Symbian Browser and Web Runtime 7.3 for Developers

The Symbian Browser and Web Runtime 7.3, which are delivered as part of the Symbian Anna refresh, bring substantial performance enhancements and improved compatibility with CSS3 and HTML5. This article provides an overview of the changes, and their implications for developers and web content providers.

25 Apr 2011

Target devices

The new Symbian Browser and Web Runtime 7.3 are delivered as part of the Symbian Anna refresh. This refresh will be available in:

- Initial releases of Nokia C7 Astound, Nokia X7 and Nokia E6
- Upcoming firmware updates of in-market Symbian^3 devices, including N8

Web Developer Documentation at Nokia Developer

Web Developer's Library

Symbian Browser and Web Runtime discussion boards

Web Development at Nokia Developer discussion boards

User Agent string

User agent string structure

Mozilla/5.0 (OSName/OSVersion; PlatformName/PlatformVersion DeviceModel/DeviceSoftwareVersion; Profile/MIDPVersion Configuration/ConfigurationVersion) WebEngineType/WebEngineversion (KHTML, like Gecko) BrowserName/BrowserVersion Mobile Safari/WebKitversion

Example: N8

Mozilla/5.0 ('*Symbian/3*; Series60/5.2 NokiaN8-00/012.002; Profile/MIDP-2.1 Configuration/CLDC-1.1) AppleWebKit/533.4 (KHTML, like Gecko) **NokiaBrowser/7.3**.0 Mobile Safari/533.4 3gpp-gba

Targeting Symbian Browser 7.3 with cross platform web content

The Symbian WebKit port update brings substantial performance and feature compatibility upgrades to e.g. content rendering.

Tip: Web content providers are in many cases better off by serving content optimized for other smartphone platforms to Symbian Anna devices.

When fine graining application behavior, there are several things developers must keep in mind when supporting Symbian Anna. Below is a list of the most commonly encountered differences to other WebKit based smartphone browsers:

- The only reliable user interaction event is the single click
  - touch events and advanced gestures are not supported
- No device API access (e.g. GeoLocation or accelerometer) from browser context
  - Wrap your web content into a Symbian WRT widget to get access to device APIs
- No HTML5 video or audio support
  - Symbian Browser 7.3 supports Flash Lite 4, thus you can use Flash Video for your video playback and streaming
- CSS3 animations, transforms and transitions may have performance issues, especially with complex content

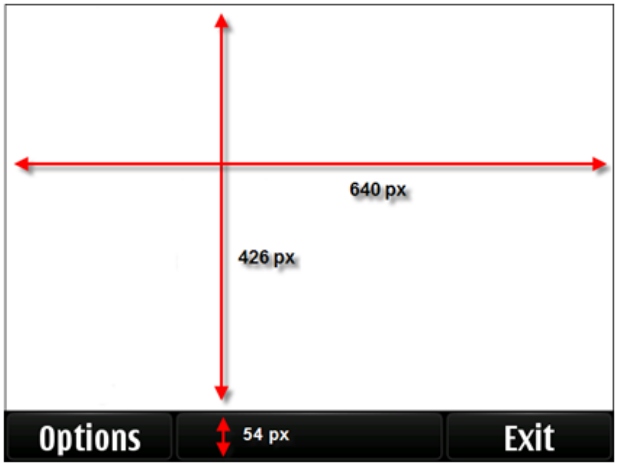
VGA resolution

Until now, Symbian^3 devices and Series 60 5th edition devices have shared a common screen resolutionof 640 x 360px. The Nokia E6 introduces VGA resolution (640 x 480px) with a very high pixel density (screen size is 2.4 inches)

Greater pixel density leads to situation where touchable elements should be a lot bigger in terms of pixel size to be usable. For example the smallest touchable element should be 7 x 7mm. On N8 this is about 60 x 60px, but to get similar dimensions in E6you would need to have an element sized 90 x 90px.

The VGA resolution menu bar height is 54px and the display is fixed to landscape mode.

(C) Copyright Nokia 2013. All rights reserved.



New and updated features for developers

Updated WebKit

Symbian WebKit port has been updated (v533.4). The update brings significant performance and compatibility benefits, compared to earlier Symbian browsers.

As a benefit of the updates to Symbian WebKit the underlying graphics adaptation layer, several CSS3 visual elements and effects , including 2D transforms, rounded corners and gradients, are at least partially supported.

In addition to these developers can utilize CSS media queries and viewport meta tags available e.g. in iOS and Android browsers.

HTML5

Support for a small subset of HTML5 features, including the doctype, section-level markup tags, form input attributes, placeholder text attributes for input fields, the scrollintoview method, and the contenteditable attribute.

Offline capabilities (web storage, application cache), HTML5 audio/video and HTML5 geolocation are **not** supported.

