Backend as a Service (BaaS)

Seminar TIE-11406, Fall 2013
Tampere University of Technology

Kari Systä, Antero Taivalsaari

Motivation and Background

• The landscape of mobile software development has changed considerably in the past ten years.

• Today’s mobile applications are rarely standalone; rather, application must commonly
  1. talk with various third party data services,
  2. work on a number of different platforms, and
  3. stay in sync across a number of devices that the user has.

• The dependence on various backend features has led to the emergence of *Backend as a Service (BaaS)* systems.
Practical Arrangements
Practical Arrangements

• The seminar will be held on **Tuesdays in Tietotalo TB222, 14:15 – 15:45**.

• Seminar is based on presentations given by participants.

• Active participation expected.

• **Getting volunteer presenters for next weeks is essential for the overall success of the seminar.**

• Dates are flexible at this point.
How to Get Credit Points (Opintopisteet)?

- **Number of credit points**: 1-5 op
- **Regular attendance**: 1 op
- **Seminar presentation on selected BaaS technology**: 2 op
- **Successfully written, new demo application using selected BaaS technology**: 2 op
Choosing Presentation & Demo Topics

• Please choose your presentation topic, possible demo application and preferred presentation date(s) as soon as possible.

• Send e-mail to: kari.systa@tut.fi

• Topics are allocated on a “first-come-first-server” basis.

• Seminar web page will be updated regularly to list the reserved presentation topics:
  • http://www.cs.tut.fi/~taivalsa/kurssit/BaaS2013/
Action Items for Next Week

1. Choose your presentation topic.

2. Optionally or in addition, come up with a great idea for a demo application.

3. Pick a date for your presentation(s).

• Send e-mail to: kari.systa@tut.fi
Backend as a Service

Overview
Backend as a Service (BaaS)

- **Backend as a Service**: a model for providing web and mobile app developers with a way to link their applications to hosted backend cloud storage while also providing features such as user management, push notifications, and integration with social networking services, all as an integrated offering.

- Key goal is to **abstract away all the complexity related to cloud and cloud management**, and provide simple APIs that can be used across all popular mobile platforms.
  - “Don’t worry about the server side, we’ll take care of it for you.”

- This is a very popular area these days; 40+ startup companies in this area have sprung up recently.
<table>
<thead>
<tr>
<th>BaaS Systems / Companies</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.appery.io/">http://www.appery.io/</a></td>
</tr>
<tr>
<td><a href="http://www.cloudmine.me/">http://www.cloudmine.me/</a></td>
</tr>
<tr>
<td><a href="http://www.mobdb.net/">http://www.mobdb.net/</a></td>
</tr>
</tbody>
</table>
Key Selling Points for BaaS Systems

• Writing and hosting a full-fledged cloud backend can be very tedious and expensive.
• Small developers do not have the resources or skills to write or host their own backends anyway.
• Backend offerings from major providers such as Amazon can be unnecessarily complex for small developers.

• Many BaaS systems are very easy to take into use.
  • To get started, often just a line or two of code in your web app is needed; excellent online documentation provided by most systems.

• Part of the value proposition is pricing. Most systems use the freemium model; small-scale use is usually free.
BaaS – Commonly Provided Features

- Data Storage
- Cross-Platform Support
- Push Notifications
- User Management
- 3rd Party Data Integration
- Versioning, analytics, etc.
Data Storage

- Most of the BaaS systems offer a *Data API* for storing application data in the cloud.
- File storage is usually provided as well (for large images, file attachments, etc.)
- Commonly, these APIs are offered in a number of different forms.
  - JavaScript API
  - REST API
  - native APIs for iOS, Android, (sometimes) WP
- The actual APIs / databases are still rather different (JSON or key-value stores, table-oriented DBs, SQL support, etc.).
A push notification API allows users to be instantly notified of data changes in 3rd party services as well as changes across the user’s own devices.

- No battery consuming polling required.

On device side, push support is often limited to iOS and Android devices:
- Google Cloud Messaging (GCM)
- Apple Push Notification (APN)
- Some systems (e.g., QuickBlox) support Microsoft Push Notifications as well
Most of the BaaS systems offer APIs for third-party data integration.

Depending on platform focus, 3rd party support is either consumer-oriented or enterprise-oriented.

**Consumer-oriented:**
- Integration with social networking (Facebook, Twitter, Flickr) and location services (Foursquare, Google Places, etc.)

