Blueworks Live API

Contents

Blueworks Live REST API

Use Blueworks Live REST APIs to access a limited subset of its available functions, such as search, list process details, and launch work instances. Depending on the API, authentication might be required, and there are several ways of completing it. Most Blueworks Live API calls return responses in JSON format; therefore, an example is provided to describe some of the JSON responses.

Overview

The Blueworks Live API provides access to a limited subset of the available Blueworks Live functions, such as search, list process details, and launch work instances. All access requires a valid user name and password and is restricted based on the normal Blueworks Live roles and permissions.

You can experiment with API calls by using the curl program. The following example demonstrates how to make an authentication call using curl:

```
curl --basic --user {username}:{password} https://www.blueworkslive.com/api/Auth
```

Authentication

Authentication is required for Blueworks Live API calls with URLs that start with /scr.

A user can complete authentication in one of the following ways:
* Using basic authentication in combination with the `account` or the `accountId` parameter
* Providing the `JSESSIONID` cookie that is returned as a result of the previous authenticated call

It is not required to call the `/api/Auth` before making other calls: The user name, password, and account, or a valid `JSESSIONID` cookie can be provided in all calls.

However, consider starting with the `/api/Auth` for the following reasons:
* If the authentication fails, a response from the `/api/Auth` call provides additional information. This information is useful in the following cases:
  – A user is locked out
  – A disallowed account or IP is used to access the system
* If the user has more than one account, the response from authentication call lists all accounts associated with the user.

Responses

> **401 UNAUTHORIZED**
>> The user is unauthorized at the time to complete the action, but user authorization might be possible. Use the `/api/Auth` method to determine the cause. One of the possible causes is that the user's password expired.

> **403 FORBIDDEN**
>> The user is authenticated, but the action is disallowed for the user. One of the possible causes is that the user without an administrator license tried to call the `/scr/api/UserList`.

Example

Use the following Java template as a basis for your own projects. The example uses the JSON4J library from the Apache Wink project to decode the JSON responses sent back by the API.

1. Download the following Java files:
   - The following example programmatically executes the `/scr/api/UserList` and parses the results. `RestApiClientTemplate.java`
   - The following example uses the `/api/Auth` to validate user credentials and list accounts that the user name belongs to. `RestApiClientAuthTemplate.java`
   - The following example calls the `/scr/api/Search`, which is based on POST method, and uses the `/api/Auth` for redirection. APIs that use POST method can encounter issues with redirection if the user's account is not hosted on the main server. The example shows how to use the `/api/Auth` to programmatically find out the address where the user's account is hosted and then to issue the API directly to that address. `RestApiClientPostTemplate.java`

2. Download the following Java archive (JAR) file:

   `json4j`

3. Compile the following example:
   - For Windows operating systems: `javac -cp .;wink-json4j-1.3.0.jar RestApiClientTemplate.java`
   - For Linux operating systems: `javac -cp ./wink-json4j-1.3.0.jar RestApiClientTemplate.java`
   - For Windows operating systems: `javac -cp .;wink-json4j-1.3.0.jar RestApiClientAuthTemplate.java`
   - For Linux operating systems: `javac -cp ./wink-json4j-1.3.0.jar RestApiClientAuthTemplate.java`
   - For Windows operating systems: `javac -cp .;wink-json4j-1.3.0.jar RestApiClientPostTemplate.java`
   - For Linux operating systems: `javac -cp ./wink-json4j-1.3.0.jar RestApiClientPostTemplate.java`

4. Run the example and change credentials to valid values.
   - For Windows operating systems: `java -cp .;wink-json4j-1.3.0.jar RestApiClientTemplate`
   - For Linux operating systems: `java -cp .:./wink-json4j-1.3.0.jar RestApiClientTemplate`
   - For Windows operating systems: `java -cp .;wink-json4j-1.3.0.jar RestApiClientAuthTemplate`
   - For Linux operating systems: `java -cp .:./wink-json4j-1.3.0.jar RestApiClientAuthTemplate`
   - For Windows operating systems: `java -cp .;wink-json4j-1.3.0.jar RestApiClientPostTemplate`
   - For Linux operating systems: `java -cp .:./wink-json4j-1.3.0.jar RestApiClientPostTemplate`

For more information, contact Blueworks Live Support.

