INNES

User manual

Rooming'it connector

1 Introduction

This document provides installation and deployment information for administrators planning to deploy the **Roomingit-calendar** compatible with Innes digital signage.

The **Roomingit -calendar** is the bridge component allowing the Innes players to communicate with **Rooming'it** servers.

Rooming'it-calendar system must be installed and configured properly.

2 Requirements

The requirements must be filled before the installation.

- **Innes PlugnCast Server V2.50.31** (or above) or
- **Innes Briva Server V2.10.10** (or above)
- **Rooming'it 7.0**

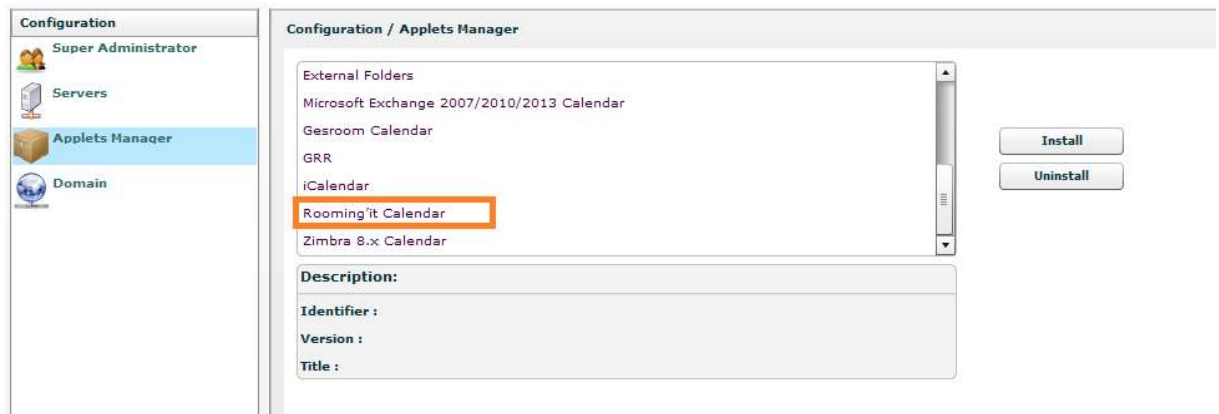# 3 Installation

- Go to the Innes Briva (or PlugnCast) Server Web Interface with your Web browser by typing:

  http://<host>/.configuration

  <host>: IP address (or DNS name) of your Innes Briva or PlugnCast Server.

- Log-in as "Super Administrator"
- Go to "Applets Manager" and click "Install" button.
- Select "*Rooming'it-calendar.saz*" file and follow the installation steps.
- Result:

4   Configuration

The configuration file *configuration.xml* for **Rooming'it-calendar** is located here:

```
<Innes Data Folder>\Server\.accounts\<domain>\.applets\.roomingit-
calendar\configuration.xml
```

- 🔸 The ***Innes Data Folder*** is the data folder of Innes Briva or PlugnCast Server.
- 🔸 The <domain> is your currently domain.

In these examples ***<domain>*** value is **my_company**.
- Vista Example

```
C:\Users\Public\Documents\Innes\Server\.accounts\my_company\.applets\.roomingit-calendar\configuration.xml
```
- XP Example

```
C:\Documents and Settings\All Users\Documents\Innes\Server\.accounts\ my_company\.applets\.roomingit-calendar\configuration.xml
```

- XML Syntax

After editing the configuration file make sure that it respects the XML specification.

The configuration file must be encoded in UTF-8. Be sure that you editing software don't changes the characters encoding.

Be careful with XML escaping ([1]). Observe the following entities:

code	character
&lt	<
&gt	>
&amp	&
&apos	'
&quot	"

- Disabled/enabled

```
<!-- Example 1: -->
<!--<scc:server>
…
</scc:server>-->
```

```
<!-- Example 1: -->
<scc:server>
…
</scc:server>
```

The configuration of **Rooming'it-calendar** has two parts:

- Server Configuration
- Calendar Configuration

---

[1] For detailed information http://www.w3.org/TR/xml-entity-names/ or
http://en.wikipedia.org/wiki/List_of_XML_and_HTML_character_entity_references

4.1   Server Configuration

It is possible to configure multiple servers in the configuration file. If multiple servers are configured, be sure that the id attribute of each server is unique.

Within the tag <scc:servers>, take a look to the *Example 1* in the configuration file. In this example it is disabled. In order to enable it, remove the xml commentary tags at the beginning and the end of the block.

