



Getting Started with HTML5 on BlackBerry

Tom Anderson

thanderson@blackberry.com

@Noctivagan

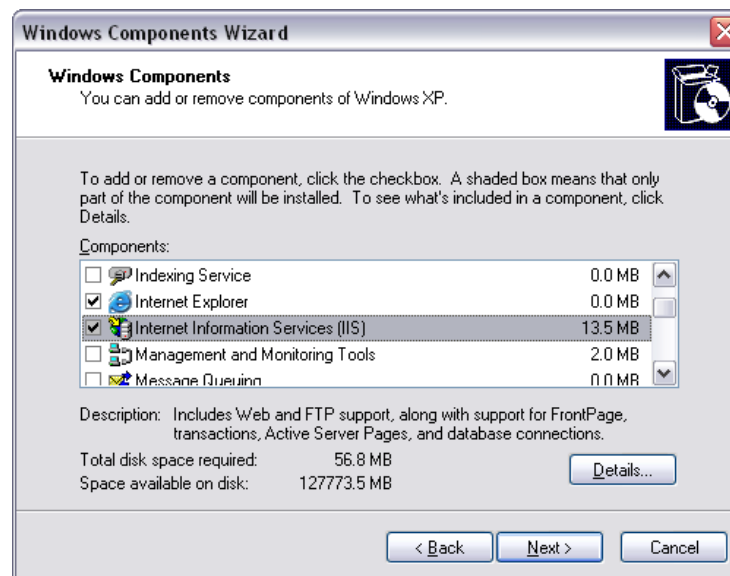


Intro to Tools and Environment

Setting up a web server



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



Setting up a web server



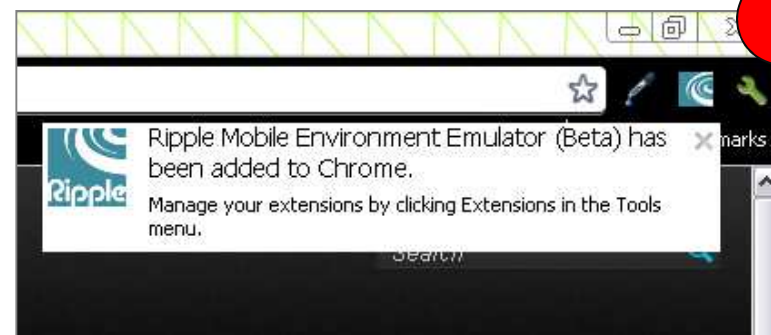
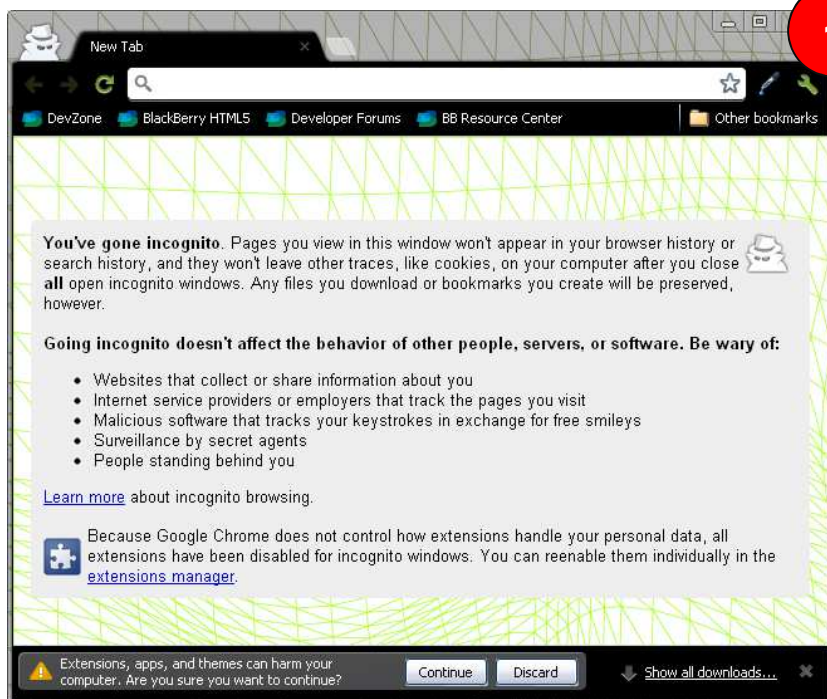
- Mac: Enable Web sharing
 - ▶ System Preferences → Sharing → Web Sharing



