BlackBerry Jam Asia

Optimization and Performance for Web Developers

JAM843

Adam Stanley @n_adam_stanley

Justin Lee @triplez82

November 29, 2012

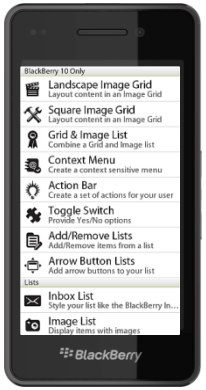
Intros and demo

What are we going to do today?

Today's Tasks

BlackBerry Jam Asia

- Lab Setup [10 mins]
- Task 1: Elements panel [20 mins]
- Task 2: Resources panel [20 mins]
- Task 3: Timeline and Network panels [25 mins]
- Task 4: Sources panel [25 mins]
- Task 5: Console panel [10 mins]
- Task 6: Profiles panel [10 mins]
- Bonus: Advanced features [10 mins]



Lab setup

Installing sample applications

[10 mins]

Lab requirements

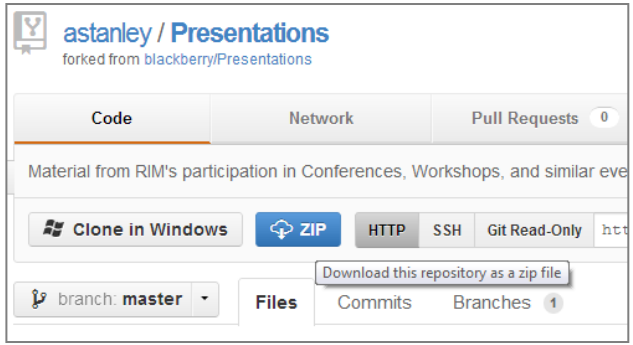
BlackBerry Jam Asia

- Desktop / Laptop:
  - Windows XP, 7 or Mac OS
  - Browser – Chrome or Safari
  - WebWorks SDK for BlackBerry 10
- BlackBerry 10 Dev Alpha
  - USB cable
  - No device? Use the BlackBerry 10 simulator

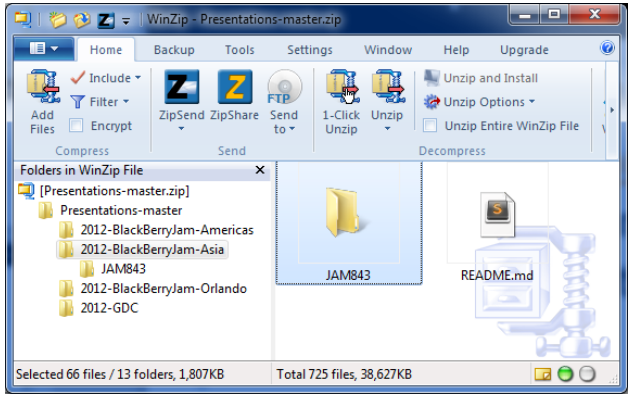
Lab setup

BlackBerry Jam Asia

- ***TASK:*** *Download and setup Lab materials*
  - https://github.com/astanley/Presentations
  - Lab presentation slides
  - Sample applications
  - Installer script



- Extract source files from ZIP
  - Open **Presentations-master.ZIP**
  - Browse to /2012-BlackBerryJam-Asia
  - Extract JAM843 folder to **C:\JAM843**



Lab setup

BlackBerry Jam Asia

- Enable development mode on your device
  - Settings → Security and Privacy → Development mode

- Connect your device to PC
  - USB
  - No connection step required if you are using a simulator

Lab setup

BlackBerry Jam Asia

- Load sample application

- Run loading script from an open command prompt:
  - Start → Run … → cmd
  - cd **C:\JAM843**
  - load.bat [device IP address] [device password]
    - E.g. **load.bat 169.254.0.1 pass**

- Confirm sample loaded successfully

Lab setup

BlackBerry Jam Asia

- Test connection to remote Web Inspector
  - Start WIC sample application (accept all prompts)

- Open a desktop browser
  - Browse to http://169.254.0.1:1337

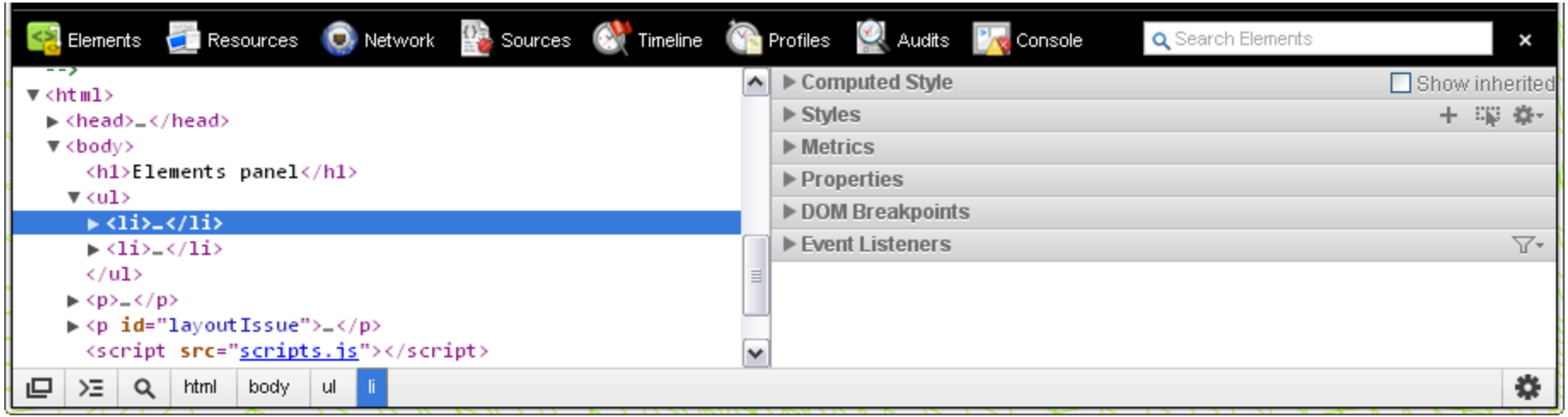
Task 1. Elements panel

Inspecting DOM elements and properties

[20 mins]

Task 1: Elements Panel

BlackBerry Jam Asia



- Interact with and edit the live Document Object Model (DOM)
  - Page markup displayed on the left.
  - Additional DOM sub panels on the right.