4.1.1   Server baseuri

- Example

```
<!-- Example 1 : -->
<scc:servers>
    <!-- Server (Cardinality=0..n, Type=ics)-->

    <!-- Example 1 : plugncast server -->
    <scc:server id="serverId1" type="roomingit">
      <scc:label>Rooming'it server</scc:label>
   <scc:baseuri>http://192.168.1.130:8080</scc:baseuri>
      <scc:charset>UTF-8</scc:charset>
    </scc:server>

</scc:servers>
```

4.2   Calendar Configuration

Within the tag <scc:calendars>, take a look to the *Example 1* in the configuration file. This example is disabled, to enable it, remove the xml commentary tags at the beginning and the end of the block.

4.2.1   Mandatory parameters

4.2.1.1   Id attribute

The id attribute identifies a calendar. It is used lately as a reference to the calendar, we recommend assigning ids easy to remember.

4.2.1.2   Server attribute

The server attribute is a reference to a configured server. The server attribute is formed like this: url(#**<serverid>**).

Example : *url(#myServer)*

4.2.1.3   Plugncast / calendarui

- Example

```
<scc:server id="plugncast" type="ics">
      <scc:label>My plugncast server</scc:label>
</scc:server>
```

The text plugncast is free text but this name will need to be pasted inside **url(#.....)**

Please see below below

```
<scc:calendar id="room" server="url(#plugncast)">
<param name="resource">room.ics</param>
</scc:calendar>
```

Please see calendar configuration

4.2.1.4   Resource

The resource parameter is the internal resource ID in the **Rooming'it** system. For each calendar it is mandatory to define the resource(s) that are present. The procedure to follow to get the resource IDs is defined in the system requirement chapter.

Get a calendar with a single resource "room1"

Example:

```
<scc:calendar id="one_room" server="url(#innes_Rooming'it)">
        <param name="resource">room1</param>
</scc:calendar>
```

```
<scc:calendar id="all_rooms" server="url(#innes_Rooming'it)">
        <param name="resource">room1</param>
        <param name="resource">room2</param>
        <param name="resource">room3</param>
</scc:calendar>
```

All others parameters of the calendar are optional (or for advanced use).

4.2.2   Optional parameters (classic)

4.2.2.1   *display all the meetings or a part of the meetings in the day*

4.2.2.1.1   StartOffset StartRelated

StartOffset specifies whether the meetings can be displayed before they occur. Default value is zero.

StartRelated specifies whether the meeting can be displayed just before the meeting occurs (**event-start**) or at the beginning of the day (**day-start**).

startRelated	startOffset	Result
event-start	-600	The meeting is visible 600 sec before the event starts
day-start	0	The meetings are visible at the beginning of the day *(00h00)*

Attention: the value start-offset can be negative !

4.2.2.1.2   EndOffset / EndRelated

EndOffset specifies whether the meeting can be display after the event finished with n offset in second. Default value is zero.

EndRelated specifies whether the meeting need to be hidden immediately after they have finished (**event-end**) or only at the end of the day (**day-end**).

endRelated	endOffset	Result

event-end	600	Keep displayed the meetings until there are finished and 600 seconds (10 minutes) after the end of the event.
day-end	0	Keep displayed the meetings until the end of the day *(23:59:59)*

- Display 10 minutes before the meeting starts / hide 10 minutes after the meeting is finished

```
<scc:calendar id="all_rooms" server="url(#innes_Rooming'it)">
        <param name="resource">room1</param>
        <param name="resource">room2</param>
        <param name="startRelated">event-start</param>
        <param name="startOffset">-600</param>
        <param name="endRelated">event-end</param>
        <param name="endOffset">600</param>
</scc:calendar>
```

Attention the value startOffset can be negative

- Display all the meeting at the beginning of the day / hide 10 minutes after the meeting is finished

```
<scc:calendar id="all_rooms" server="url(#innes_Rooming'it)">
        <param name="resource">room1</param>
        <param name="resource">room2</param>
        <param name="startRelated">day-start</param>
        <param name="endRelated">event-end</param>
        <param name="endOffset">600</param>
</scc:calendar>
```

4.2.3   Optional parameters (advanced)

4.2.3.1   RequestWindow

Permits to define the requested time window for the ICS. If the value in days is 5, the ICS will contain the meeting for the next 5 days. In case network anomaly, MeetingRoom should be then able to display the meeting for the next 5 days anyway.

```
<scc:calendar id="one_room" server="url(#innes_Rooming'it)">
        <param name="resource">room1</param>
        <param name="requestWindow">5</param>
</scc:calendar>
```
If not defined, the default value is 7.

