

Snapshot

© 2017 All rights reserved by Metrological

This document contains information which is proprietary and confidential to Metrological. It is provided with the expressed understanding that the recipient will not divulge its content to other parties or otherwise misappropriate the information contained herein. This information is furnished for guidance; specifications and availability of goods mentioned in it are subject to change without notice. No part of this publication may be reproduced, stored in a database, retrieval system or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the written prior permission of Metrological.

History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 0.1 | 08-11-2017 | O.Deveci | Initial version |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Scope 4

1.2 Case sensitivity 4

1.3 Acronyms, Abbreviations and Terms 4

1.4 Standards 4

1.5 References 4

1.6 Open Issues 5

1.7 Limitations 5

2. Snapshot Plugin 6

2.1 Configuration 6

2.2 Snapshot Image 6

2.3 Supported Platforms 6

2.3.1 Raspberry PI 6

2.3.2 Broadcom Nexus 6

2.4 Application Programming Interface (API) 6

2.4.1 General information 6

2.4.2 Capture action 6

2.5 Events 7

2.6 JSON definitions 7

# Introduction

## Scope

This document describes the Plugin Snapshot API interface. This plugin can be configured to be loaded and executed in the WPEFramework and offers to capture the current displayed screen on the platform. For details on the WPEFramework API, refer to: [WPEF]

## Case sensitivity

All identifiers on the interface described here are case-sensitive. E.g. an id known in the plugin as 'C0FFEE' is not the same as 'c0ffee'.

All keywords, entities, properties, relations and actions should be treated as case-sensitive.

## Acronyms, Abbreviations and Terms

The next list provides an overview of acronyms and abbreviations used in this document and their definitions.

|  |  |
| --- | --- |
| **Acronym** | **Definitions** |
| API | Application Programming Interface |
| JSON | JavaScript Object Notation |

Below terms are listed with their definitions, as used in this document.

|  |  |
| --- | --- |
| **Term** | **Definitions** |
| Callsign | The callsign is the name given to an instance of a plugin. One plugin can be instantiated multiple times, but each instance the instance name, callsign, must be unique. |

## Standards

Date time formats between the systems shall be in UTC time and W3C (ISO 8601 profile) formatting [ISO 8601], e.g.: 2004-11-05T13:15:30Z. This way time discontinuities can be avoided due to daylight savings. Note that all interfacing systems must decode/encode the date time to the correct local time.

Languages used in the WPEFramework will be conform [ISO 639-1] using two letter language codes. If WPEFramework encounters a language code it does not recognize, it will use ‘xx’ instead. For a list of available two letter ISO language codes, please visit:  
<http://www.loc.gov/standards/iso639-2/php/code_list.php>

## References

This section lists the references made in this document:

|  |  |
| --- | --- |
| [WPEF] | WPEFramework API Reference  <https://github.com/WebPlatformForEmbedded/WPEFramework> |
| [HTTP] | Hypertext Transfer Protocol  <http://www.w3.org/Protocols> |
| [ISO 8601] | Date and time format  http://www.iso.org/iso/date\_and\_time\_format |
| [ISO-3166] | Country code specification  <http://www.iso.org/iso/country_codes.htm> |
| [ISO-639-1] | Language code specification (Alpha-2 code)  <http://www.loc.gov/standards/iso639-2/php/code_list.php> |
| [JSON] | JavaScript Object Notation  http://www.json.org |
| [URLENC] | URL Encoding  <http://www.w3schools.com/tags/ref_urlencode.asp> |

## Open Issues

This is a list of open issues that needs to be resolved:

* This document is still a work in progress.

## Limitations

The information described in this document is preliminary and subject to change in the future.

Legend:

****

**Be aware of:** implementation choice is needed or side-effect needs to be handled.



**Implementation advice:** Guide line for implementation mostly related to performance.

# Snapshot Plugin

## Configuration

|  |  |
| --- | --- |
| callsign | [string] the instance name for the plugin. Default: Snapshot. |
| classname | [string] Snapshot. |
| locator | [string] libWPEFrameworkSnapshot.so |
| autostart | [bool] should the Snapshot plugin be instantiated at the moment the WPEFramework is starts up. |

## Snapshot Image

The Capture API will return PNG image which is based on RGBA pixel formation.

## Supported Platforms

### Raspberry PI

This plugin uses Dispmanx which is the windowing system of Raspberry PI. The capture functionality of Dispmanx is utilized to capture the current framebuffer. It provides the composition of video and graphic layers as a snapshot.

### Broadcom Nexus

On Nexus based platforms, this plugin captures graphic layer by fetching the current framebuffer of graphic layer.

## Application Programming Interface (API)

### General information

Using this method, the current status of plugin can be retrieved from the plugin.

|  |  |
| --- | --- |
| Request: | GET /Service/Snapshot |
| Success: | HTTP/1.1 200 Plugin is up and running |
| Failure | HTTP/1.1 405 Method not allowed |

### Capture action

Using this method, the capture request will be retrieved by plugin.

|  |  |
| --- | --- |
| Request: | GET /Service/Snapshot/Capture |
| Success: | HTTP/1.1 202 <capture\_device>[[1]](#footnote-1) |
| Failure | HTTP/1.1 412 Plugin is already in progress  HTTP/1.1 412 Could not create a capture on <capture\_device> |

## Events

Events are not applicable for Snapshot plugin.

## JSON definitions

There are no JSON objects applicable for Snapshot plugin.

1. capture\_device indicates the platform name [↑](#footnote-ref-1)