Install Ripple extension



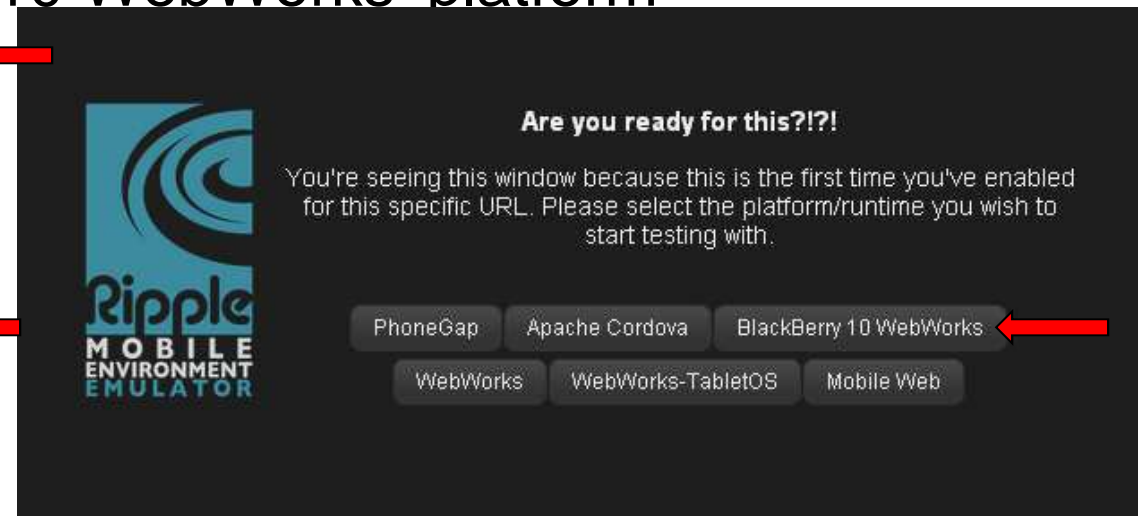
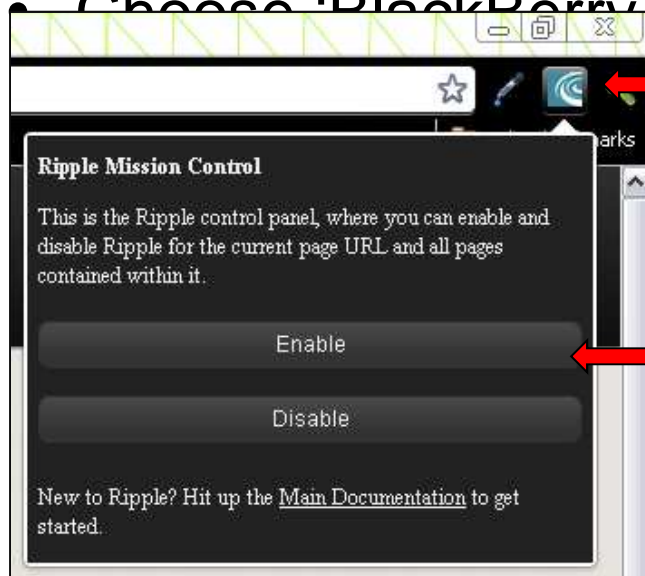
- Open your Google Chrome browser and go to **Tools > Extensions**.
- Drag **ripple_ui.crx** into the Chrome browser



