
  - Pickers

Sencha Touch data

- **Models, stores and proxies**
  - associations
  - validation
- **Easily consume (web) services**
  - JSON/P
  - XML
  - YQL

Sencha Touch components

- **Lists**
  - nested, grouped, sortable
- **Carousel**
- **Picker**
- **Overlay**
- **Slider**
- **Forms & fields**
- **Toolbars & buttons**
- **HTML5**
  - audio
  - video
  - GeoLocation

Sencha Touch theming

- **CSS3**
- **SASS & Compass**
  - flexible themes
  - highly optimized
- **300+ pre-included icons**
- **Robust animations**
- **Resolution independent**

Offline web applications

- **One step closer to ‚real‘ apps**

- **HTML5 Offline web application**
  - list of URLs — HTML, CSS, JavaScript, images, or any other kind of resource
  - home page of the offline web application points to this list, called a manifest file, which is just a text file located elsewhere on the web server
- **Web browsers that implement HTML5 offline applications**
  - read the list of URLs from the manifest file
  - download the resources
  - cache them locally
  - automatically keep the local copies up to date as they change

Offline web applications

- **When the time comes that you try to access the web application without a network connection, your web browser will automatically switch over to the local copies instead**
- **Browser events when going offline or online**
  - you need to handle this change
- **Storage**
  - local
  - session
  - WebSQL database
  - IndexedDB
  - file API
- **Storage abstraction with e.g. persist-js**

```
CACHE MANIFEST

/main/home
/main/app.js
/settings/home
/settings/app.js
http://img.example.com/logo.png
http://img.example.com/check.png
```

PhoneGap

- **“Bridge the gap!”**
- **Set of templates for building native iOS, Android, Blackberry, Symbian, and WebOS using HTML, CSS, and JavaScript**
  - open source

- **phonegap.js contains a device neutral javascript API for accessing native device API’s, e.g.**
  - camera
  - accelerometer
  - GPS, compass
  - address book / contacts
  - media (audio / video)
  - events
  - … and more…

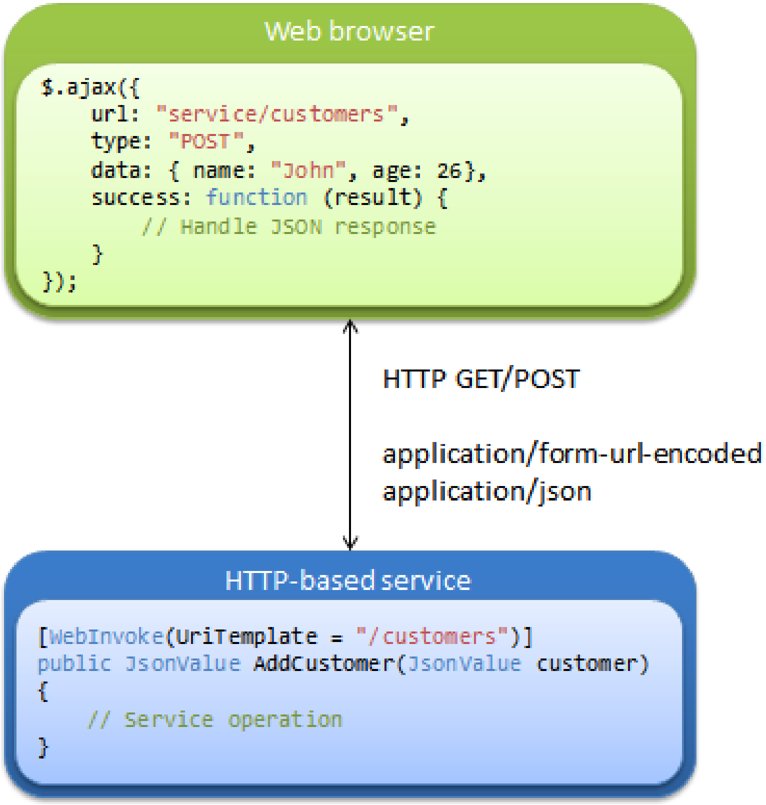
PhoneGap plugins

- **Several plugins for accessing functionality that may be platform specific, e.g.**
  - keychain access (for storing sensitive data on iOS)
  - PayPal API plugin
  - native iPhone controls

- **https://github.com/phonegap/phonegap-plugins**

Communication

- **Mobile apps need to communicate with services**
  - AJAX (maybe encapsulated by e.g. jQuery or some other API)
  - think about cross-site access
- **ASP.NET MVC**
  - controller approach very popular for creating REST services
  - rendering e.g. JSON views
- **WCF**
  - today we have simple web programming functionality
  - in vNext WCF will offer an all new and shining web-enabled programming model