**Enterprise-oriented:**
- Integration with LDAP, Oracle, PeopleSoft, Salesforce.com, SAP, Siebel, etc.
Cross-Platform Support

• Most of the BaaS backends offer Web APIs (JavaScript, REST) as well as native APIs for iOS and Android.

• Most of the functionality and APIs offered can be used from pure web apps, as well as from iOS and Android applications.

• At the moment, only a few systems provide support for Windows Phone development.
User Management

- User management is needed for multi-device use as well as for third-party data integration.
- BaaS systems typically offer [a subset of] the following mechanisms:
  - OAuth and/or OAuth 2.0, OpenID
  - Google Account, Apple ID, Microsoft Live, Twitter, Facebook logins.
  - Enterprise integration with LDAP.
- For security, the systems rely on common technologies (HTTPS, SSL/TLS); server-side data is usually encrypted.
Additional Features

Versioning, analytics, etc.

Additional differentiating features available in some systems:

1. Data analytics / logging capabilities
2. API / Data versioning support
3. Server-side code execution support
4. Application / plugin marketplace
5. Multi-platform UI generation / UI authoring tools / widget libraries
Broader Classification

**DBaaS**
- Cloud database
- Cloud file storage
- Access control
- Push notifications
- Offline support (sync)

**BaaS**
- Integration to third party services and authentication systems (via OAuth, LDAP)
- Server-side computation
- Analytics & monetization
- Client-server communication
- User management
- Device management

**PaaS**
- All-in-One IDE/SDK for multi-platform development
- Multi-platform UI/application framework
- Multi-platform building and deployment

AWS DynamoDB
Cloudant
Cloudbase
Dropbox
Firebase
...

CloudMine
Buddy
FeedHenry
Kinvey
Parse
...

Appcelerator
Appery.io
ApplicationCraft
Icenium
Sencha.io
...

...
Example Feature Set: Parse

*Parse Data*
Store your app's data in the cloud. No servers necessary.

*Parse Hosting*
A powerful web presence without all the hassle.

*Parse Push*
Creating, scheduling, and segmenting push notifications just got a whole lot easier.

*Parse Social*
Make your app social. Instantly.

*Cloud Code*
Run custom app code in the Parse Cloud. Say goodbye to servers.
Usage Example – Parse Data API

https://parse.com/docs/js_guide

```javascript
var GameScore = Parse.Object.extend("GameScore");
var gameScore = new GameScore();

gameScore.save(
    {
        playerName: 'New Player',
        score: 1000, cheatMode: false
    },
    {
        success: function(gameScore) {
            // Execute any logic that should take place after the object is saved.
        },
        error: function(gameScore, error) {
            // Execute any logic that should take place if the save fails.
            alert('Failed to create new object, error code: ' + error.description);
        }
    });
```
Business Term Example – Parse

**Basic**
Great for developers to get started

- **FREE**
  - Requests: 1 million/month
  - Pushes: 1 million/month
  - Burst Limit: 20/second

**Pro**
For production applications

- **$199 per month**
  - Requests: 15 million/month
  - Pushes: 5 million/month
  - Burst Limit: 40/second

**Enterprise**
The most advanced features at a custom annual price

- Discover all the benefits of the enterprise plan
- Learn more

**Parse Core Platform**
- Collaborators and security features
- Powerful marketing features

**Parse Core Platform**
- Parse Pro features
- Enterprise-grade SLA
- High performance infrastructure
- Dedicated support

Choose plan

30-day Free trial

Choose plan
Business Term Example – Kinvey

<table>
<thead>
<tr>
<th>Building it</th>
<th>Growing it</th>
<th>Scaling it</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>$0</strong> per month</td>
<td><strong>$20</strong> per month</td>
<td><strong>+$7</strong> per month</td>
</tr>
<tr>
<td>0-100 active users</td>
<td>101-1,000 active users</td>
<td>For each additional 1,000 active users (up to 50,000)</td>
</tr>
</tbody>
</table>

Developing on Kinvey is always free. Period.

Once you pass 100 users, you will incur a monthly charge of at least $20 until you remove your app.

Please contact us at (617) 505-4096 or write to us to discuss rates for even bigger scale (more than 50,000 active users).
We are currently preparing a research report on popular BaaS systems.