Install Ripple extension



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



First App



First App



Hello

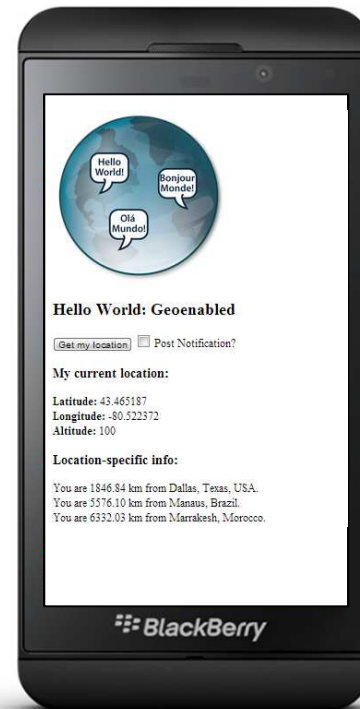


World

Anatomy of a WebWorks App



- Web Assets
- Configuration File
- WebWorks SDK
- Favorite Text Editor



Intro to config.xml



- An .xml file that contains the elements to define the BlackBerry WebWorks application namespace
- The name of the application
- Application permissions
- The start page
- The icons to use for the application.
- General information (author and description)
- Name: config.xml (yes, it is case sensitive)

Create config.xml



- User your editor to create a file named config.xml
- Put this file in the root of your Web Project
 - ▶ C:\inetpub\wwwroot\helloworld
- As the first two lines add:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<widget xml:lang="en" id="HelloWorld"
```

```
xmlns=http://www.w3.org/ns/widgets
```

```
xmlns:rim=http://www.blackberry.com/ns/widgets version="1.0.0.0">
```

Finish config.xml



- Add the application information

```
<name>Hello World</name>
```

```
<author>Noctivagan</author>
```

```
<description>Hello World location display</description>
```

```
<icon src="images/icon.png"/>
```

```
<content src="index.html"/>
```

```
</widget>
```

Final config.xml



```
<?xml version="1.0" encoding="UTF-8"?>
<widget xml:lang="en" id="HelloWorld"
xmlns=http://www.w3.org/ns/widgets
xmlns:rim=http://www.blackberry.com/ns/widgets version="1.0.0.0">
  <name>Hello World</name>
  <author>Noctivagan</author>
  <description>Hello World location display</description>
  <icon src="images/icon.png"/>
  <content src="index.html"/>
</widget>
```

Completed Source Code



- <https://github.com/Noctivagan/HelloWorld>

PUBLIC Noctivagan / HelloWorld Pull Request Unwatch Star 0 Fork 0

Code Network Pull Requests 0 Issues 0 Wiki Graphs Settings

BlackBerry Starter Session Sample

Clone in Windows ZIP HTTP SSH Git Read-Only <https://github.com/Noctivagan/HelloWorld.git> Read+Write access

branch: master Files Commits Branches 1 Tags

HelloWorld / 4 commits

Removing App.zip

Noctivagan authored just now latest commit 79805634a3

images	4 days ago	Initial Load [Noctivagan]
scripts	5 minutes ago	Adding Checkbox [Noctivagan]
.gitattributes	4 days ago	Initial Load [Noctivagan]
.gitignore	4 days ago	Initial Load [Noctivagan]
config.xml	5 minutes ago	Adding Checkbox [Noctivagan]
debugtoken.bar	4 days ago	Initial Load [Noctivagan]
index.html	5 minutes ago	Adding Checkbox [Noctivagan]
styles.css	4 days ago	Initial Load [Noctivagan]





Building Index.html



Walk before we run....



- Build the skeleton of our index.html
- Globe image at top (or whatever image you want)
- Title Line
- Get Location Button
- Save File to your root



Hello World: Geoenabled

Get my location

Here's mine...

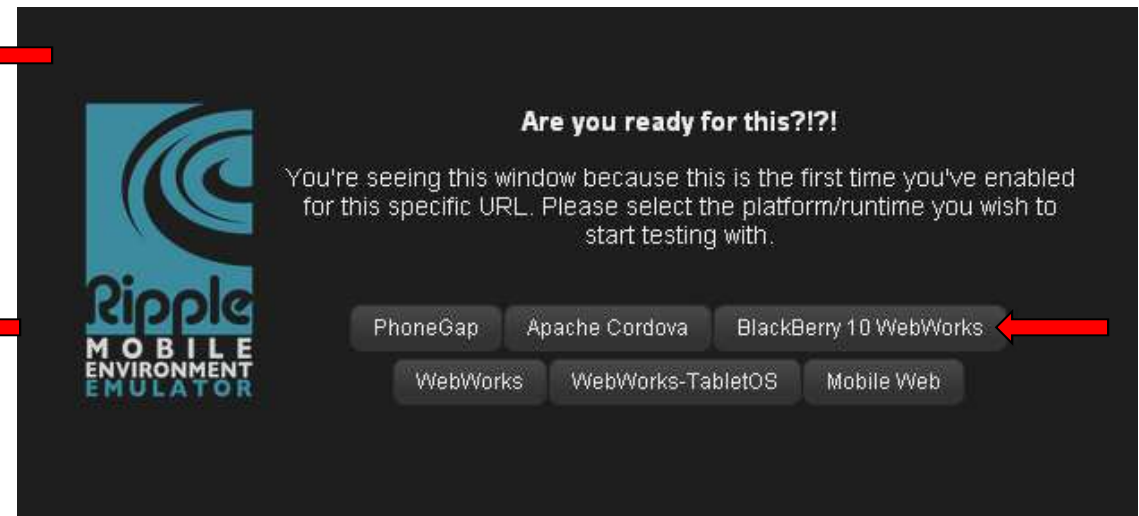