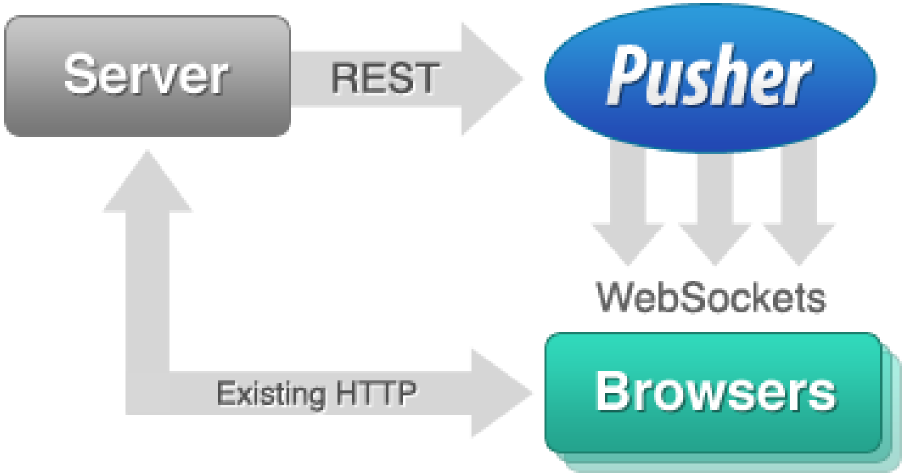


```
$.ajax({
    url: "service/customers",
    type: "POST",
    data: { name: "John", age: 26},
    success: function (result) {
        // Handle JSON response
    }
});
```

```
[WebInvoke(UriTemplate = "/customers")]
public JsonValue AddCustomer(JsonValue customer)
{
    // Service operation
}
```

Advanced communication

- **Advanced communication patterns like push**
  - Comet-style (reverse AJAX) with long polling
  - HTML5 WebSockets
- **Can leverage the Cloud for push-style notifications**
- **PusherApp.com cloud service**
  - exposes public REST API
  - offers JavaScript & .NET libraries (and more)



Cloud

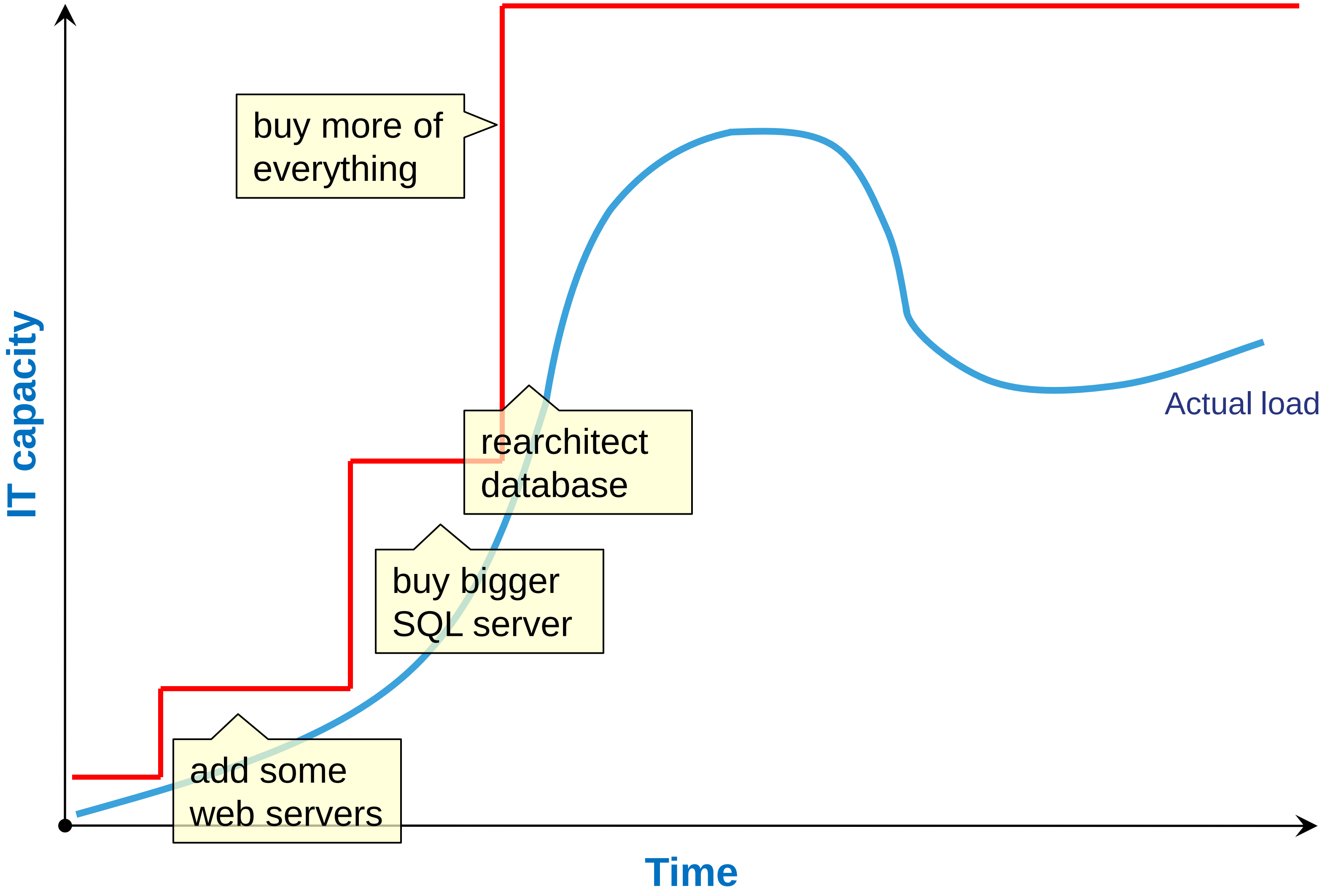
What would Bart do?

Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?
Cloud & Azure - des brauch ich doch nicht! Oder..?