Task 1: Elements Panel

BlackBerry Jam Asia

- Start WIC sample application
  - Open “Elements” page

- Open http://169.254.0.1:1337 in Chrome
  - Click on “Web Inspector Companion”

  - Verifying your connection to an actual device
    - Click on different page elements in the markup view of web inspector
    - Observe how selected elements become highlighted **on the device**.

Task 1: Elements Panel

BlackBerry Jam Asia

- Live DOM editing
  - Very useful for rapid UI testing
  - Tip: expand or collapse any elements

```
▼<div data-bb-type="round-panel" cla
  <div data-bb-type="panel-header"
  ▶<div>…</div>
  <h2> Elements panel </h2>
  ▼<ul>
    ▶<li>…</li>
    ▶<li>…</li>
    </ul>
    <div data-bb-type="panel-header"
  ▶<div>…</div>
```

- ***TASK:*** *Modify page content*
  - Double click DOM elements in the markup view.
  - Change the contents of an element and press ENTER.
    - E.g. `<h2>Elements Panel</h2>` → `<h2>Let's rock and roll this!</h2>`
  - Click and drag a page element to reorder its position in the DOM.

Task 1: Elements Panel

BlackBerry Jam Asia

- Live DOM interaction
  - Click on the first <li> element

```
▶ <div>…</div>
  <h2>Elements panel</h2>
▼ <ul>
  ▶ <li>…</li>
  ▶ <li>…</li>
  </ul>
  <div data-bb-type="panel-header"
▶ <div>…</div>
```

- ***TASK:*** *Expand right-side sub panels*
  - Enable "**show inherited**" checkbox in the Computed Style view
    - What is the value of the inherited **background-color** style?
  - Locate 'Matched CSS Rules' section within Styles view
    - What happens when you deselect the "**float: left**" checkbox?
  - Properties view
    - What is the **clientWidth** value of the first **HTMLLIElement** item?

Task 1: Elements Panel

BlackBerry Jam Asia

- Discovering information about event listeners
  - Click on element

```
▶ <div>…</div>
▶ <div data-bb-type="button" id="noaction" cl
▶ <div data-bb-type="button" id="cantaction"
▶ <div data-bb-type="button" id="fullaction"
  </div>
</div>
▶ <div style="position: absolute; z-index: 100; w
opacity: 0; ">…</div>
```

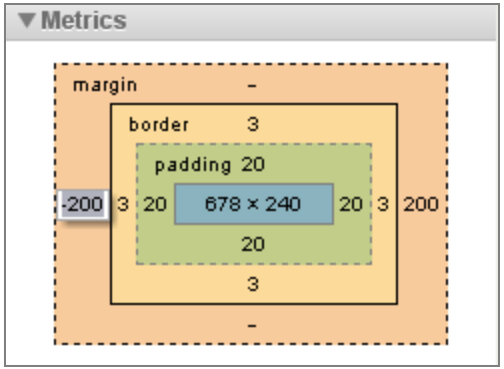
```
▶ Computed Style        ☐ Show inherited
▶ Styles                + 
▶ Metrics
▶ Properties
▶ DOM Breakpoints
▶ Event Listeners
```

  - ***TASK:*** *Open the* ***Event Listeners*** *view*
    - How many event listeners are set for this button?
    - Find the source of the event handler by expanding each event listener.

Task 1: Elements Panel

BlackBerry Jam Asia

- Use metrics to troubleshoot layout issues
  - Select the <p id="layoutIssue"> element markup
  - Metrics view displays margin, border and padding



  - ***TASK:*** *Open* ***Metrics*** *view*
    - What happens when the mouse is hovered on each region in Metrics view?
    - Why is the left side of this element hidden off the screen on the device?
    - Double click **left** margin value. Change number to a positive value.
    - What did this change do to the <p id="layoutIssue"> element **on device**?

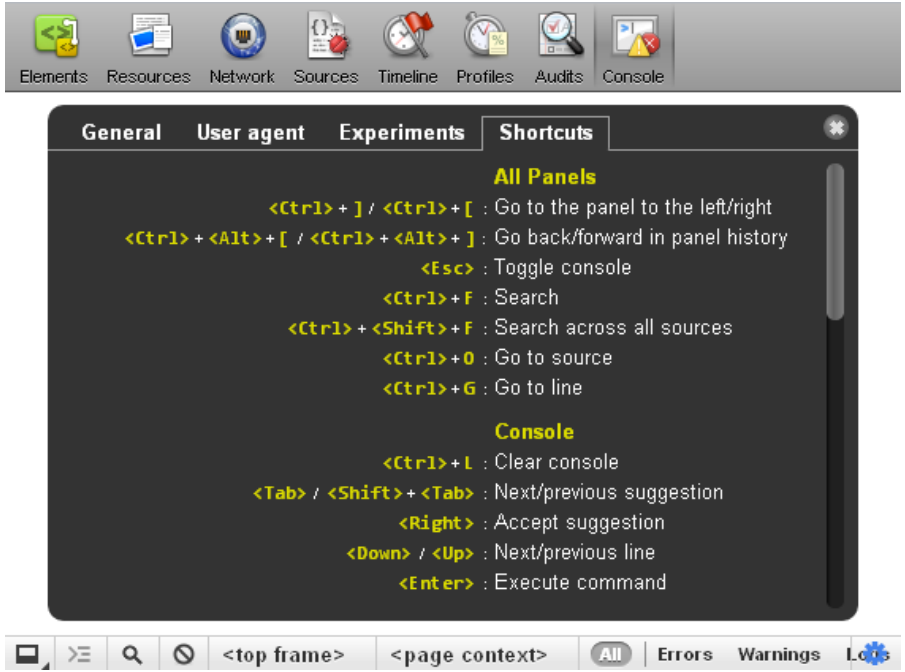
Bonus task: Short cuts

Using Web Inspector more efficiently

[5 mins]