Survey was started by narrowing down the number of potential systems based on a DeveloperEconomics study:


15 systems chosen for more detailed inspection.
BaaS Landscape – January 2013

**Mindshare**

% of developers using service among developers using mobile Back-end services (n=387, median, weighted)

- 28% Parse
- 11% CloudMine
- 10% sencha.io
- 10% ACS
- 9% mobDB
- 8% Buddy
- 7% StackMob
- 6% Trestle Flurry

**Selection Criteria**

Reasons for selecting a Back-end service (n=384, weighted)

- Availability across platforms: 37%
- Feature set / capabilities: 33%
- Flexibility (e.g. Custom business logic): 30%
- Ease of use & integration: 25%
- Cost: 24%
- Performance: 24%
- Stability: 24%

% of developers using Back-end services on 5 major platforms:

- Android: 15%
- BlackBerry: 9%
- HTML5: 13%
- iOS: 18%
- Windows: 15%

*Source: Developer Economics 2013 | www.DeveloperEconomics.com | January 2013*
BaaS Landscape – May 2013

Percentage of developers using each solution as their primary BaaS (n=259).

Average developer ratings across the 7 criteria rated most important for this type of service (n=195).
## Summary of Most Prominent BaaS Systems

<table>
<thead>
<tr>
<th>Most Popular Systems</th>
<th>Most Recommended Systems</th>
<th>Most Feature-Rich Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Parse</td>
<td>1. Parse</td>
<td>1. Parse</td>
</tr>
<tr>
<td>7. StackMob</td>
<td>7. Feedhenry</td>
<td>7. Apigee</td>
</tr>
<tr>
<td>10. QuickBlox</td>
<td>10. Applicasa</td>
<td>10. QuickBlox</td>
</tr>
</tbody>
</table>
Highlights of Most Popular BaaS Systems

- **Parse**: The most popular BaaS platform for consumer apps. Has a good set of features and a wide range of target platforms, but does not support conflict resolution or enterprise features. Recently acquired by Facebook.

- **Sencha.io**: Originally a HTML5 Touch UI toolkit, later an app framework, and more recently adding BaaS capabilities (“the first mobile HTML5 cloud”). BaaS features limited to push notifications and social networking (Facebook) integration. Own app marketplace supported.

- **CloudMine**: Enterprise-oriented system that provides an impressive set of data connectors for enterprise software, including Oracle, PeopleSoft, SAP and Siebel.
Highlights of Most Popular BaaS Systems

- **Appcelerator ACS**: Uses its own authoring tool (Titanium) and UI language, although app logic is written in JavaScript. Rich SDK with fairly limited core functionality; extendable with various third party plugins. Own app marketplace supported.

- **Kinvey**: Comprehensive platform offering both consumer and enterprise features. Integrates also with Google App Engine.

- **Google Mobile Backend Starter**: A new, minimalistic, generic storage and messaging backend that can be auto-deployed on Google App Engine. A sample Android app that includes a client library for the backend is provided. By extending the sample backend you can access all the Google Cloud APIs.
Highlights of Other BaaS Systems
With Consumer/UI/Gaming Orientation

- **Applicasa**: Mainly oriented towards games; tries to minimize coding, e.g., by providing a drag-and-drop interface for defining database tables. Supports Unity as a separate platform in addition to iOS and Android.

- **QuickBlox**: Similar to Applicasa; oriented towards "thick client, thin server" apps focusing on client-side features. Supports video chat, augmented reality, maps and Unity.

- **Deployd**: An easy-to-use open source Node.js module that makes it easy to write self-hosted web app backends running on MongoDB. There are plans to provide Deployd as a hosted service (currently in alpha).
**Highlights of Other BaaS Systems**

With Enterprise Orientation

- **Apigee**: Enterprise-oriented system that was recently positioned as a BaaS offering; main differentiator is the ability to write server-side API adapters.

- **FeedHenry**: Enterprise-oriented system with well thought-out core features, e.g., data sync can be adapted to other backends. Some features missing, e.g., no built-in social network integration is provided.

- **StackMob**: Comprehensive platform with offline support and synchronization capabilities with custom conflict resolution (iOS only for now), API versioning, and own marketplace for extending platform functionality with 3rd party add-ons.
Highlights of Other BaaS Systems

With Data Analytics Orientation

• **Buddy**: Simple platform with analytics as the main selling point.

• **Flurry**: Purely an advertisement and analytics platform.

