

# BRAVIA B2B HTML5 Solution Technical Manual

Version 0.8

Copyright (C) 2013 Sony Corporation  
All rights reserved.

## 1. Introduction

This document describes specifications, behaviors and restrictions of BRAVIA B2B function related to web based technology. It mainly focus on system integration technology with HTML5 and HTTP. It does not cover serial control function, HDMI CEC and so on. The information described in this document may be subject to change on late notice.

**Keywords:** HTML, HTML5, HTTP, W3C, RFC, TCP/IP, Web Browser, JavaScript, CSS

*BRAVIA B2B HTML5 Solution* offers boxless flat panel display integration platform. It has embedded full-screen web browser which runs HTML5 application without any additional devices. It allows not only to run applications but to control display via TCP/IP network by sending HTTP request.

Put simply, *BRAVIA B2B HTML5 Solution* capable BRAVIA has the following functions.

- HTML5 Application Runtime
- IP Control Function

## 2. References

[HTML5]	HTML5 - A vocabulary and associated APIs for HTML and XHTML <a href="http://www.w3.org/TR/html5/">http://www.w3.org/TR/html5/</a>
[RFC2616]	IETF RFC 2616, Hypertext Transfer Protocol - HTTP/1.1 <a href="http://www.ietf.org/rfc/rfc2616.txt">http://www.ietf.org/rfc/rfc2616.txt</a>
[ISO3166]	ISO 3166-1:2006, Codes for the representation of names of countries and their subdivisions -- Part 1: Country codes <a href="http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39719">http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=39719</a>
[RFC3986]	IETF RFC 3986, Uniform Resource Identifier (URI): Generic Syntax <a href="http://www.ietf.org/rfc/rfc3986.txt">http://www.ietf.org/rfc/rfc3986.txt</a>
[JSONRPC]	JSON-RPC 1.0 Specifications <a href="http://json-rpc.org/wiki/specification">http://json-rpc.org/wiki/specification</a>

## 3. Device Configuration

### 3.1 Hotel Mode

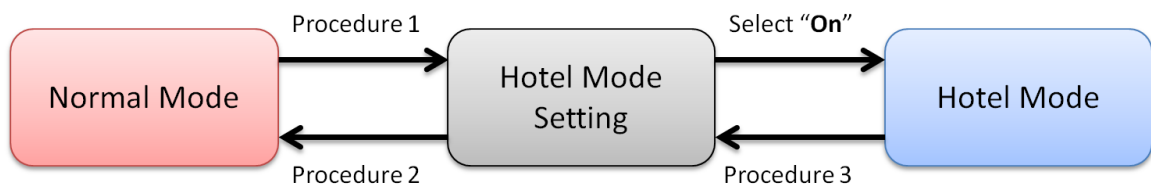
BRAVIA B2B functions are available under B2B special mode called **Hotel Mode**. Professional users need to understand the following three states:

- **Normal** : Normal mode for consumer use
- **Hotel Mode Setting** : Special setting mode for professional users
- **Hotel Mode** : Professional mode for B2B system works under special settings

### Table of Contents

<a href="#">1. Introduction</a>
<a href="#">2. References</a>
<a href="#">3. Device Configuration</a>
<a href="#">3.1 Hotel Mode</a>
<a href="#">4. HTML5 Application Runtime</a>
<a href="#">4.1 HTML Rendering Engine</a>
<a href="#">4.2 Web Standard Compliance</a>
<a href="#">4.3 Supported URL Schemes</a>
<a href="#">4.4 SSL/TLS Root CA Certificates</a>
<a href="#">4.5 User Input</a>
<a href="#">4.6 Media Elements</a>
<a href="#">4.7 Broadcast Video</a>
<a href="#">5. Application Packaging</a>
<a href="#">5.1 Manifest</a>
<a href="#">5.2 Boot from USB Storage</a>
<a href="#">5.3 Storing Application in USB Storage</a>
<a href="#">6. IP Control</a>
<a href="#">6.1 Access Control</a>
<a href="#">6.2 Services and Methods</a>
<a href="#">6.3 Examples</a>

The state transition is illustrated in Figure 1.



**Figure 1. State Transition of Hotel Mode**

To switch to "Hotel Mode Setting" state, follow **Procedure 1** with IR remote controller.