Short cuts

- ***TASK:*** *Press F1 key within Web Inspector*



Task 2. Resources panels

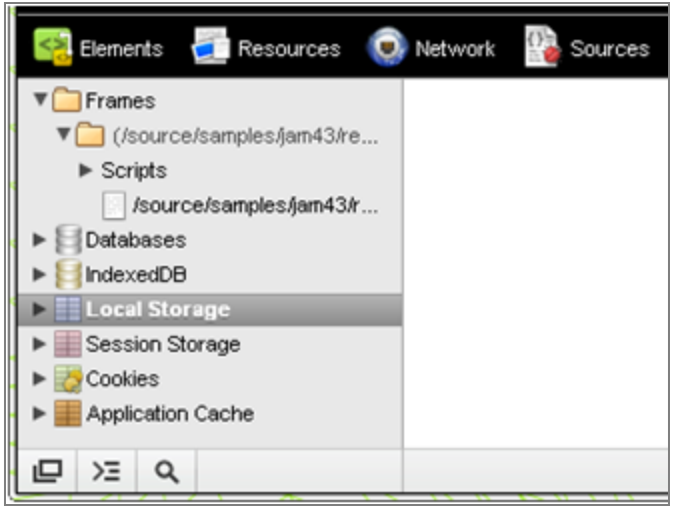
Inspecting page resources

[20 mins]

Task 2: Resources Panel

BlackBerry Jam Asia

- Start WIC sample application
  - Open “Resources” page
- Open http://169.254.0.1:1337 in Chrome
  - Click on “Web Inspector Companion”
- Can interact with page resources
  - Page files
  - Local storage / Web database
  - Application cache
  - Cookies



Task 2: Resources panels

BlackBerry Jam Asia

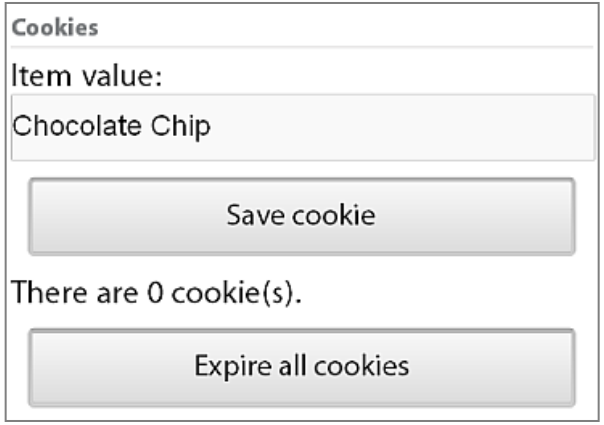
- Local storage
  - Enables offline application data

- ***TASK:*** *Viewing and modifying data*
  - Click on **Local Storage** item in resources panel
  - From device, click **Insert into localstorage** button
    - What happens to local storage results in resources panel?
  - Double click key or value, modify values and press ENTER
    - E.g. “Foo” → “Bacon”
  - Right click a local storage record and select **Delete**

Task 2: Resources panels

BlackBerry Jam Asia

- Interacting with cookies
  - HTTP header added to all requests

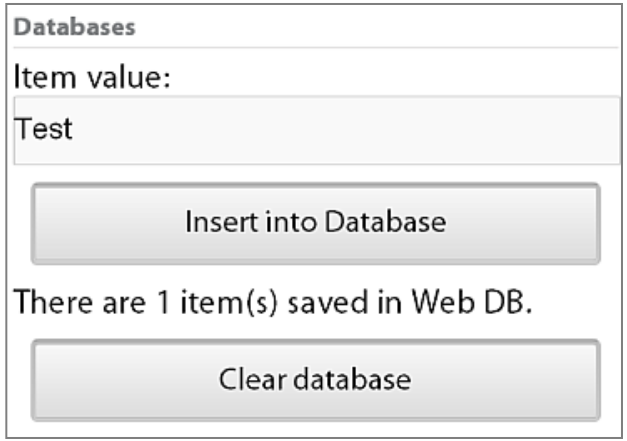


- ***TASK:*** *Viewing, deleting saved data*
  - Click on **Cookies** resource
  - From device, click **Save cookie** button
    - Where is the new cookie added to the resources panel?
  - Right click cookie and select **Delete**

Task 2: Resources panels

BlackBerry Jam Asia

- Interacting with Web DB storage
  - SQL database

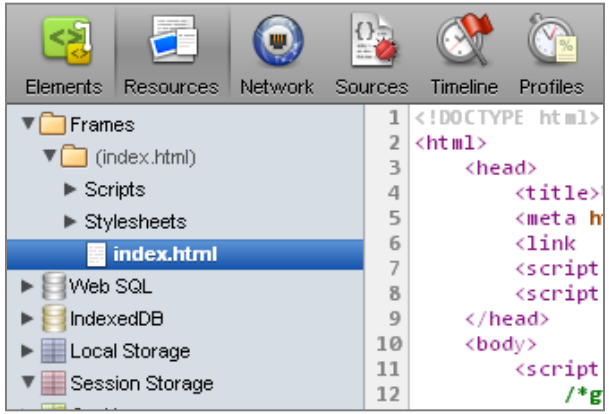


- ***TASK:*** *Viewing saved data*
  - Click on **Database** resource.
    - What database has been created?
    - What is the name of the newly created table?
  - From device, click **Save** button next to Web DB sample
    - What are the 3 fields of the newly created database record?

Task 2: Resources panels

BlackBerry Jam Asia

- Viewing frames tree
  - Helpful way to view raw page resources before they were modified by CSS or JavaScript



- ***TASK:*** *Expand Frames tree*
  - Select **index.html**
  - What framework is being used to manage the display of pages?
  - What is the name of the HTML file where the content for the elements page is stored?

