

# WF-system Web services guide

---

Version 3.0

© 2004-2012, ZAO Runa, this document is available under GNU FDL license. RUNA WFE is an open source system distributed under a LGPL license (<http://www.gnu.org/licenses/lgpl.html>).

RUNA WFE is an open source end-user oriented system for enterprise business process management, fully implemented in Java. Designed for large and middle-size enterprises.

## Introduction

Web service is a method of platform independent system interaction. It can use HTTP and performs information exchanges in XML format. SOAP over HTTP is used for remote calls.

SOAP message looks like this:

SOAP envelope

SOAP header

Header element 1

Header element 2

....

Header element N

SOAP body

Body element N

....

Body element 2

Body element 1

Each web service has a unique WSDL document that describe web-service. He defines the service location and operations (or methods) that it provides.

## RunaWFE web services

Web services start automatically when RunaWFE server is started. Web services are located in 2 subsystems.

### Af subsystem

To access af services use URI: [http://host:port/af.webservice/webservice\\_name](http://host:port/af.webservice/webservice_name).

AuthenticationWebService (<http://localhost:8080/af.webservice/AuthenticationBean?wsdl>) – performs executor authorization on the server

BotWebService (<http://localhost:8080/af.webservice/BotBean?wsdl>) – performs bots and botstations manipulations

ExecutorWebService (<http://localhost:8080/af.webservice/ExecutorBean?wsdl>) – performs manipulations with executors

ProfileWebService (<http://localhost:8080/af.webservice/ProfileBean?wsdl>) – copies executor profile to all other executors

---

SubstitutionBean (<http://localhost:8080/af.webservice/SubstitutionBean?wsdl>) – performs manipulations with substitution rules

## Wf subsystem

To access wfe services use URI: [http://host:port/wf.webservice/webservice\\_name](http://host:port/wf.webservice/webservice_name).

ArchivingWebService (<http://localhost:8080/wf.webservice/ArchivingBean?wsdl>) – performs manipulations with archives

DefinitionWebService (<http://localhost:8080/wf.webservice/DefinitionBean?wsdl>) – performs manipulations with process definitions

ExecutionWebService (<http://localhost:8080/wf.webservice/ExecutionBean?wsdl>) – performs manipulations with process instances

PermissionWebService (<http://localhost:8080/wf.webservice/PermissionBean?wsdl>) – performs manipulations with permissions

## Invoking RunaWFE web services from java application

As opposed by JAX-WS we should generate interoperability classes from WSDL. In Jboss we can use wsconsume.bat(sh). Also you can download generated classes with sources (runawfe-ws-client.xxx.jar) from <https://sourceforge.net/projects/runawfe/files/SRC%20and%20BIN%20files/extras/>.

After this add them to classpath and use API.

This sample demonstrates authentication and task list receiving:

```
try {
    AuthenticationWebService authenticationWebService = new AuthenticationWebService(new URL(
        "http://172.16.124.3:8080/af.webservice/AuthenticationBean?wsdl"));
    Authentication authentication = authenticationWebService.getAuthenticationPort();
    ActorPrincipal principal = authentication.authenticateDB("Administrator", "wf");
    System.out.println("Authenticated as " + principal);
    ExecutionWebService executionWebService = new ExecutionWebService(new URL(
        "http://172.16.124.3:8080/wf.webservice/ExecutionBean?wsdl"));
    Execution execution = executionWebService.getExecutionPort();
    List<WfeTask> tasks = execution.getTasks(principal);
    System.out.println("Your task list size: " + tasks.size());
} catch (Exception e) {
    e.printStackTrace();
}
```

## Authentication web service description.

Can be accessed by name "AuthenticationBean".

## AuthenticateDB command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
name	User name	yes
password	Password	yes

As a result ActorPrincipal object is returned. This object is necessary in order to run each data manipulation command via web service.

Description of ActorPrincipal object

ActorPrincipal	
<i>Attribute</i>	<i>Description</i>
key	Encryption key
<b>Embedded elements:</b>	
Actor:	
<i>Attribute</i>	<i>Description</i>
id	User identifier
version	User Version
name	User name
description	User Description
fullName	User full name
code	User code
isActive	User status
email	User email

## Description of Bot web service.

Accessible via BotBean name.

