

Introduction

Notation

- Text that is to appear exactly as shown is shown in Courier font.
- Required input is shown in **bold**.
- XML elements that may be repeated are shaded.
- Choices are indicated by [choice1|choice2]. The first is the default value.

Table Tree

A table is represented by an XML tree of the form:

```
<table name=" table name">
  <title>table title</title>
  <sdesc>short description</sdesc>
  <ldesc>long description</ldesc>
  <link>link header</link>
  <maxdown>download limit</maxdown>
  <cols>
    <th name="column name"
      type="type"
      format="format" >
      column heading
    </th>
  </cols>
  <data>
    <tr>
      <td>value</td>
    </tr>
  </data>
</table>
```

- <th> under <cols> is repeated for each column.
- type is i (integer), f (float) or a (alphanumeric).
- <tr> under <data> is repeated for each row.
- <td> under <tr> is repeated for each column.

Transaction Details

Log In (login)

Query String:

```
api=login&apiversion=2&uid=userid
&pswd=password &kill=[no|yes]
```

Specification (XML):

None

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

```
<sid>session ID</sid>
<pswd>encrypted password</pswd>
```

Log Out (logout)

Query String:

```
api=logout&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

None

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

Directory Listing (dir)

Query String:

```
api=dir&apiversion=2&uid=userid
&pswd=encrypted-password&sid=session-ID
```

Specification (XML):

```
<name>directory name</name>
```

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
<table>directory listing</table>
```

- The result table contains two columns, both alphanumeric: name and type. type has the value "table" or "dir".

Table Information (tabinfo)

Query String:

```
api=tabinfo&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
<name>table name</name>
```

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
<table>table with no rows</table>
```

Apply Query (query)

Query String:

```
api=query&apiversion=2&uid=userid
&pswd=encrypted-password&sid=session-ID
```

Specification (XML):

```
<name>table name</name>
<ops>query operations</ops>
```

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

```
<nrows>number of rows in result</nrows>
<table>table with no rows</table>
```

- query operations is the macro language used on the Edit Actions page of the browser interface.

Get Query Results (getdata)

Query String:

```
api=getdata&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
<cols>
  <col>column name</col>
</cols>
```

and one of

```
<rows mode="1">
  <next>number of rows</next>
</rows>
```

or

```
<rows mode="2">
  <from>row number</from>
  <to>row number</to>
</rows>
```

and one of

```
<format type="xml"
  values=[raw|formatted]>
```

or

```
<format type="csv">
  values=[raw|formatted]>
  <sep>column separator</sep>
  <linesep>line separator</linesep>
</format>
```

- <col> is repeated for each desired column.
- The default format is XML.
- The default column separator is a comma.
- The default line separator is a newline.

Result (XML):

```
If format is XML:
<rc>return code</rc>
<msg>system message</msg>
<table>table with requested data</table>
If format is CSV:
<rc>return code</rc>
<msg>system message</msg>
<csv>
<![CDATA[character-separated data ]]>
</csv>
```

- The first row of CSV data contains the column headings.

Save as File (savefile)

Query String:

```
api=savefile&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
<file>file name</file>
<content>
  <meta type=[headers|names]/>
</content>
<format>
  <sep>field separator</sep>
  <linesep>record separator</linesep>
</format>
```

- <meta> is repeated for each desired type of information.
- The default field separator is a comma.
- The default record separator is a newline.

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

Save as Table (savetable)

Query String:

```
api=savetable&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
<name mode="replace?">name</name>
<title>title</title>
<sdesc>short description</sdesc>
<ldesc>long description</ldesc>
<link>link header</link>
<maxdown>download limit</maxdown>
and one of
<users type="private"/>
or
<users type="inherit"/>
or
<users type="list">
  <user>user ID</user>
</users>
```

- <user> is repeated for each user.

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

Upload Table (upload)

Query String:

```
api=upload&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
<table>table</table>
<name mode="replace?">name</name>
and one of
<users type="private"/>
or
<users type="inherit"/>
or
<users type="list">
  <user>user ID</user>
</users>
```

- <user> is repeated for each user.

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

Merge Tables (merge)

Query String:

```
api=merge&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
<tabs>
  <tab>name</tab>
</tabs>
<name mode="replace?">name</name>
<title>title</title>
<sdesc>short description</sdesc>
<ldesc>long description</ldesc>
<link>link header</link>
<maxdown>download limit</maxdown>
and one of
<users type="private"/>
or
<users type="inherit"/>
or
<users type="list">
  <user>user ID</user>
</users>
```

- <user> is repeated for each user.

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

Delete Table (droptable)

Query String:

```
api=droptable&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
<name>table name</name>
```

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```

Clear the Cache (clear)

Query String:

```
api=clear&apiversion=2
&uid=userid &pswd=encrypted-password
&sid=session-ID
```

Specification (XML):

```
None
```

Result (XML):

```
<rc>return code</rc>
<msg>system message</msg>
```