Plugin XMLUPDATE

Introduced in [Tiki15](#) and backported to Tiki 12.5 and Tiki14.2

Use this [wiki plugin](#) to easily update selected, multiple elements of an XML file stored in a File Gallery, by using a form in the wiki page.

This plugin was originally developed so that Tiki managed 'control' XML files could be downloaded by multiple Raspberry Pi's that are used for data collection and other IoT purposes, and where the updating of individual elements of the control file could be delegated to individual Tiki users.

Many other uses will however be possible.

XML file requirements

The XML file is stored in a File Gallery and File Galleries must (at present) use the option to store all files in a Directory.

The XML file must also be a simple structure that does not repeat any of the node names as names are searched for irrespective of what 'level' they are at.

The following file is used as a worked example in this documentation:

```
<?xml version="1.0"?>
<management>
        <servername description="the host name of the Pi server">pitiki01</servername>
        <control description="">
                <stopimaging description="">no</stopimaging>
                <starttime description="">07:00</starttime>
                <stoptime description="">22:30</stoptime>
                <suspendimaging description="">no</suspendimaging>
        </control>
        <sitedata description="">
                <uploadgallery description="">4</uploadgallery>
                <uploadname description="">kitchen_image</uploadname>
                <downloadgallery description="">5</downloadgallery>
                <downloadXMLfileId description="tiki fileId for the XML
 file">1</downloadXMLfileId>
                <cookiepath description="">/home/pi/pirepo_cookies/</cookiepath>
                <cookiefile description="">pitiki01-cookies.txt</cookiefile>
        </sitedata>
        <intervals description="">
                <uploadinterval description="">300</uploadinterval>
                <downloadXMLinterval description="">1800</downloadXMLinterval>
        </intervals>
</management>
```

Parameters

Introduced in Tiki 15. Required parameters are in `bold`.
[Go to the source code](#)

Parameters	Accepted Values	Description	Default	Since
(body of plugin)		list (one per line) of the XML element (node) names irrespective of their path and must (at present) be unique in the file		
`fileId`	digits	File Id of the XML file stored in a File Gallery which is assumed to store its data in a directory		15.0
`attribute`	text	optional use of an attribute for the XML nodes - used as a label in the input form. If used then all nodes in the XML file should have the attribute text set even if it is a blank/space		15.0
`namelisted`	yes no	yes/no option to include the XML node name in the plugin output - default is yes - and should always be yes if the attribute parameter is not used or not all the nodes have their attribute text set	yes	15.0

Example

The following is the code for the worked example:

```
{XMLUPDATE(fileId="1281" attribute="description" namelisted="yes")}
servername

notinXMLfile

downloadXMLfileId
downloadXMLinterval

{XMLUPDATE}
```

As you can see the node names that are being updated are simply listed in the plugin body, and as illustrated in this example, blank lines are ignored and any node names that are not in the XML file will be flagged as 'not found'.

The resulting output screen shot from this example is below

The current values for the 4 items from the XML management file are shown in the input fields below.

Change any individual values and click the 'Update' button to submit the revised values and to resave them in the File Gallery XML file.

xml node name	xml node description	xml node current/new value
servername	the host name of the Pi server	pitiki01
notinXMLfile		This item not found in the XML file
downloadXMLfileId	tiki fileId for the XML file	1
downloadXMLinterval		1800

Update

The output format for the form includes the text just above the form which confirms the number of node names being considered and provides basic text for what to do.

The form is structured as a table and depending upon what theme is being used some padding may need to be applied to the table cells to make the form look 'nice'.