- **Procedure 1**

1. Turn on the device
2. Press **POWER** button (Screen is turned off)
3. Press **DISPLAY** button
4. Press **TEST** button (Ask your account manager for detail)
5. Press **VOL +** button
6. Press **POWER** button (Screen is turned on)

Then, the banner "Hotel Mode Settings in progress" is shown on the upper part of the screen as Figure 2.



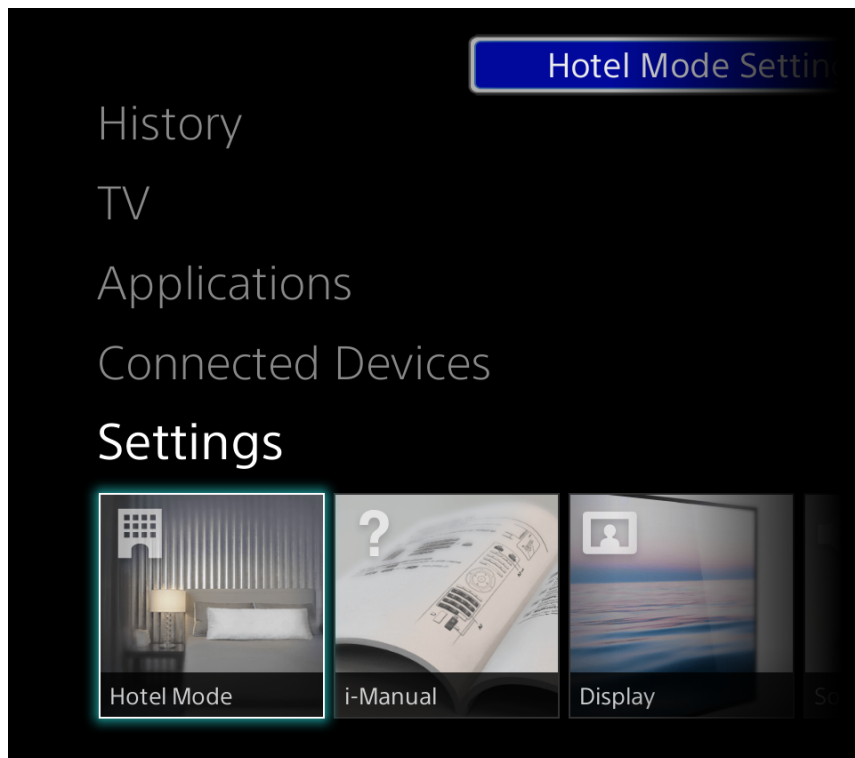
**Figure 2. Hotel Mode Settings banner**

To switch back to Normal mode, follow **Procedure 2** during Hotel Mode Setting state.

- **Procedure 2**

1. Turn on the device
2. Press **POWER** button (Screen is turned off)
3. Press **DISPLAY** button
4. Press **TEST** button (Ask your account manager for detail)
5. Press **VOL -** button
6. Press **POWER** button (Screen is turned on)

In the Hotel Mode Setting state, **Hotel Mode** icon is added into "Settings" category in the home menu as shown in Figure 3. The "Hotel Mode" icon allows professional users to configure device in detail.



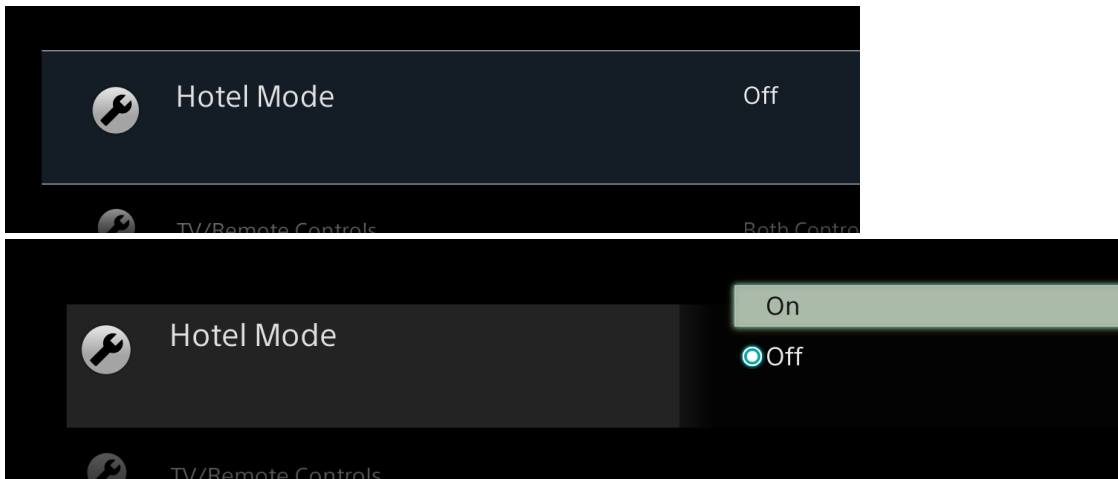
**Figure 3. Hotel Mode icon in Settings Category**

The following setting items related to the HTML5 solution are available under this state. These items are shown in hotel mode setting state only.