Account activity

Use this resource to retrieve information about activities in the account. The user's permissions in the account define restrictions on returned information.

Method summary

HTTP Method	Path	Description
GET	/scr/api/activity	Retrieves information about activities in the account.

GET /scr/api/activity

Description

Use this method to retrieve information about activities in the account. Specify the types of activities to retrieve and the time period in which they occurred. The user's permissions in the account define restrictions on returned information.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an account administrator.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
endDate	Query	The end date of the account activities to return. The date must be in ISO 8601 format (example: 2013-04-01T08:00:00.000-06:00). The **endDate** value must be after the **startDate** value, but by not more than 31 days.	Yes	String
startDate	Query	The start date of the account activities to return. The date must be in ISO 8601 format (example: 2013-04-01T08:00:00.000-06:00).	Yes	String

Name	Location	Description	Required	Type
type	Query	Specify one or more types of account activity to return. Any combination of types can be specified as comma-separated values. Valid values are `LOGINS`, `PROCESSES_VIEWED`, `POLICIES_VIEWED`, `DECISIONS_VIEWED`, `SPACES_VIEWED`, `ITEMS_VIEWED`, `PROCESS_COMMENTS`, `DECISION_COMMENTS`, `COMMENTS`, `USERS_JOINED`, `PROCESSES_CHANGED`, `POLICIES_CHANGED`, `DECISIONS_CHANGED`, `SPACES_CHANGED`, `ACCOUNT_CHANGED`, `ITEMS_CHANGED`, `PROCESS_SNAPSHOTS`, `POLICY_SNAPSHOTS`, `DECISION_SNAPSHOTS`, `SNAPSHOTS`. Some types are logical groupings of other types so they can return collections of similar data: • `ITEMS_VIEWED` includes `PROCESSES_VIEWED`, `POLICIES_VIEWED`, `DECISIONS_VIEWED`, `SPACES_VIEWED` • `COMMENTS` includes `PROCESS_COMMENTS`, `DECISION_COMMENTS` • `ITEMS_CHANGED` includes `PROCESSES_CHANGED`, `POLICIES_CHANGED`, `DECISIONS_CHANGED`, `SPACES_CHANGED` • `SNAPSHOTS` includes `PROCESS_SNAPSHOTS`, `POLICY_SNAPSHOTS`, `DECISION_SNAPSHOTS`	Yes	String

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

The results are returned as JSON data contains an array that is mapped by the key 'records'. Each item in the array corresponds to an account activity that meets the criteria for the requested types and dates.

Example input

```
https://www.blueworkslive.com/scr/api/activity?startDate=2014-04-01T08:00:00.000-0
```

Example output

```
{
  "records": [
    {
      "message": "user user_email_1@website.com opened a login session at 2014-04-
      "time": "2014-04-01T16:58:39.184-05:00",
      "timeStamp": "2014-04-01T16:58:39.184-05:00",
      "endTime": "2014-04-01T17:40:47.134-05:00",
      "type": "SESSION_TIMEOUT",
      "user": "user_email_1@website.com"
    },
```

```
{
  "message": "user user_email_2@website.com opened a login session at 2014-04-02
  "time": "2014-04-02T13:28:37.156-05:00",
  "timeStamp": "2014-04-02T13:28:37.156-05:00",
  "endTime": "2014-04-02T13:28:51.567-05:00",
  "type": "USER",
  "user": "user_email_2@website.com"
},
{
  "message": "user user_email_3@website.com opened a login session at 2014-04-02
  "time": "2014-04-02T14:57:40.374-05:00",
  "timeStamp": "2014-04-02T14:57:40.374-05:00",
  "endTime": "2014-04-02T16:01:40.676-05:00",
  "type": "USER",
  "user": "user_email_3@website.com"
}
  ]
}
```

Response properties

activityName
The name of activity item to which the record applied when the action occurred.

activityType
The major type of activity to which the record applied when the action occurred.

copiedFromProcessName
The link to process that it was copied from.

invitedByUserName
The full name of the user who sent the invite.