• **MobDB**: Storage backend running on AWS that lets you easily define and access database tables. Supports push notifications and analytics. Provides a web interface for backend management.
# Feature Comparison

<table>
<thead>
<tr>
<th>System</th>
<th>Data API</th>
<th>Push notif</th>
<th>User mgmt</th>
<th>3rd party integr.</th>
<th>Analy tics</th>
<th>Server code</th>
<th>Market place</th>
<th>WP support</th>
<th>Differentiators</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apigee</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>(X)</td>
<td>(X)</td>
<td>(X)</td>
<td>Enterprise focus</td>
</tr>
<tr>
<td>Appcelerator</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>(X)</td>
<td>X</td>
<td>(X)</td>
<td>Own UI authoring SDK</td>
</tr>
<tr>
<td>Applicasa</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>(X)</td>
<td>Gaming, analytics focus</td>
</tr>
<tr>
<td>Buddy</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>Analytics focus</td>
</tr>
<tr>
<td>CloudMine</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>(X)</td>
<td>(X)</td>
<td>Enterprise connectors</td>
</tr>
<tr>
<td>Deployd</td>
<td>X</td>
<td>(X)</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>(web)</td>
<td></td>
<td>Open source, self-host</td>
</tr>
<tr>
<td>FeedHenry</td>
<td>X</td>
<td></td>
<td>(X)</td>
<td></td>
<td>X</td>
<td>X</td>
<td>(X)</td>
<td>X</td>
<td>Enterp focus; device APIs</td>
</tr>
<tr>
<td>Flurry</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>Analytics focus</td>
</tr>
<tr>
<td>Google MBS</td>
<td>X</td>
<td>(X)</td>
<td>(X)</td>
<td></td>
<td>(X)</td>
<td>(X)</td>
<td></td>
<td></td>
<td>Android/App Eng focus</td>
</tr>
<tr>
<td>Kinvey</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td>Excellent documentation</td>
</tr>
<tr>
<td>MobDB</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>Storage/analytics focus</td>
</tr>
<tr>
<td>Parse</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td>Excellent documentation</td>
</tr>
<tr>
<td>QuickBlox</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td>X</td>
<td></td>
<td>(X)</td>
<td>X</td>
<td>Chat/video focus</td>
</tr>
<tr>
<td>Sencha.io</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>(X)</td>
<td>X</td>
<td>(X)</td>
<td></td>
<td></td>
<td>Client-side focus</td>
</tr>
<tr>
<td>StackMob</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>API versioning, offline</td>
</tr>
</tbody>
</table>
Database as a Service (DBaaS)

• There are a number of companies that offer subsets of BaaS functionality, usually focusing on storage and push notifications:
  • https://cloudant.com/
  • http://www.cloudbase.io/
  • https://cloud.google.com/products/cloud-storage
  • http://www.dropbox.com/
  • http://aws.amazon.com/simpledb/
  • http://www.firebase.com/

• The basic value proposition is the same – build cloud-oriented applications fast without worrying about backend development or management.
Usage Example – Firebase

```javascript
// Boilerplate
<script src='https://cdn.firebase.com/v0/firebase.js'></script>

// Open database
var db = new Firebase('https://mydb.firebaseapp.com/');

// Write data
db.push({name: "New Player", gameScore:1000 });

// Read data & listen to change notifications
db.on("value", function(data) {
  renderMessage(data.val());
});
```
Summary and Key Takeaways

• Mobile software development is changing (again).
• Developers no longer write applications against a specific mobile platform.
• Rather, they must actively:
  • target number of platforms (iOS, Android, WP, Web),
  • write apps that talk with numerous 3rd party data services,
  • stay in sync across multiple devices.
• BaaS systems aim at providing the necessary backend functionality, and relieving the programmers from having to develop, host or maintain backend functionality.
Summary and Key Takeaways

- There is a lot of competition in this area (40+ companies).
- At the moment, the features and the maturity of the systems still vary considerably.
- Core functionality has started converging (e.g., data APIs, push notifications, 3rd party API integration, platform support, analytics capabilities).
- At the moment, only a few systems support Windows Phone development.
- At least the following systems deserve a closer study: Parse, Kinvey, Deployd, Firebase.
Action Items for Next Week

1. Choose your presentation topic.

2. Optionally or in addition, come up with a great idea for a demo application.

3. Pick a date for your presentation(s).

   • Send e-mail to: kari.systa@tut.fi
1. Introduction
2. Overview of the system and its design principles
3. Small examples or existing demos
4. Walkthrough of a more complete example illustrating the use of the technology
5. Evaluation (benefits, drawbacks, peculiarities, usefulness of the system to relevant problems, possible measurements, ...)
6. Summary
Kiitos!
Thank You!
Backend as a Service – Ecosystem Map