Task 3. Timeline and Network panels

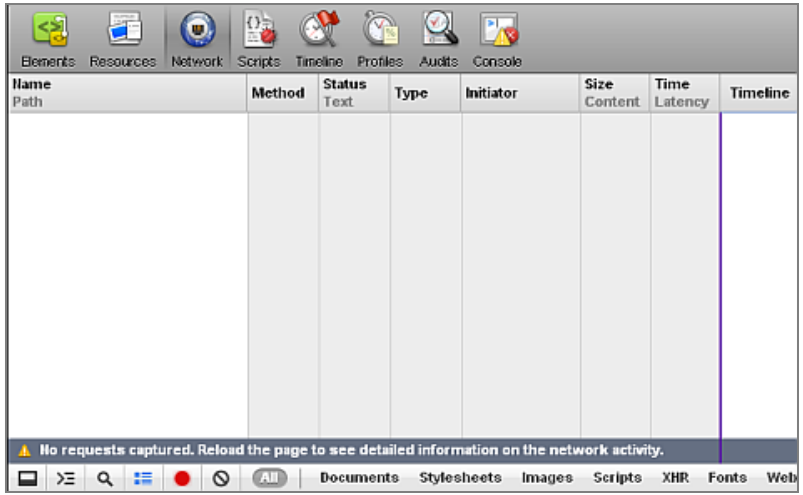
Measuring page and HTTP events

[25 mins]

Task 3: Timeline & Network

BlackBerry Jam Asia

- Start WIC sample application
  - Open “Timeline & Network ” page
- Open http://169.254.0.1:1337 in Chrome
  - Click on “Task 3: Timeline & Network Panels”
- Master the timeline:
  - Record
  - Filter Events
  - Save/Load
  - Drill Down



Task 3: Timeline & Network

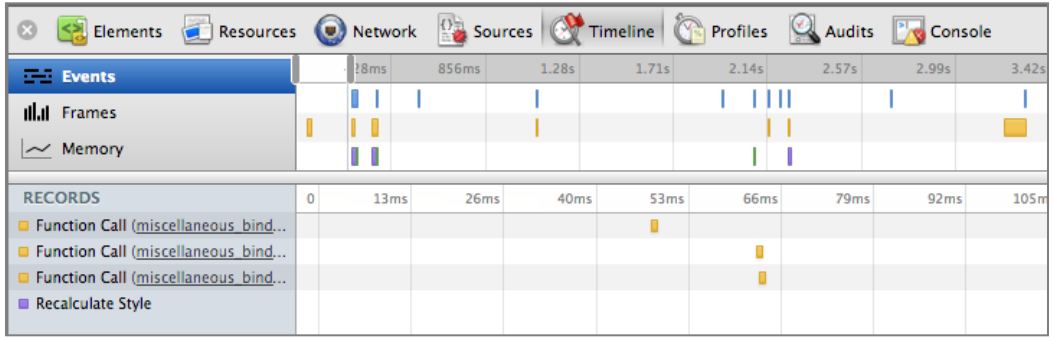
- Toolbar

- ***TASK:*** *Get to know the toolbar*
  - Click the **Record** button and scroll the page then click the button again.
    - Which events occur most often?
  - Filter out events shorter than 15ms.
    - Which events type is usually longer than 15ms? (most often)

Task 3: Timeline & Network

BlackBerry Jam Asia

- Timeline Events

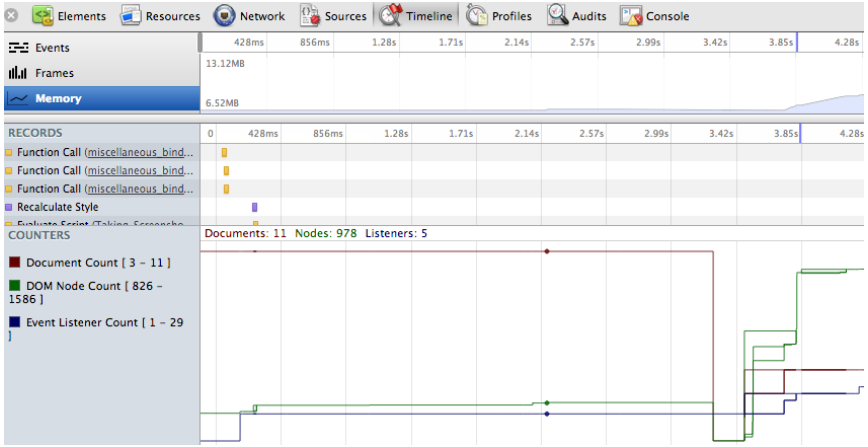


- ***TASK**: Get Event Details*
  - Record a timeline and hover over the events
    - What function name cause the final Layout after scrolling?
  - Remove Network events from the timeline.
  - Painting

Task 3: Timeline & Network

BlackBerry Jam Asia

- ***TASK**: Reading the memory view*
  - Record a timeline of going to the Main page and back to Sources.
  - Click on the memory view.
    - What's the highest amount of memory used?
    - What's the biggest Event Listener count?
    - How many Documents are there?



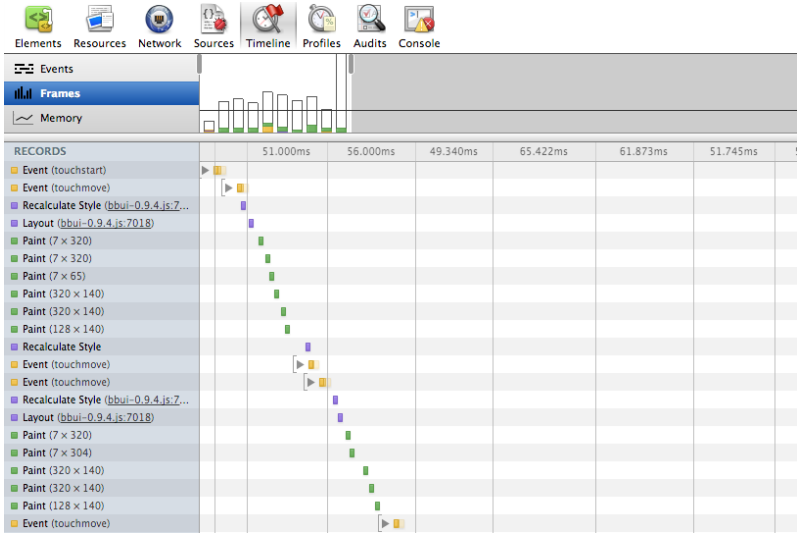
Task 3: Timeline & Network

BlackBerry Jam Asia

- ***TASK**: Reading the Frames View*

  - Click on the Frames View
    - Which frame is the longest/tallest?
    - How many milliseconds?
    - What's the average frame rate?

