

# BB10 and WebWorks 101 – Intro to building BB10 applications using Ripple and the WebWorks SDK

DEV144

@n\_adam\_stanley, @ken\_wallis, @confusement

May 1-3, 2012

# 1. Intros and demo

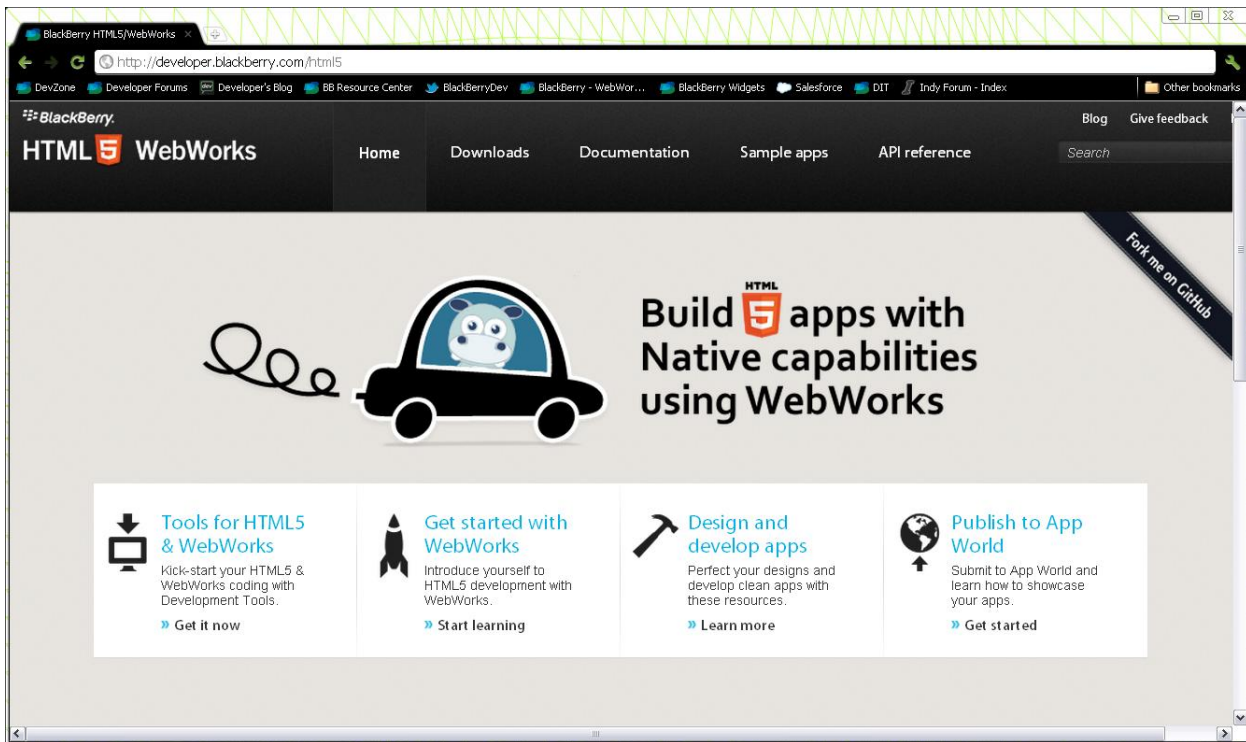
What are we going to do today?

- Must Have:
  - ▶ Laptop (Win XP, Win 7 or Mac OS)
  - ▶ Chrome browser
  - ▶ Ripple extension; WebWorks SDK for BB10
    - Available on USB Key
- Would be nice:
  - ▶ Internet connection
  - ▶ Work in groups if you prefer

Bookmark this page

 **BlackBerry** 10 Jam

<http://developer.blackberry.com/html5>



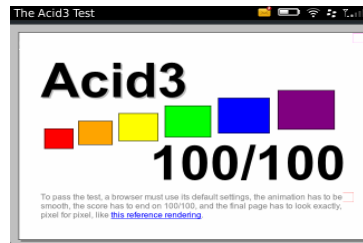
# BlackBerry Web Platform

 **BlackBerry 10** Jam

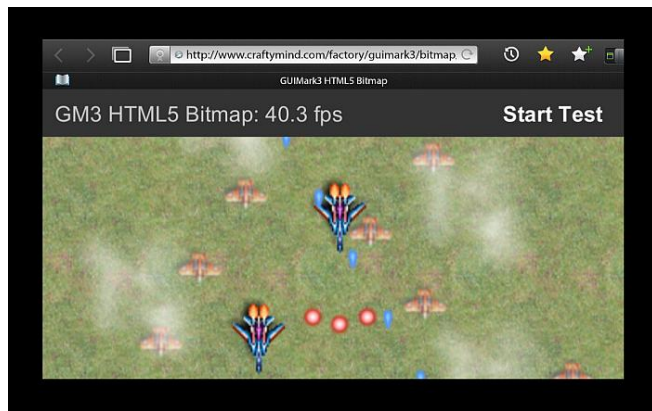
- Driven by strong Web standards support
- Powered by WebKit
- HTML5 and CSS3
- Flash 11 and WebGL



<http://html5test.com>

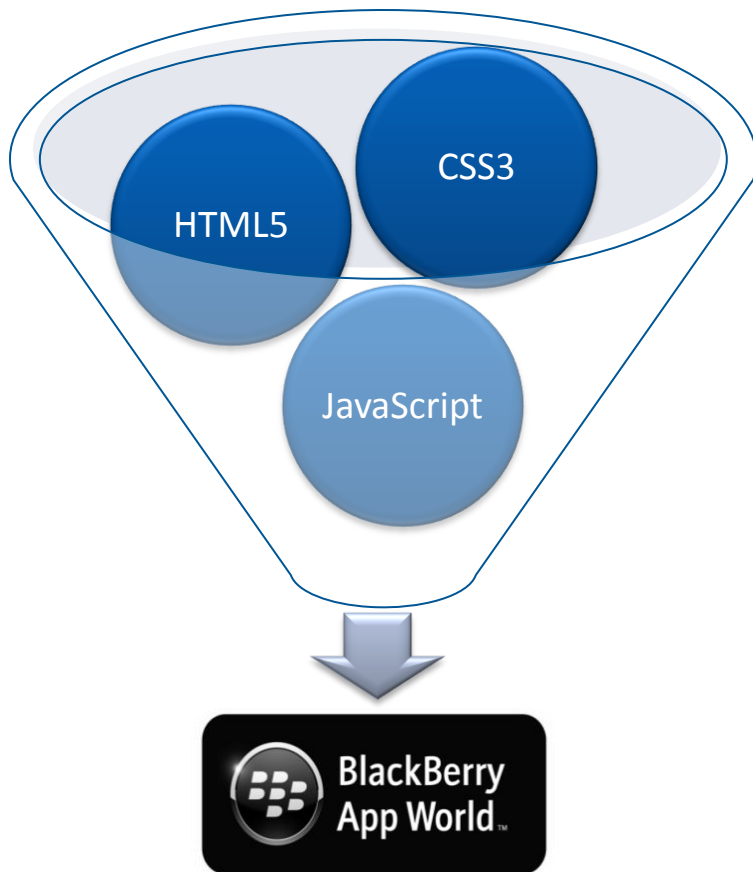


<http://acid3.acidtests.org>



<http://www.craftymind.com/guimark3/>





What if you could build a BlackBerry application using Web technologies?

# Real Examples in App World

**BlackBerry 10 Jam**



Rugby World Cup 2011



News360



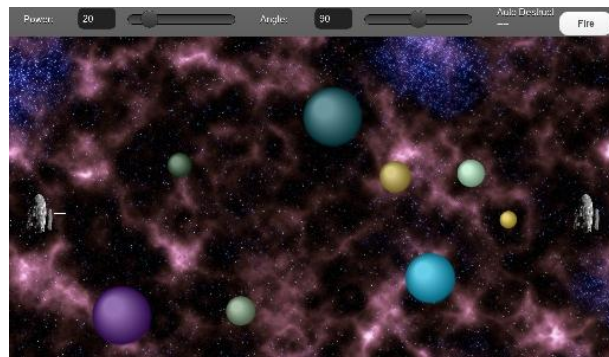
Mahjong for PlayBook



Huffington Post



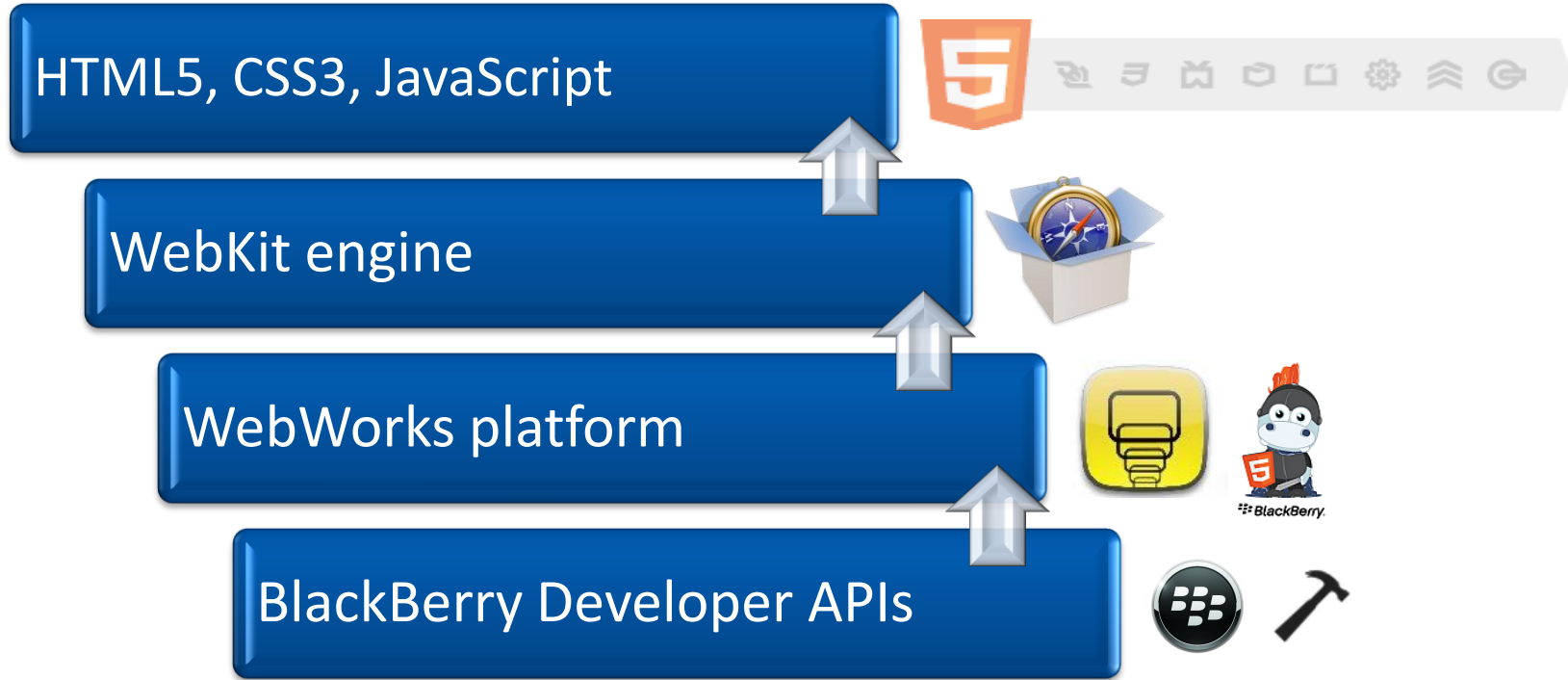
Italian Party



Gravitation

# HTML5 powered by native capabilities

BlackBerry 10 Jam



<https://bdsc.webapps.blackberry.com/html5/api>



## 2. Install Ripple

Lets emulate

# What is Ripple?

- A multi-platform mobile emulator
  - ▶ Render Web content similar to a web browser
  - ▶ Emulate device-specific APIs and capabilities
  - ▶ Web inspector debugging & profiling
  - ▶ Package and sign BlackBerry applications



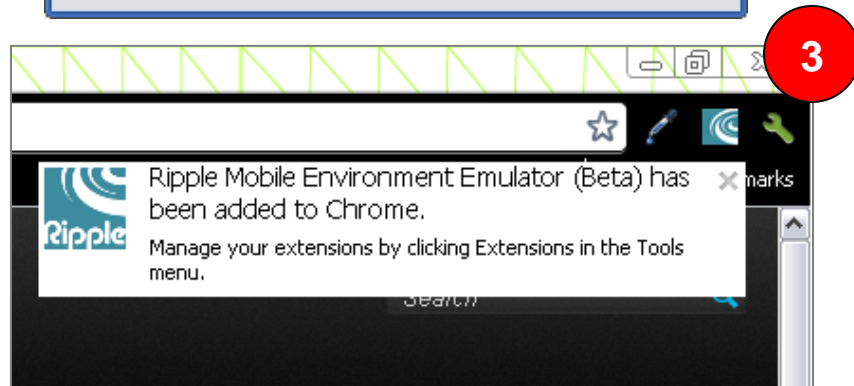
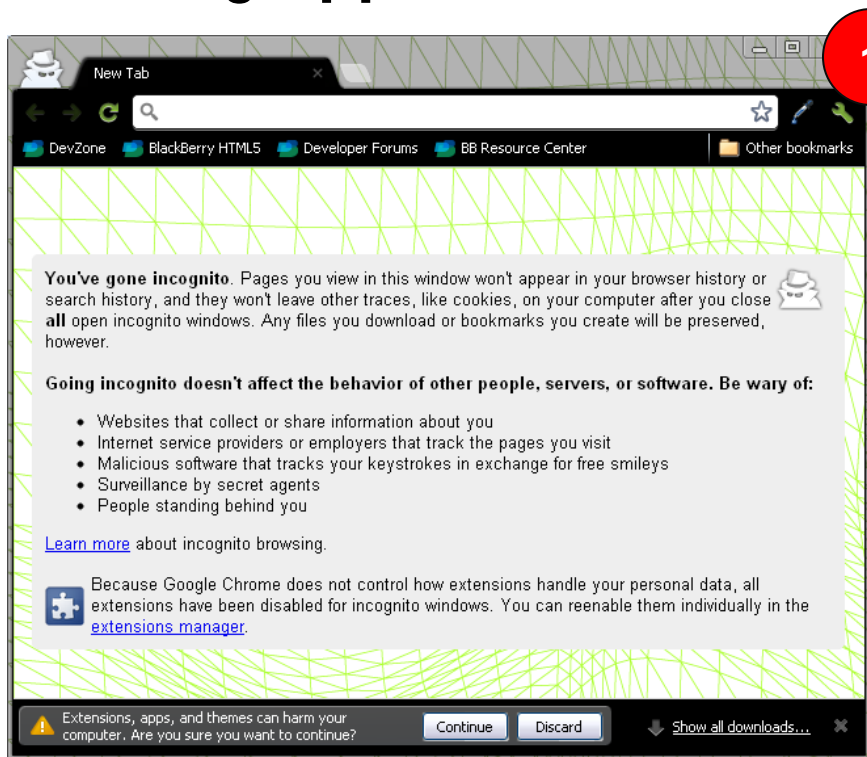
- Download and install the Ripple extension
  - ▶ <http://developer.blackberry.com/html5>
- Recommended installation folder
  - ▶ Windows: C:\program files\Research In Motion\Ripple <version>
  - ▶ Mac: /Developer/SDKs/Research In Motion/Ripple <version>