4.2.3.2   CachePersistence

MeetingRoom is fetching through **Rooming'it** applet by default ICS every 1 minutes. To avoid network traffic, it is possible to allow applet to update ICS every 5 minutes (or more). In this case, set a cache persistency to 300 (sec) to authorize MeetingRoom to use an existing ICS old of 5 minutes (or less).

```
<scc:calendar id="one_room" server="url(#innes_Rooming'it)">
        <param name="resource">room2</param>
        <param name="cachePersistence">300</param>
</scc:calendar>
```
Default value is 300 seconds

4.2.3.3   Label

The value of this parameter is used if no classroom resource is found

```
<scc:calendar id="room" server="url(#plugncast)">
        <param name="resource">.medias/.applets/.calendarui/room.ics</param>
        <param name="label">My Room</param>
</scc:calendar>
```

4.2.3.4   SendCacheOnServerError

The default value is false. If this parameter is enabled (value true), the connector sends the calendar from the cache (if present) in case of error while retrieving the new calendar information.

This example shows how to enable sendCacheOnServerError parameter.

```
<scc:calendar id="rooms" server="url(#innes_Rooming'it)">
        <param name="sendCacheOnServerError">true</param>
</scc:calendar>
```

5   Troubleshooting / debug

You can test the whole configuration accessing to the following address:

```
http://<your_domaine>/plugnCast/.applets/.roomingit-
calendar/2ical.php?calendarId=<calendarId>
```

- The **<host>** is the IP address or DNS of your Innes Briva or PlugnCast Server.
- The **<calendarId>** is the reference of the calendar to test.

Example:

Ex : if you are on the PlugnCast server (or Briva server)
http://localhost/plugnCast/.applets/.roomingit-calendar/2ical.php?calendarId=all_rooms

This address returns a *meetingroom.ics* file that contains the result of the configured calendar.

Please, if any error message is displayed check your configuration.

You can also test your configuration by using the console mode of the view.

It is possible to run the connector in debug mode by using the following URL

Ex : if you are on the PlugnCast server (or Briva server)
Ex : http://localhost/plugnCast/.applets/.roomingit-calendar/2ical.php?calendarId=all_rooms&debug=yes

This will show the connector running step by step which can allow more detailed troubleshooting.

6 Appendix

6.1 Mapping

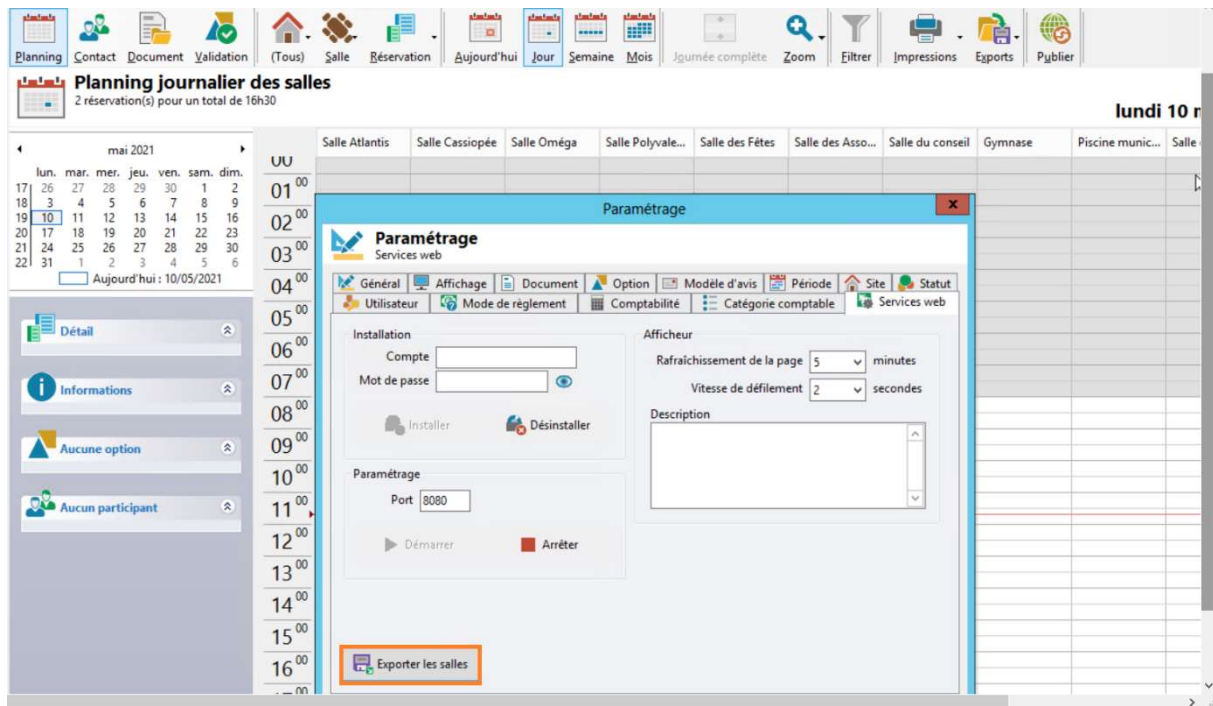The following data mapping is used between the **Rooming'it** XML and the ICS format.