### CreateBotStation command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Botstation name	yes
address	Bostation name	no

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:createBotStation>
```

```

<web:actorPrincipal>
<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:name>testbotstation</web:name>
<web:address>localhost</web:address>
</web:createBotStation>
</soapenv:Body>
</soapenv:Envelope>

```

## UpdateBotStation command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot station name	yes
newName	New bot station name	No
address	Botstation's new address	No

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://runa.ru/workflow/webservices">
<soapenv:Header/>
<soapenv:Body>
<web:updateBotStation>
<web:actorPrincipal>
<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>

```

```

<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:name>test</web:name>
<web:newName>?</web:newName>
<web:address>?</web:address>
</web:updateBotStation>
</soapenv:Body>
</soapenv:Envelope>

```

## RemoveBotStation command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot station name	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removeBotStation>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:name>testbotstation</web:name>
    </web:removeBotStation>
  </soapenv:Body>
</soapenv:Envelope>

```

```
</soapenv:Body>
```

```
</soapenv:Envelope>
```

## CreateBot command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot's name	yes
botStation	Bot station name	yes
startTimeout	Start timeout	No
password	Password to login to the system	No

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:createBot>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:botStationName>testpkbotstation</web:botStationName>
      <web:name>testpkbot</web:name>
      <web:password>321</web:password>
      <web:starttimeout>200</web:starttimeout>
    </web:createBot>
  </soapenv:Body>
</soapenv:Envelope>
```

## UpdateBot command

Element attributes		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot's name	yes
newName	New bot's name	No
botStation	Bot station name	No
newBotStation	New bot station name	No
startTimeout	New start timeout	No
password	New password	No

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:updateBot>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:botStationName>testpkbotstation</web:botStationName>
      <web:name>testbot</web:name>
      <web:newName>newtestbot</web:newName>
      <web:password>321</web:password>
      <web:starttimeout>300</web:starttimeout>
    </web:updateBot>
  </soapenv:Body>
</soapenv:Envelope>

```

## DeleteBot command

Element attributes		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot's name	yes
botStation	Bot station name	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removeBot>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:botStationName>testbotstation</web:botStationName>
      <web:name>testbot</web:name>
    </web:removeBot>
  </soapenv:Body>
</soapenv:Envelope>

```

## AddConfigurationsToBot command



<b>Element attributes</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot's name (the one we want to add a configuration to.)	yes
botStation	Bot station name	no
<b>Embedded elements:</b>		
botConfiguration	The configuration to be added	Not less than one
<i>Attribute</i>	<i>Description</i>	
name	Configuration name	yes
handler	Configuration handler	yes
config	Configuration file	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:addConfigurationsToBot>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:botStationName>testpkbotstation</web:botStationName>
      <web:botName>testpkbot</web:botName>
      <web:configurations>
        <web:name>DoNothing</web:name>
        <web:handler>ru.runa.wf.logic.bot.DoNothingTaskHandler</web:handler>
        <web:config></web:config>
      </web:configurations>
      <web:configurations>
        <web:name>create text file</web:name>

```

```

<web:handler>ru.runa.wf.logic.bot.TextReportTaskHandler</web:handler>
<web:config>/bot/handler/textFile.xml</web:config>
</web:configurations>
</web:addConfigurationsToBot>
</soapenv:Body>
</soapenv:Envelope>

```

## RemoveConfigurationsFromBot command

Element attributes		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot's name (from which we want to remove configuration)	yes
botStation	Bot station name	no
<b>Embedded elements:</b>		
botConfiguration	The configuration you want to remove	One or more
<i>Attribute</i>	<i>Description</i>	
name	Configuration name	yes
handler	Configuration handler	yes
config	Configuration file	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removeConfigurationsToBot>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:botStationName>testbotstation</web:botStationName>
    </web:removeConfigurationsToBot>
  </soapenv:Body>
</soapenv:Envelope>

```

```

<web:botName>testpkbot</web:botName>
<web:configurations>
<web:name>DoNothing</web:name>
<web:handler>ru.runa.wf.logic.bot.DoNothingTaskHandler</web:handler>
<web:config></web:config>
</web:configurations>
</web:removeConfigurationsToBot>
</soapenv:Body>
</soapenv:Envelope>