Motivation

- **What is the status quo of today's software landscape?**
  - architecture
  - deployment, maintenance
  - lifecycle
  - integration
- **Can we do better under certain circumstances?**
  - separating concerns
  - enforce architectural practices
  - manageable lifecycle
  - predictable costs
- **What is the Cloud and how can it help?**
  - motivations for going into and leveraging the Cloud
  - enabling & controlling ‘ilities’
- **Is Windows Azure platform the answer to all Cloud-iness?**

Need for dynamic scale-out



Cloud computing

- **Umbrella term and concept unifying different ideas**
  - "Dynamic IT", “On-Demand", "Utility Computing", "Software-as-a-Service“, “Software + Services”, “Cloud Services”, "Virtualization“
- **Promised advantages**
  - reduce capital & operations costs
  - lower capital lockup and usage-bound billing
  - cost effective handling of peeks
  - simplify application deployment & management
  - always on
  - simplify scaling to possible Internet scale
  - focus on new features & functionality, not infrastructure
- **Vision: "IT like power from the socket“**

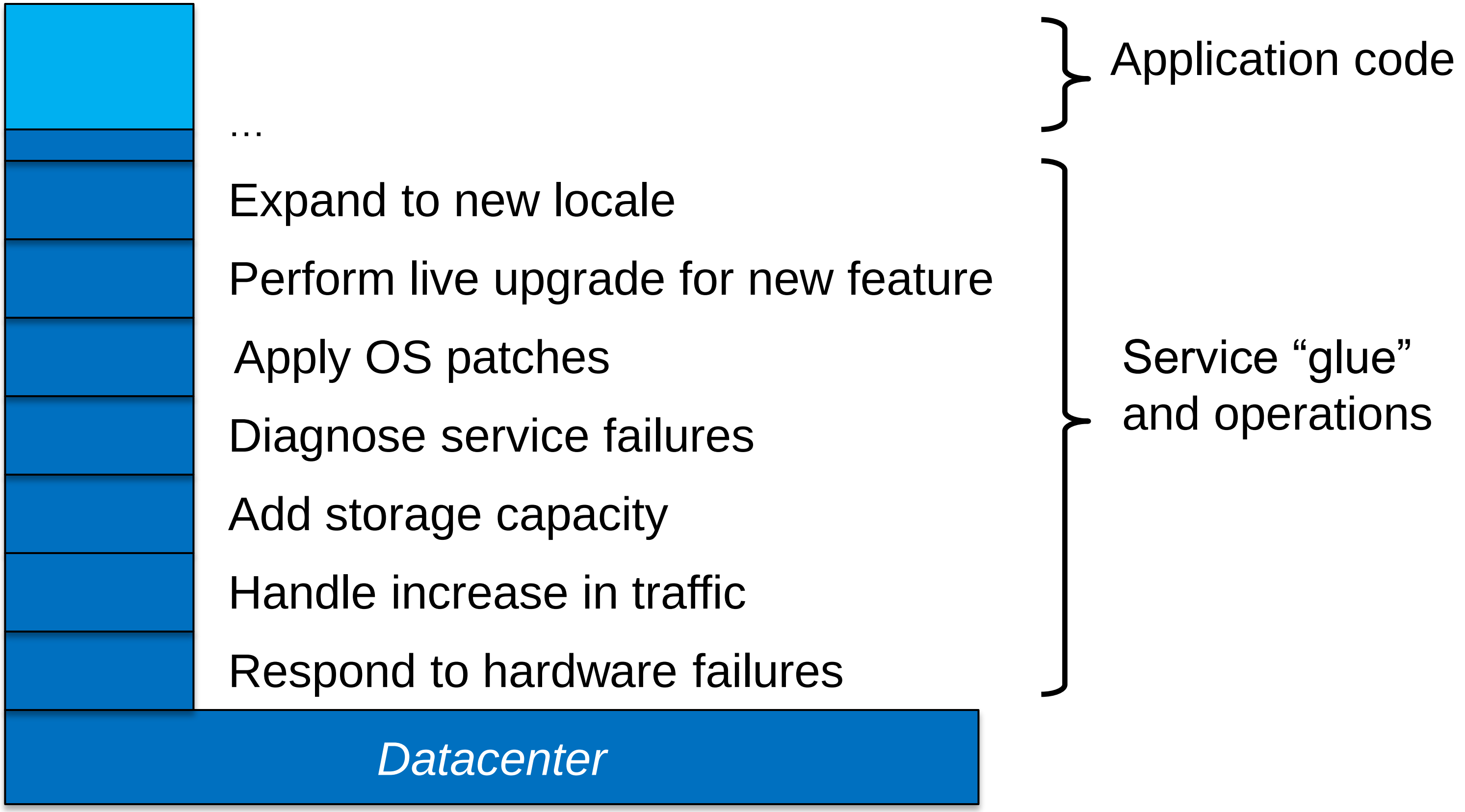
Cloud computing II

- **Cloud computing seems more than just a hype topic**
  - cloud services will have their place in the „IT sourcing mix“
  - each shop needs to find its own good balance of leveraging in-house IT, outsourcing and cloud services
- **Users are sceptical**
  - what happens with my data?
  - is my data stored safely and securely?
  - is the service capability and availability warranted?
  - am I really saving costs below the line?
  - can I customize cloud services to my individual needs?
- **Not everybody will need cloud services**
- **Not everybody needing it will immediately jump on the bandwagon – we are at the beginning**