**Table 1. Setting Items for HTML5 Solution**

Name	Type	Default	Detail
Startup Application	Boolean	On	If "Startup Application URI" setting value is valid, the application is launched when turning on the TV in Hotel Mode
Startup Application URI	String		Application URI Nothing happens if the value is empty e.g.) <a href="http://example.com/myapp.html">http://example.com/myapp.html</a>
IP Control > Authentication	Selection	Normal	Access control method - None - Normal - Pre-Shared Key - Normal and Pre-Shared Key Normally "Pre-Shared Key" is used for B2B system
IP Control > Pre-Shared Key	String		User-defined secret key used for access control of IP control function

To switch to Hotel Mode from Hotel Mode Settings, select the first item in Hotel Mode icon and select **On**.



**Figure 4. Switching to Hotel Mode**

Then, the "Hotel Mode Settings in progress" banner disappears. To switch back to Hotel Mode Setting state, follow **Procedure 3**.

- **Procedure 3**

1. Press **DISPLAY** button
2. Press **TEST** button (Ask your account manager for detail)
3. Press **VOL +** button
4. Press **HOME** button

Then, the "Hotel Mode Settings in progress" banner is shown on the screen.

## 4. HTML5 Application Runtime

### 4.1 HTML Rendering Engine

"BRAVIA B2B HTML5 Solution" uses **Opera Presto** for its rendering engine. The rendering engine version can be found in User-Agent as below.

```
User-Agent: Opera/9.80 (Linux armv7l; U; en) Presto/2.10.287 Version/12.00
SonyCEBrowser/1.0 (KDL-55W800A; CTV/PKG3.005EUA; GBR)
```

The detailed information about Opera Presto engine can be found in the following link.

- <http://www.opera.com/docs/specs/presto2.10/>

Note that some of the functions in Presto depends on platform device integration. For example, BRAVIA has its own specification in the following functions. See 4.2 for more detail.

- Supported media format (<video> and <audio> tags)
- Availability of the storage (Application Cache, WebStorage, etc.)
- Availability of the graphic function especially in WebGL function

BRAVIA supports only one window at a time, and window size is fixed to 1920x1080.

### 4.2 Web Standard Compliance

BRAVIA supports some of the standardized APIs as described in Table 2.

**Table 2. Supported APIs**

Item	Status	Detail
MediaElements	Supported	See 4.6 for more detail
Canvas	Supported	
SVG	Supported	

WebGL	Supported	
WebStorage	Not Supported	
Offline Apps	Supported	Up to 5MB per domain, 10MB in total
Web Fonts	Supported	
Selectors	Supported	
WebSocket	Supported	
Web Workers	Supported	
Viewport	Not Supported	
Indexed DB	Not Supported	
Geolocation	Not Supported	
W3C Widget	Not Supported	

BRAVIA supports the following HTTP authentication methods and JavaScript dialogs.

- HTTP Authentication methods
  - Basic
  - Digest
- JavaScript dialogs
  - alert
  - prompt
  - confirm

Note that "file dialog" isn't supported.

Accept-Language field holds main language as the first priority and English as the second.

e. g. ) ja, en;q=0.8

### 4.3 Supported URL Schemes

BRAVIA supports the following url schemes.

- http
- https
- data

### 4.4 SSL/TLS Root CA Certificates

The UA contains the root CA certificates listed below. When SSL/TLS server authentication fails, the UA refuses access to the server without user interaction.