  - Select a group of Frames to compare
    - What contributed to the most to time?



Task 3: Timeline & Network

- Networking: seeing more detailed results
  - HTML request / response headers
  - Timings – how long does the request take
  - Cookies
  - Initiator column – seeing where resources came from
  - Load events
  - Viewing raw HTML send from the server, before it was modified by CSS or JavaScript
  - WebSocket Frames view
  - Filter Requests by Type

Task 3: Timeline & Network

BlackBerry Jam Asia

- ***TASK**: Networking - more details*
  - Record a timeline of navigating between pages (Main -> Sources -> Main -> Networking).
  - Click on the Network view.
    - How many document requests are there?
    - What's the status of the red request?
    - What caused the invalid request?
  - Click on the **Preview** pane.
  - Click on the **Response** pane.

Task 3: Timeline & Network

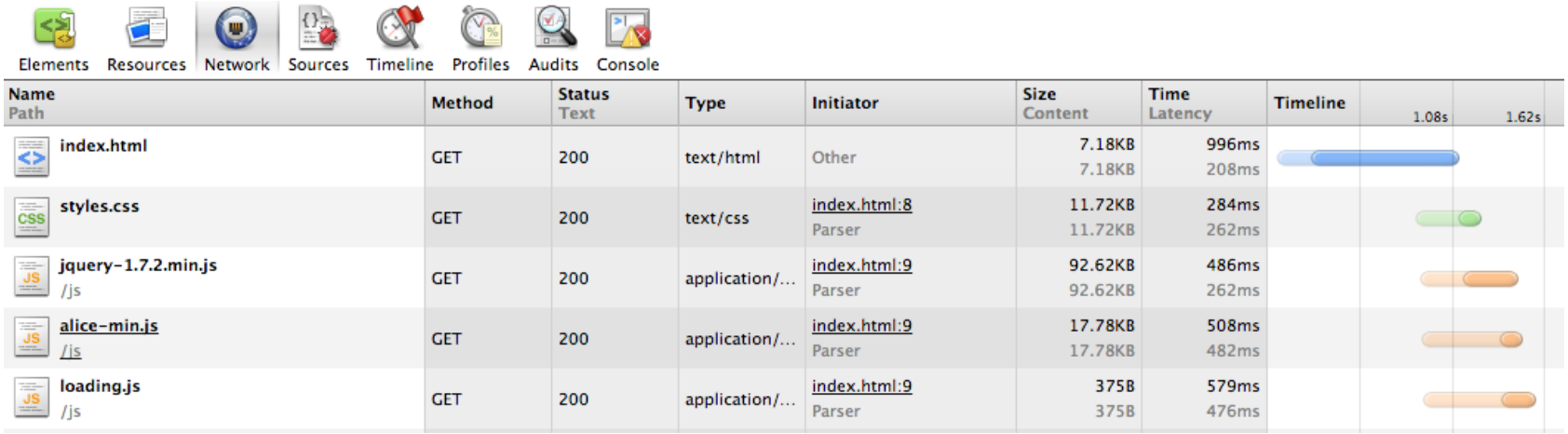
BlackBerry Jam Asia

- ***BONUS TASK**: Networking – Web Inspector*
  - In the inspector window press **CTRL + SHIFT + C**
  - In the newly opened Inspector window press **CTRL + R**
    - Do this in the **new** window not your old one.
  - Filter the requests by Web Socket.
    - What is the **Request Header** status?
    - How many Web Socket Frames do you see?

Task 3: Timeline & Network

BlackBerry Jam Asia

- Network Panel



Task 4. Sources panel

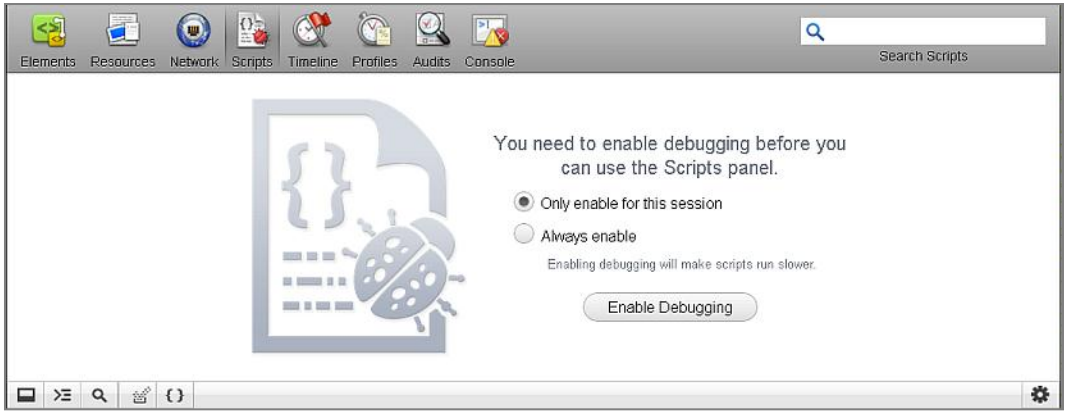
Setting breakpoints and stepping through JavaScript

[30 mins]

Task 4: Sources Panel

BlackBerry Jam Asia

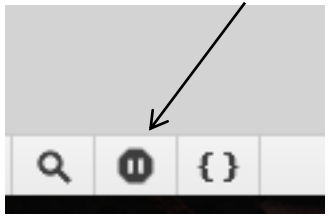
- Start WIC sample application
  - Open “Sources ” page
- Open http://169.254.0.1:1337 in Chrome
  - Click on “Task 4: Sources Panel”

