

module.xml

What is module.xml

The module.xml file is the basic descriptor for each module. Used throughout FreePBX, it's primary usage is within Module Admin as when users navigate to Module Admin the module.xml file of each module is reloaded into the database

- [Module.xml Setup](#)
 - Root Blocks
 - Rawname
 - Repo
 - Name
 - Version
 - Type
 - Category
 - Publisher
 - Description
 - Changelog
 - Info
 - License
 - License Link (Framework 12+)
 - Location
 - Md5sum
 - Sha1
 - Menu Items
 - Depends
 - Requirements
 - Supported
 - Popovers (Framework 2.11+)
 - FileInclude (Framework 2.11+)
 - Commercial (Framework 13+)
 - Out of Box Experience (Framework 13+)
 - Database (Framework 14+)
 - Hooks (Framework 12+)
 - Methods
 - Update URL (Framework 13.0.191+)

Module.xml Setup

Root Blocks

Every module.xml file starts off between `<module>`/`</module>` root tags

```
<module>

</module>
```

Rawname

The rawname tag is the raw name of the module. It **MUST** be unique from any other module

```
<module>
  <rawname>joeshmoe</rawname>
</module>
```

Repo

The repo tag is an internal tag used for the FreePBX repo. The options for this tag can be 'standard', 'extended', 'unsupported' or 'commercial'

Note: If you are developing a third-party module this can normally be excluded

```
<module>
  <repo>standard</repo>
</module>
```

Name

The name tag is the pretty name of the module displayed in Module Admin

```
<module>
  <name>Joe Schmozes Module</name>
</module>
```

Version

The version tag is the version of the module. If it changes between viewings of module admin then module admin will mark said module for an 'upgrade'

Note: Words like alpha/beta and rc are OK

Note: Modules are considered 'core' normally match the version of FreePBX they were developed for. Eg. the module below was developed for FreePBX 2.11

```
<module>
  <version>2.11.0.1</version>
</module>
```

Type

The type tag is deprecated it has been replaced with the category tag starting in FreePBX 2.10. It was used in FreePBX 2.9 and lower to determine visual placement of the module within FreePBX. It had two options, either 'setup' or 'tools'

```
<module>
  <type>setup</type>
</module>
```

Category

The category tag is used to determine the visual placement of the module in the drop down category items in FreePBX 2.10+. Options are 'Admin', 'Applications', 'Connectivity', 'Reports', 'Settings', 'Other'.

Note: If a category is specified here but is not defined in FreePBX then the module will be placed in the 'other' category

```
<module>
  <category>Connectivity</category>
</module>
```

Publisher

The publisher tag is used as a vanity tag.

```
<module>
  <publisher>This Guy</publisher>
</module>
```

Description

The description tag is used to describe the module in module admin. Note that it's possible, but discouraged, to use HTML in the Description tag, by replacing < and > with < and >

```
<module>
  <description>My module does some sweet sweet stuff. &lt;br/&gt;
  This is on a new line.</description>
</module>
```

Changelog

The changelog tag is used to describe changes to the module. This is only seen in module admin

```
<module>
  <changelog>This Guy</changelog>
</module>
```

Info

The info tag is used as an external reference to get more information about the module.

```
<module>
  <info>http://joe.com</info>
</module>
```

License

The license tag is used as the license descriptor. All licenses must be compatible with [GPL](#) in order to be contributed back to FreePBX. You can define any license you want for your modules but in order for us to consider your module submissions, they must be licensed as GPL. Not all modules on the FreePBX Mirror servers and available via Module Admin are GPL, such as Commercial Modules distributed by the FreePBX team.

```
<module>
  <license>GPL</license>
</module>
```

License Link (Framework 12+)

It is used to display a clickable link to the license of a given module in Module Admin and to generate a LICENSE file which sits inside the given module folder. All licenses must be compatible with [GPL](#) in order to be contributed back to FreePBX. Please note that the license link should be a plain text version of the license and not HTML or Markup.

```
<module>
  <licenselink>http://www.gnu.org/licenses/gpl-2.0.txt</licenselink>
</module>
```

Location

The location tag is an internal FreePBX reference for where module admin should find the module when downloading it from the external repository server

```
<module>
  <location>release/2.11/</location>
</module>
```

Md5sum

The md5sum tag is an internal FreePBX reference used to compare the downloaded module to the external repository server

```
<module>
  <md5sum>2832e12476c29c6a72c85225fb7ad143</md5sum>
</module>
```

Sha1

The sha1 tag is an internal FreePBX reference used to compare the downloaded module to the external repository server

```
<module>
  <sha1>6fb9f879c5ce62c6de3a03596f89ce94c8a45ee8</sha1>
</module>
```

Menu Items

The Menu Item tag

```
<module>
  <menuitems>
    <epm_devices sort="1">End Point Device List</epm_devices>
    <epm_templates sort="2">End Point Template Manager<
/epm_templates>
    <epm_config sort="3">End Point Configuration</epm_config>
    <epm_advanced sort="4">End Point Advanced Settings<
/epm_advanced>
  </menuitems>
</module>
```

```
<module>
  <menuitems>
    <userpaneltab category="User Panel" href="/recordings"
target="ari" access="all">User Panel</userpaneltab>
  </menuitems>
</module>
```

```
<module>
  <menuitems>
    <backup needsengineedb="yes">Backup & Restore</backup>
    <backup_servers needsengineedb="yes" hidden="true">Backup &
Restore - Servers</backup_servers>
    <backup_templates needsengineedb="yes" hidden="true">Backup
& Restore - Templates</backup_templates>
    <backup_restore needsengineedb="yes" hidden="true">Backup &
Restore - Restore</backup_restore>
  </menuitems>
</module>
```