invitedUserName
The full name of the invited user.

message
A brief description of an activity that took place.

newGoalName
The new name of the project goal.

newLaneName
The new name of the swim lane.

newParentActivityName
The name of the new parent of an activity item.

newParentActivityType
The major type of the new parent of an activity item.

oldActivityType
The major type of an activity item.

oldGoalName
The old name of the project goal.

oldLaneName
The old name of the swim lane.

oldParentActivityName
The name of the old parent of an activity item.

oldParentActivityType
The major type of the old parent of an activity item.

parentActivityName
The name of the parent of an activity item.

parentGoalName
The name of the parent of a project goal.

subType
The sub-type of the record. The value might be `null`.

timeStamp
The date and time when operation took place.

type The type of change.

user The user's full name.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.
404	Activities with the specified parameters were not found.
405	The specified method is not allowed. Select GET method for this request.

Artifact details

Query artifact and activity stream details, update favorites, and post comments using the Blueworks Live artifact details APIs.

Library artifact

Use this resource to retrieve information about library artifacts (space, blueprint, process app, decision, and policy) in CSV format. A user can specify which fields to return and optionally filter on which artifacts are returned.

Method summary

HTTP Method	Path	Description
GET	/scr/api/LibraryArtifact	Retrieves information about library artifacts (space, blueprint, process app, decision, and policy) in CSV format.

GET /scr/api/LibraryArtifact

Description
Use this method to retrieve information about library artifacts (space, blueprint, process app, decision, and policy) in CSV format. A user can specify which fields to return and optionally filter on which artifacts are returned.

Resource information

Requirements	Description
Response format	CSV
Requires authentication	Yes. The user must be an account administrator.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
returnFields	Query	A list of fields to return. Any combination of fields can be specified as comma-separated values. Valid values are * (return all fields), NAME, VERSION_NAME, ID, TYPE, HIERARCHY, SPACE, TAGS, IS_ACTIVE, IS_PUBLISHED, PUBLISHED_DATE, PUBLISHED_NAME.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
activeState	Query	Filter on the state of the artifact. Valid values are ACTIVE, ARCHIVED.	No	String
publishedState	Query	Filter on the published state of the artifact. Valid values are PUBLISHED, UNPUBLISHED.	No	String
spaceId	Query	Filter on the space ID. All child artifacts of the space are returned. (The space itself is also returned.)	No	String
spaceName	Query	Filter on the space name. All child artifacts of the space are returned. (The space itself is also returned.)	No	String
topSpaceOnly	Query	Filter on whether to return only artifacts in top-level spaces. Valid values are true and false.	No	String

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Activity documentation

Use this resource to retrieve documentation for an activity.

Method summary

HTTP Method	Path	Description
GET	/scr/api/ ActivityDocumentation	Retrieves documentation for an activity.

GET /scr/api/ActivityDocumentation

Description

Use this method to retrieve documentation for an activity.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must have at least participant access to the process that contains the activity for which the documentation is retrieved.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
`activityId`	Query	The ID of the activity for which the documentation is retrieved.	Yes	String
`processId`	Query	The ID of the process that contains the activity.	Yes	String

Name	Location	Description	Required	Type
type	Query	Filter on the type of the artifact. Valid values are SPACE, BLUEPRINT, PROCESS_APP, DECISION, POLICY.	No	String

Response

The results are returned as CSV data.

Example input

The following example retrieves all active library artifacts of SPACE type.

```
https://www.blueworkslive.com/scr/api/LibraryArtifact?returnFields=*&type=SPACE&acti
```

Example output

```
"Name","Version Name","ID","Type","Hierarchy","Space Name","Tags","Is Active","Is Pul
"child2","child2","25008a","Space","sp1 > child1","child1","","true","false","",""
"child1","child1","250061","Space","sp1","sp1","","true","false","",""
"Process Examples","Process Examples","90001","Space","","","","true","false","",""
"sp1","sp1","e0f38","Space","","","","true","false","",""
```

Response properties

Name The name of the artifact.

Version Name
> The current version name of the artifact.

ID The ID of the artifact.

Type The type of artifact.