```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Hello World: Geo-enabled</title>
  <meta name="description" content="" />
  <meta name="author" content="thanderson" />
</head>
<body>
<div id="container">
  <div id="world">
    
  </div>
  <div id="button">
    <h2>Hello World</h2>
    <button type="button" id="btnGPSDefault">
      Get my location
    </button>
  </div>
</div>
</body>
</html>
```

Load your new app in Ripple



- Point Chrome to your root
 - ▶ <http://localhost/helloworld>
- Enable Ripple



Play around in Ripple



- Live load of your code
- Change the title in your code
- Refresh Ripple
- If you make changes to config.xml, need to clear cache



Initialize WebWorks



Initializing WebWorks



- Load WebWorks JavaScript file
- Wait for JavaScript to be ready
- Warning: Code in Ripple will work ***without initializing***

Add Reference



- Copy webworks-1.0.4.11.js to script directory
 - ▶ It's found in the WebWorks SDK client folder
 - ▶ C:\Program Files\Research In Motion\BlackBerry 10 WebWorks SDK 1.0.4.11\Framework\clientFiles
- Add a reference to the index.html

```
<script src="./scripts/webworks-1.0.4.11.js" type="text/javascript"></script>
```

Here's mine...



```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Hello World: Geo-enabled</title>
  <meta name="description" content="" />
  <meta name="author" content="thanderson" />
  <script src="/scripts/webworks-1.0.4.11.js" type="text/javascript"></script>
</head>
<body>
<div id="container">
  <div id="world">
    