- Find **ripple\_ui.crx** file:
  - ▶ Windows: C:\program files\Research In Motion\Ripple <version>
  - ▶ Mac: /Developer/SDKs/Research In Motion/Ripple <version>
- Launch chrome browser

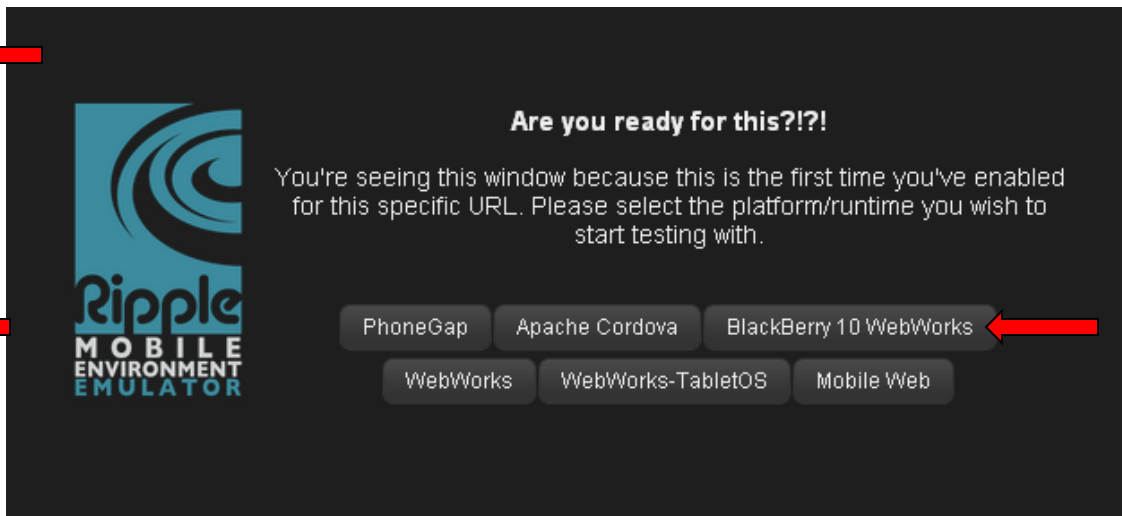
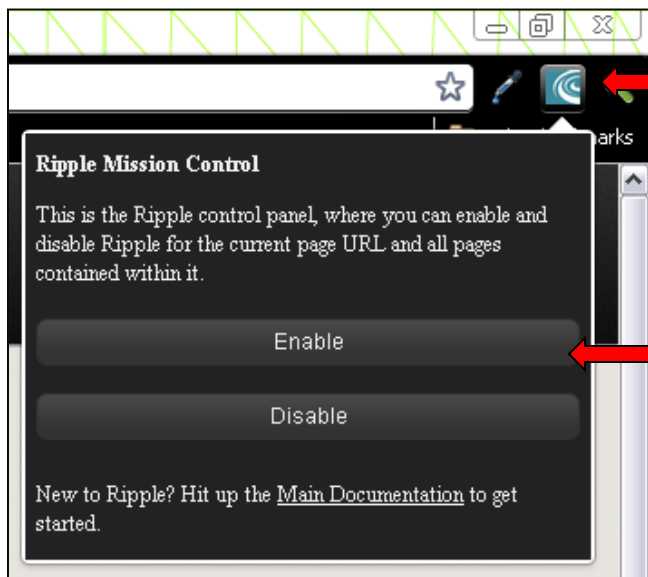
# Installing Ripple

- Drag `ripple_ui.crx` into the Chrome browser



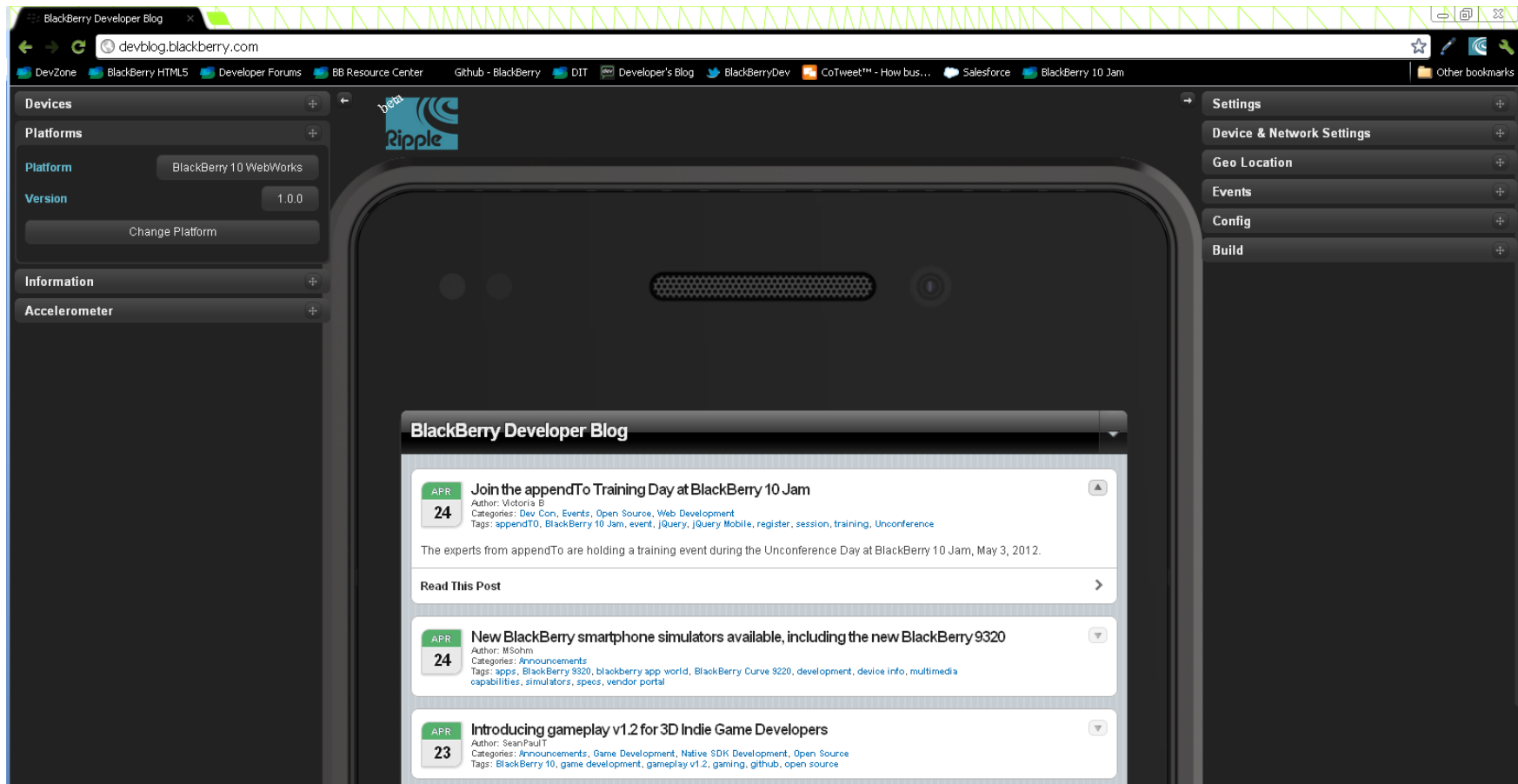
# Using Ripple

- Open <http://devblog.blackberry.com> using Chrome browser
- Click on Ripple extension icon → Enable
- Choose 'BlackBerry 10 WebWorks' platform



# Using Ripple

 **BlackBerry 10 Jam**



The screenshot shows a BlackBerry 10 device displaying the BlackBerry Developer Blog. The browser address bar shows `devblog.blackberry.com`. The page features a left sidebar with navigation links: Devices, Platforms, Platform (BlackBerry 10 WebWorks), Version (1.0.0), Change Platform, Information, and Accelerometer. A right sidebar contains a Settings menu with options: Device & Network Settings, Geo Location, Events, Config, and Build. The main content area displays the BlackBerry Developer Blog header and a list of three blog posts:

- Join the appendTo Training Day at BlackBerry 10 Jam**  
Author: Victoria B  
Categories: Dev Con, Events, Open Source, Web Development  
Tags: appendTO, BlackBerry 10 Jam, event, jQuery, jQuery Mobile, register, session, training, Unconference  
The experts from appendTo are holding a training event during the Unconference Day at BlackBerry 10 Jam, May 3, 2012.  
Read This Post
- New BlackBerry smartphone simulators available, including the new BlackBerry 9320**  
Author: M Sohm  
Categories: Announcements  
Tags: apps, BlackBerry 9320, blackberry app world, BlackBerry Curve 9320, development, device info, multimedia capabilities, simulators, specs, vendor portal
- Introducing gameplay v1.2 for 3D Indie Game Developers**  
Author: SeanPaulT  
Categories: Announcements, Game Development, Native SDK Development, Open Source  
Tags: BlackBerry 10, game development, gameplay v1.2, gaming, github, open source

## 3. Sample application

Kitchen Sink



- Sample application for developers
  - ▶ Over 100 samples: “Everything but the kitchen sink”
  - ▶ HTML5
  - ▶ CSS3
  - ▶ WebWorks APIs
  - ▶ BlackBerry web platform capabilities
- Download the source code
  - ▶ <http://github.com/blackberry/WebWorks-Samples>
  - ▶ ZIP archive will contain a kitchenSink folder

- Next we'll configure Ripple to load kitchenSink



## 4. Environment setup

Configure to load WebWorks project using a web server or file system access

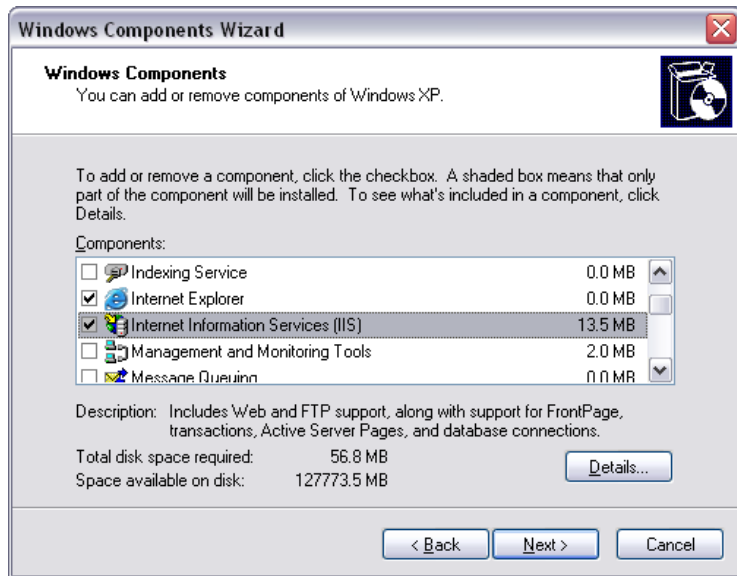
Use one of two ways to load a WebWorks project in Ripple:

1. Setting up a web server
2. Accessing local file system directly

- How it works:
  - ▶ A web server provides a response to an HTTP request
  - ▶ Developer saves WebWorks project to local machine
  - ▶ Load and display WebWorks project files using Ripple
    - E.g. <http://localhost/kitchenSink>

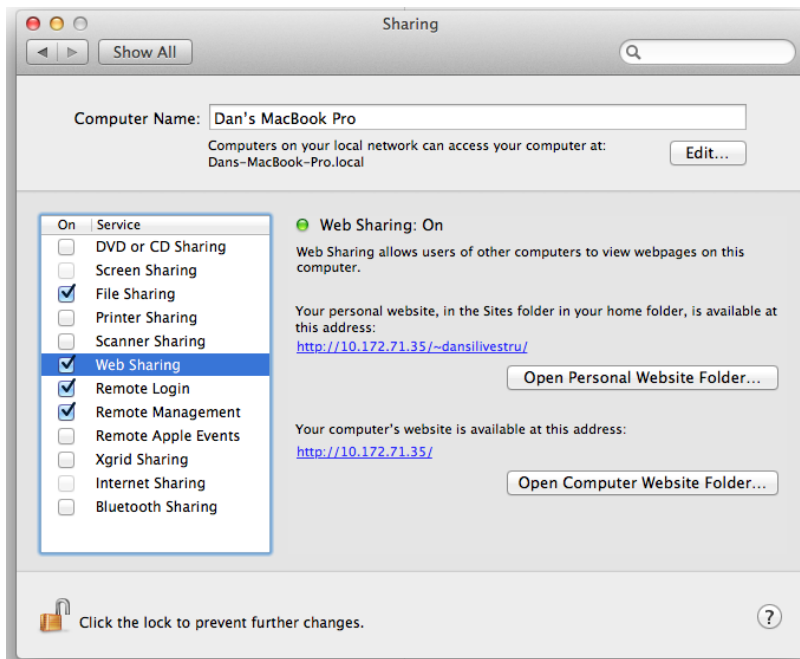
# Setting up a web server

- Windows: Enable Internet Information Services (IIS)
  - ▶ **XP** : Control Panel → Add / remove programs → Windows Components
  - ▶ **Win7**: Control Panel → Programs → Windows Features