Hierarchy
> The space hierarchy of the artifact. This value is blank if the artifact is a root space.

Space Name
> The name of the space that contains the artifact. This value is blank if the artifact is a root space.

Tags The tags of the artifact.

Is Active
> The active state of the artifact. The values is false if the artifact is archived.

Is Published
> The published state of the artifact.

Published Date
> The published date of the artifact. This value is blank if the current version of the artifact is published and not a named snapshot.

Published Name
> The published name of the artifact. This value is blank if the current version of the artifact is published and not a named snapshot.

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/ActivityDocumentation?processId=80012&activity
```

Example output

```
{
    "activityId":"5f500e10928f0",
    "description":"<b><i><u>activity documentation api test<\/u><\/i><\/b>"
}
```

Response properties

activityId
> The ID of the requested activity.

description
> The activity documentation that contains HTML markup entered by the user. This property is not present if there is no documentation.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.
403	The user does not have access to the process. The process requires at least participant access.

Followed processes

Use this resource to determine the processes that the user is following.

Method summary

HTTP Method	Path	Description
GET	/scr/api/FollowedProcesses	Retrieves a list of processes that the user is following.

GET /scr/api/FollowedProcesses

Description

Use this method to determine the processes that the user is following.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/FollowedProcesses
```

Example output

```
{
  "followedProcesses": [
    {
      "processId": "1eeabc7ee8",
      "name": "process name 1",
      "modifiedUserName": "user name",
      "modifiedDate": "8/28/15"
    },
    {
      "processId": "1deabcc2aa",
      "name": "process name 2",
      "modifiedUserName": "user name",
      "modifiedDate": "9/10/15"
    },
    {
```

```
            "processId": "deabc338f",
            "name": "process name 3",
            "modifiedUserName": "user name",
            "modifiedDate": "7/27/15"
        }
    ]
}
```

Response properties

followedProcesses
> The key for the array that contains information about the processes that the user follows.

modifiedDate
> The date when the process was last modified.

modifiedUserName
> The name of the user who last modified the process.

name The name of the process.

processId
> The ID of the process.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.
404	The specified processes were not found.

News feed

Use this resource to retrieve an activity stream of a user.

Method summary

HTTP Method	Path	Description
GET	/scr/api/NewsFeed	Retrieves an activity stream of a user.

GET /scr/api/NewsFeed

Description
> Use this method to retrieve an activity stream of a user.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/NewsFeed
```

Example output

```
{
    "messages":[
        {
            "processId":"5f500e12a00ea",
            "name":"user name",
            "description":"received <i>1 task<\/i> in <i>ProcessApp with multiple tasks
            "group":"Today"
        },
        {
            "processId":"5f500e12a00ea",
            "name":"user name",
            "description":"launched <i>ProcessApp with multiple tasks A - New for test.
            "group":"Today"
        },
        {
```

```
                              "processId":"5f500e12a000f",
                              "name":"user name",
                              "description":"received <i>1 task<\/i> in <i>ProcessApp with 1 task A - N
                              "group":"Today"
                          },
                          {
                              "processId":"5f500e12a000f",
                              "name":"user name",
                              "description":"launched <i>ProcessApp with 1 task A - New for test.<\/i>
                              "group":"Today"
                          }
                      ]
                  }
```

Response properties

messages
> The array of messages in reverse chronological order.

name The full name of the user who is addressed in the message. If the user makes this API call, then the word You appears instead.

description
> The text of the message that includes <i> and <span> markup. The span has a class of date.

group States when the event occurred and contains one of the following possible values:
> - Today
> - Yesterday
> - The actual date, if the event occurred from the day before yesterday to 7 days ago
> - Older, if the event occurred more than 7 days ago

processId
> Optional parameter. If the message is associated with a process, this property value is the ID of the process.

comment
> Optional parameter. If the message is associated with a comment, this property value is the text of the comment.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.

Post comment

Use this resource to add a comment to an activity or to respond to an existing comment.

Method summary

HTTP Method	Path	Description
POST	/scr/api/PostComment	Adds a comment to an activity or responds to an existing comment.

POST /scr/api/PostComment

Description

Use this method to add a comment to an activity or to respond to an existing comment.

Use the `WorkComment` call to add comments to work instances.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must have at least participant access to the process that contains the activity to which a comment is added.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
activityId	Query	The ID of the activity to which a comment is attached.	Yes	String
comment	Query	The text of the comment.	Yes	String
parentCommentId	Query	The ID of the comment that the user replies to with a new comment.	Yes, if the user is posting a comment.	String
processId	Query	The ID of the process that contains the activity to which a comment is attached.	Yes	String

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

Post a comment *asdf* to the activity with *800c4* ID that is located in the process with the *80012* ID: `/scr/api/`
`PostComment?processId=80012&activityId=800c4&comment=asdf`