- SCEI DNAS Root 01
- SCEI DNAS Root 02
- SCEI DNAS Root 03
- SCEI DNAS Root 04
- SCEI DNAS Root 05
- Thawte Server CA
- Thawte Premium Server CA
- VeriSign Class 3 Public Primary Certification Authority
- Equifax Secure Certificate Authority
- GTE CyberTrust Global Root
- Entrust.net Certification Authority (2048)
- Entrust.net Secure Server Certification Authority
- ValiCert Class 2 Policy Validation Authority
- GlobalSign Root CA
- VeriSign Class 3 Public Primary Certification Authority - G5
- thawte Primary Root CA
- GeoTrust Primary Certification Authority
- Entrust Root Certification Authority
- AddTrust External CA Root

- COMODO Certification Authority
- GlobalSign Root CA - R2
- DigiCert High Assurance EV Root CA
- DigiCert Global Root CA

## 4.5 User Input

Key event is delivered when being signaled from TV remote controller as described in Table 3. VK\_\* values are the constant properties owned by the window object.

**Table 3. Key definition for user input**

RC key	Virtual Key	Support
Left	VK_LEFT	Mandatory
Up	VK_UP	Mandatory
Right	VK_RIGHT	Mandatory
Down	VK_DOWN	Mandatory
Enter (Confirm)	VK_ENTER	Mandatory
Return	VK_BACK_SPACE	Mandatory
Blue	VK_BLUE	Optional
Red	VK_RED	Optional
Green	VK_GREEN	Optional
Yellow	VK_YELLOW	Optional
0	VK_0	Optional
1	VK_1	Optional
2	VK_2	Optional
3	VK_3	Optional
4	VK_4	Optional
5	VK_5	Optional
6	VK_6	Optional
7	VK_7	Optional
8	VK_8	Optional
9	VK_9	Optional
Play	VK_PLAY	Optional
Pause	VK_PAUSE	Optional
Stop	VK_STOP	Optional
Next	VK_TRACK_NEXT	Optional
Prev	VK_TRACK_PREV	Optional
FF (Fast-Forward)	VK_FAST_FWD	Optional
FR (Fast-Rewind)	VK_REWIND	Optional
Subtitle	VK_SUBTITLE	Optional
Information	VK_INFO	Optional
Exit	VK_EXIT	Optional

## 4.6 MediaElements

BRAVIA supports <audio> and <video> tags defined in [HTML5]. See Table 4, Table 5 and Table 6. for the supported media format.

### Restrictions

- width and height attributes are necessary for <video> tag
- type attribute which has mime type in source element is needed for identifying proper media format

e.g.)

```
<video width="960" height="540">  
  <source src="http://example.com/video.mp4" type="video/mp4"/>  
</video>
```

These restrictions are going to be removed in the future.

**Table 4. Supported Video Formats**

Protocol	Container	Video	Audio	Manifest	MIME type
HTTP(S)	MP4	AVC	AAC-LC/HE-AAC	-	video/mp4
HTTP(S)	MPEG2 TS	AVC/MPEG2	AAC-LC/HE-AAC/MP1L2	-	video/m2ts
HLS/HTTP(S)	MPEG2 TS	AVC/MPEG2	AAC-LC/HE-AAC/MP1L2	M3U8	application/vnd.apple.mpegurl
MSSS/HTTP(S)	MP4	AVC	AAC-LC/HE-AAC	MSSS	application/vnd.ms-sstr+xml

**Table 5. Supported Audio Formats**

Protocol	Container	Audio	MIME type
HTTP(S)	MP4	AAC-LC/HE-AAC	audio/aac
HTTP(S)	MP3	MP3	audio/mp3

**Table 6. Codec Information**

Codec	Supported Specification
MPEG4 part10 (AVC/H.264)	BP@L3, MP@L4, HP@L4
AAC-LC	32k, 44.1k, 48k / 384kbps(max) / up to 5.1ch
HE-AAC	32k, 44.1k, 48k / (ISO/IEC 14496-3 compliant / profile level3) / up to 2ch

## 4.7 Broadcast Video

BRAVIA allows to embed broadcast video and external input into html application as described below. width and height attributes are mandatory for this object.

```
<object type="application/x-decimated-video" width="960" height="540"/>
```

The last input source (broadcast video or external input) is shown in the rectangle. **INPUT** button and **CH +/-** buttons on IR remote controller work for changing input source.

If a page contains two or more broadcast video objects, only the first one is available. If a page contains a broadcast object and one or more media elements (<video> and <audio> elements), the broadcast object doesn't work and a black rectangle is drawn.

## 5. Application Packaging

### 5.1 Manifest

A manifest contains application information.