# Setting up a web server

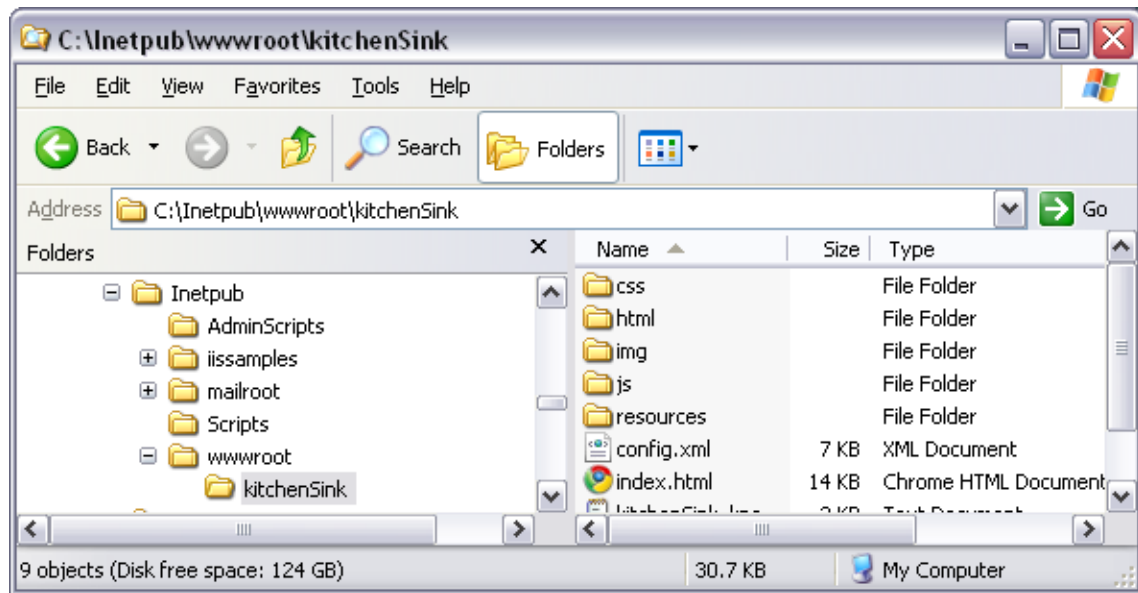
- Mac: Enable Web sharing
  - ▶ System Preferences → Sharing → Web Sharing



# Setting up a web server

 **BlackBerry** 10 **Jam**

- Extract **kitchenSink** folder into web server “working” folder
  - ▶ Windows default: **C:\inetpub\wwwroot\kitchenSink**
  - ▶ Mac default: **~/Sites/kitchenSink**



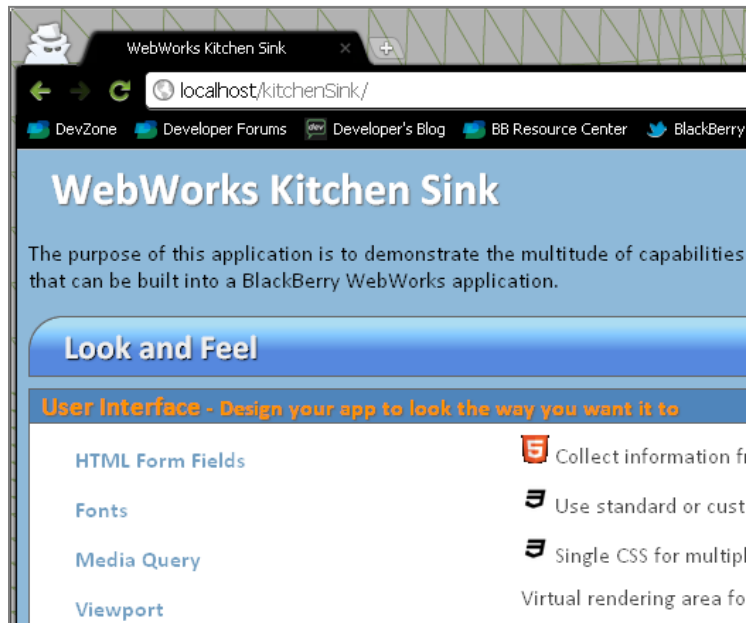


# Setting up a web server

 **BlackBerry** 10 **Jam**

- Can now load <http://localhost/kitchenSink>

## Browser



## Ripple extension



\* Required only if you cannot setup a local Web server

# Accessing local file system directly

 **BlackBerry** 10 **Jam**

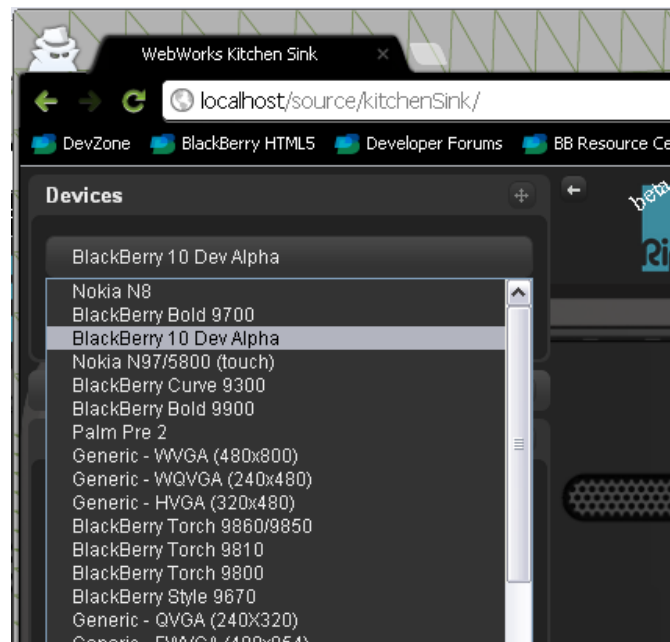
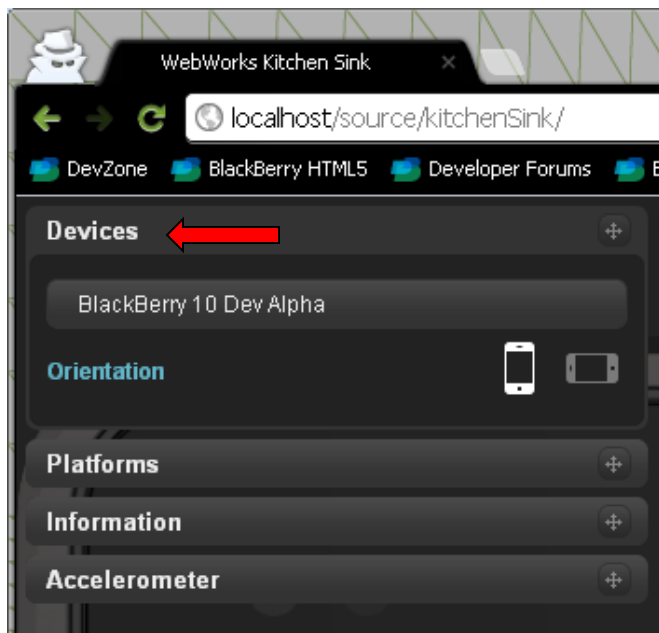
- “Accessing a project from the file system in Ripple”
  - ▶ Tutorial <https://bitly.com/>
- How?
  - ▶ Create a local **RippleSites** folder
  - ▶ Deploy your WebWorks project to this folder
  - ▶ Load using port 9900 <http://localhost:9900/myApp/index.html>

## 5. Using Ripple

Its more than just a pretty face

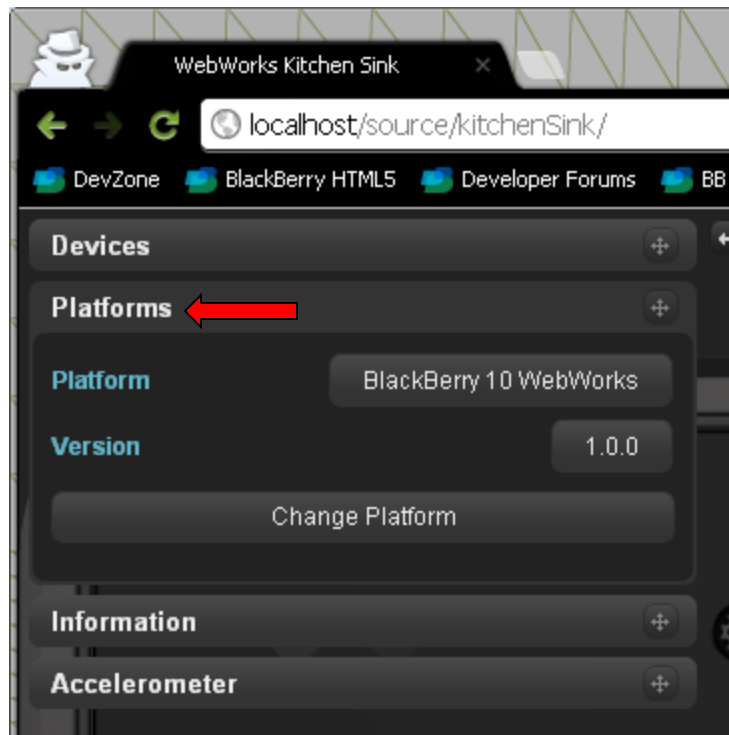
# Using Ripple

- Switch Devices
  - ▶ Change between device profiles and screen orientation



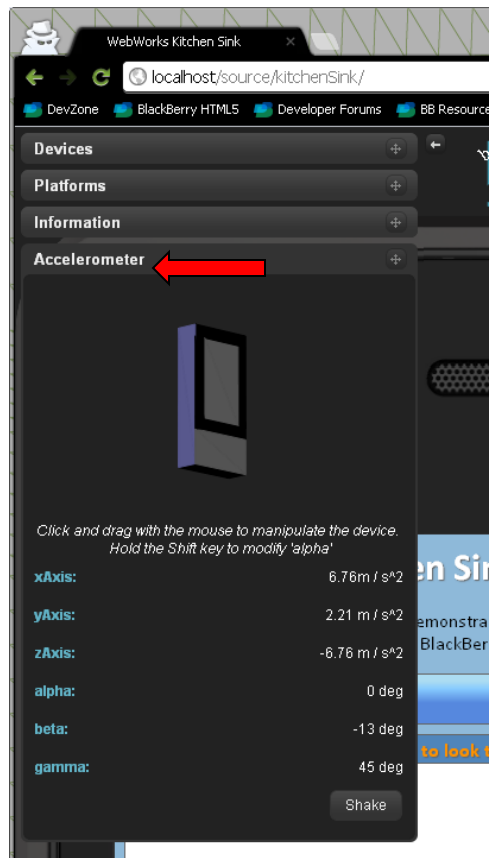
# Using Ripple

- Change Platforms
  - ▶ WebWorks
  - ▶ PhoneGap
  - ▶ Apache Cordova
  - ▶ Mobile Web
- Platform-specific features
  - ▶ WebWorks APIs
  - ▶ PhoneGap APIs



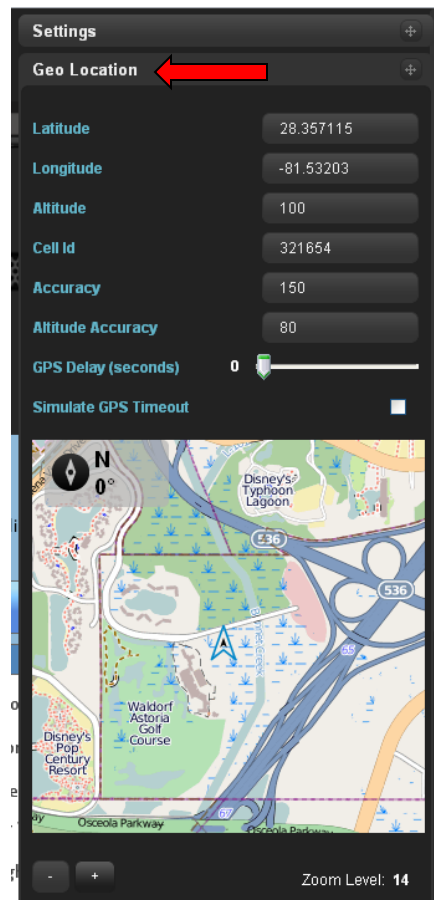
# Using Ripple

- Accelerometer
  - ▶ Emulate physical device movement
  - ▶ Move virtual device with mouse
- Example test page:
  - ▶ <http://rem.io/run-away.html>



# Using Ripple

- Geolocation
  - ▶ Emulate GPS coordinates
  - ▶ Emulate GPS delay
- GPS examples:
  - ▶ BB10Jam: 28.357115, -81.53203
  - ▶ Taj Majal: 27.175057, 78.042068
  - ▶ Niagara Falls: 43.08337, -79.073925
  - ▶ Stonehenge: 51.17859, -1.826134





## 6. Installing the WebWorks SDK

Transform your HTML5 into a BlackBerry application

- Package WebWorks assets into a BlackBerry application
- Required: BlackBerry WebWorks SDK for BB10
- Optional: BlackBerry 10 simulator
  - VMware Player (Windows)
  - VMware Fusion (Mac)

# Step 1: Set up the SDK

- Install the BlackBerry WebWorks SDK for BB10
  - ▶ <http://developer.blackberry.com/html5/download>

## Package and Distribute

Are you ready to start packaging your BlackBerry WebWorks applications? Choose one of the following SDKs to help you package your application files and distribute your app.