Rooming'it	ICS
UNID	UID
codesSalle	LOCATION
Title	SUMMARY
Description	DESCRIPTION
startDateTime	DTSTART
endDateTime	DTEND
Organizer	ORGANIZER
Attendees	ATTENDEES

6.2 Room export

Launch the **Rooming'it** configuration interface.

In the **File > Settings** menu, use the **Room export** button of the **Web services** tab to get the room ids.
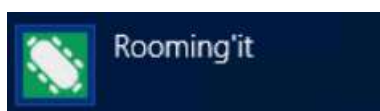

For example:

```
CleSalle;NomSalle
-2098340312;Gymnase
-1528772257;Salle Atlantis
-577481325;Piscine municipale
```
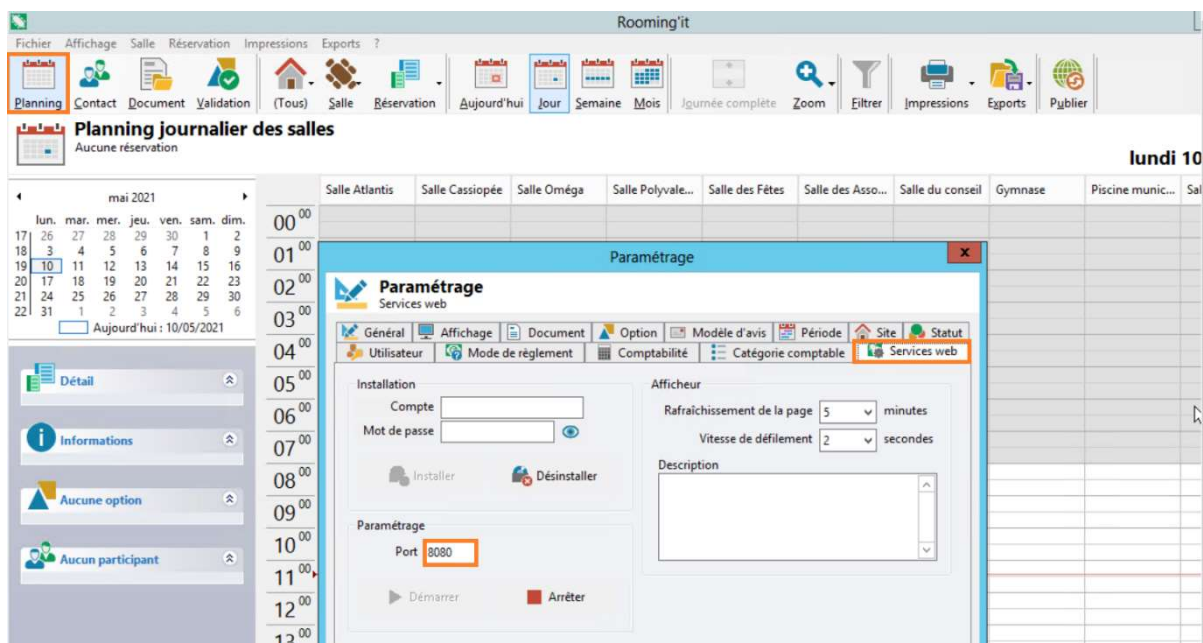
-467875551;Salle Oméga
-461993305;Véhicule AB-123-CD (Livraison)
-419900099;Salle Polyvalente
-100642142;Salle du conseil
986452603;Salle des Fêtes
1396364368;Véhicule DC-312-PM
1493369038;Salle des Associations
1538037070;Dojo
1807597091;Salle de danse
1849015526;Salle Cassiopée

6.3   Rooming'it : change the port

To change the port of the **Rooming'it** Web services, launch the **Rooming'it** configuration interface.
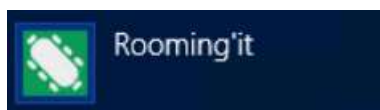

In the **File > Settings** menu, use the **Web service** tab to change the **Rooming'it** port (in the example: 8080).

6.4   Rooming'it : register the MAC address of the Briva server

To register the MAC address of your Briva server in **Rooming'it**, launch the **Rooming'it** configuration interface.

In the **File > Settings** menu, select the **User** tab.

Select **Rooming Web services** value in the drop list, and press on the **Add** button to add a server entry. Fill the *label* and the *MAC address*.