PARSE

BACKEND AS A SERVICE

Ville Seppänen
CONTENTS

- What is Parse?
- Parse Data & Parse Push
- Android Demo
- Other services: Cloud code, Social, Analytics
- Evaluation & Summary
WHAT IS PARSE?
PARSE IS A MBAAS

Parse is a hosted closed-source, proprietary mobile-backend-as-a-service (mBaaS)

It offers BaaS over APIs for several platforms
SERVICE CATALOGUE

- Parse Database
- Parse Push notifications
- Parse User management and Social web integration
- Cloud Code server-side code hosting
- Parse Analytics
SUPPORTS MOST PLATFORMS

Main platforms: Android, iOS, OS X, JavaScript, Windows 8, Windows Phone 8, Unity, REST

3rd party libraries: .NET, Java, Clojure, jQuery, Node, PHP, Python, Objective-C, Qt, Ruby, ...
April 2013: "Facebook bought mobile cloud services company Parse for a reported $85 million"
readwrite.com/2013/04/29/parse-acquisition-makes-its-rivals-very-happy

Even when you are not in Facebook, you are part of Facebook?
SUBSCRIPTION LEVELS

Basic
Great for developers to get started
FREE
Requests: 1 million/month
Pushes: 1 million/month
Burst Limit: 20/second
Parse Core Platform

Pro
For production applications
$199 per month
Requests: 15 million/month
Pushes: 5 million/month
Burst Limit: 40/second
Parse Core Platform
+ Collaborators and security features
+ Powerful marketing features

Enterprise
The most advanced features at a custom annual price
Discover all the benefits of the enterprise plan
Parse Core Platform
+ Parse Pro features
+ Enterprise-grade SLA
+ High performance infrastructure
+ Dedicated support
Upgrade to Enterprise
PARSE RUNS ON AMAZON WEB SERVICES

aws.amazon.com/solutions/case-studies/parse
PARSE SUPPORT

- Getting Started guides, Tutorials and API documentation
- StackOverflow (843 parse.com questions)
- Parse's own "StackOverflow clone" parse.com/help
- Parse Developer Day conference held in September 2013
APP BASICS

- A single Parse developer account can hold many Apps, paying customers can collaborate on an app
- Apps are isolated from each other
- Dashboard for simple management and monitoring
DASHBOARD

Parse

Dashboard
Quickstart
Tutorials
Documentation
Downloads
Help
Pricing
Products

Welcome Ville

Baas
Analytics
Data Browser
Cloud Code
Push Notifications
Settings

General settings
App name, delete your app

Application keys
App ID, Client key, Master key...

Push notifications
Manage your push settings

Web hosting
Manage your hosted domains

User authentication
User auth., Facebook, Twitter

Email settings
Manage your email templates

Collaborators
Share your app with others

Export data
Get all your app's data in a zip

(●) Push Notification Settings

Configure your push notification settings here. For detailed instructions, take a look at the Push Notification Guide.

Client push enabled?

OFF

Apple Push Certificate

This is only necessary if you plan on pushing to iOS devices. Having more than one push certificate for a single app is only supported by Pro and Enterprise accounts.

Certificate file (.p12)

Select your certificate
PARSE DATA
EFFORTLESS DATABASE

- JSON-based, auto-generated schema, semi-relational
- not real-time, not auto-syncing (started making a chat app around the db until I realized this)
- A Parse dev suggested creating sync with push notifications [parse.com/questions/real-time-updates-like-firebase](https://parse.com/questions/real-time-updates-like-firebase)
- ParseObjects can only be 128kb, ParseFile object can be 10 megabytes
SMALLEST WORKING EXAMPLE

// Init Parse and auto-generate an anonymous user for this installation
Parse.initialize(this, YOUR_APPLICATION_ID, YOUR_CLIENT_KEY);
ParseUser.enableAutomaticUser();

// Upload a new object to the Parse database
ParseObject gameScore = new ParseObject("GameScore");
gameScore.put("score", 1337);
gameScore.put("playerName", "Sean Plott");
gameScore.saveInBackground();
SUBCLASSING PARSEOBJECT