XaaS – Anything as a Service

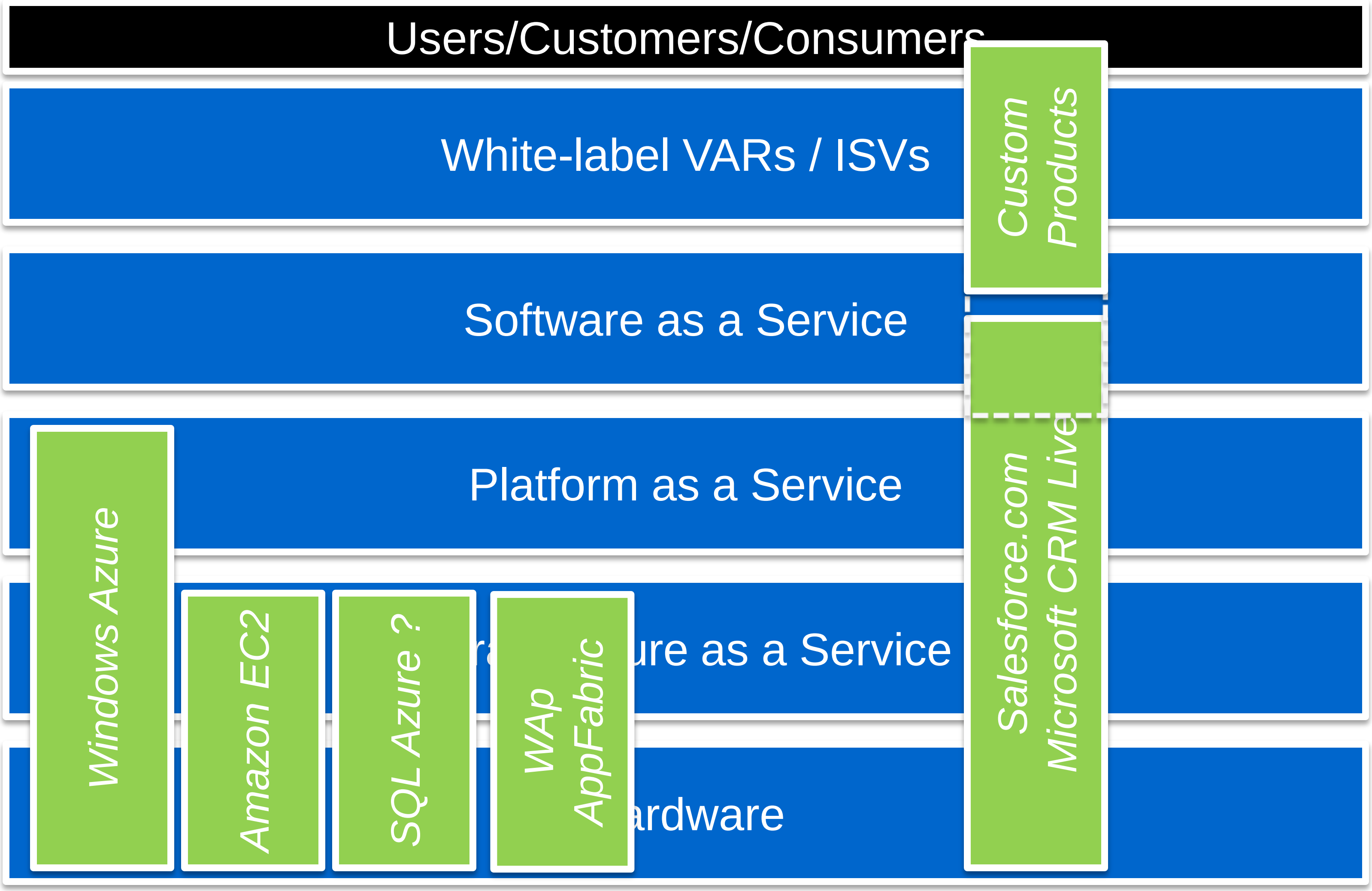
- **Infrastructure-as-a-Service (IaaS)**
  - organization outsources equipment used to support operations (storage, hardware, servers and networking components)
  - service provider owns equipment and is responsible for housing, running and maintaining it
- **Platform-as-a-Service (PaaS)**
  - paradigm for delivering operating systems and associated services over the Internet without downloads or installation
- **Software-as-a-Service (SaaS)**
  - software distribution model in which applications are hosted by a vendor or service provider
  - made available to customers over the Internet

Cloud operating system (PaaS)



Application code
…
Expand to new locale
Perform live upgrade for new feature
Apply OS patches
Diagnose service failures
Add storage capacity
Handle increase in traffic
Respond to hardware failures
Service “glue”
and operations
Datacenter

Cloud stack



Users/Customers/Consumers

White-label VARs / ISVs

Software as a Service

Platform as a Service

ure as a Service

ardware

Windows Azure

Amazon EC2

SQL Azure ?

WAp
AppFabric

Custom
Products

Salesforce.com
Microsoft CRM Live

Windows Azure platform

- **An end-to-end cloud platform**
  - cloud OS
  - platform services
  - infrastructure services
  - tooling
- **Four critical concepts**
  - roles
  - storage
  - messages
  - connectivity

- **Web of geographically distributed data centers**
- **Approached and administered through web portals & APIs**
- **Integrated purchasing and billing platform**