### BlackBerry 10 WebWorks SDK

Want to get started on developing your WebWorks based BlackBerry 10 application? This is the package you want to have.

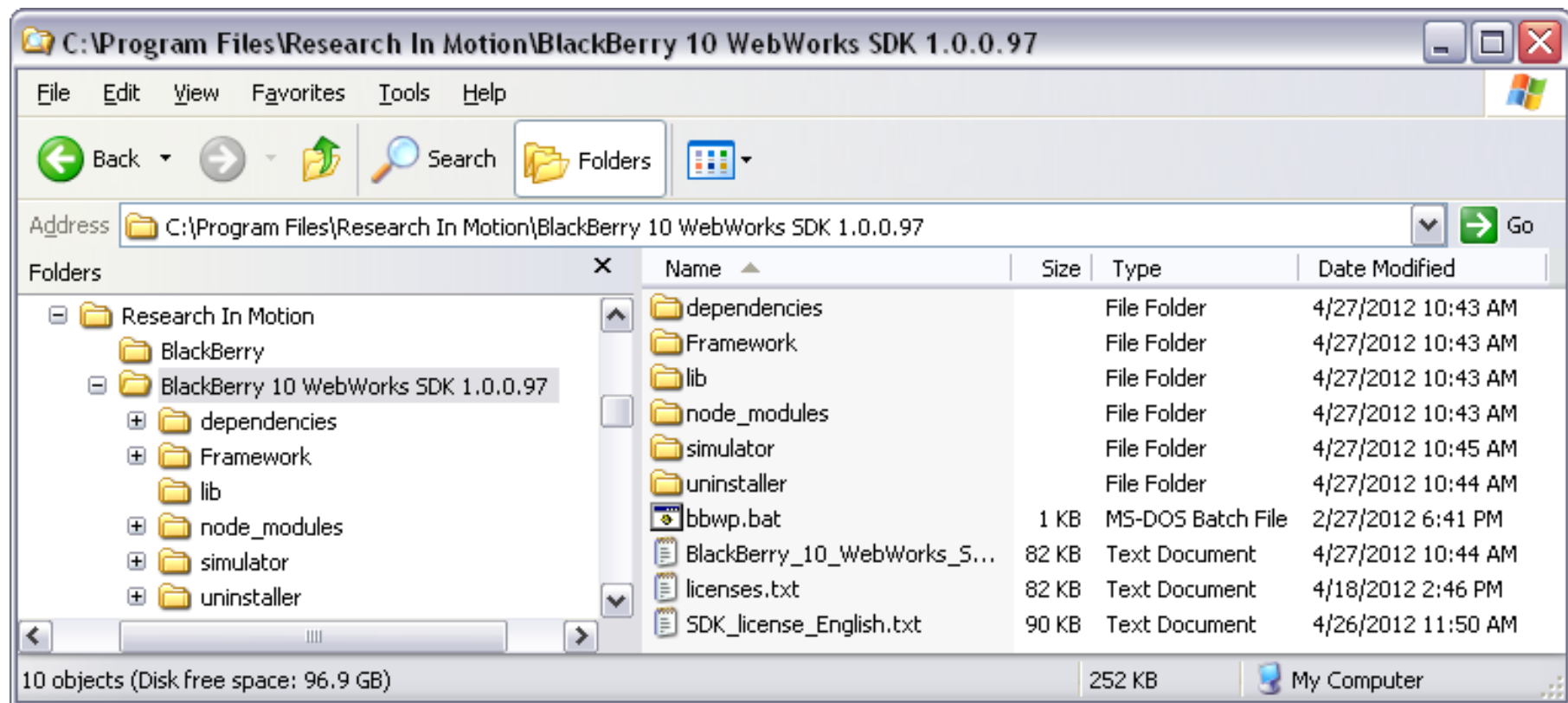


#### BlackBerry 10 WebWorks SDK

1.0.0 beta for Windows (305 Mb)

[For Mac?](#) | [System Requirements](#) | [Previous Versions](#)

# Step 2: Set up the SDK

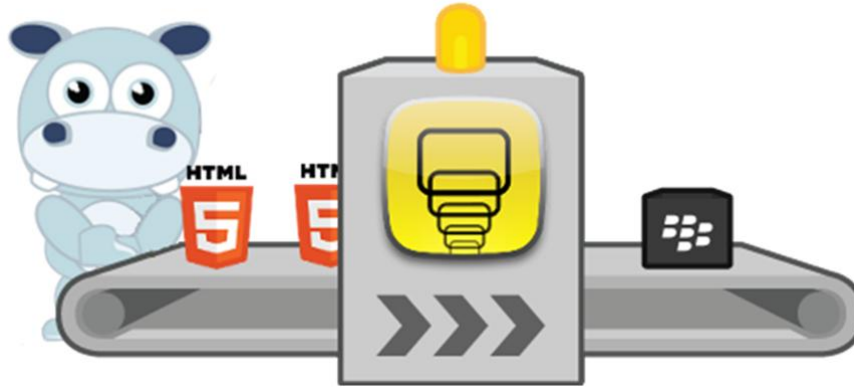
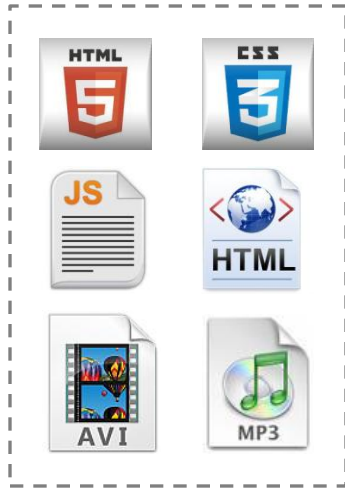


## 7. Build the kitchenSink app

Do it

# How to get there?

**BlackBerry 10 Jam**



Web Assets

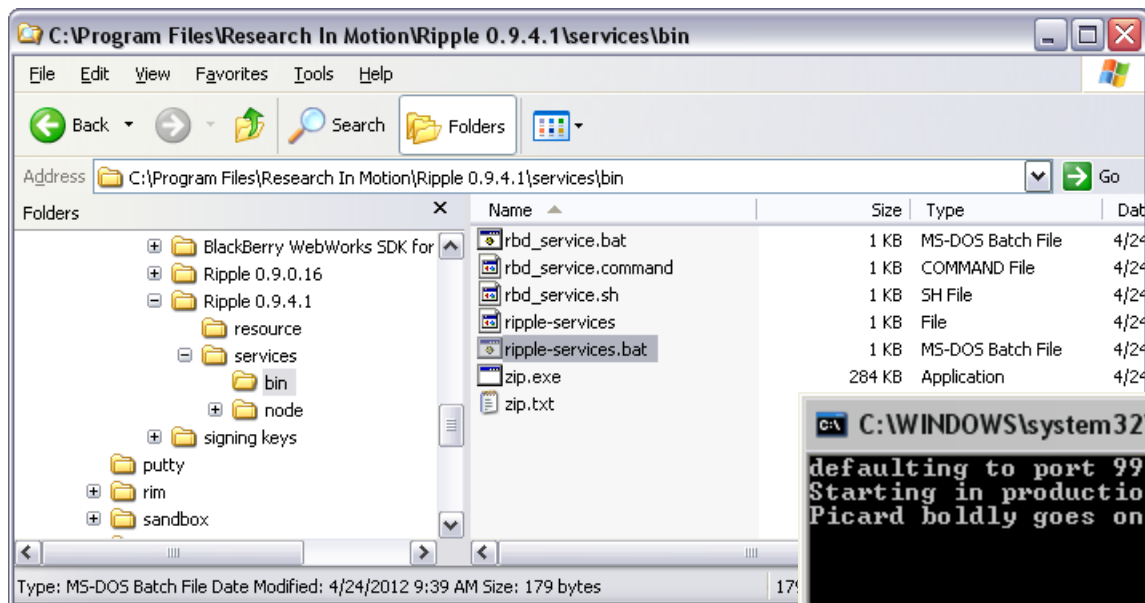
WebWorks

BlackBerry Applications

- Ripple extension can compile, sign and deploy apps!
  - ▶ Start **ripple-services** command line utility
  - ▶ Opens port 9910 for use
- Run services\bin\**ripple-services.bat**:
  - ▶ Windows: C:\program files\Research In Motion\Ripple <version>
  - ▶ Mac: /Developer/SDKs/Research In Motion/Ripple <version>
- Keep command window open

# Build environment setup

 **BlackBerry** 10 Jam

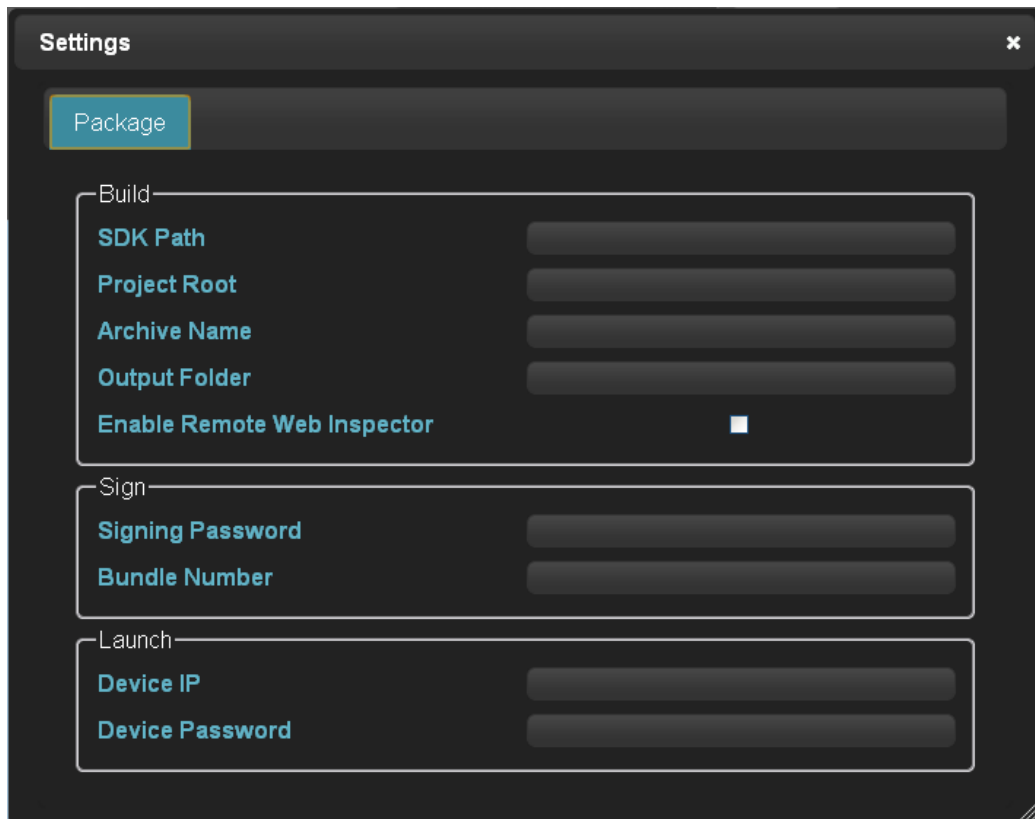
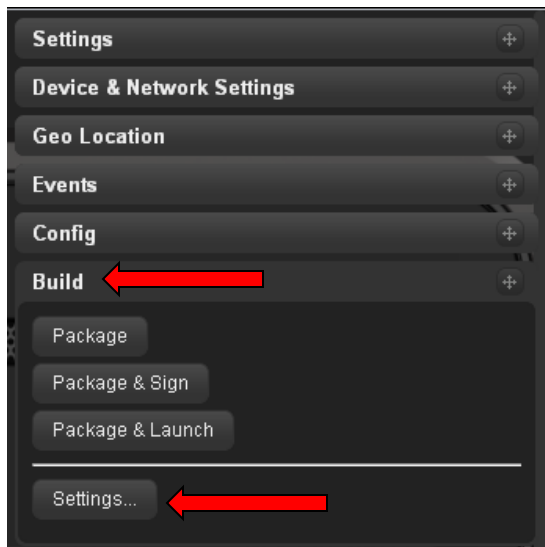




- **TEMPORARY STEP (As of May 1, 2012)**
  - ▶ WebWorks for BB10 does not yet support *all* APIs
- Edit kitchenSink/**config.xml**
- Comment / remove ALL <feature> elements **except**:
  - ▶ blackberry.app
  - ▶ blackberry.invoke
  - ▶ blackberry.system
  - ▶ blackberry.identity