Instead of:

```java
ParseObject shield = new ParseObject("Armor");
shield.put("displayName", "Wooden Shield");
shield.put("fireproof", false);
```

We can have:

```java
Armor shield = new Armor();
shield.setDisplayName("Wooden Shield");
shield.setFireproof(false);
```

```java
@ParseClassName("Armor")
public class Armor extends ParseObject { /* ... */ }
```

```java
ParseObject.registerSubclass(Armor.class);
```
<table>
<thead>
<tr>
<th>objectID</th>
<th>username</th>
<th>password</th>
<th>authData</th>
<th>emailVerified</th>
<th>email</th>
</tr>
</thead>
<tbody>
<tr>
<td>yyGwZ1sTZR</td>
<td>3gmovqojmopy...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>OYrGxHRZn</td>
<td>zmi2meg03d92...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>vZRdk0nZ1p</td>
<td>mpame7mcq8s...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>Jsouir21Yo</td>
<td>4heapy53xan9...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>9s8cGfyhq5</td>
<td>nod5w50p4cr1n...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>Mq4aZnt1Jo</td>
<td>sndzm8sj8ta50d...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>Pk9Ap9QdLy</td>
<td>lr01sL7zwsc1ho...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>wiUjHBhgle</td>
<td>d97bymkpxseke...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>rGLK0iQ12p</td>
<td>9n3wj53dvg8b...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>D58ezQXT0Y</td>
<td>5rzmxp1iia020k...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>7mX1t2Wmoa</td>
<td>7w2vq1b2fck2...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>siwxB3PdC</td>
<td>mtryc9af2caipn...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>JiuNN1DHuY</td>
<td>rsje24q7vm2e4...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>uV0E5mlG5u</td>
<td>j02mpdfo9wb6...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
<tr>
<td>X6j8dUSYL</td>
<td>u7n4blcz7qd6f...</td>
<td>(hidden)</td>
<td>Anonymous</td>
<td>(undefined)</td>
<td>(undefined)</td>
</tr>
</tbody>
</table>
# SPECIAL CLASS: INSTALLATION