```
curl -i --location-trusted --user admin:password -H "Version:1.0" -H "Content-Type
"https://www.blueworkslive.com/scr/api/PostComment?processId=80012&activityId=800c
```

Example output

```
{
        "commentId":"5f504e13c80d6"
}
```

Response properties

> **commentId**
>> The ID of the newly created comment.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.
405	The specified method is not allowed. Select POST method for this request.

Process data

Use this resource to retrieve process details, such as milestones, activities, and activity properties.

Method summary

HTTP Method	Path	Description
GET	/scr/api/ProcessData	Retrieves details about a process.

GET /scr/api/ProcessData

Description

Use this method to retrieve process details, such as milestones, activities, and activity properties.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must have at least participant access to the process.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
processId	Query	The ID of the process to retrieve.	Yes	String

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

The data is returned as a map of items. Each item in an account has a unique ID; the ID values link all items in the account together.

Example input

```
https://www.blueworkslive.com/scr/api/ProcessData?processId=100005
```

Example output

```
{
  "processId":"100005",
  "accessControl":{
    "isAuthor":"true"
  },
  "items":{
    "5f500e10929de":{
      "positionFlowsRight":"false",
      "participants":"3d9200f4",
      "majorType":"activity",
      "uiTaskIncludeAttach":"false",
      "uiTaskIncludeDoc":"false",
```

```
        "name":"Generate Conterct Document",
        "itemType":"ACTIVITY"
},
"5f500e110054b":{
        "name":"myProb",
        "itemType":"PROBLEM"
},
"5f500e1092907":{
        "businessOwners":[
                {
                        "val":"5f500e1100507"
                }
        ],
        "activityDecisionDiagram":"5f500e109293d",
        "experts":[
                {
                        "val":"5f500e110050f"
                }
        ],
        "uiTaskIncludeDoc":"false",
        "name":"Prequalify Vehicle",
        "cost":"10.00",
        "risk":"1",
        "outputs":[
                {
                        "val":"3d92069b"
                }
        ],
        "majorType":"activity",
        "uiTaskIncludeAttach":"false",
        "customers":[
                {
                        "val":"5f500e107002f"
                }
        ],
        "ruleType":"ruleDecision",
        "participants":"3d9200ee",
        "positionFlowsRight":"false",
        "waitTime":".64532m",
        "fileItems":[
                {
                        "val":"5f500e11000fd",
                        "fileItemId":"5f500e11000fb"
                }
        ],
        "systems":[
                {
                        "val":"5f500e1100517"
                }
        ],
        "suppliers":[
                {
                        "val":"5f500e1070032"
                }
        ],
        "inputs":[
                {
                        "val":"3d920697"
                }
        ],
        "itemType":"ACTIVITY",
        "valueAdd":"3",
        "comments":[
                {
                        "val":"5f500e1100574"
                }
        ],
```