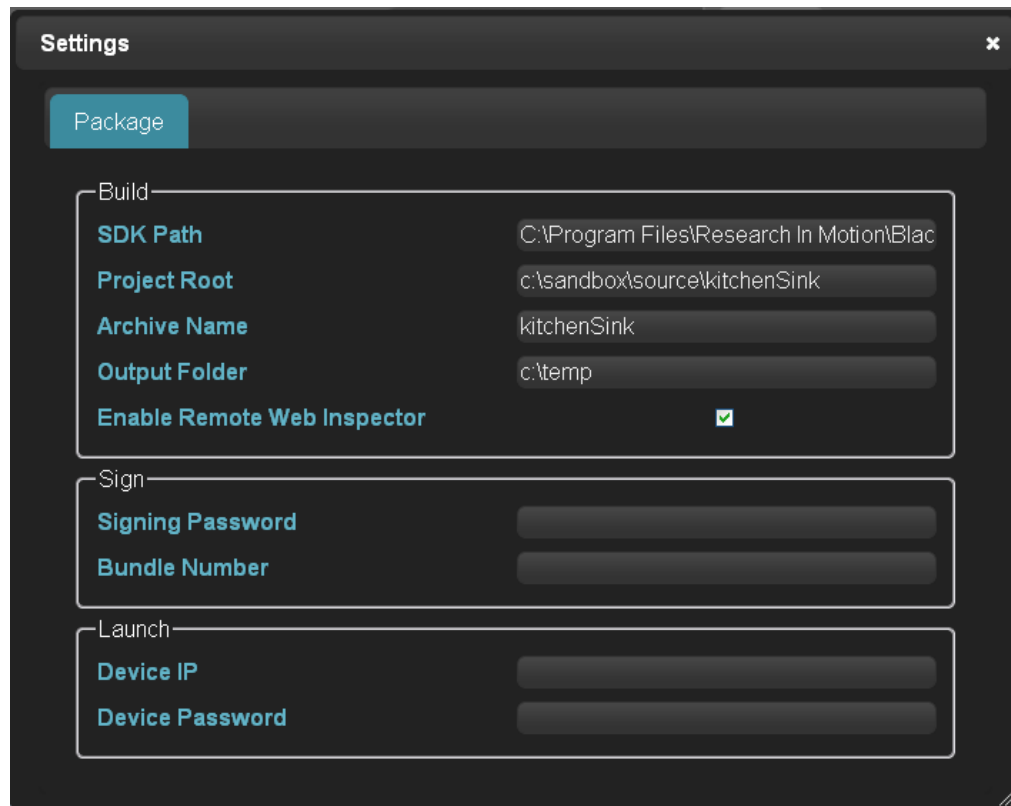
# Build environment setup

- Open Ripple extension
- Expand Build tab
- Click Settings button



# Configure settings

- SDK Path
  - ▶ BBWP installer directory
- Project Root
  - ▶ Application source code
- Archive Name
  - ▶ BAR file name
- Output Folder
  - ▶ Different than project root

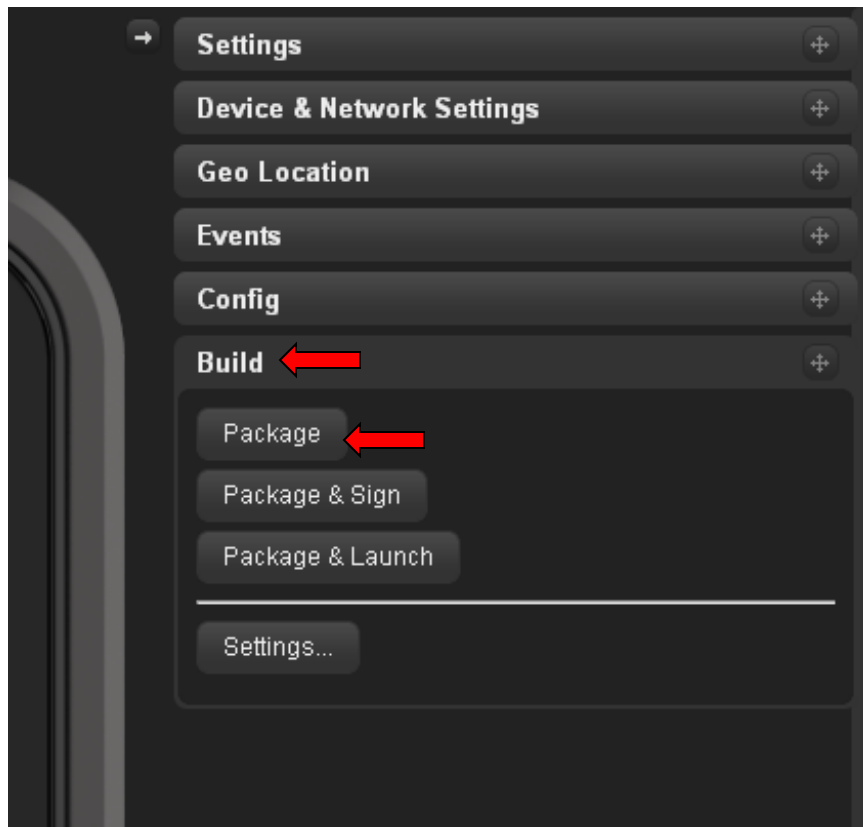


The screenshot shows the 'Settings' window for BlackBerry 10 Jam. It has a dark theme and a title bar with a close button. A 'Package' tab is selected. The settings are organized into three sections: 'Build', 'Sign', and 'Launch'. The 'Build' section contains five settings: 'SDK Path' (C:\Program Files\Research In Motion\Blac), 'Project Root' (c:\sandbox\source\kitchenSink), 'Archive Name' (kitchenSink), 'Output Folder' (c:\temp), and 'Enable Remote Web Inspector' (checked). The 'Sign' section contains 'Signing Password' and 'Bundle Number'. The 'Launch' section contains 'Device IP' and 'Device Password'. All text input fields are currently empty except for the pre-filled values in the 'Build' section.

Section	Setting	Value
Build	SDK Path	C:\Program Files\Research In Motion\Blac
	Project Root	c:\sandbox\source\kitchenSink
	Archive Name	kitchenSink
	Output Folder	c:\temp
	Enable Remote Web Inspector	<input checked="" type="checkbox"/>
Sign	Signing Password	
	Bundle Number	
Launch	Device IP	
	Device Password	

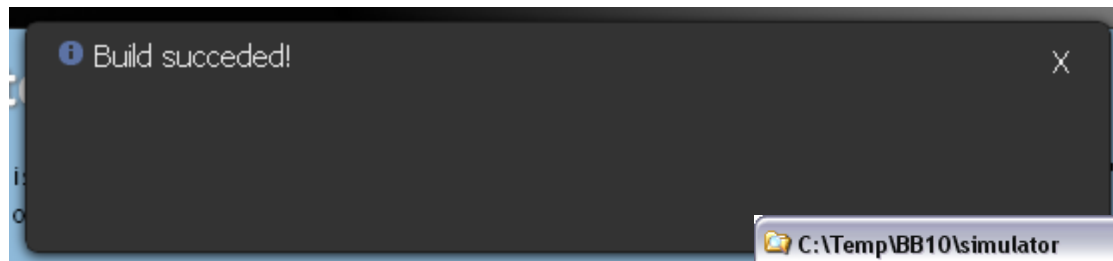
# Build using Ripple

- Open Build tab
  - ▶ Package
    - For simulators
  - ▶ Package & Sign
    - For live devices
  - ▶ Package & Launch
    - Deploy to simulator

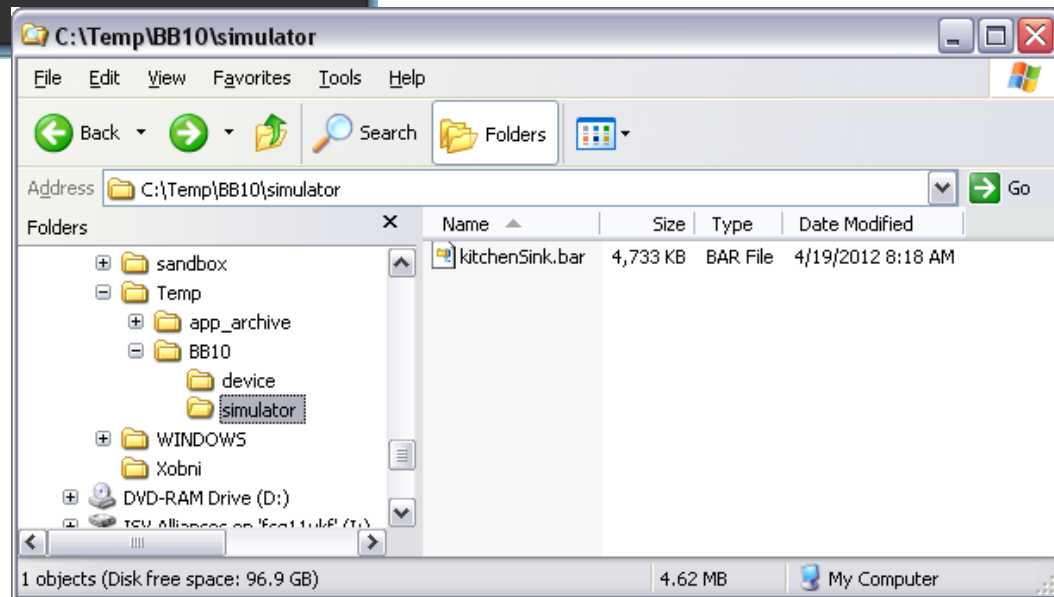


# Build using Ripple

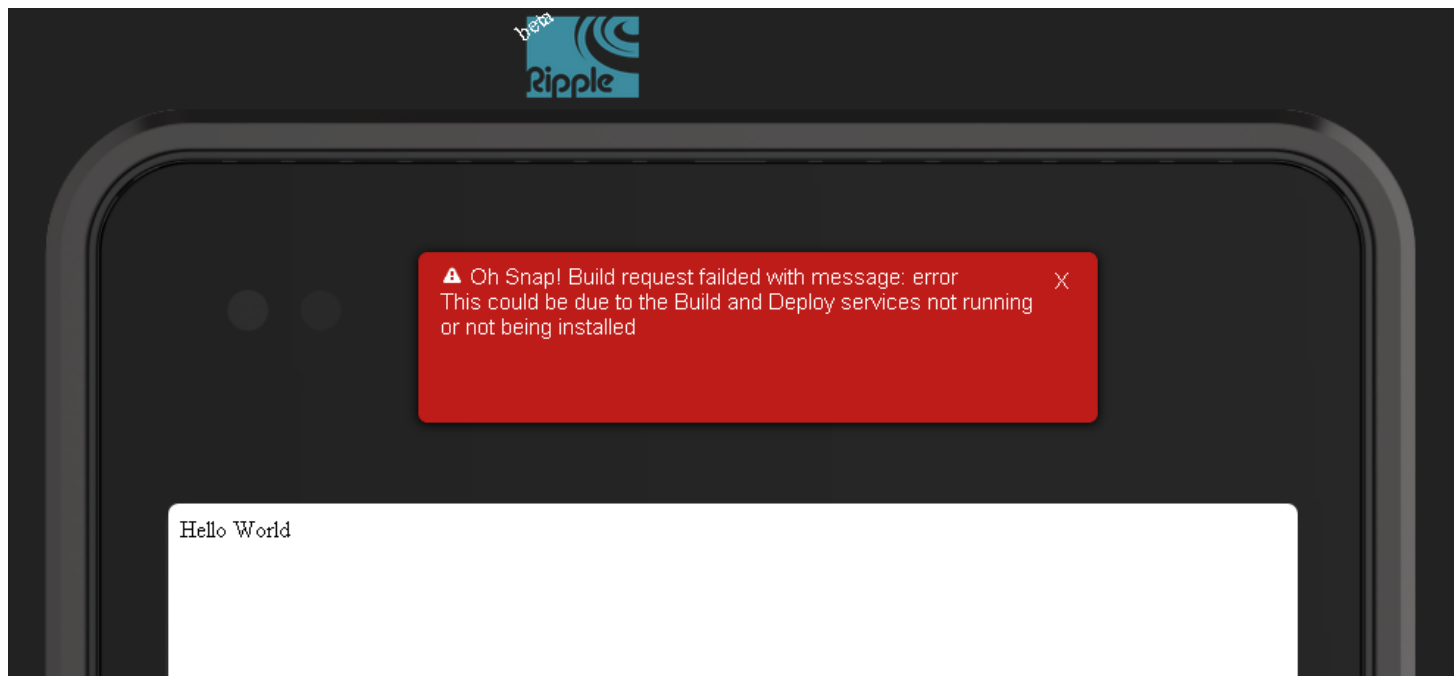
 **BlackBerry** 10 Jam



- Two BAR files created
- Deployment targets:
  - device
  - simulator



- Troubleshooting:
  - ▶ What happens if **ripple-services** isn't running?

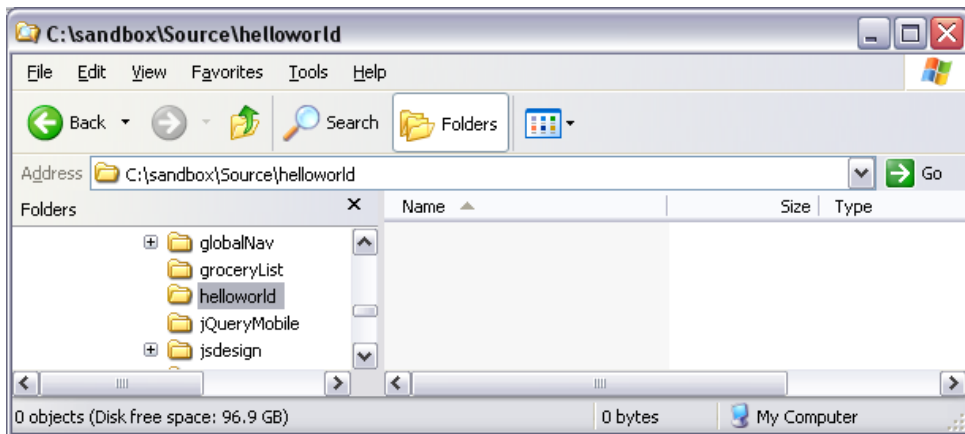


## 8. Creating your 1<sup>st</sup> app

Hello World?