<table>
<thead>
<tr>
<th>objectid</th>
<th>appId</th>
<th>appVersion</th>
<th>deviceID</th>
<th>installationId</th>
<th>owner</th>
<th>Pointer&lt;User&gt;</th>
<th>parse</th>
<th>timeZone</th>
<th>createdAt</th>
<th>updatedAt</th>
<th>acl</th>
</tr>
</thead>
<tbody>
<tr>
<td>ke3SLvm5SD</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>40291200-cb...</td>
<td>yjGwZ1sTZR</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 18:33</td>
<td>Oct 20, 2013, 18:34</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>5tgqdkMeeW</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>820c0154-e7c...</td>
<td>OYrQxGRtZh</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 18:33</td>
<td>Oct 20, 2013, 18:33</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>9s4eFh667Tp</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>f219f97-1947...</td>
<td>vZRDkOnZ1p</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 18:32</td>
<td>Oct 21, 2013, 10:45</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>RaYpCQmpAO</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>8ec152c3-486...</td>
<td>Jscull2Y0</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 18:31</td>
<td>Oct 20, 2013, 18:38</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>T64EVUmNRF</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>f43b612-4c3d...</td>
<td>98s0Cfhq5</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 17:36</td>
<td>Oct 20, 2013, 17:40</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>fN55KdC4u</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>0b1ca1f-0563...</td>
<td>Mq4as7n4e</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 17:31</td>
<td>Oct 20, 2013, 17:31</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>chCkJouS2h</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>144615b-17...</td>
<td>PkBap9CdLy</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 18:46</td>
<td>Oct 20, 2013, 18:46</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>Pn15F45oe3</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>9d818998-87...</td>
<td>VIUH3ngie</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 18:25</td>
<td>Oct 20, 2013, 18:25</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>YkFutHfY3S</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>7d84b2f-07ef...</td>
<td>rGLdQ12p</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 13:41</td>
<td>Oct 20, 2013, 16:41</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>U0E3oPh5UT4</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>cad99a3-3cb...</td>
<td>D6Bk:string</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 18:37</td>
<td>Oct 20, 2013, 17:27</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>TVEJGRUU3k</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>30e14413-3b...</td>
<td>siwxEBFDC</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 10:05</td>
<td>Oct 20, 2013, 10:18</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>9usxQoJu</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>android</td>
<td>a10b6f-ae4...</td>
<td>JiuN1dUv</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 20, 2013, 10:16</td>
<td>Oct 20, 2013, 16:03</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>7wSTPe18...</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>e44c7043-c0...</td>
<td>uYy8anm105u</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 18, 2013, 19:14</td>
<td>Oct 20, 2013, 16:09</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>1YF56Cwmp</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>751d9c81-e...</td>
<td>X8j98JUuY</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 18, 2013, 19:01</td>
<td>Oct 19, 2013, 19:01</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>kR2ZYbrgqjW</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>e58444cb-5858...</td>
<td>KrkRgynS</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 18, 2013, 19:01</td>
<td>Oct 18, 2013, 19:01</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>IE57HY3oaC</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>39b1f450-052...</td>
<td>TYbSxGJ6zL</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 16, 2013, 19:01</td>
<td>Oct 20, 2013, 09:39</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>KE0JeWEnuEt</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>da210f4e-44a...</td>
<td>2A08zXtCX</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 18, 2013, 18:28</td>
<td>Oct 18, 2013, 18:51</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>eXPr8Iah1Hy</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>cc8a3607-f41...</td>
<td>Xw0XcmCOt</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 18, 2013, 15:05</td>
<td>Oct 18, 2013, 18:45</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>ViiHnP2b3T</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>37306b72-17...</td>
<td>0mSCBu6psx</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 18, 2013, 10:55</td>
<td>Oct 18, 2013, 14:52</td>
<td>(undefined)</td>
<td></td>
</tr>
<tr>
<td>AejdLkJHGf</td>
<td>Peekaboo</td>
<td>0.6</td>
<td>android</td>
<td>7c1d340f-5ec...</td>
<td>QBPbo4AsV5</td>
<td>Europe/He...</td>
<td>1.3.4</td>
<td>Oct 17, 2013, 19:19</td>
<td>Oct 18, 2013, 11:32</td>
<td>(undefined)</td>
<td></td>
</tr>
</tbody>
</table>
REST CALLS

curl --request GET \
   --header "X-Parse-Application-Id: Xlsvm0...WL4in" \
   --header "X-Parse-REST-API-Key: 00PlRuE...5lk" \
   --get \
   --data-urlencode 'where={"playerName":"Sean Plott","cheatMode":false}' \
   https://api.parse.com/1/classes/GameScore

{  "results": [  
    {  
      "playerName": "Sean Plott",  
      "updatedAt": "2011-08-21T18:02:52.248Z",  
      "cheatMode": false,  
      "createdAt": "2011-08-20T02:06:57.931Z",  
      "objectId": "Ed1nuqPvcm",  
      "score": 73453  
    }  
  ]}
Each object has its own ACL of users/roles/everyone and their read/write access rights

```json
{
  "*: {
    "read": true
  },
  "F4ori0jU9r": {
    "write": true,
    "read": true
  }
}
```
CLASS-SPECIFIC ACCESS CONTROL

- Get: fetching an object by its objectId.
- Find: issuing a query to fetch objects.
- Update: saving an object that already exists and has been modified.
- Create: saving an object that is new and hasn't been created yet.
- Delete: deleting an object.
- Add fields: adding fields to the class.
ROLE-BASED ACCESS CONTROL

- Users can be grouped into roles
- Subroles can be grouped into roles
- Roles can be referred to in ACLs

// Create new role object
ParseACL roleACL = new ParseACL();
roleACL.setPublicReadAccess(true);
ParseRole role = new ParseRole("Administrator", roleACL);

// Add user and child role to role
role.getUsers().add(newUser);
role.getRoles().add(childRole);
role.saveInBackground();
PARSE PUSH
PUSH NOTIFICATION

Compose message

Hello valued customer! Buy moar stuff!