  </div>
  <div id="button">
    <h2>Hello World</h2>
    <button type="button" id="btnGPSDefault">
      Get my location
    </button>
  </div>
</div>
</body>
</html>
```

Add Listener



- Can't access any of the groovy WebWorks stuff until it is ready
- Add the Event Listener in a script block

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


On your own...



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

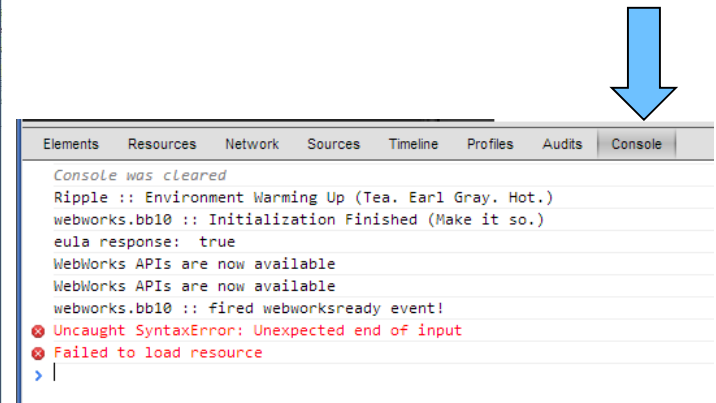
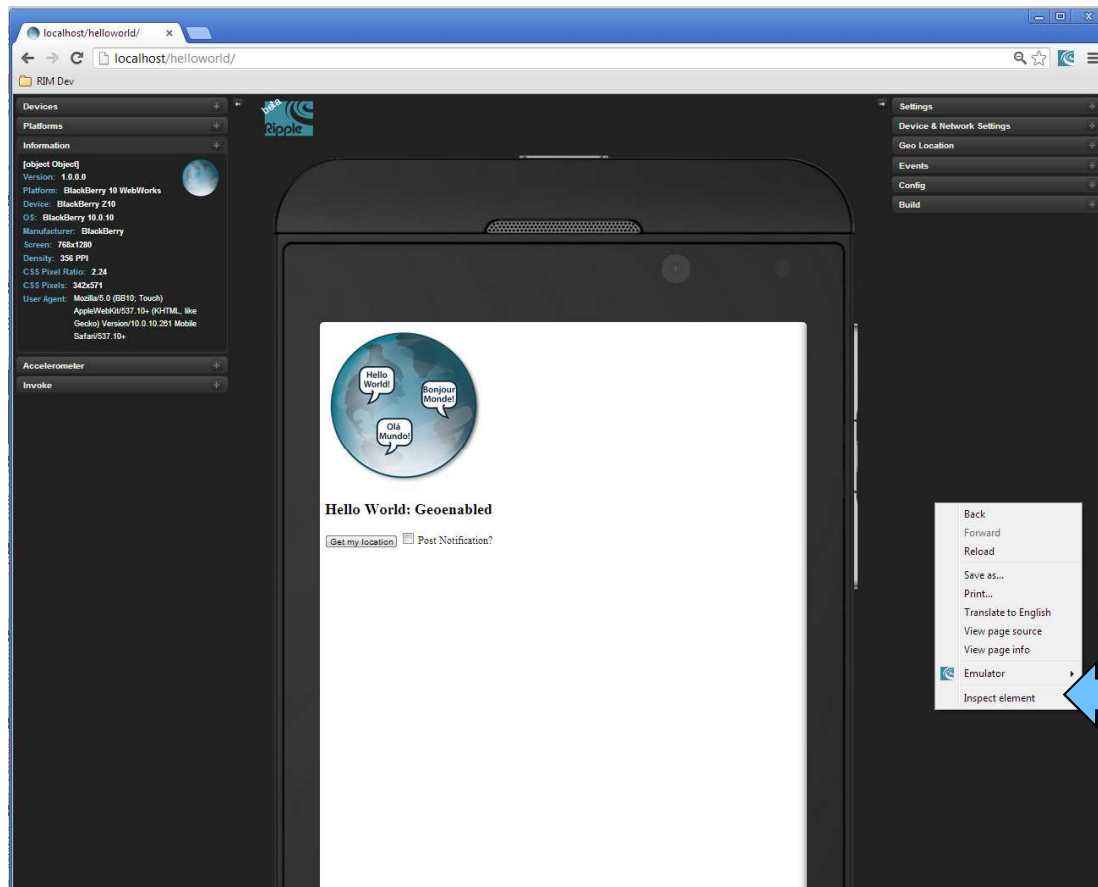
- Create something to happen in the ready event.

Here's mine...



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

Loading Console



Add Geolocation



Where are we?



- Add an onclick to the Button
- Get your location
- Show the coordinates (alert or console)
- Api signature
 - ▶ static void navigator.geolocation.getCurrentPosition
(positionCallback: function(position : Position),
*[positionErrorCallback: function([error: PositionError])], [options :
PositionOptions])*)
- <https://developer.blackberry.com/html5/apis/navigator.geolocation.html>

The call



```
if (navigator.geolocation !== null)
{
    navigator.geolocation.getCurrentPosition(geolocationSuccess, geolocationError, options);
}
else {
    errorMessage("HTML5 geolocation is not supported.");
}
```

Ripple...oops...