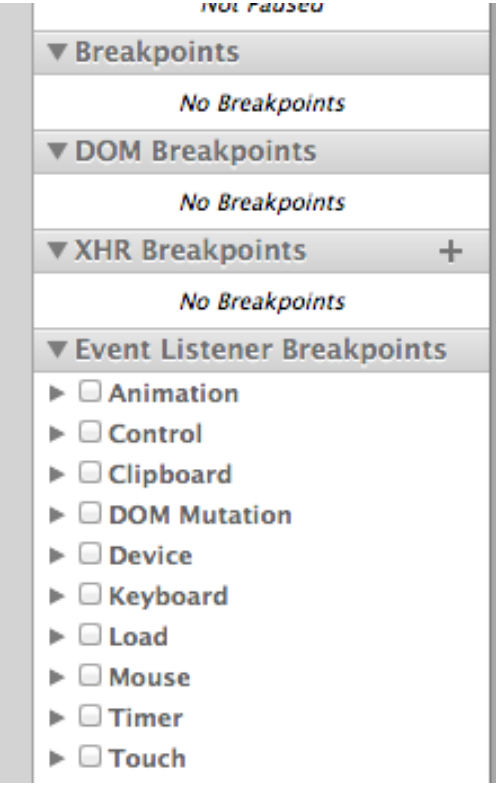


Task 4: Sources panel

BlackBerry Jam Asia

- ***TASK**: Set Normal Breakpoints*
  - *Open sources.js file in Sources.*
  - Click on line 43 of sources.js
    - What's method called this function?
    - What is the value of variable ***log***?



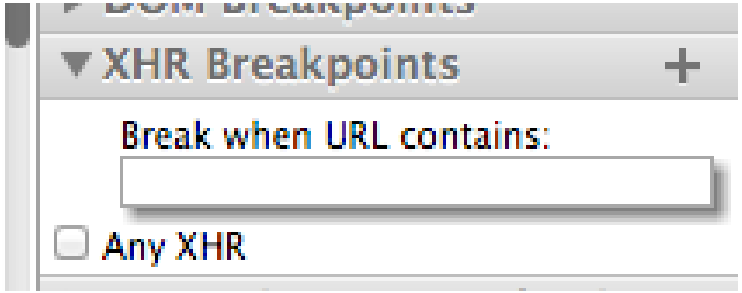


Task 4: Sources panel

BlackBerry Jam Asia

- ***TASK**: XHR breakpoints*

  - Open the XHR Breakpoints pane and click on the **Any XHR** checkbox.

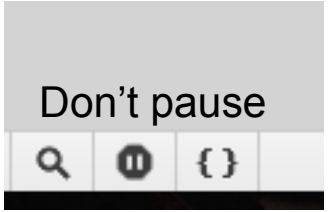


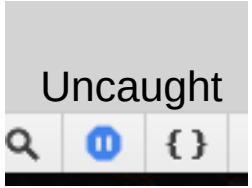
  - Navigate to a new page.
    - What's the call stack?
    - What are the values of the local variables?

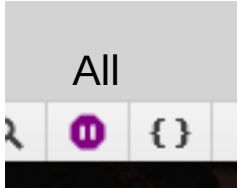
Task 4: Sources panel

BlackBerry Jam Asia

- ***TASK**: Exception breakpoints*
  - Click on the **Pause** button at the bottom of the **Sources Panel**.
  - Go to the main page and navigate back to **Sources** again.
    - What function did the exception occur on?
    - What line number?
    - What's the call stack look like?



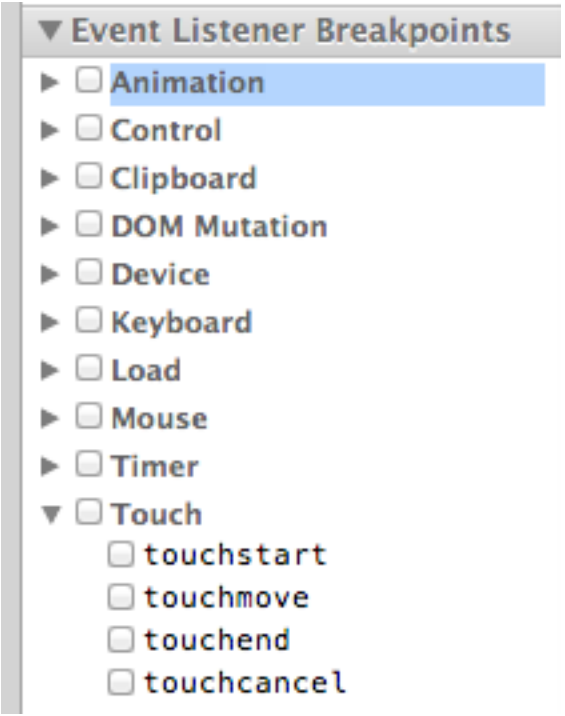




Task 4: Sources panel

BlackBerry Jam Asia

- ***TASK**: Event listener breakpoints*
  - Expand the **Event Listener Breakpoints** pane and expand the **Touch** section.
  - Check the **touchstart** event.
  - Tap on the page.
    - What happens when you're at a breakpoint and you type a variable name at the given scope in the console?
    - Does it match what's shown in the **Scope Variables** pane?

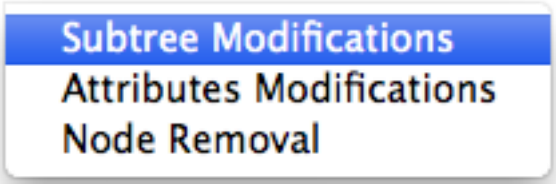


Task 4: Sources panel

BlackBerry Jam Asia

- ***TASK**: DOM breakpoints*

  - Right click on DOM Node in Elements panel.
  - Navigate back.
    - Observe the call stack.