Message  JSON  142 character(s) left

Read Push Docs

Hello valued customer! Buy moar stuff!

Google

Parse
PUSH NOTIFICATIONS

- Parse offers scheduling, expiration time and target segmenting
- Can be sent from the dashboard and from all SDKs, but only received in iOS, Android and Win8 "Metro"
- Requires Apple Push Certificate on iOS and Windows Push Credentials on Win8 apps. On Android Parse has its own service, with pros and cons
CUSTOMIZING PUSH NOTIFICATIONS

- **alert**: the notification's message.
- **badge**: (iOS only) the value indicated in the top right corner of the app icon. This can be set to a value or to Increment in order to increment the current value by 1.
- **sound**: (iOS only) the name of a sound file in the application bundle.
- **content-available**: (iOS only) if you are using Newsstand, set this value to 1 to trigger a background download (more here).
- **action**: (Android only) the Intent should be fired when the push is received. If not title or alert values are specified, the Intent will be fired but no notification will appear to the user.
- **title**: (Android only) the value displayed in the Android system tray notification.
# PUSH CHANNELS

<table>
<thead>
<tr>
<th>objectid</th>
<th>appName</th>
<th>appVer</th>
<th>channels</th>
<th>owner</th>
</tr>
</thead>
<tbody>
<tr>
<td>ke3SLvnhB0</td>
<td>Peekaboo</td>
<td>0.8.4</td>
<td>&quot;Control&quot;, &quot;user_yrGwZ1STZr&quot;</td>
<td>yrGwZ1STZr</td>
</tr>
<tr>
<td>5tqGdA0xM</td>
<td>Peekaboo</td>
<td>0.8.4</td>
<td>&quot;Control&quot;, &quot;user_OYrGxHIrZn&quot;</td>
<td>OYrGxHIrZn</td>
</tr>
<tr>
<td>s9444h67Tp</td>
<td>Peekaboo</td>
<td>0.8.4</td>
<td>&quot;Control&quot;, &quot;user_vZrd0Z1g&quot;</td>
<td>vZrd0Z1g</td>
</tr>
<tr>
<td>Ra1v9C0mpA</td>
<td>Peekaboo</td>
<td>0.8.4</td>
<td>&quot;Control&quot;, &quot;user_zsclir21yO&quot;</td>
<td>zsclir21yO</td>
</tr>
<tr>
<td>T64BVUrmrF</td>
<td>Peekaboo</td>
<td>0.8.2</td>
<td>&quot;Control&quot;, &quot;user_Gs8oGyhq5&quot;</td>
<td>9s8oGyhq5</td>
</tr>
<tr>
<td>6NG5kdtc4u</td>
<td>Peekaboo</td>
<td>0.8.1</td>
<td>&quot;Control&quot;</td>
<td>Mc4aZntAe</td>
</tr>
<tr>
<td>o1CHPouS2h</td>
<td>Peekaboo</td>
<td>0.8.1</td>
<td>&quot;Control&quot;, &quot;user_Fk3Ap9QdLy&quot;</td>
<td>Fk3Ap9QdLy</td>
</tr>
<tr>
<td>Pn1uF4b6e3</td>
<td>Peekaboo</td>