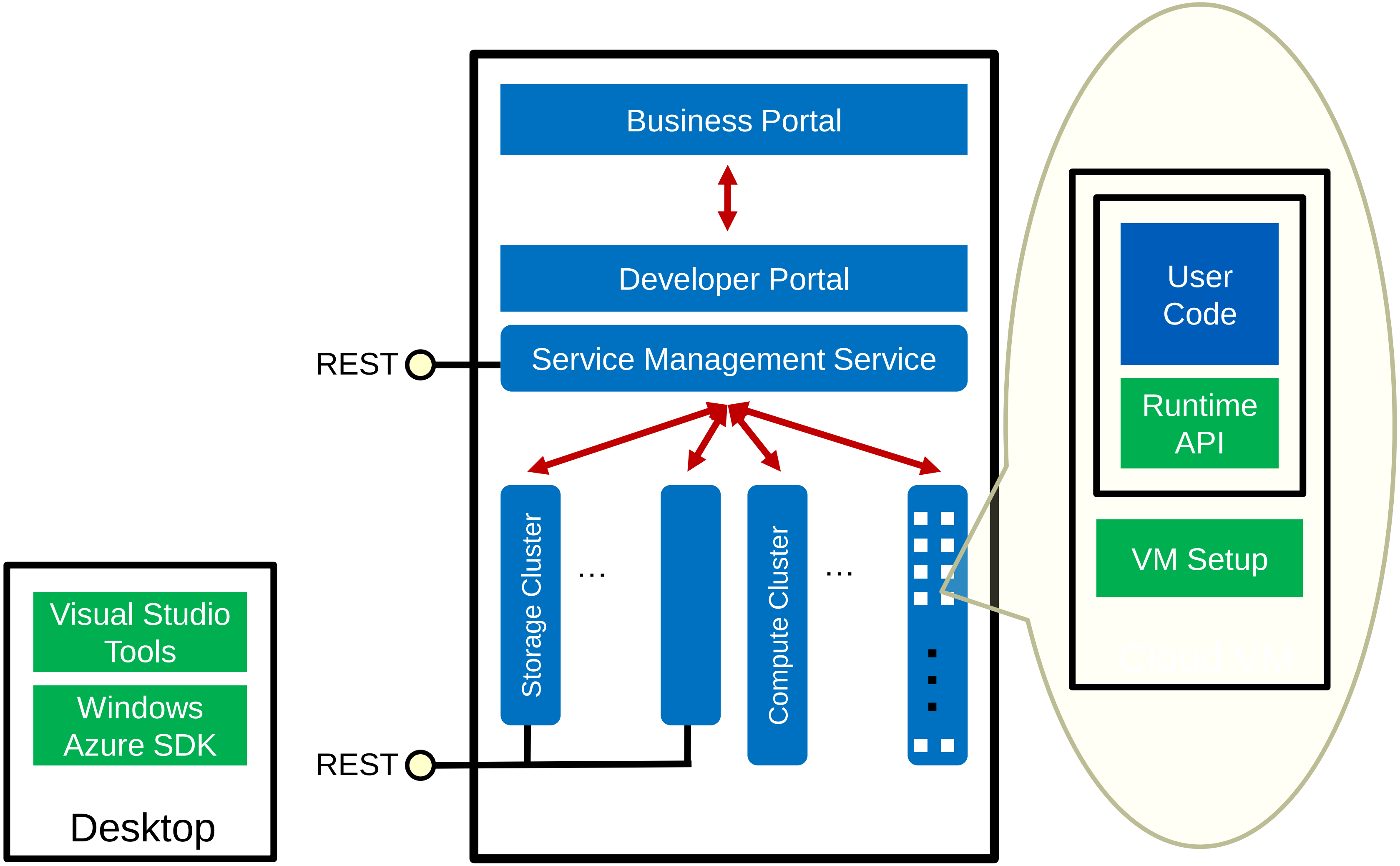
Windows Azure platform parts

- **Windows Azure**
  - compute & storage
  - local and cloud fabric
  - local and cloud storage
- **Windows Azure AppFabric**
  - Access Control Service
  - Service Bus
- **SQL Azure**

Windows Azure

- **Windows Azure is analogous to an OS for the cloud**
- **Windows Azure provides**
  - application hosting
  - automated service management
  - durable storage at massive scale
- **Runs in a virtualized environment**
  - VMs are provisioned in an automated process

Windows Azure high level view



Windows Azure Compute

- **App-centric development and execution model**
- **Applications can be**
  - fault-tolerant
  - highly available
  - highly scalable
- **Application (aka service) requirements modelled through DSL**
  - roles
  - instances
  - interfaces
- **Provides elasticity in compute**
- **Monitoring and management built-in**
- **Different VM sizes available**

Windows Azure Storage

- **Table, Blob and Queue storage capabilities**
- **Data can be**
  - fault-tolerant
  - highly available
  - highly scalable
- **Goal is having data close to applications**
- **Independently accessible**
  - can be used from any platform, on-premise or cloud-based
- **Independently scalable**
  - does not depend on Windows Azure compute
- **Partitions are key concept for scalability**
- **Runs on Windows Azure compute since ever**