CSS3

Support for a subset of CSS3 features, including animation, backgrounds, borders, color, gradients, media queries, multi-column layout, selectors, text effects, transitions, and 2–D transforms.

Usernames and passwords can not be automatically passed as part of a address of http/https resources

Http/https resources protected with basic authentication can not anymore be accessed by inserting the username and password as a part of the requested address.

E.g. while in earlier Symbian browsers, a resource accessible with username and password credentials could be accessed directly by calling *http://username:password@mytestsitexyz.com/protected*, the same resource will force the user to manually enter the username and password in Browser 7.3.

DPI based scaling

Browser 7.3 introduces a mechanism to enhance readability on devices with high pixel density screens.

160 DPI is treated as a default value, meaning that content will be scaled to 100%, whereas e.g. N8 (220 DPI) and E6 (330 DPI) will get scaling factors 1.25 and 1.50 respectively.

Note:

- This change exists only in the browser context so existing WRT widgets are not affected
- Developers should utilize the viewport meta tag, just introduced to Symbian Browser 7.3, to explicitly control scaling of the web content served through the browser.

Viewport meta tag in Web pages

Using the viewport meta tag is recommended if you want to control the size of your page and the ability to zoom. If you want to define the viewport we recommend this basic syntax:

```
<meta name="viewport" content="width=device-width"/>
```

This syntax will fix the size to the actual device width and is compatible with other browsers such as Android and iPhone. Our browsers will apply an initial

(C) Copyright Nokia 2013. All rights reserved.

scale to make the text more readable, but you may define the *initial-scale* and also the *target-densityDpi*, if you want to control in more detail how the browser renders the page. We discourage blocking the ability to zoom as this is normally a bad user experience, it is better to just define the initial scale.

Viewport meta tag in Web Runtime

Setting scaling in a widget, which is placed into a Home Screen, will cause all of Home Screen widget mini views to be scaled accordingly. This will result into a badly rendered Home Screen. Therefore usage of viewport meta tag is strongly discouraged on Home Screen enabled widgets.

Web Runtime Homescreen dimensions

Observed Home Screen widget size stays unaffected to keep the compatibility with N8 or N97 i.e. 312x82px.

The actual resolution of the home screen widget slot is 440 x 116px on a VGA device.

The home screen widget image is automatically scaled to fit the available resolution.

Old code for detecting if widget is shown on home screen, can still be used without modifications, because Web Runtime reports window.innerHeight as 82px instead of actual 116px.

Renewed WRT security prompts

WRT 7.3 supports a new API group all of the platform service (device API) permission prompts into a single prompt. Hence the end user doesn't have to allow use of platform services APIs one by one during the application usage.

The new API is part of the device-object.

```
device.getServicePermissions( requestedServices)
```

`RequestedServices` is an object which holds the names of the services that are used by the widget. The service provider's name is used as an object's property name and interface name as its value.

```
//check for getServicePermissions for backward comptibility
 if(device && typeof device.getServicePermissions == "function"){
  try {
   device.getServicePermissions({
    "Service.AppManager": "IAppManager",
    "Service.Calendar": "IDataSource",
    "Service.Contact": "IDataSource"
   });
  }
  catch (e) {
   //notify the user that access to services has been declined
   return;
  }
 }
 //continue by using old per service permission model
```

JavaScript execution change

KIS001646 - JavaScript execution change in Browser 7.3 on Symbian devices

Known Issue for XMLHttpRequest on Browser 7.3

When a Symbian Web Runtime Widget, started from Applications menu, is trying to access Internet for a first time "Allow connection to Network?" – prompt is shown. In previous Browser versions answering "No" will fire onoffline event and XMLHttpRequest will complete with a status code 0. In Browser 7.3 denying network access will fire onoffline event, but XMLHttpRequest won't complete.

Solution is to implement a timer that aborts the XMLHttprequest if it is taking too long to complete. This will ensure that request won't be left hanging forever. Please note that in addition usage of online/offline API is recommended in all WRT Widgets.

```
var timeOutDelay = 10000; // 10 sec
var timerId;
var xmlHttpRequest = new XMLHttpRequest();
xmlHttpRequest.open("GET", "http://www.nokia.com", true);

xmlHttpRequest.onreadystatechange = function(){
   if (xmlHttpRequest.readyState == 4 && xmlHttpRequest.status == 200) {
        clearTimeout(timerId);
        alert(xmlHttpRequest.responseText);
    }
}
xmlHttpRequest.send(null);
timerId = setTimeout(xmlHttpRequestOnTimeOut, timeOutDelay);

function xmlHttpRequestOnTimeOut(){
   xmlHttpRequest.abort();
   alert("XMLHttpRequest timed out");
}
```


(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.