<td>0.8.1</td>
<td>&quot;Control&quot;, &quot;user_vuH2hge&quot;</td>
<td>vuH2hge</td>
</tr>
<tr>
<td>YkFutHNY8S</td>
<td>Peekaboo</td>
<td>0.8.1</td>
<td>&quot;Control&quot;, &quot;user_rLk0Iq12p&quot;</td>
<td>rLk0Iq12p</td>
</tr>
<tr>
<td>U0EzP5UT4</td>
<td>Peekaboo</td>
<td>0.8.1</td>
<td>&quot;Control&quot;, &quot;user_D58ezQXT0Y&quot;</td>
<td>D58ezQXT0Y</td>
</tr>
<tr>
<td>1VBJGRLzgB</td>
<td>Peekaboo</td>
<td>0.8</td>
<td>&quot;Control&quot;, &quot;user_suwxt3PdC&quot;</td>
<td>suwxt3PdC</td>
</tr>
<tr>
<td>8usx9GjJl1</td>
<td>Peekaboo</td>
<td>0.8.5</td>
<td>&quot;Control&quot;, &quot;user_jiuNn1HdUy&quot;</td>
<td>jiuNn1HdUy</td>
</tr>
<tr>
<td>7vW7N10Aq</td>
<td>Peekaboo</td>
<td>0.8.5</td>
<td>&quot;Control&quot;, &quot;user_u0D5mlQ5u&quot;</td>
<td>u0D5mlQ5u</td>
</tr>
<tr>
<td>1F56CWpmp</td>
<td>Peekaboo</td>
<td>0.8.5</td>
<td>&quot;Control&quot;, &quot;user_x6j0g8u8yL&quot;</td>
<td>x6j0g8u8yL</td>
</tr>
<tr>
<td>kRtzybVjM</td>
<td>Peekaboo</td>
<td>0.8.5</td>
<td>&quot;Control&quot;, &quot;user_k1hRqxynt5&quot;</td>
<td>k1hRqxynt5</td>
</tr>
<tr>
<td>1Es7Hy3oC</td>
<td>Peekaboo</td>
<td>0.8.6</td>
<td>&quot;Control&quot;, &quot;user_TJbc9X96zl&quot;</td>
<td>TJbc9X96zl</td>
</tr>
<tr>
<td>KEOeNeUat</td>
<td>Peekaboo</td>
<td>0.8.5</td>
<td>&quot;Control&quot;, &quot;user_2A0t8zTcX&quot;</td>
<td>2A0t8zTcX</td>
</tr>
<tr>
<td>eXnV8haHy</td>
<td>Peekaboo</td>
<td>0.8.5</td>
<td>&quot;Control&quot;, &quot;user_xwoXcn00Zt&quot;</td>
<td>xwoXcn00Zt</td>
</tr>
<tr>
<td>VfHnUpRb3T</td>
<td>Peekaboo</td>
<td>0.8.5</td>
<td>&quot;Control&quot;, &quot;user_0mGCBuVsP&quot;</td>
<td>0mGCBuVsP</td>
</tr>
<tr>
<td>AojdLKdHQF</td>
<td>Peekaboo</td>
<td>0.6.1</td>
<td>&quot;Control&quot;, &quot;user_QBbpq4A45V&quot;</td>
<td>QBbpq4A45V</td>
</tr>
</tbody>
</table>

Installations (not users) subscribe to pushes
Parse.initialize("RNh7...vfJtTc", "1CdI01...21iU");

var recent = new Date(new Date().getTime() - (24 * 3600*1000));
var query = new Parse.Query(Parse.Installation);
query.equalTo('channels', 'Control');
query.greaterThanOrEqualTo("updatedAt", recent);

Parse.Push.send({
  where: query,
  data: {
    alert: "Hello JavaScript!"
  }
}, {
  success: function() {
    /* Push was successful */
  },
  error: function(error) {
    /* Handle error */
  }
});
JSON DATA PUSH