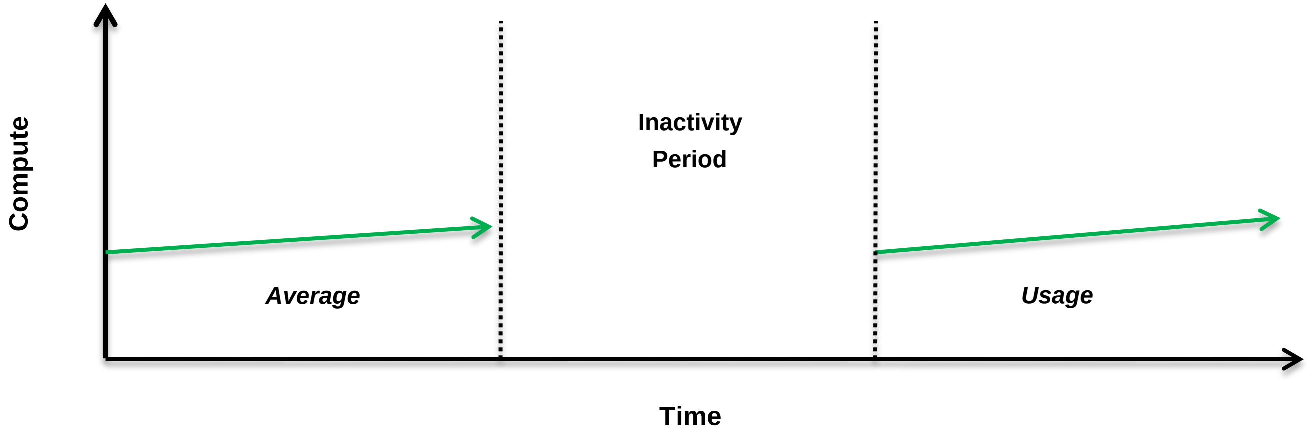
Scenarios for Windows Azure

- **Windows Azure is not just your common Windows Server in the cloud**
- **Windows Azure is not a replacement for your common hosting provider**
- **Windows Azure is a perfect fit for solving certain scenarios in the cloud**
  - on and off
  - growing fast
  - unpredictable bursts
  - predictable bursts

Scenario: On and Off

- **Only needed in certain situations (e.g. batch jobs)**
- **Operating party reserves too much resources & capacity**
- **Provisioning can be cumbersome**
- **Examples**
  - payroll accounting, payment cycles, reporting, simulations

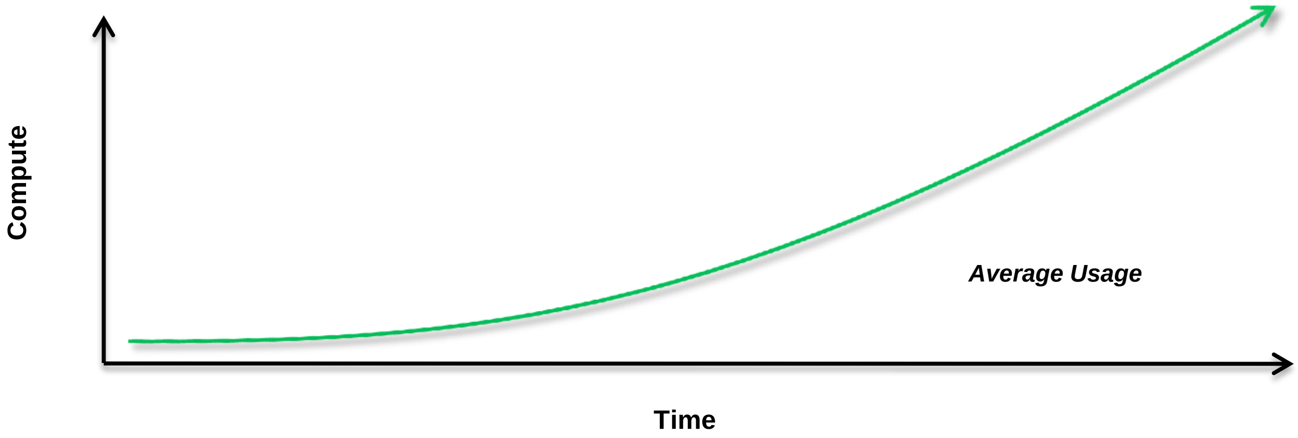
On and Off



Scenario: Growing fast

- **Successful services need to scale**
  - big challenge
  - hard to pre-plan
- **Examples**
  - social games, viral services like Twitter, Facebook, consumer apps

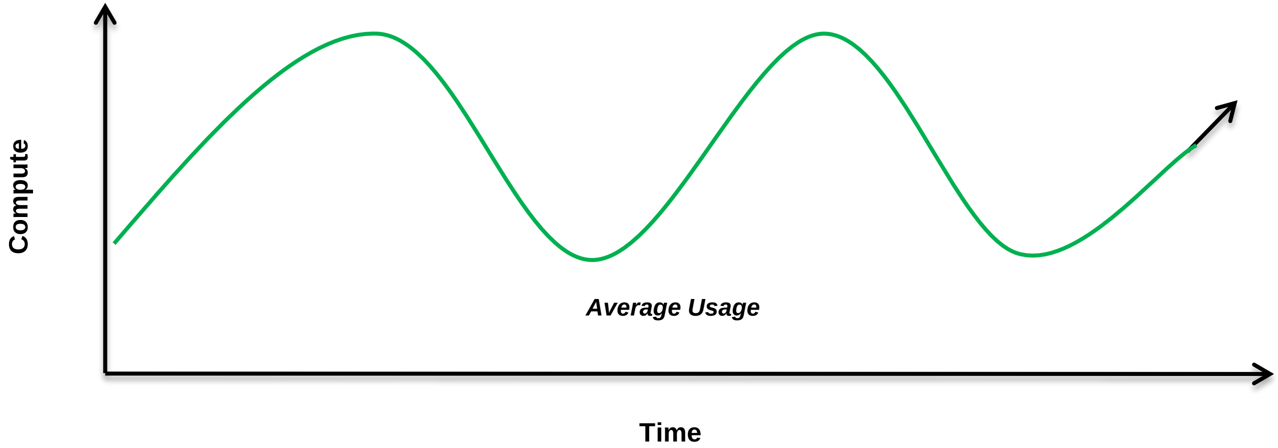
Growing Fast



Scenario: Predictable bursting

- **Services with seasonal bursts**
- **Peaks predictable, can be planned**
- **High IT complexity and low efficiency**
- **Examples**
  - online shops, order systems, Amazon, world cup information systems, live streams