```

## RemoveAllConfigurationsFromBot command

Element attributes		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Bot's name	yes
botStation	Bot station name	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removeAllConfigurationsFromBot>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:botStationName>testbotstation</web:botStationName>
      <web:botName>testbot</web:botName>
    </web:removeAllConfigurationsFromBot>
  </soapenv:Body>

```

</soapenv:Envelope>

## Description of Executor web service.

Accessible via ExecutorBean name.

### CreateActor

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	User name	yes
fullName	Full user name	no
description	Description of user	no
password	User's password	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:createActor>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:name>test2</web:name>
      <web:fullName>test2</web:fullName>
      <web:description></web:description>
      <web:passwd>123</web:passwd>
    </web:createActor>
  </soapenv:Body>

```

</soapenv:Envelope>

## CreateGroup command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Group name	yes
description	Description of group	no

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:createGroup>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:name>testGroup</web:name>
      <web:description></web:description>
    </web:createGroup>
  </soapenv:Body>
</soapenv:Envelope>
```

## DeleteExecutor command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the executor that should be removed	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:deleteExecutor>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:name>testGroup</web:name>
    </web:deleteExecutor>
  </soapenv:Body>
</soapenv:Envelope>
```

## DeleteExecutors command

<b>Embedded elements:</b>		
Element	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
deleteExecutor	Executors that are to be deleted	1 or more

The command makes it possible to delete several executors in one query. It's just like to delete each executor individually, but it takes less time to complete.

Example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:deleteExecutors>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:names>test</web:names>
      <web:names>test2</web:names>
    </web:deleteExecutors>
  </soapenv:Body>
</soapenv:Envelope>
```

## AddExecutorsToGroup command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the group to which the executor is added.	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
executor	Executor (actor or group)	0 or more

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:addExecutorsToGroup>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:executors>test</web:executors>
      <web:executors>test2</web:executors>
      <web:groupName>testGroup</web:groupName>
    </web:addExecutorsToGroup>
  </soapenv:Body>
</soapenv:Envelope>

```

## RemoveExecutorsFromGroup command



<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of a group from which executors are removed	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
executor	Executor (actor or group)	0 или более

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removeExecutorsFromGroup>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:executors>test</web:executors>
      <web:executors>test2</web:executors>
      <web:groupName>testGroup</web:groupName>
    </web:removeExecutorsFromGroup>
  </soapenv:Body>
</soapenv:Envelope>

```

## Description of Archiving webservice.

Accessible via ArchivingBean name.

### RemoveOldProcessInstances command

Element attributes		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
name	Process definition name	No
version	The max version number to remove process definition up to	No
id	ID of process instance for removal	No
idTill	If this parameter is present, all the instances with ids in a range from "id" and up to "idTill" (inclusively) will be removed	No
startDate	Remove all the process instances with the start date prior to startDate	No
endDate	Remove all the process instances with the end date prior to endDate	No
onlyFinished	Removed only finished process instances	No
dateInterval	Remove all the process instances with the start date in the interval between startDate and endDate	No

Parameters for process instances removal

1. onlyFinished - Set true to filter only finished processes for removal and false to turn off the filter. (The default value is true)
2. endDate - If dateInterval parameter is set to true then endDate is used to set the (excluded) end of the interval for process instances starting date.
3. startDate - If dateInterval parameter is set to true then startDate is used to set the (excluded) beginning of the interval for process instances starting date.
4. name - Business process name
5. version - Business process version for the processes with name set in parameter number 4. If you want to set version parameter the name parameter must also be set.
6. id - of a particular business process instance.
7. idTill - if this parameter is set then all the process instances with ids in the range from "id" to "idTill" (inclusively) will be removed.
8. dateInterval – if this parameter set to true, the parameters 2 and 3 must also be set. Those parameters values will define the interval for the process instances start date. (By default dateInterval = false)

If parameter 6 is not set, then parameter 2 or 3 (or both) must be set

The process instance will be removed only if all the conditions set in parameters are true for it.

Note. In order to set a date value in the parameters use the following format:

"2007-02-22T23:59:59"

Example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removeOldProcessInstances>
```

```

<web:actorPrincipal>
<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:name></web:name>
<web:version></web:version>
<web:id>1</web:id>
<web:idTill></web:idTill>
<web:startDate></web:startDate>
<web:endDate></web:endDate>
<web:onlyFinished></web:onlyFinished>
<web:dateInterval</web:dateInterval>
</web:removeOldProcessInstances>
</soapenv:Body>
</soapenv:Envelope>

```

## RemoveOldProcessDefinitionVersion

Element attributes		
Attribute	Description	Mandatory
name	The name of process definition	yes
version	The max version number that can be removed	No

The process definitions with the given name and version less or equal to given will be removed only if:

- A. There are no process instances (both completed and not completed) for those definitions in the system
- B. This is not the latest business process version.