{
    "title": "BaasChat",
    "alert": "Hello mobile client!",
    "myCustomData": "foobar"
}
EXECUTE (ANDROID) CLIENT CODE VIA PUSH

```java
{
    "action": "fi.vilsepi.peekaboo.TAKE_PHOTO",
    "sneakyPayload": "any parameters for our function call"
}

<!-- the function onReceive(context, intent) of this class will be called -->
<receiver android:name="fi.vilsepi.peekaboo.BackgroundPushReceiver">
    <intent-filter>
        <action android:name="fi.vilsepi.peekaboo.TAKE_PHOTO"></action>
    </intent-filter>
</receiver>

UiPushReceiver pushReceiver = new UiPushReceiver(new Handler());
registerReceiver(pushReceiver, new IntentFilter("fi.vilsepi.peekaboo.PHOTO_TA

public class UiPushReceiver extends BroadcastReceiver {
    public void onReceive(final Context context, Intent intent) {
        String action = intent.getAction();
        String channel = intent.getExtras().getString("com.parse.Channel");
        JSONObject json = new JSONObject(intent.getExtras().getString("com.parse.
        // etc..
    }
}
```
UNDER THE HOOD

- On Android, pushes use Parse's own background service
- Pushes are cleartext HTTP over open TCP connection from push.parse.com

```json
{
    "time": "2013-10-17T12:33:00.224Z",
    "oauth_key": "RNh7DvetG0eZtiffApJ2yTUv9g9wY0rq4ULVfJtTC",
    "data": {
        "alert": "Hello Wireshark!",
        "push_hash": "3616ff34c02d2e63b895ec112765f270"
    }
}
```
DEMO TIME!
1. Enable Settings -> Security -> Unknown sources
2. Install vilsepi.dyndns.org/baas

(tested on Gingerbread 2.3.3, Jelly Bean 4.2 and 4.3)
PEEKABOO: CAMERA SHARING APP

vilsepi.dyndns.org/baas
JAVASCRIPT CLIENT

Latest photos from all users
1. on first start, auto-generate an anonymous user
2. update the user into the database and subscribe to push channels
3. query for users that have been modified lately, pick one at random
4. send client push to the target via Parse
5. on receiving TAKE_PHOTO, take a photo with camera and upload it to Parse
6. once photo has been uploaded, send client push to requester via Parse
7. on receiving PHOTO_TAKEN, query for Photos that current user has requested
OTHER SERVICES
CLOUD CODE: HOSTED SERVER-SIDE CODE

- server-side Javascript programs
- code is uploaded to Parse using CLI tool (simple as "parse deploy")
Lorem Ipsum on yksinkertaisesti testausteksti, jota tulostus- ja ladontateollisuudet käyttävät. Lorem Ipsum on ollut teollisuuden normaali testausteksti jo 1500-luvulta asti, jolloin tuntematon tulostaja otti kaljuunan...
NOT JUST WEB APPS

- Client apps can make function calls to server-side code
  parse.com/docs/cloud_code_guide#functions
- Cloud code can run scheduled jobs
- JS modules for 3rd party services: photochecking, email, credit cards, sms, voice calls
CLOUD CODE LIMITS

- Functions have strict time limits (15 seconds, 3 seconds for beforeSave and afterSave)
- Background jobs are killed in 15 minutes
- Only one job running at a time (pay for whopping two jobs), rest are queued
- This is not a computation IaaS but rather cleaning up the data that is stored
OTHER CLIENT SDK FEATURES

- Social and identity integration with Facebook and Twitter
- Convenience methods for adapting data queries into UI elements
- Geolocation helper functions
- Caching data in client devices (saveEventually, saveInBackground...)
- Analytics, custom metrics (errors, user behavior)
EVALUATION

- database is not "real-time" and auto-syncing like Firebase
- data can be exported easily, but code is specific to the proprietary API
• Varying delay in push notifications
• Sometimes often the Android client's push service dies silently
parse.com/questions/ios-push-notifications-very-unreliable
citrrus.com/blog/what-going-on-with-parse-dot-com
CloudCode snippet for Google Cloud Messaging
parse.com/questions/gcm-support
• cannot change/reset API keys
• cannot easily copy/fork/clone an app
• can only view push statistics in the dashboard
WHAT PARSE IS GOOD FOR

- backend for (multi-platform) mobile apps
- push notifications for marketing and user engagement
- probably not the best option if you only need a database
QUESTIONS?

da fuq did i just watch?
THANK YOU!