  - Walk the call stack.
    - Do the variables

  - NOTE: Doesn’t work on HEAD, SCRIPT or META tag

Task 5. Console panel

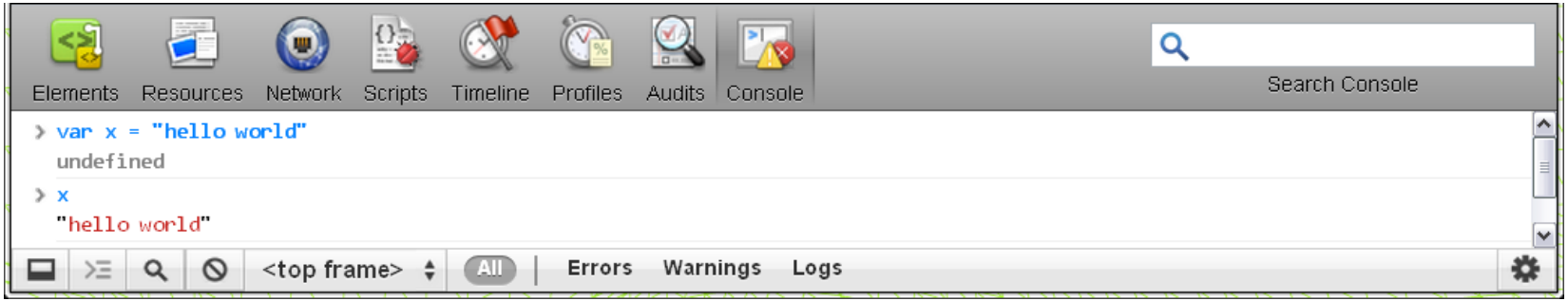
Interact with the application runtime

[10 mins]

Task 5: Console Panel

BlackBerry Jam Asia

- Start WIC sample application
  - Open “Console” page
- Open http://169.254.0.1:1337 in Chrome
  - Click on “Web Inspector Companion”



Task 5: Console panel

BlackBerry Jam Asia

- Viewing messages in the console

- ***TASK: application logging***
  - Open the console. Type **console.log("0 div 0")** and press **enter**
    - What output do you see?
  - Tap on the **Log Message** button in the sample application
    - What message(s) are displayed to the console?
    - Where in the JavaScript code did these message(s) come from?
  - Click the **Log Warning** button in the sample application
    - What do warning messages look like compared to log messages?

Task 5: Console panel

BlackBerry Jam Asia

- ***TASK:*** *Tracking down runtime errors quickly*
  - Click the **Open Browser** button from the sample application.
    - What error is causing the browser not to open?
    - Where in the JavaScript code does this error originate from?

- ***TASK:*** *Testing a fix*
  - Open the elements panel
  - Select the element
  - Change click event handler to **click="openBrowser()"**
  - What happens when you click the **Open Browser** button again?

Task 5: Console panel

BlackBerry Jam Asia

- Running JavaScript from the console
  - You can access any variable or method that the web page can.
- ***TASK:*** *Viewing runtime data*
  - What do you see when you type **window.localStorage** or **document.body** and press enter?
- ***TASK:*** *Manually running methods*
  - Type **window.location.reload()** to reload the current page.
  - What happens when you type **openBrowser**?
  - What happens when you type **openBrowser()**?

Task 5: Console panel

BlackBerry Jam Asia

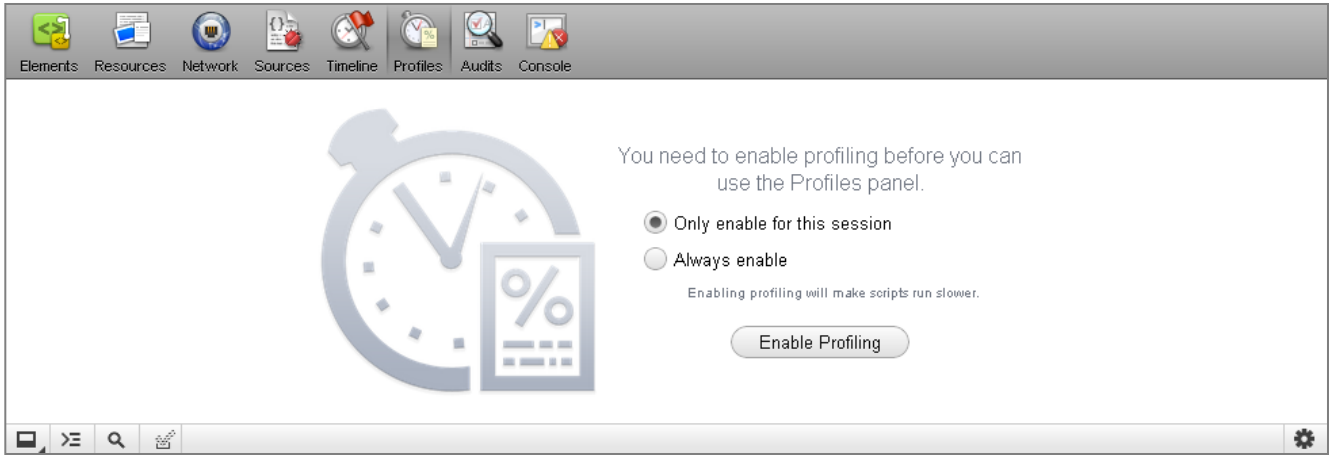
- Console tips and tricks:
  - Clear the console type **CTRL + L** or click
  - Type **$0** to access the currently selected element
  - Type **ESC** to open / close the console when viewing other panels
  - Use auto complete: press the right → arrow key.

Task 6. Profiles panel

Measuring CPU and memory load

[10 mins]

Task 6: Profiles Panel



- Measuring JavaScript or CSS performance
  - A profile is a snapshot in time of CPU load.
  - Results can be analyzed in % or milliseconds (ms).