# Create a project folder

- Create a new folder named **helloWorld**



- Hints:
  - ▶ Put in working directory of web server.
  - ▶ Tutorial: “Creating your first application” <http://bit.ly/J3N6sS>



# Create the main page

- Create a new file in **helloworld** folder
  - ▶ Name it **index.html**

```
<!DOCTYPE html>
<html>
<head>
    <title>Hello World</title>
</head>
<body>
    <div id="message">Hello World</div>
</body>
</html>
```

# Create the configuration file

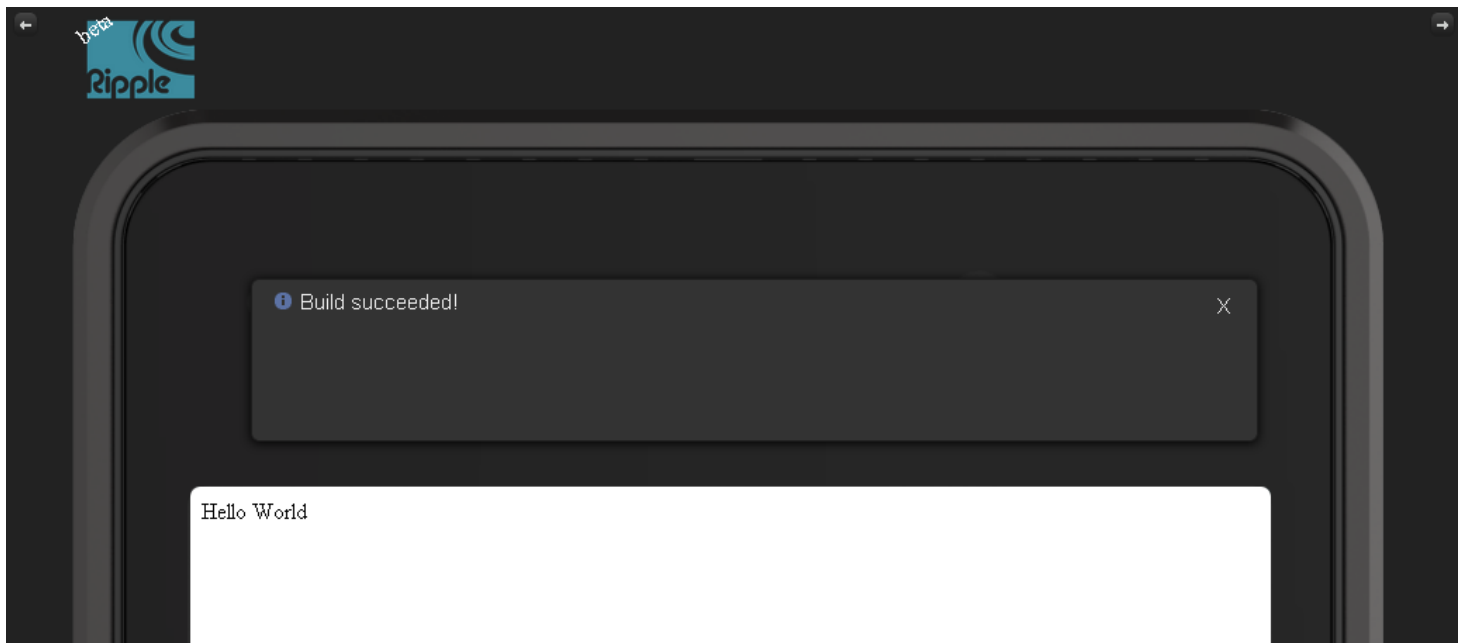
- Create a new file in **helloworld** folder
  - ▶ Name it **config.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<widget xmlns="http://www.w3.org/ns/widgets"
        version="1.0.0.1"
        id="helloWorld">

    <name> Hello World </name>
    <content src="index.html" />
    <author>Your name here</author>

</widget>
```

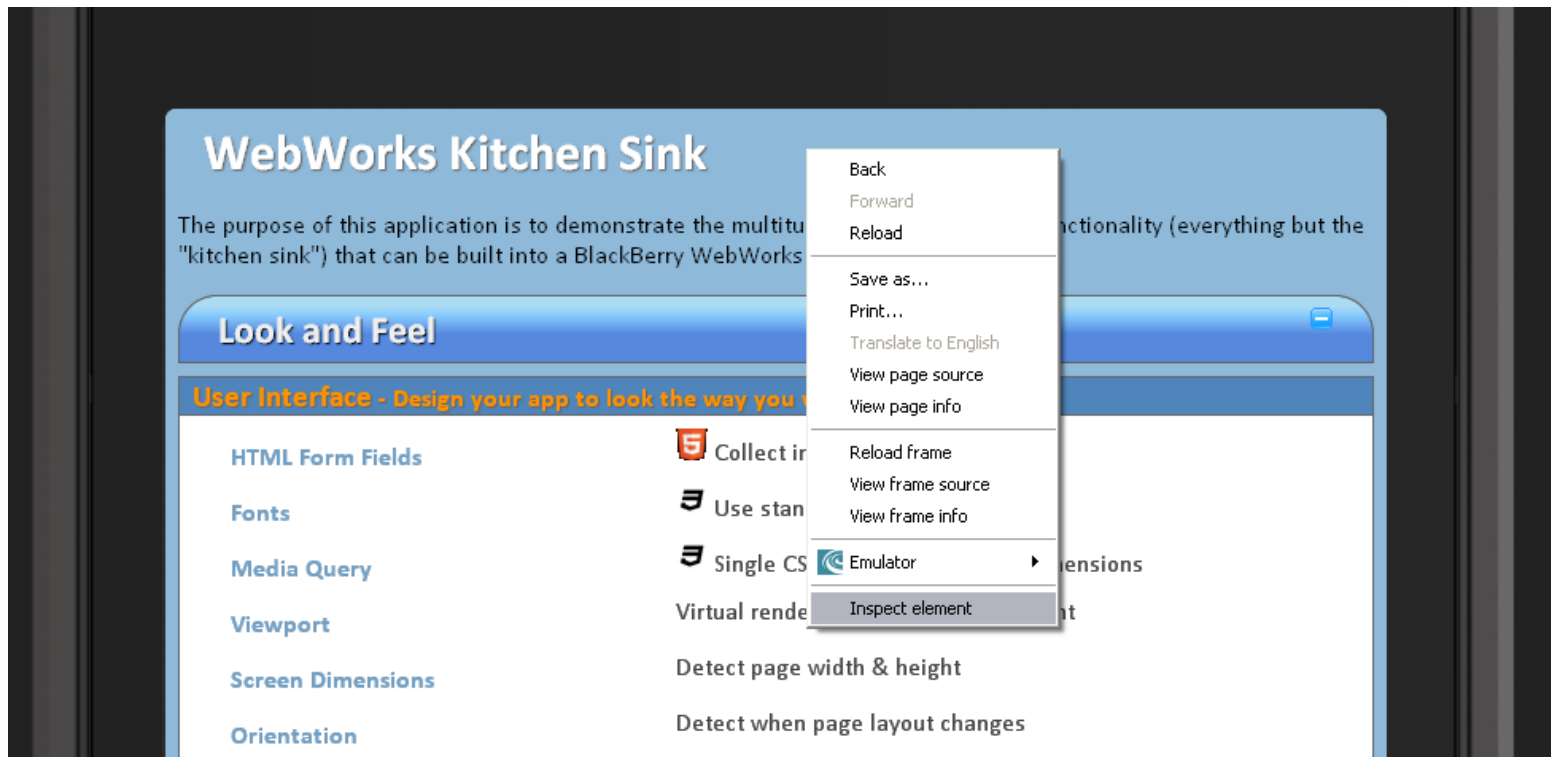
- Configure Ripple to build **helloWorld** into an application
  - ▶ Using WebWorks for BB10



## 9. Debugging

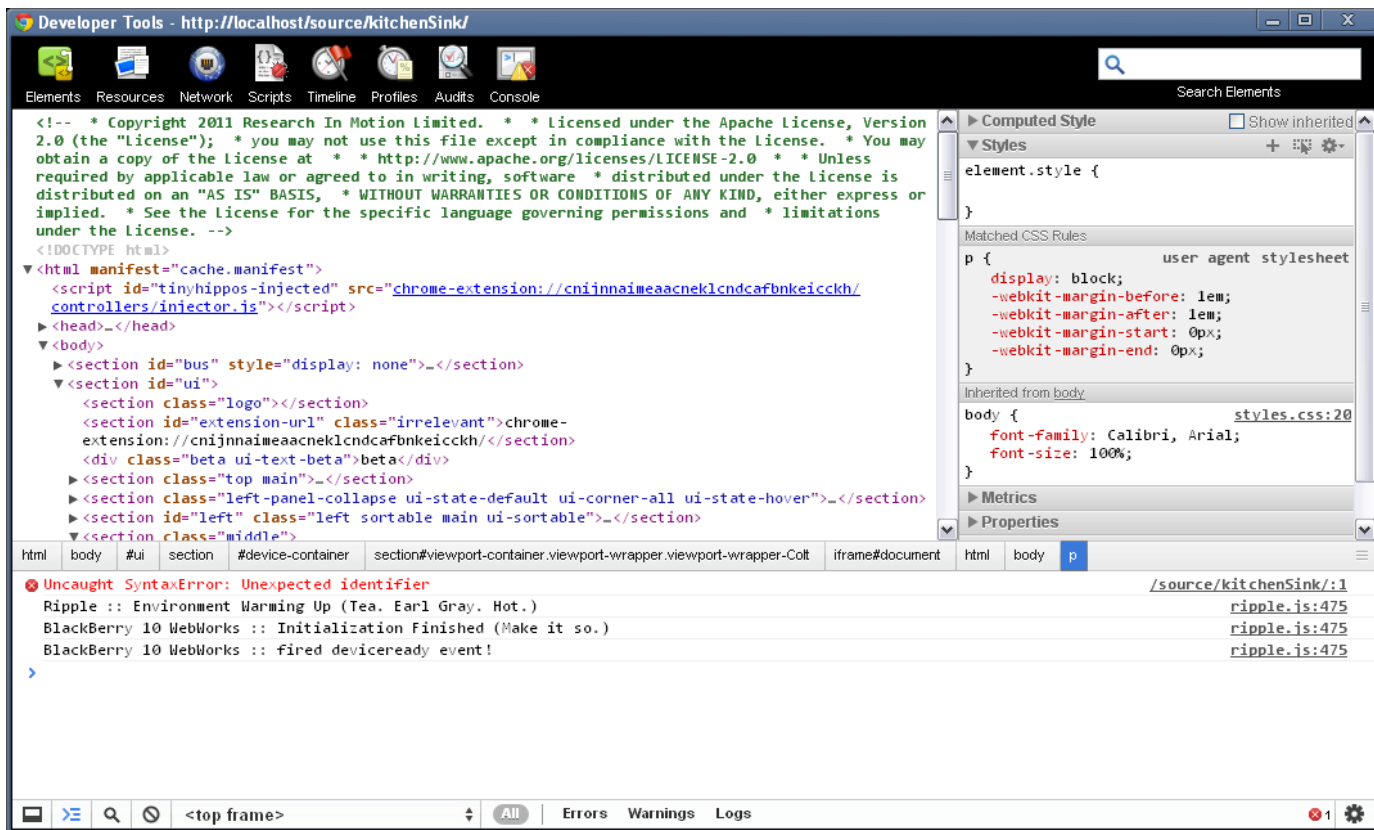
Use Web Inspector to troubleshoot bugs and performance issues.

- Right click content window → “Inspect Element”



# Web Inspector debugging

 BlackBerry 10 Jam



The screenshot displays the BlackBerry 10 Web Inspector interface. The top toolbar includes icons for Elements, Resources, Network, Scripts, Timeline, Profiles, Audits, and Console. The main content area shows the HTML document structure, with a syntax error highlighted: "Uncaught SyntaxError: Unexpected identifier". The error message is "Uncaught SyntaxError: Unexpected identifier" and the location is "/source/kitchenSink:1". The console log shows the following messages:

- Ripple :: Environment Warming Up (Tea. Earl Gray. Hot.)
- BlackBerry 10 WebWorks :: Initialization Finished (Make it so.)
- BlackBerry 10 WebWorks :: fired deviceready event!

The right-hand pane shows the "Computed Style" panel, displaying the "element.style" and "Matched CSS Rules" for the selected element. The "Matched CSS Rules" section shows the "user agent stylesheet" with the following rules:

- display: block;
- webkit-margin-before: 1em;
- webkit-margin-after: 1em;
- webkit-margin-start: 0px;
- webkit-margin-end: 0px;

The "Inherited from body" section shows the following rules:

- font-family: Calibri, Arial;
- font-size: 100%;

The bottom status bar shows the current frame as "<top frame>" and the number of errors as 1.

- Web Inspector features:
  - ▶ Elements – DOM elements and properties.
  - ▶ Resources –images, scripts, css, storage, cookies, cache.
  - ▶ Network – HTTP request and response data.
  - ▶ Scripts – step through JavaScript.
  - ▶ Timeline – download speeds.
  - ▶ Profiles – CPU and memory load.
  - ▶ Audits – feedback about network and web page performance.
  - ▶ Console –error and log statements.

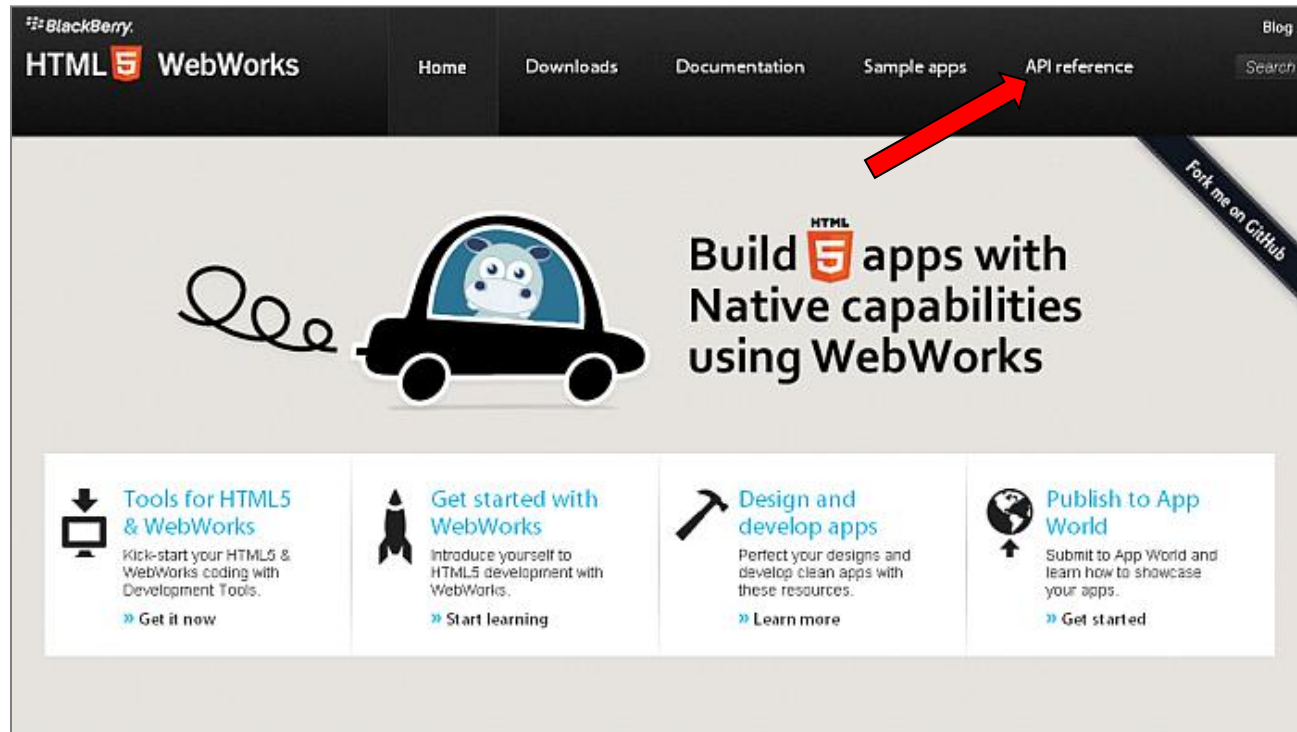
# 10. WebWorks APIs

Connecting your HTML5 content with native BlackBerry 10 device capabilities



# BlackBerry 10 WebWorks APIs

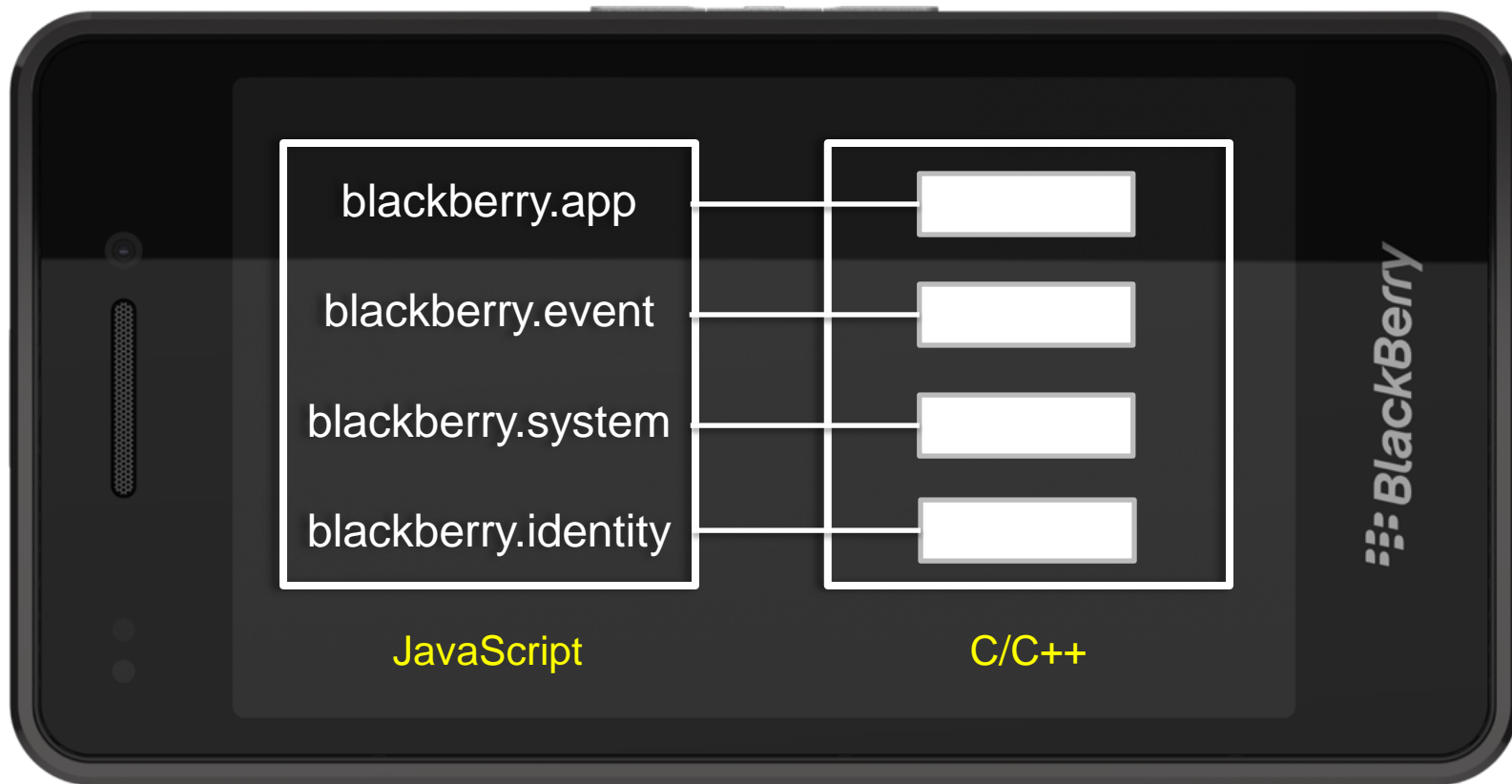
 **BlackBerry** 10 **Jam**



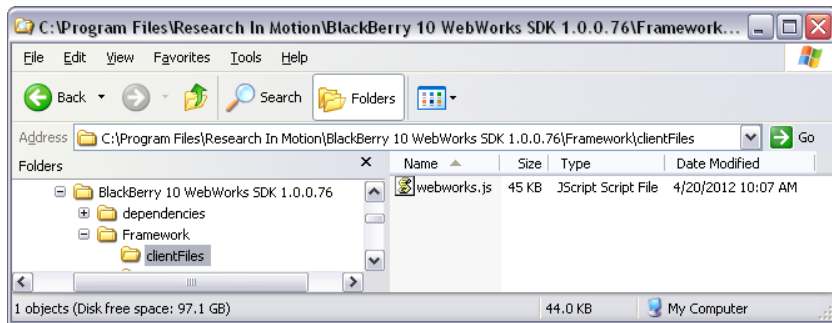
<http://developer.blackberry.com/html5/api>

# BlackBerry 10 WebWorks APIs

 **BlackBerry** 10 **Jam**



- Add **webworks.js** to your project
  - ▶ Copy from ../BlackBerry 10 WebWorks SDK <version>/framework/clientFiles/



- Add a reference to **webworks.js** in your code
  - ▶ Best practice: Put JS at the end of your HTML page