Depends

The depend tag

```
<module>
  <depends>
    <version>2.6</version>
  </depends>
</module>
```

Requirements

The requirements tag

```
<module>
  <requirements>
    <module>core</module>
    <module>fw_ari</module>
  </requirements>
</module>
```

Supported

The supported tag, used only in the FreePBX mirror server to determine what module to serve up to what FreePBX client. This will be the lowest possible supported FreePBX Framework version.

```
<module>
  <supported>12.0</supported>
</module>
```

Popovers (Framework 2.11+)

The popovers tag.

```
<module>
  <popovers>
    <announcement>
      <display>announcement</display>
    </announcement>
  </popovers>
</module>
```

NOTE: That if there are multiple forms on the page that is being displayed, you should add the popover-form class to the form which is being made available to the user via popovers such as

```
<form class="popover-form" name="editGRP" action="<?php $_SERVER
['PHP_SELF'] ?>" method="post" onsubmit="return checkGRP(editGRP);">
```

FileInclude (Framework 2.11+)

For more information about it's usage see [Restricting Module Installation](#).

```
<fileinclude>
  <install>functions.inc.php</install>
</fileinclude>
```

```
<fileinclude>
    <uninstall>functions.inc.php</uninstall>
</fileinclude>
```

```
<fileinclude>
    <install>functions.inc.php</install>
    <uninstall>functions.inc.php<uninstall>
</fileinclude>
```

Commercial (Framework 13+)

Commercial modules have two sub tags, 'type' which can be 'free', 'paid' or 'upgradeable', and 'link' which should be a protocol-independent link (eg, it should start with two slashes, not with http or https)

When the commercial tag is present, a 'Buy' or 'Obtain' button appears in module admin, depending on the value of 'type'. When clicking on the button, a new window is opened with the target being link.

```
<commercial>
    <type>free</type>
    <link>//website.com/linkto/buypage</link>
</commercial>
```

Out of Box Experience (Framework 13+)

Any modules that wish to provide an Out of Box Experience page must provide an 'oobe' tag in their XML. This is documented fully on the OOBE page - [Out of Box Experience](#)

```
<oobe method="oobeHook" priority="200" />
```

Database (Framework 14+)

Database information. This let's FreePBX know how the database should be generated. The database structure is setup/executed before the install() function/install.php is run. For more information see: [Database 13#Creating/Altering/UpdatingaDatabaseThroughmodule.xml \(FreePBX14+\)](#)

```

<database>
  <table name="trunks">
    <field name="trunkid" type="integer" default="0"
primaryKey="true"/>
    <field name="name" type="string" length="50" default=""/>
    <field name="tech" type="string" length="20" primaryKey="
true"/>
    <field name="outcid" type="string" length="40" default=""/>
    <field name="keepcid" type="string" length="4" default="
off" notnull="false"/>
    <field name="maxchans" type="string" length="6" default="
notnull="false"/>
    <field name="failscrip" type="string" length="255"
default=""/>
    <field name="dialoutprefix" type="string" length="255"
default=""/>
    <field name="channelid" type="string" length="255"
default="" primaryKey="true"/>
    <field name="usercontext" type="string" length="255"
notnull="false"/>
    <field name="provider" type="string" length="40" notnull="
false"/>
    <field name="disabled" type="string" length="4" default="
off" notnull="false"/>
    <field name="continue" type="string" length="4" default="
off" notnull="false"/>
  </table>
</database>

```

Hooks (Framework 12+)

Hooking information to let FreePBX know that this module has functions/methods that need to hook into other modules. For more information on how this works see: [BMO Hooks#GeneralModuleHooks](#)

```

<hooks>
  <bulkhandler namespace="FreePBX\modules" class="Bulkhandler"
priority="100">
    <method namespace="FreePBX\modules" class="Core"
callingMethod="getTypes">bulkhandlerGetTypes</method>
    <method namespace="FreePBX\modules" class="Core"
callingMethod="getHeaders">bulkhandlerGetHeaders</method>
    <method namespace="FreePBX\modules" class="Core"
callingMethod="validate">bulkhandlerValidate</method>
    <method namespace="FreePBX\modules" class="Core"
callingMethod="import">bulkhandlerImport</method>
    <method namespace="FreePBX\modules" class="Core"
callingMethod="export">bulkhandlerExport</method>
  </bulkhandler>
</hooks>

```


Methods

The methods tag helps FreePBX to prioritize function calls during certain processes. In the call below we are setting a priority of 480 to core's core_go_get_config function. This happens during reload where-as all other get_config functions are set at a default priority of 500

```
<methods>
  <get_config pri="480">core_do_get_config</get_config>
</methods>
```

Update URL (Framework 13.0.191+)

This allows you to define an updateURL for this module. When "Check Online" is run Framework will reach out to the server defined in this URL to ask it for updates to this module. For more information about the Update URL and it's functionality see: [Third Party Module Hosting](#)

```
<updateurl>http://link.to.json.file.json</updateurl>
```