If the A or B condition is not satisfied the business process definition version is not removed. If the version parameter is not set, all the process definitions with given name, with no instances and of not the latest version are removed.

Example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://runa.ru/workflow/webservices">

```

```

<soapenv:Header/>
<soapenv:Body>
<web:removeOldProcessDefinitionVersion>
<web:actorPrincipal>
<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:name>TimerDemo</web:name>
<web:version>5</web:version>
</web:removeOldProcessDefinitionVersion>
</soapenv:Body>
</soapenv:Envelope>

```

## ArchiveOldProcessInstances command

All the attributes and parameters are the same as for removeOldProcessInstances

Additional settings:

In the wf\_logic.properties file there is an archive.db.enabled parameter of boolean type. This parameter must be set to true in order to enable operations with archive database.

In {\${JBoss\_HOME}/server/default/conf} it's necessary to add hibernate.archive.cfg. This hibernate.cfg copy differs from hibernate.cfg only in the B hibernate.archive.cfg parameter <property name="hibernate.connection.datasource"> value, where the archive datasource name is set. Make sure that the datasource file is present in jboss server

See "installation and configuration guide" for detailed info on how to create datasource.

There is only one archive database that we archive data into and retrieve data from.

When an object is archived its id stays intact. If there's an object with such id already exists in the archive it will be rewritten. When an object is retrieved the id might change. If there's an object with the same id the source database already the retrieved object's id will be the current id counter + 1 value.

You can view archived processes data via web interface. In order to do it you will need a separate RunaWFE server, where in hibernate.cfg the hibernate.connection.datasource property is set to archive database data source name. You can not add new elements via web interface if you use certain databases (e.g. MSSQL).

## ArchiveOldProcessDefinitionVersion command

Attributes and parameters are the same as for removeOldProcessDefinitionVersion.

Additional settings are the same as for archiveOldProcessInstances element.

## RetrieveOldProcessInstances command

Attributes and parameters are the same as for removeOldProcessDefinitionVersion.

Additional settings are the same as for archiveOldProcessInstances element.

## RetrieveOldProcessDefinitionVersion command

Attributes and parameters are the same as for removeOldProcessDefinitionVersion.

Additional settings are the same as for archiveOldProcessInstances element.

## Description Permission веб – сервиса.

Accessible via "PermissionBean" name.

## AddPermissionsOnActor command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the user the permissions are added on	yes
executor	The executor that receives permissions on the given actor	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:addPermissionsOnActor>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
```

```

<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:actor>test2</web:actor>
<web:executor>test</web:executor>
<web:permissions>permission.update_permissions</web:permissions>
<web:permissions>permission.update_executor</web:permissions>
</web:addPermissionsOnActor>
</soapenv:Body>
</soapenv:Envelope>

```

## SetPermissionsOnActor command

Element attributes:		
Attribute	Description	Mandatory
actorPrincipal	Authorized user	yes
name	The name of the user the permissions are added on	yes
executor	The executor that recieves permissions on the given actor	yes
Embedded elements:		
Element	Description	Mandatory
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:setPermissionsOnActor>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>

```

```

<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:actor>test2</web:actor>
<web:executor>test</web:executor>
<web:permissions>permission.read</web:permissions>
</web:setPermissionsOnActor>
</soapenv:Body>
</soapenv:Envelope>

```

## RemovePermissionsOnActor command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the user the permissions are edited on	yes
executor	The executor whose permissions on the given actor are being revoked	yes
Embedded elements:		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removePermissionsOnActor>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:actor>test2</web:actor>
      <web:executor>test</web:executor>
    </web:removePermissionsOnActor>
  </soapenv:Body>
</soapenv:Envelope>

```

```

<web:permissions>permission.update_permissions</web:permissions>
</web:removePermissionsOnActor>
</soapenv:Body>
</soapenv:Envelope>

```

## AddPermissionsOnGroup command

See below

## SetPermissionsOnGroup command

See below

## RemovePermissionsOnGroup command

Attributes for commands \*PermissionsOnActor and \*PermissionsOnGroup are the same and summed up here:

Possible permission types for Actors:

- *permission.read*
- *permission.update\_permissions*
- *permission.update\_actor\_status*
- *permission.update\_executor*

Possible permission types for Groups:

- *permission.read*
- *permission.update\_permissions*
- *permission.update\_executor*
- *permission.list\_group*
- *permission.list\_group*
- *permission.remove\_from\_group*

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the executor the permissions are edited on	no
executor	The executor that receives/loses permissions on the given executor	yes
Embedded elements:		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	0 или более

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removePermissionsOnGroup>
      <web:actorPrincipal>

```



```

<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:group>testGroup</web:group>
<web:executor>test</web:executor>
<web:permissions>permission.remove_from_group</web:permissions>
</web:removePermissionsOnGroup>
</soapenv:Body>
</soapenv:Envelope>

```

## AddPermissionsOnSystem command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
executor	The executor that receives permissions on the system	yes
Embedded elements:		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://runa.ru/workflow/webservices">
<soapenv:Header/>
<soapenv:Body>
<web:addPermissionsOnSystem>
<web:actorPrincipal>
<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>

```

```

<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:executor>employee </web:executor>
<web:permissions>permission.login_to_system </web:permissions>
</web:addPermissionsOnSystem>
</soapenv:Body>
</soapenv:Envelope>

```

## SetPermissionsOnSystem command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
executor	The executor that receives permissions on the system	yes
Embedded elements:		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:setPermissionsOnSystem>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>

```

```

</web:actorPrincipal>
<web:executor>employee</web:executor>
<web:permissions>permission.read</web:permissions>
</web:setPermissionsOnSystem>
</soapenv:Body>
</soapenv:Envelope>

```

## RemovePermissionsOnSystem command

Element attributes		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
executor	The executor whose permissions on the system are revoked	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removePermissionsOnSystem>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:executor>employee </web:executor>
      <web:permissions>permission.login_to_system </web:permissions>
    </web:removePermissionsOnSystem>
  </soapenv:Body>
</soapenv:Envelope>

```

Attributes for commands \*PermissionsOnSystem are the same and descriptions are summed below:

Valid permission types:

- permission.read
- permission.update\_permissions
- permission.login\_to\_system
- permission.change\_self\_password
- permission.create\_executor
- permission.deploy\_definition
- permission.change\_self\_password

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
executor	The executor that receives/loses permissions on the system	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

## AddPermissionsOnDefinition command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the process definition on which the permissions are added	yes
executor	The executor that receives permissions on the definition	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:addPermissionsOnDefinition>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
        </web:actor>
      </web:actorPrincipal>
    </web:addPermissionsOnDefinition>
  </soapenv:Body>
</soapenv:Envelope>

```

```

<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:definition>validProcess</web:definition>
<web:executor>employee</web:executor>
<web:permissions>permission.read</web:permissions>
</web:addPermissionsOnDefinition>
</soapenv:Body>
</soapenv:Envelope>

```

## SetPermissionsOnDefinition command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the process definition on which the permissions are set	yes
executor	The executor that receives permissions on the definition	yes
Embedded elements:		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:setPermissionsOnDefinition>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>

```

```

<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:definition>validProcess</web:definition>
<web:executor>employee</web:executor>
<web:permissions>permission.read</web:permissions>
</web:setPermissionsOnDefinition>
</soapenv:Body>
</soapenv:Envelope>

```

## RemovePermissionsOnDefinition command

Element attributes:		
Attribute	Description	Mandatory
actorPrincipal	Authorized user	yes
name	The name of the process definition on which the permissions are removed	yes
executor	The executor whose permissions are being revoked	yes
Embedded elements:		
Element	Description	Mandatory
permission	Permission type	no

пример тега:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removePermissionsOnDefinition>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:definition>validProcess</web:definition>
      <web:executor>employee</web:executor>
    </web:removePermissionsOnDefinition>
  </soapenv:Body>
</soapenv:Envelope>

```