```
<script src="webworks.js"></script>
```

- Initialize the **webworks.js** framework:
  - ▶ Must create a handler for **webworksready** event
- Only use WebWorks APIs *after* this event has occurred

```
<script>
    function ready() {
        //APIs are now available
    }
    window.addEventListener("load", function(e) {
        document.addEventListener("webworksready", ready);
    })
</script>
```

# Task: Display uuid (PIN)

- Read and display device **uuid** in helloWorld application
  - ▶ **uuid** = Universally unique identifier
  - ▶ Equals the BlackBerry PIN, but using Cordova API signature
- Hints:
  - ▶ Add a feature element for blackberry.identity to **config.xml**
  - ▶ Use JavaScript to read blackberry.identity.uuid
  - ▶ Display **uuid** in a new DIV element on the page

# Solution: Display uuid (PIN)

- config.xml:

```
<?xml version="1.0" encoding="UTF-8"?>
<widget xmlns="http://www.w3.org/ns/widgets"
        version="1.0.0.1"
        id="helloWorld">

    <name> Hello World </name>
    <content src="index.html" />
    <author>Your name here</author>
    <feature id="blackberry.identity" required="true"
            version="1.0.0.0"/>

</widget>
```

# Solution: Display uuid (PIN)

- index.html:

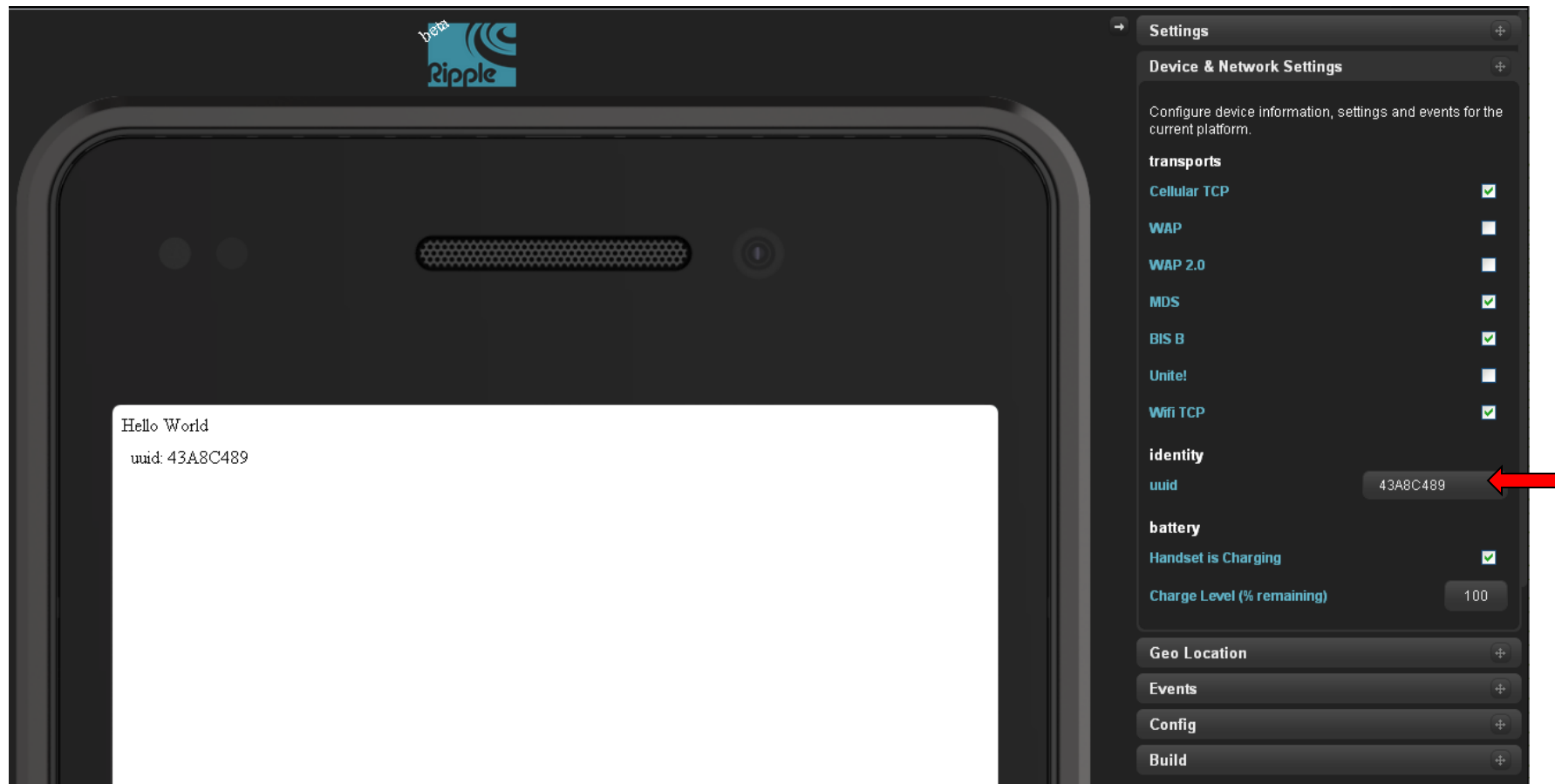
```
...
<script type="text/javascript" src="webworks.js"></script>