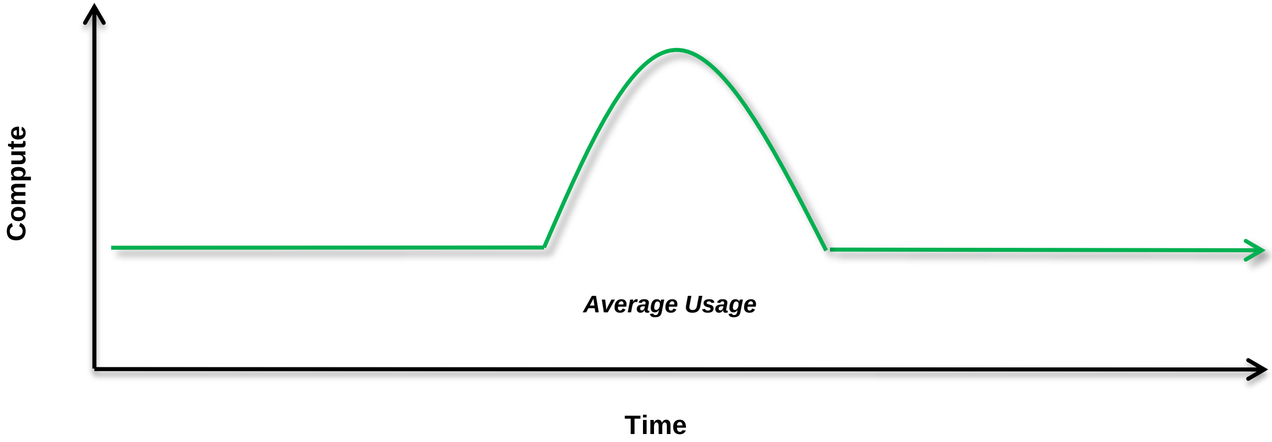
Predictable Bursting



Scenario: Unpredictable bursting

- **Sudden performance degradation**
- **Unknown extremal values**
- **Examples**
  - stock trading systems, news portals, search engines

Unpredictable Bursting



SQL Azure

- **Database as a service**
- **Relational database management system in the Cloud**
- **Compatible with known SQL Server tooling**
- **Not a full-blown SQL Server instance**
  - only the core database engine
- **No full support for all database features and T-SQL**

Identity management

- **Various identity silos**
- **Various ways of authenticating users**
- **Every application needs to implement authentication and authorization itself**
  - essential piece for application's functionality
  - repetitive, error-prone task
- **Need to factor out authentication and authorization**
  - have a separate role in the architecture, a security token service
- **Modern applications also need personalization support**
- **Claims-based security can fulfil all these needs**

Windows Azure AppFabric

- **Access Control**
  - service for issuing access tokens based on authorization rules
  - resource STS in the cloud
  - can be federated with different identity providers, e.g. Active Directory, Windows Live
  - REST-based programming interface
  - used by the Service Bus
- **Service Bus**
  - application messaging bus infrastructure
  - enables application integration beyond physical boundaries
  - implements open format and protocols
  - supports REST and WS-*
  - uses Access Control to allow sending messages to and listening on endpoints

Design for the Cloud

- **Systems need to be designed for the cloud to get the most out of it**
  - you can just move applications to the Cloud, but they may just not be a big success
- **Windows Azure platform components offer powerful features**
  - Azure services implement various patterns on their own
  - developer needs to opt-in to developing based on proved patterns
- **The Cloud (and Windows Azure) enforces certain architectural patterns**

Cloud-related patterns

- **Separation of concerns**
- **Layering**
- **Provider**
- **Scale-out**
- **Elasticity**
- **Asynchronous processing**
- **Idempotency**
- **MapReduce**
- **Multi tenancy**
- **Hybrid applications**
- **Caching**
- **Sharding**

Cloud-related anti-patterns

- **Session state**
- **Assuming single instance in general**
- **2PC transactions**

Billing

- **Five things to bill on**
  - compute time/hours
  - bandwidth
  - storage
  - activities/transactions
  - connections
- **Three ways to bill**
  - pay as you go
  - subscription
  - volume licensing
- **Predictability through up-front modelling of costs**
  - based on customer's own estimations

Windows Azure Platform Pricing - I

- **Windows Azure Compute VM pricing**
  - CPU cost is allocated when the instance is deployed

	X-Small	Small	Medium	Large	X-Large
Service hour	0.05 $	0.12 $	0.24 $	0.48 $	0.96$
Processor	1x1,0 GHz	1x1,6 GHz	2x1,6 GHz	4x1,6 GHz	8x1,6 GHz
RAM	768 MB	1,75 GB	3,5 GB	7,0 GB	14,0 GB
Local storage	20 GB	225 GB	490 GB	1000 GB	2040GB
Peak Mbps	5	100	200	400	800

- **Data transfer outside of datacenter**

Direction	North America	Europe	Asia Pacific
Ingress	0.10 $ / GB	0.10 $ / GB	0.30 $ / GB
Egress	0.30 $ / GB	0.30 $ / GB	0.45 $ / GB

Pricing II

- **Other services**

Service	Unit	Cost
Windows Azure Storage	Data & transactions	0.15 $ / GB/month 0.01 $ / 10K transactions

SQL Azure	Database	Web edition (0-5GB):
		9.99 – 49.95 $/month
		Business edition (0-50GB):
		99.99 – 499.95 $/month