```

<web:permissions>permission.read</web:permissions>
</web:removePermissionsOnDefinition>
</soapenv:Body>
</soapenv:Envelope>

```

Attributes of commands \*PermissionsOnDefinition are the same and description for them is summed up below.

Valid permissions are:

- permission.read
- permission.update\_permissions
- permission.undeploy\_definition
- permission.redeploy\_definition
- permission.start\_process
- permission.read\_instance
- permission.cancel\_instance.

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	The name of the process definition on which the permissions are changed	no
executor	The executor that receives/loses permissions on the definition	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

## AddPermissionsOnProcessInstances command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Process instance on which the permissions are changed	yes
executor	The executor that receives permissions on the process instances	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://runa.ru/workflow/webservices">
<soapenv:Header/>
<soapenv:Body>
<web:addPermissionsOnProcessInstances>
<web:actorPrincipal>
<web:actor>

```

```

<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:processInstance>Vacation</web:processInstance>
<web:executor>employee </web:executor>
<web:permissions>permission.cancel_instance</web:permissions>
</web:addPermissionsOnProcessInstances>
</soapenv:Body>
</soapenv:Envelope>

```

## SetPermissionsOnProcessInstances command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Process instance on which the permissions are changed	yes
executor	The executor that receives permissions on the process instances	yes
Embedded elements:		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:setPermissionsOnProcessInstances>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>

```



```

<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:processInstance>Vacation</web:processInstance>
<web:executor>employee </web:executor>
<web:permissions>permission.cancel_instance</web:permissions>
</web:setPermissionsOnProcessInstances>
</soapenv:Body>
</soapenv:Envelope>

```

## RemovePermissionsOnProcessInstances command

Element attributes:		
Attribute	Description	Mandatory
actorPrincipal	Authorized user	yes
name	Process instance on which the permissions are changed	yes
executor	The executor whose permissions on the process instances are being revoked	yes
Embedded elements:		
Element	Description	Mandatory
permission	Permission type	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removePermissionsOnProcessInstances>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>

```

```

</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:processInstance>Vacation</web:processInstance>
<web:executor>employee</web:executor>
<web:permissions>permission.cancel_instance</web:permissions>
</web:removePermissionsOnProcessInstances>
</soapenv:Body>
</soapenv:Envelope>

```

Синтаксис и семантика команд \*PermissionsOnProcessInstances одинаковы и Description всех команд собрано в одно. Полномочия, допустимые в командах:

- permission.read
- permission.update\_permissions
- permission.cancel\_instance

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Process instance on which the permissions are changed	yes
executor	The executor whose permissions on the process instances are being revoked	yes
<b>Embedded elements:</b>		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
permission	Permission type	no

## RemoveAllPermissionsFromExecutor command

see below

## RemoveAllPermissionsFromProcessDefinition command

see below

## RemoveAllPermissionsFromProcessInstances command

Those 3 commands have the same attributes:

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	A name of the object the permissions on which are to be revoked from all the users (but Administrators?)	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:removeAllPermissionsFromProcessInstances>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:processes>Vacation</web:processes>
    </web:removeAllPermissionsFromProcessInstances>
  </soapenv:Body>
</soapenv:Envelope>

```

## Description Profile веб – сервиса.

Accessible via "ProfileBean" name.

## ReplicateBatchPresentation command

Element attributes		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
batchName	BatchPresentation name that will be created during coping process	yes
setActive	Determines if the new presentation should be active: Possible values: 'all' - yes, for all profiles 'changed' - yes only if BatchPresentation was changed in a profile; 'none' - no, for all profiles	No, default value is none
useTemplates	If template BatchPresentation should be used. If no is set and a copied BatchPresentation is present it is unconditionally substituted	No, by default templates are not used
<b>Embedded elements:</b>		
batchPresentation		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
Name	Element name	yes
actorName	Actor's name to whom batchPresentation belongs.	No
batchId	BatchPresentation type. (See available values in AFProfileStrategy and WFProfileStrategy)	yes
batchName	BatchPresentation name	No

replicateBatchPresentation element must have only one child element batchPresentation with attribute name='source' and any number of child elements batchPresentation with attribute name='template'. BatchPresentation is copied in the following cases:

1. There's no copying BatchPresentation yet.
2. useTemplate attribute is set to 'no'.
3. useTemplate attribute is set to 'yes', but the substituted BatchPresentation is equal to one of the BatchPresentation templates.

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:replicateBatchPresentation>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
```

```

<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:batchName>replicatedNew</web:batchName>
<web:useTemplates>no</web:useTemplates>
<web:activeMode>all</web:activeMode>
<web:batchPresentations>
<web:name>source</web:name>
<web:actorName>attila</web:actorName>
<web:batchId>listTasksForm</web:batchId>
<web:batchName>replicateMe</web:batchName>
</web:batchPresentations>
</web:replicateBatchPresentation>
</soapenv:Body>
</soapenv:Envelope>

```

## Description Substitution веб – сервиса.

Accessible under "SubstitutionBean" name.