<script>
    function ready() {
        var ele = document.createElement("div");
        ele.innerHTML = "uuid: " + blackberry.identity.uuid;
        document.documentElement.appendChild(ele);
    };

    window.addEventListener("load", function(e) {
        document.addEventListener("webworksready", ready);
    }, false);
</script>
...
```

# Task: Display uuid (PIN)

 **BlackBerry** 10 *Jam*

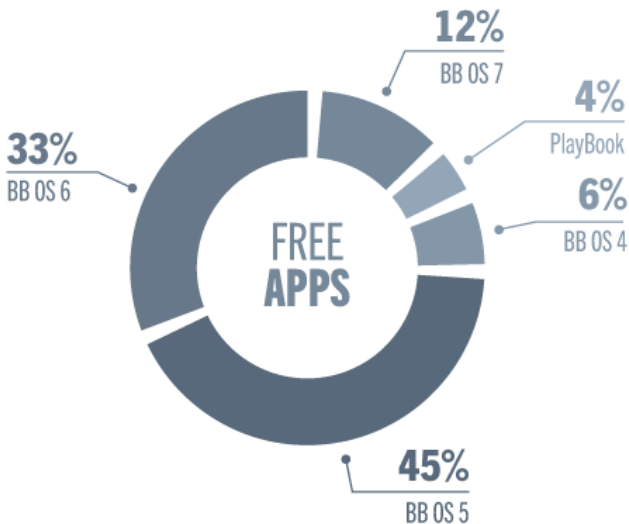




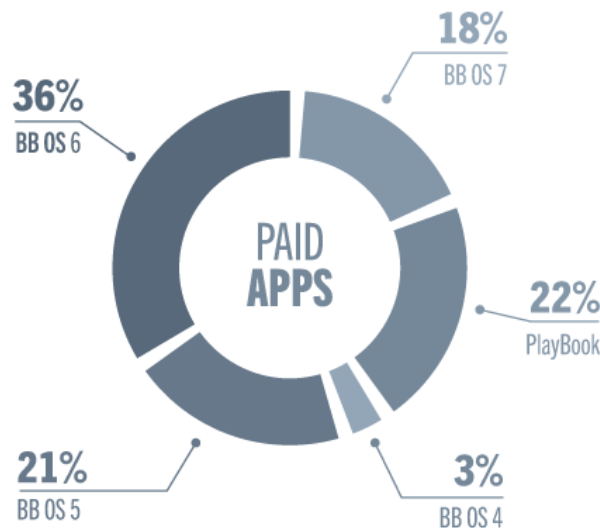
# 11. Deployment

How to get your BAR file on a simulator or live device

- BlackBerry App World
  - ▶ 5.5 million daily downloads
  - ▶ 40+ carriers supporting carrier billing



NOV 1, 2011 - FEB 1, 2012

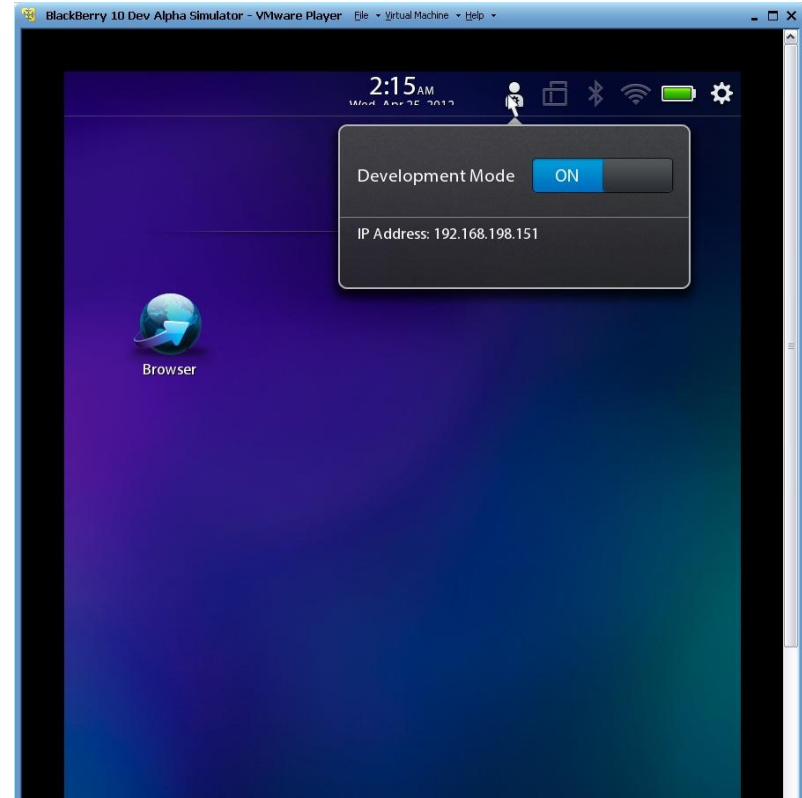


- What about deploying apps directly?
  - ▶ Live devices
  - ▶ simulators

- BlackBerry 10 Dev Alpha simulator
  - ▶ BlackBerry10Simulator.vmx
  - ▶ C:\Program Files\Research In Motion\BlackBerry 10 WebWorks SDK <version>\simulator
- VMware player is available from:
  - ▶ <http://www.vmware.com/products/player>

# Deployment

- Enable development mode
  - ▶ Open security settings
  - ▶ Enter system password
  - ▶ IP address



- Use Ripple to deploy unsigned app to simulator
  - ▶ Enter IP address and Password in settings screen
  - ▶ Select “Package & Launch” option
  - ▶ Deploy to VMWare simulator

# Deployment

 **BlackBerry** 10 Jam

**Settings** ✕

Package

Build

SDK Path

C:\Program Files\Research In Motion\Blac

Project Root

c:\sandbox\source\kitchensink

Archive Name

kitchensink

Output Folder

c:\temp

Enable Remote Web Inspector

☒

Sign

Signing Password

Bundle Number

Launch

Device IP

192.168.1.7

Device Password

.....

→ **Settings** ⊕

Device & Network Settings

⊕

Geo Location

⊕

Events

⊕

Config

⊕

Build

⊕

Package

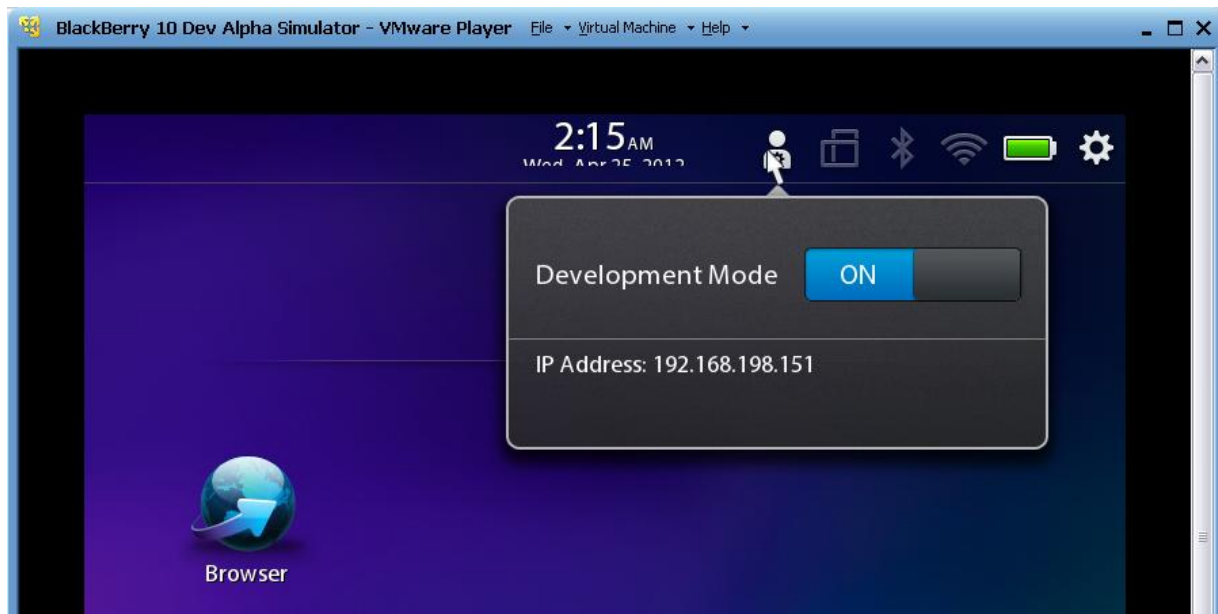
Package & Sign

Package & Launch

Settings...

# Task:

- Deploy kitchenSink and helloWorld to simulator
- Hint:
  - ▶ Don't forget to enable development mode





- Use blackberry-deploy to side-load a signed app
  - ▶ Command line tool found in
    - C:\Program Files\Research In Motion\BlackBerry 10 WebWorks SDK <version>\dependencies\tools\bin
- Deploy to
  - ▶ a live device (app must be signed)
  - ▶ a simulator (app does not have to be signed)

```
blackberry-deploy -installApp -device <Device IP> -package  
<Compiled BAR> -password <Device PWD>
```

```
C:\Program Files\Research In Motion\BlackBerry 10 WebWorks SDK  
1.0.0.76\dependencies\tools\bin>blackberry-deploy -installApp  
-device 192.168.198.134 -package "c:\temp\kitchenSink.bar"  
-password 1234
```

```
Sending Install request...
```

```
Info: Action: Install
```

```
Info: File size: 40731
```

```
Info: Installing ...
```

```
actual_dname::DEV8281a833da63a6b7e2098dae6d0662e1.MjA5OG
```

```
RhZTZkMDY2MmUxICAgICA
```

```
actual_id::MjA5OGRhZTZkMDY2MmUxICAgICA
```

```
actual_version::1.0.0.0
```

```
result::success
```

# 12. Code signing

BlackBerry device capabilities

- Apps must be signed to run on a live device
  - ▶ Required in order to deploy to BlackBerry App World
- Register for keys
  - ▶ <http://developer.blackberry.com/html5/signingkey>
- Install keys
  - ▶ <http://bit.ly/uLLS0R>

- Open command prompt and navigate to
  - ▶ C:\Program Files\Research In Motion\BlackBerry 10 WebWorks SDK <version>\dependencies\tools\bin
- Install the keys:

```
blackberry-signer -register -csjpin <csj pin>  
                  -storepass <KeystorePassword> <client-RDK-xxxxxx.csj file>  
                  <client-PBDT-xxxxxx.csj file>
```

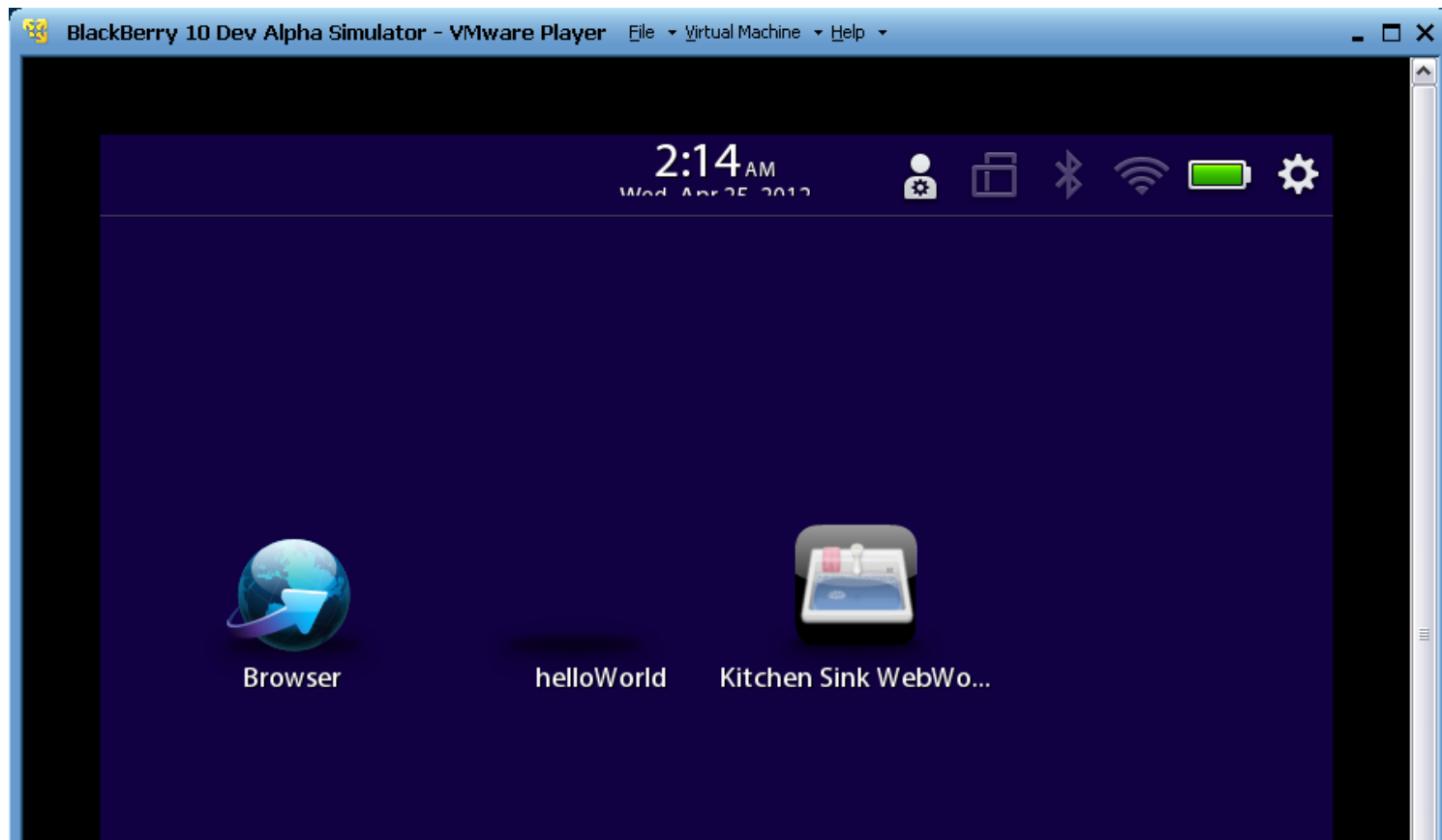
- Use Ripple to “Package & Sign”

# Task: Deploy your app

- Deploy kitchenSink or helloWorld to live device
- Hint:
  - ▶ Register for keys
  - ▶ Install keys
  - ▶ Don't forget to sign it

# Task: Deploy your app

 **BlackBerry** 10 Jam



# For more information

**BlackBerry** 10 Jam

The screenshot shows the BlackBerry HTML5 WebWorks developer website. The header is dark with the BlackBerry logo and 'HTML5 WebWorks' on the left, and navigation links (Home, Downloads, Documentation, Sample apps, API reference, Blog) and a search bar on the right. The main content area features a cartoon car with a blue cow inside, leaving a trail of loops. To the right of the car, the text reads 'Build HTML5 apps with Native capabilities using WebWorks'. A diagonal banner on the right says 'Fork me on GitHub'. Below this, there are four white boxes with icons and text: 1. 'Tools for HTML5 & WebWorks' with a download icon, text 'Kick-start your HTML5 & WebWorks coding with Development Tools.', and a 'Get it now' link. 2. 'Get started with WebWorks' with a rocket icon, text 'Introduce yourself to HTML5 development with WebWorks.', and a 'Start learning' link. 3. 'Design and develop apps' with a hammer icon, text 'Perfect your designs and develop clean apps with these resources.', and a 'Learn more' link. 4. 'Publish to App World' with a globe icon, text 'Submit to App World and learn how to showcase your apps.', and a 'Get started' link.

<http://developer.blackberry.com/html5>



# THANK YOU

DEV144

@n\_adam\_stanley, @ken\_wallis, @confusement

May 1-3, 2012