Task 6: Profiles Panel

BlackBerry Jam Asia

- Start WIC sample application
  - Open “Profiles” page
- Open http://169.254.0.1:1337 in Chrome
  - Click on “Web Inspector Companion”

- What is profile information?
  - Used to measure JavaScript or CSS profiling
  - New profiles are manually started & stopped
  - Results can be analyzed in % or milliseconds (ms)

Task 6: Profiles panel

BlackBerry Jam Asia

- ***TASK:*** *Manually collecting JavaScript CPU profile info*
  - Click **Start** button from Profiles panel in Web Inspector.
  - Click **Start 5s Profile Script** button in sample on device.
  - Once alert box is displayed, click **Stop** button in Web Inspector

- ***TASK:*** *Select* ***Profile 1*** *from CPU profiles results*
  - Which 3 functions used the most total CPU time %?
  - How many times were these methods called?
  - How would you improve performance based on these results?

Task 6: Profiles panel

BlackBerry Jam Asia

- ***TASK:*** *Automated JavaScript CPU profiling*
  - Can use console.profile() and console.profileEnd() in your code.
  - Click **Inline Profiling** button in sample on device.
  - A new **runSimulation** result will be displayed in Web Inspector

- ***TASK:*** *Select* ***runSimulation*** *from CPU profiles results*
  - Where in the code was this profile ran from?
  - What is the performance issue identified by this profile result?
  - How would you improve performance based on these results?

Task 6: Profiles panel

- ***TASK:*** *Tracing the call stack*
  - Select **Profile 1** result
  - Expand results in the **Function** column see calling methods
  - Which parent method called the **updateGraphics** method?
  - Which page event initiated the original call to **updateGraphics**?
  - Where in the JavaScript source was this function called from?

Bonus task: Advanced features

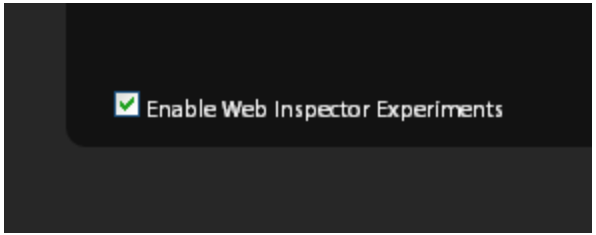
Tips and tricks

[10 mins]

Bonus Task: Advanced features

BlackBerry Jam Asia

- Start WIC sample application
  - Open “Advanced Features” page
- Open http://169.254.0.1:1337 in Chrome
  - Enable **experiments** checkbox



  - Click on “Web Inspector Companion”

Bonus Task: Advanced features

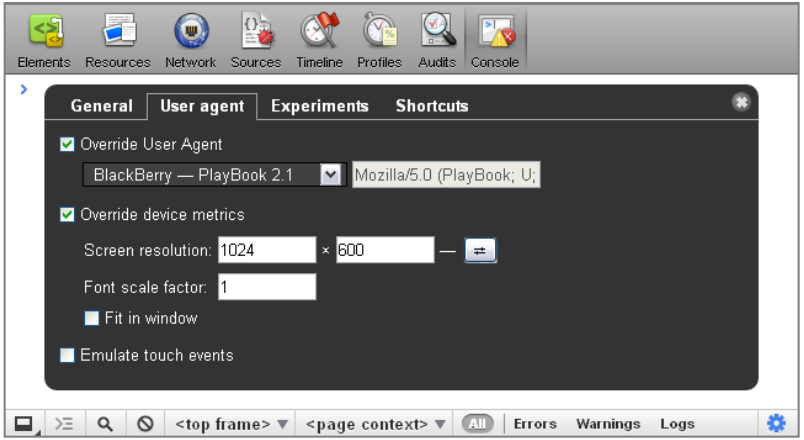
BlackBerry Jam Asia

- ***TASK:*** *Overriding user agent*
  - Click **Display user agent** button in the sample application.
    - What user agent value is displayed?
  - Open **Settings** screen
  - Select the **User Agent** tab
  - Click **Override User Agent** checkbox
  - Select a different option user agent drop-down box
  - Click **Display user agent** button in the sample application
    - What new user agent value is displayed?

Advanced features

BlackBerry Jam Asia

- ***TASK:*** *Overriding device Geolocation values*
  - Click **Display GPS** button in sample application
    - What values are displayed?
  - Open **Settings** screen in Web inspector
  - Enable **Override Device Geolocation** checkbox
  - Change GPS values to 43.642722, -79.387207
  - Close **Settings** screen
  - Click **Display GPS** button again in sample application
    - What values are displayed?



BlackBerry Jam Asia

THANK YOU

JAM843

Adam Stanley @n_adam_stanley

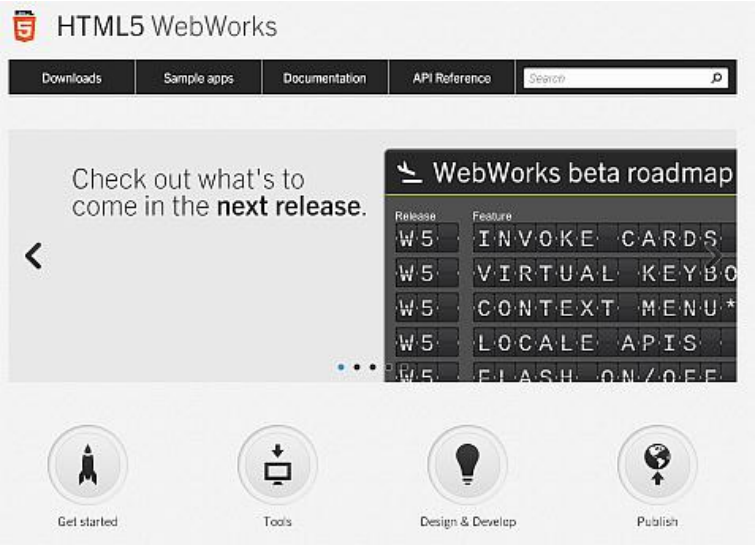
Justin Lee @triplez82

November 29, 2012

For More Information…

BlackBerry Jam Asia

- Sessions:
  - JAM826 – Native Look & Feel in Web
  - JAM836 – PIM Web API
  - JAM839 – Invocation Web API
  - JAM840 – HTML5 Gaming



- Resources:
  - http://developer.blackberry.com/html5
  - http://github.com/blackberry