### ChangeSubstitutions command

Element attributes:		
<i>Element</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
executors	Executors whose substitution rules are to be changed	1 or more
addedSubstitutions	Substitution rules that to be added	0 or more
deletedSubstitutionsDescr	Substitution rules that to be added	0 or more

There may be more than one executor element, but at least one must be present. Executor can be a user or a group.

Substitutor changes will be applied to all executors set in executor

AddedSubstitutions and DeletedSubstitutionsDescr elements may be omitted or included as many times as it is necessary.

Orgfunction and criteria can be set in the deletedSubstitutionsDescr subelement. If one of this subelements is not set the other is used. If none is set then all substitution rules are to be deleted.

Orgfunction in addedSubstitutions in not a mandatory subelement. The default values of the rest subelements are criteria = always, isFirst = true, isEnabled = true.

Substitutions will be inserted in the order in which they appear. That is the first element will be the first rule, the second element will be the second rule etc if isFirst subelement set to true.

The same goes to isFirst=false but the last element will be the first rule ect.

If orgfunction is not set in addedSubstitutions then a terminator is inserted.

Supported shortcuts: %self\_code% %self\_id% %self\_name%

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web: changeSubstitutions >
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web: executors>employee</web: executors>
      <web: addedSubstitutions >
        <web:orgFunc></web:orgFunc>
        <web: criteria >attila</web: criteria >
        <web: isEnabled >listTasksForm</web: isEnabled >
        <web: isFirst >replicateMe</web: isFirst >
      </web: addedSubstitutions >
    </web: changeSubstitutions >
  </soapenv:Body>
</soapenv:Envelope>
```

## Description of Definition web service.

Accessible under "DefinitionBean" name.

## DeployProcessDefinition command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
file	Path to the file with definition (.par)	yes
type	Process type. If not set "Script" type is used	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:deployProcessDefinition>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:file>../samples/TimerDemo.par</web:file>
      <web:type>Script</web:type>
    </web:deployProcessDefinition>
  </soapenv:Body>
</soapenv:Envelope>

```

## RedeployProcessDefinition command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
definitionId	Identifier of business process definition	yes
file	Path to the file with definition (.par)	yes
type	Process type. If not set "Script" type is used	no

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:redeployProcessDefinition>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:definitionId>18</web:definitionId>
      <web:file>../samples/TimerDemo.par</web:file>
      <web:type>Script</web:type>
    </web:redeployProcessDefinition>
  </soapenv:Body>
</soapenv:Envelope>

```

## UndeployProcessDefinitioncommand



Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Process definition name	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:undeployProcessDefinition>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:name> TimerDemo </web:name>
    </web:undeployProcessDefinition>
  </soapenv:Body>
</soapenv:Envelope>
```

## GetLatestProcessDefinitionStub command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
name	Process definition name	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getLatestProcessDefinitionStub>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:name> TimerDemo </web:name>
    </web:getLatestProcessDefinitionStub>
  </soapenv:Body>
</soapenv:Envelope>
```

## GetLatestProcessDefinitionStubs command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes

Returns last version of all process definitions (up to 10000).

## GetProcessDefinitionStub command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
id	Identifier of business process definition	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getLatestProcessDefinitionStub>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:id> 18 </web:id>
    </web:getLatestProcessDefinitionStub>
  </soapenv:Body>
</soapenv:Envelope>

```

## GetProcessDefinitionStubByProcessId command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
id	Identifier of process	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getLatestProcessDefinitionStub>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web: processInstanceId> 18 </web: processInstanceId>
    </web:getLatestProcessDefinitionStub>
  </soapenv:Body>
</soapenv:Envelope>
```

## Description Execution веб – сервиса.

Accessible under ExecutionBean name.

## GetProcessInstanceTasks command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
processInstanceId	Identifier of process	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getProcessInstanceTasks>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:processInstanceId>2</web:processInstanceId>
    </web:getProcessInstanceTasks>
  </soapenv:Body>
</soapenv:Envelope>
```