```
<?xml version="1.0" encoding="UTF-8"?>
<widget xml:lang="en"
id="HelloWorld" xmlns="http://www.w3.org/ns/widgets"
    xmlns:rim="http://www.blackberry.com/ns/widgets"
    version="1.0.0.0">
    <name>Hello World</name>
    <author>Noctivagan</author>
    <description>Hello World location display</description>
    <icon src="images/icon.png"/>
    <content src="index.html"/>
    <rim:permissions>
        <rim:permit>access_location_services</rim:permit>
    </rim:permissions>
</widget>
```

Demo Hello World



- Let's take a look at the geoenabled application



Geo.js



```
function getPosition(params)
{
    try
    {
        clearOutput();

        //First test to verify that the browser supports the Geolocation API
        if (navigator.geolocation !== null)
        {
            //Configure optional parameters
            var options;
            if (params)
            {
                options = eval("options = " + params + ";");
            }
            else {
                // Uncomment the following line to retrieve the most accurate coordinates available
                // options = { enableHighAccuracy : true, timeout : 60000, maximumAge : 0 };
            }
            displayOutput("<p><h3>Here we go....</h3></p>")
            navigator.geolocation.getCurrentPosition(geolocationSuccess, geolocationError, options);
        }
        else {
            errorMessage("HTML5 geolocation is not supported.");
        }
    }
    catch (e) {
        errorMessage("exception (getPosition): " + e);
    }
}
```

```
function geolocationSuccess(position)
{
    try
    {
        // The Position object contains the following parameters:
        // coords - geographic information such as location coordinates,
        //     accuracy, and optional attributes (altitude and speed).
        var coordinates = position.coords;

        // Now that we have the geographic information, what are some useful things that can be done with this info?

        // 1) Display current location information:
        displayLocationInfo(coordinates);

        // 2) Display content relevant to the users current location:
        // Identify whether a user is within range of a given location. This can be done by calculating their
        //     distance from a known location (within an allowable threshold of accuracy).
        displayContentForLocation(coordinates);

        // 3) Calculate relative direction to a point of interest
        // displayDirections(coordinates);
    }
    catch (e) {
        errorMessage("exception (geolocationSuccess): " + e);
    }
}
```

```
function displayLocationInfo(coordinates)
{
    try
    {
        var lat = coordinates.latitude;
        var lon = coordinates.longitude;
        var alt = coordinates.altitude;

        var locationInfo = "<h3>My current location:</h3>";
        locationInfo += "<b>Latitude:</b> " + coordinates.latitude + "<br/>";
        locationInfo += "<b>Longitude:</b> " + coordinates.longitude + "<br/>";
        locationInfo += "<b>Altitude:</b> " + coordinates.altitude + "<br/>";