- e.g.) Application ID, Title, Splash Image, Initial URL

A tempate manifest is shown below. Manifest file must be named "**manifest.json**".

```

{
  "name": {
    "default": "en_US",
    "en_US": "Example B2B Application"
  },
  "uuid": "com.example.app.myapp",
  "description": {
    "default": "en_US",
    "en_US": "Created for an example"
  },
  "version": "1.0",
  "manifest_version": "1.0",
  "app": {
    "launch": {
      "web_url": "http://example.com/myapp"
    }
  },
  "splash": "splash.jpg",
  "icons": {
    "128x128": "icon.png"
  },
  "minimum_web_platform_version": "1.0"
}

```

## 5.2 Boot from USB Storage

BRAVIA B2B capable devices can launch web application from USB when it's inserted. To use this function the following conditions must be satisfied.

1. A USB storage is formatted with FAT32
2. Manifest files are stored in accordance with the following manner
  - /sony/apps/webapps/<application\_folders>/app/manifest.json
3. autorun.txt file is stored under /sony directory
  - /sony/autorun.txt

autorun.txt contains an uuid to identify which application in /sony/apps/webapps directory should be launched. An example is shown below.

```

{
  "uuid": "com.example.app.myapp"
}

```

Any directory name can be used for application folders and one or more applications can be stored under /sony/apps/webapps directory.

## 5.3 Storing Application in USB Storage

Application files such as html, image, css and js files can be stored in USB storage under /sony/apps/webapps/<application\_folders>/app/doc directory. The doc directory is mounted to file://localhost/doc. By storing a whole application files in USB storage, the application works offline without any network connection. The entry name local\_path is used for file:// scheme.

An example local\_path of offline application is shown below.

```

...
  "app": {
    "launch": {
      "local_path": "file://localhost/doc/index.html",
    }
  },
  ...

```

## 6. IP Control

BRAVIA has IP control function which allows to control it via IP network. The protocol is based on [JSON-RPC 1.0](#) over HTTP and is equipped with access control feature.



IP control function is designed for being called from both web applications and standalone applications. Examples in this section is written in JavaScript for web applications.

## 6.1 Access Control

BRAVIA has the following 4 types of access control methods for IP control, which is configu

**Table 7. Access Control Methods**

No.	Method	Detail
1	None	No authentication
2	Normal	PIN-based authentication
3	Pre-Shared Key	Fixed secret string based authentication
4	Normal and Pre-Shared Key	Combination of 2 and 3

Normally "Pre-Shared Key" is used for B2B system. In this case, an IP control request is accepted if the http request header has the following header field with the value specified in **Pre-Shared Key** setting.

```
X-Auth-PSK: <value in Pre-Shared Key setting>
```

e.g.)

```
X-Auth-PSK: mysecret
```

## 6.2 Services and Methods

A service consists of one or more methods. HTTP server in a device has one or more services and listens requests on port 80. SSL isn't supported. A service is identified by the following URL syntax.

```
http://<address>/sony/<service_name>
```

A method is identified by the method field in JSON-RPC request body.

e.g.)

```
{  
  "method": "getVersions",  
  "params": [],  
  "id": 123  
}
```

All services have the methods in Table 8, which allow to inspect what methods and versions are available.

**Table 8. Common Methods for All Services**

Method	params	Description
getVersions	(none)	Returns all version numbers
getMethodTypes	version	Returns all method names and params of a specified version

e.g.)

```
{  
  "method": "getMethodTypes",  
  "params": ["1.0"],  
  "id": 4649  
}
```

When a service has two or more versions and each of them has the same method, version field is available for identifying them.

e.g.) Call doSomething method of version 2.1

```
{
  "method": "doSomething",
  "params": [],
  "version": "2.1",
  "id": 100
}
```

## 6.3 Examples

### Basic Sequence

```
// create XHR object
var xhr = new XMLHttpRequest();

// open service url with pre-shared key header
xhr.open("POST", "http://192.168.0.100/sony/system", true);
xhr.setRequestHeader("X-Auth-PSK", "mysecret");

// set callback function
xhr.onload = function() {
  console.log(this.responseText);
};

// send request
xhr.send(JSON.stringify({
  "method": "getSystemInformation",
  "params": [],
  "id": 101
})));
```

### Power-Off Function Example

```
function powerOff(ip, secret) {
  var xhr = new XMLHttpRequest();
  xhr.open("POST", "http://" + ip + "/sony/system", true);
  xhr.setRequestHeader("X-Auth-PSK", secret);
  xhr.send(JSON.stringify({
    "method": "setPowerStatus",
    "params": [{
      "status": false
    }],
    "id": 102
  })));
}
```