## StartProcessInstance command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
processDefinitionName	Process definition name	yes
variables	Variables of the process	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:startProcessInstance>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:processDefinitionName>TimerDemo</web:processDefinitionName>
      <web:variables>
        <!--Optional:-->
        <web:name>test</web:name>
        <web:booleanValue>true</web:booleanValue>
      </web:variables>
    </web:startProcessInstance>
  </soapenv:Body>
</soapenv:Envelope>

```

## CompleteTask command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
taskId	Identifier of task	yes
taskName	Task name	yes
actorId	Identifier of user	yes
variables	Variables of the process	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:completeTask>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:taskId>1</web:taskId>
      <web:taskName>name</web:taskName>
      <web:actorId>1</web:actorId>
      <web:variables>
        <web:name>test</web:name>
        <web:booleanValue>true</web:booleanValue>
      </web:variables>
    </web:completeTask>
  </soapenv:Body>
</soapenv:Envelope>

```

## GetTasks command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getTasks>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
    </web:getTasks>
  </soapenv:Body>
</soapenv:Envelope>
```

## GetVariables command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
taskId	Identifier of task	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web: getVariables>
```



```

<web:actorPrincipal>
<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>
<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:taskId>1</web:taskId>
</web: getVariables>
</soapenv:Body>
</soapenv:Envelope>

```

## GetVariable command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
taskId	Identifier of task	yes
variableName	Variable name	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:web="http://runa.ru/workflow/webservices">
<soapenv:Header/>
<soapenv:Body>
<web: getVariables>
<web:actorPrincipal>
<web:actor>
<web:id>1</web:id>
<web:version>0</web:version>
<web:name>Administrator</web:name>
<web:description>Default System Administrator</web:description>
<web:fullName> </web:fullName>
<web:code>0</web:code>
<web:isActive>true</web:isActive>

```

```

<web:email>?</web:email>
</web:actor>
<web:key>0Administrator</web:key>
</web:actorPrincipal>
<web:taskId>1</web:taskId>
<web:variableName>test</web:variableName>
</web: getVariables>
</soapenv:Body>
</soapenv:Envelope>

```

## GetProcessInstanceStub command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
processInstanceId	Identifier of process	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web: getProcessInstanceStub >
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web: processInstanceId >1</web: processInstanceId >
    </web: getProcessInstanceStub >
  </soapenv:Body>
</soapenv:Envelope>

```

## GetProcessInstanceStubs command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes

Returns all process instances (up to 10000).

## CancelProcessInstance command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
processInstanceId	Identifier of process	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web:cancelProcessInstance>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:processInstanceId>1</web:processInstanceId>
    </web:cancelProcessInstance>
  </soapenv:Body>
</soapenv:Envelope>
```

## GetProcessInstanceDiagram command

Element attributes:		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
instanceId	Identifier of process	yes
taskId	Identifier of task	yes

example:

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web: getProcessInstanceDiagram>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:instanceId>1</web:instanceId>
      <web:taskId>1</web:taskId>
    </web: getProcessInstanceDiagram>
  </soapenv:Body>
</soapenv:Envelope>
```

## AssignTask command

<b>Element attributes:</b>		
<i>Attribute</i>	<i>Description</i>	<i>Mandatory</i>
actorPrincipal	Authorized user	yes
taskId	Identifier of task	yes
taskName	Task name	yes
assigningActorId	Identifier ou user	yes

example:

```

<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:web="http://runa.ru/workflow/webservices">
  <soapenv:Header/>
  <soapenv:Body>
    <web: assignTask>
      <web:actorPrincipal>
        <web:actor>
          <web:id>1</web:id>
          <web:version>0</web:version>
          <web:name>Administrator</web:name>
          <web:description>Default System Administrator</web:description>
          <web:fullName> </web:fullName>
          <web:code>0</web:code>
          <web:isActive>true</web:isActive>
          <web:email>?</web:email>
        </web:actor>
        <web:key>0Administrator</web:key>
      </web:actorPrincipal>
      <web:taskId>1</web:taskId>
      <web:taskName>name</web:taskName>
      <web: assigningActorId>1</web: assigningActorId>
    </web: assignTask>
  </soapenv:Body>
</soapenv:Envelope>

```

# Article Sources and Contributors

**WF-system Web services guide** *Source:* <http://wf.runa.ru/doc/index.php?oldid=827> *Contributors:* Dofs, Natkinnat