        clearOutput();
        displayOutput("<p>" + locationInfo + "</p>");
    }
    catch (e) {
        errorMessage("exception (displayLocationInfo): " + e);
    }
}
```

Add Notification



Notifications



- Puts a notification in BlackBerry HUB
- Simple notifications
 - ▶ `new Notification("You have a simple notification");`
 - ▶ `new Notification("The title", {body: "The body"});`

Notifications

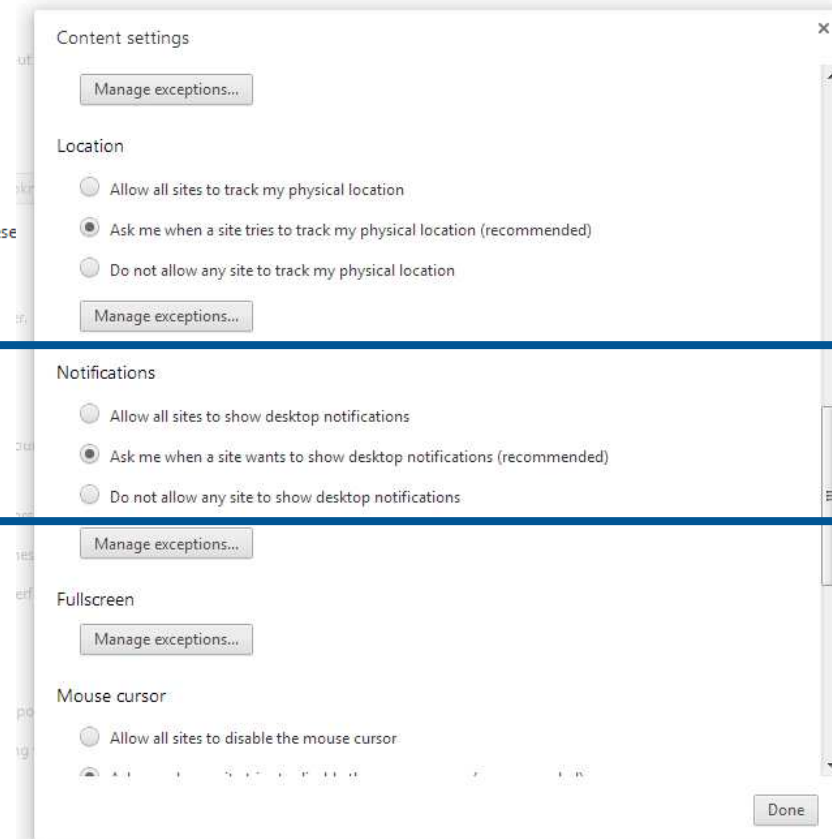


Privacy

Content settings...

Clear browsing data...

Google Chrome may use web services to improve your browsing experience. You may optionally disable these services. [Learn more](#)



Click: Manage Autofill settings

}

Notifications



- Puts a notification in BlackBerry HUB
- Simple notifications
 - ▶ `new Notification("You have a simple notification");`
 - ▶ `new Notification("The title", {body: "The body"});`
- Task:
 - ▶ Add a button to your page that posts a notification
- Fancy Task:
 - ▶ Add a checkbox to your page
 - ▶ Notification is posted when Get My Location button pressed if checkbox is checked

Notifications: Need Permission



- Add to your config.xml
 - ▶ `<rim:permit>post_notification</rim:permit>`
- How do you know when permissions are needed?

Notifications: Need Permission



- Add to your config.xml
 - ▶ `<rim:permit>post_notification</rim:permit>`
- How do you know when permissions are needed?
 - ▶ API Documentation will tell you

Permission Elements (PlayBook and BlackBerry 10+)

You must declare the permission element(s) below in your configuration document:

- `<rim:permit>post_notification</rim:permit>`
Permits your application to create notifications.



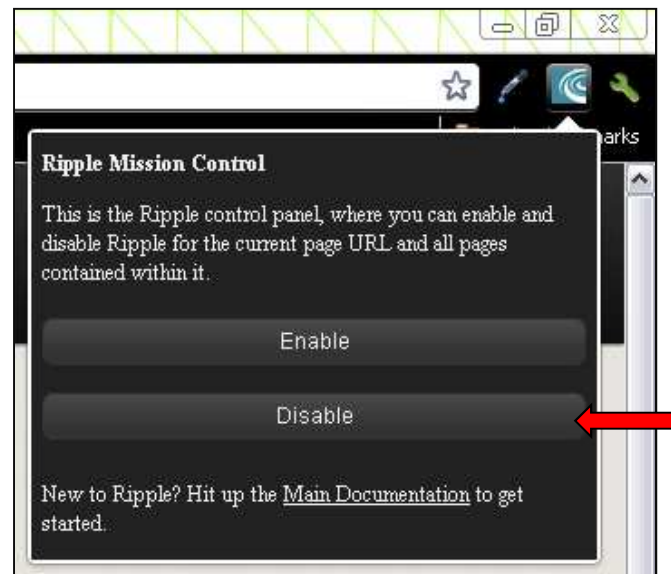
To WebWorks or Not to WebWorks



What happens?



- Load your app in Chrome.
- Turn off Ripple.
- Run your Application
- Did it work?



API Documentation



Event

Event

GPS

HTML5 Coordinates

HTML5 Geolocation

HTML5 Position

HTML5 PositionError

HTML5 PositionOptions

Identity

Identity

Phone

Service

Transport

Invoke

AddressBookArguments

BrowserArguments

Event

Event

GPS

HTML5 Coordinates HTML5 Geolocation HTML5 Position HTML5 PositionError

HTML5 PositionOptions

Identity

Identity Phone Service Transport

Invoke

AddressBookArguments BrowserArguments CalendarArguments CalendarComposerOptions

CalendarPickerOptions CameraArguments Card EmailComposerOptions FilePickerOptions

IcsViewerOptions Invoke Invoked JavaArguments MapsArguments MediaPlayerOptions

MemoArguments MessageArguments PhoneArguments Query Response

Query Response Target SearchArguments TaskArguments

- To call BlackBerry specific functions we need to register a feature
- Where do you suppose we do that?

- To call BlackBerry specific functions we need to register a feature
- Where do you suppose we do that?
 - ▶ config.xml

Device Information



- Feature is blackberry.identity
- Add to your config.xml:
 - ▶ `<feature id="blackberry.identity" />`
- Access with:
- `blackberry.identity.<property>`

Properties	
String	IMEI
String	IMSI
String	PIN
String	uuid

Here's mine...