Access Control	Transactions	1.99 $ / 100K transactions
Service Bus	Connections	3.99 $ / connection/month*

- **Current pricing targeted rather at professional cloud services for the enterprise**
  - not at garage builders, blog owners, pizza shop
- **Several promotions available**
- **It is all about TCO & ROI**

SLAs

Service	Description	Service level
Compute connectivity	• Your service is connected & reachable via web • Internet facing roles will have external connectivity	>99.95%
Instance monitoring & restart	• All running roles will be continuously monitored • Unhealthy roles detected within 2 minutes, and corrective action initiated	>99.9%
Storage availability	• Storage service will be available/ reachable (connectivity) • Your storage requests will be processed successfully	>99.9%
Database availability	• Database is connected to the internet gateway • All databases will be continuously monitored	>99.9%
Service Bus & Access Control availability	• Service bus & access control endpoints will have external connectivity • Message operation requests processed successfully	>99.9%

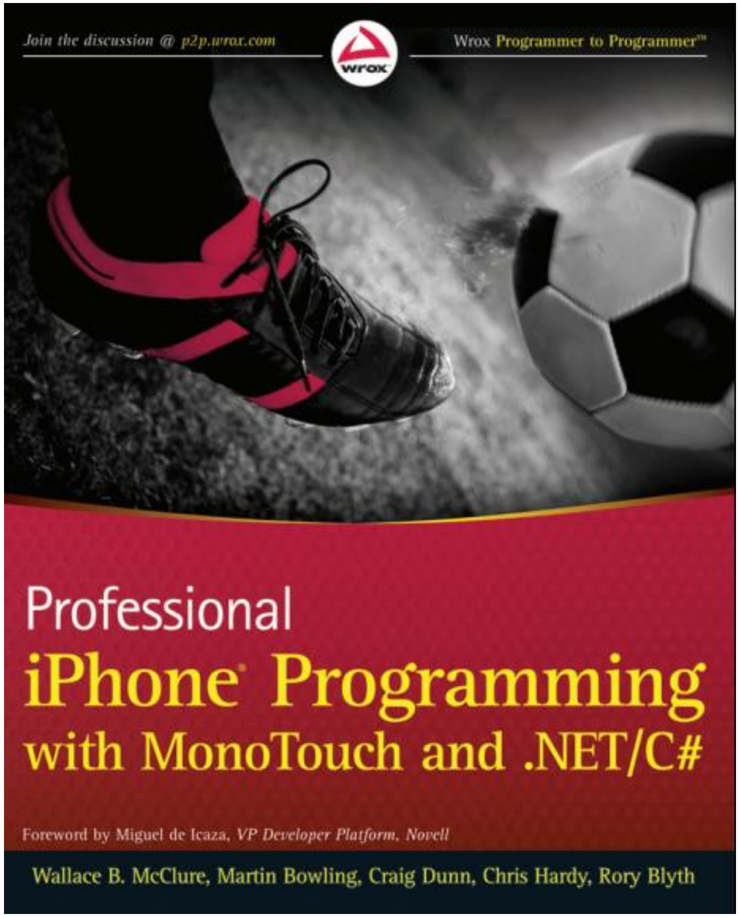
Summary

- **Cloud computing can solve certain problems**
  - number of architectural and operational headaches easier to solve in the Cloud
  - not everybody will need the Cloud, though
- **Utility computing on a pay-as-you-go base**
- **Windows Azure is an OS for the Cloud**
- **Windows Azure platform is a platform**
  - you choose the parts and services you will need
  - you determine the degree of cloudiness
- **Designing for the Cloud adds real value**
- **Windows Azure platform is not done, new services will emerge**

Resources

- **http://blogs.thinktecture.com/cweyer**
- **christian.weyer@thinktecture.com**

- **iOS Developer Center**
  - http://developer.apple.com/devcenter/ios/index.action
- **MonoTouch**
  - http://monotouch.net
- **MonoTouch.Dialog**
  - http://github.com/migueldeicaza/MonoTouch.Dialog


Join the discussion @ p2p.wrox.com
wrox
Wrox Programmer to Programmer™
Professional
iPhone Programming
with MonoTouch and .NET/C#
Foreword by Miguel de Icaza, VP Developer Platform, Novell
Wallace B. McClure, Martin Bowling, Craig Dunn, Chris Hardy, Rory Blyth

Resources

- **Android**
  - http://developer.android.com
- **Mono**
  - http://mono-project.com
- **MonoDroid**
  - http://monodroid.net/
- **MonoDroid samples**
  - https://github.com/mono/monodroid-samples
- **DroidDraw**
  - http://www.droiddraw.org/

Resources

- **HTML5**
  - http://www.w3.org/html/wg/html5/
  - http://diveintohtml5.org/
  - http://html5demos.com/
  - http://www.html5rocks.com/
- **jQuery Mobile**
  - http://jquerymobile.com/
  - http://jquerymobile.com/demos/1.0a3/
- **SenchaTouch**
  - http://www.sencha.com/products/touch/
  - http://www.sencha.com/learn/Sencha_Touch

Resources

- **WidgetBox**
  - http://www.widgetbox.com/
- **PhoneGap**
  - http://www.phonegap.com/
- **linq.js - LINQ for JavaScript**
  - http://linqjs.codeplex.com/
- **PersistJS**
  - https://github.com/jeremydurham/persist-js
- **WCF Community Site**
  - http://wcf.codeplex.com
- **Pusher**
  - http://pusherapp.com
- **SASS**
  - http://sass-lang.com/
- **Compass**
  - http://compass-style.org/