```
<script>
function postNotification() {
    if (chkPostNotification.checked) {
        new Notification("Device Information", {
            body : "uuid: " + blackberry.identity.uuid + " IMEI: " +
                blackberry.identity.IMEI
        });
    }
}
</script>
```


Turn off Ripple







```
Elements Resources Network Sources Timeline Profiles Audits Console
✖ ▶ Uncaught SyntaxError: Unexpected end of input
✖ ▶ GET http://localhost:8472/extensions/get/?hash=75e5fd7c4b2ada1a71469c74ecd4bf9d&version=1.0.4.11
✖ ▶ Uncaught ReferenceError: blackberry is not defined
>
```

Deploying Applications



Request keys



Technologies Native HTML5 AIR Android Java Platform Partners	Programs Beta Zone Alliance programs Academic programs Developer groups BlackBerry 10 Dev Alpha Built for BlackBerry Apps	Support Code Signing Help Developer YouTube Channel Developer forums Developer Issue Tracker Knowledge Base	BlackBerry World Get your signing key Submitting your app	Follow us on:     Legal & Trademarks
---	--	---	--	---

Copyright © 2013 BlackBerry, unless otherwise noted.

<https://developer.blackberry.com/>

Request keys



<https://www.blackberry.com/SignedKeys>

BlackBerry Keys Order Form

BlackBerry Code Signing Keys are now absolutely free!

* Indicates a required field

☒ For BlackBerry OS 7.x and Lower

☐ I also require access to the secure element ([What's this?](#))

☒ For BlackBerry PlayBook OS and BlackBerry 10 and Higher

Personal Information

First name*

Last name*

Company*

Email*

Country *

Register your computer



https://developer.blackberry.com/html5/documentation/signing_setup_bb10_apps_2008396_11.html

- client-RDK-xxxxxx.csj - This is the file that you use to sign your applications and publish to the BlackBerry App World storefront.

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

- client-PBDT-xxxxxx.csj - This file is used to generate a debug token that you can use to test your application on a device.

```
blackberry-signer -register -csjpin 123456 -storepass rocket11  
C:\keys\client-RDK-4888888888.csj C:\keys\client-PBDT-4888888888.csj
```

Download Tools



<http://blackberryjamsessions.com/blackberryjamsessions/getting-ready-for-the-hackathon/html5/>
<http://tinyurl.com/bxurxpa>

- **BlackBerry Tablet OS SDK:** to create apps for the current BlackBerry PlayBook tablet and its Simulator.
- **BB10 WebWorks SDK:** to create apps for the BlackBerry 10 Dev Alpha device or BlackBerry 10 Simulator.
- **Ripple:** to emulate and package apps using Google Chrome.
- **BlackBerry Graphical Aid:** to visually install and build applications for the BlackBerry platform.
- **WebWorks Samples (with Kitchen Sink):** to see and test all functionality available on the WebWorks SDK with practical samples, so if you are creating an HTML5 app packaged with WebWorks, just copy and past the code to have GPS, Accelerometer, in-app-purchase and a lot more. You can view some sample apps [also here](#).



THANK YOU

Tom Anderson

thanderson@blackberry.com

@Noctivagan