```
                                "problems":[
                                    {
                                        "val":"5f500e110054b",
                                        "frequency":"9",
                                        "severity":"9"
                                    },
                                    {
                                        "val":"5f500e110055e"
                                    },
                                    {
                                        "val":"5f500e1100563",
                                        "frequency":"3",
                                        "severity":"3"
                                    }
                                ],
                                "workTime":"20h",
                                "activityPolicies":[
                                    {
                                        "val":"3d920015"
                                    }
                                ]
                            },
                            "3d920015":{
                                "policyAbrvDesc":"This policy describes the special considerations for ha
                                "name":"Legal Handling of Military Family Moves out of State",
                                "itemType":"POLICY",
                                "itemParent":"3d920002"
                            },
                            "5f500e1092598":{
                                "name":"Validate Vehicle",
                                "itemType":"DECISION_DIAGRAM",
                                "itemParent":"5f500e10900d3"
                            },
                            "5f500e10928e2":{
                                "playbackPath":[
                                    {
                                        "val":"5f500e10928fd"
                                    }
                                ],
                                "playbackParent":"5f500e10928e0",
                                "name":"Test Playback",
                                "itemType":"PLAYBACK"
                            },
                            "5f500e1081072":{
                                "name":"Alternate Start Event",
                                "itemType":"PROJECT"
                            },
                            "5f500e1100576":{
                                "isRead":"true",
                                "createdByUserId":"7f0002",
                                "createdDate":"1427234810125",
                                "createdByName":"admin",
                                "name":"reply",
                                "itemType":"COMMENT",
                                "ownerId":"5f500e10900d2"
                            },
                            "3d92069b":{
                                "name":"Activity Output 1",
                                "itemType":"INPUTOUTPUT"
                            },
                            "5f500e11000fd":{
                                "itemType":"FILEITEM"
                            },
                            "5f500e10928f4":{
                                "positionFlowsRight":"false",
                                "majorType":"milestone",
                                "uiTaskIncludeAttach":"false",
```

```
                                    "uiTaskIncludeDoc":"false",
                                    "name":"Make final decisions",
                                    "itemType":"ACTIVITY",
                                    "activities":[
                                        {
                                            "val":"5f500e109297f"
                                        },
                                        {
                                            "val":"5f500e1092981"
                                        },
                                        {
                                            "val":"5f500e1092983"
                                        },
                                        {
                                            "val":"5f500e1092985"
                                        }
                                    ]
                                },
                                "5f500e107002f":{
                                    "name":"Activity Customer Search API",
                                    "itemType":"SUPPLIERCUSTOMER"
                                },
                                "5f500e109292e":{
                                    "itemType":"FILEITEM"
                                },
                                "files":{
                                "5f500e11000fb":{
                                    "size":"3.0",
                                    "deleted":false,
                                    "type":"9",
                                    "name":"bar.txt"
                                }
                            }
                        }
                    }
```

Response properties

accessControl
> The access properties that the requesting user has for the process.

activities
> The array that contains IDs of sub activities of an activity. For the root activity, sub activities are milestones.

businessOwners
> The array that contains IDs of any business owners who are linked to the activity. Each entry in the array is an object with a single `val` property that holds ID of the business owner `ACTOR` item.

childComments
> The array that contains IDs of replies to the comment. Each entry in the array is an object with a single `val` property that holds the ID of the `COMMENT` item.

comments
> The array that contains IDs of any comments that are linked to the activity. Each entry in the array is an object with a single `val` property that holds the ID of the `COMMENT` item.

cost   A number that represents the cost.

createdByName
> The name of the user who created the comment.

createdByUserId
> The ID of the user who created the comment.

createdDate
> The millisecond value that represents when the comment was created.

customers
> The array that contains the IDs of customers associated with the activity. Each entry in the array is an object with a single `val` property that holds the ID of the `SUPPLIERCUSTOMER` item.

cycleTime
> The cycle time value that is expressed with a number followed by a character unit marker.

Character unit marker	Description
m	minutes
h	hours
d	days
w	weeks
M	months

deleted
> The indication of whether the data for the file was deleted. The possible values are `true` or `false`.

diagram
> The ID of the diagram, if the process has one. The `diagram` item itself is not present in the data.

experts
> The array with the IDs of any experts that are linked to the activity. Each entry in the array is an object with a single `val` property that holds the ID of the expert `ACTOR` item.

files The `files` map. The `key` is the ID of the file, and the `value` is the object with the file's properties.

fileItemId
> The ID of the file in the `files` map. This property can be used with the `FileDownload` to download contents of the file.

fileItems
> The array with the IDs of file items that are linked to the activity. Each entry is an object with a `val` property that holds the ID of the `fileItem` item and the `fileItemId` property that holds the ID of an entry in the `files` map. The `files` map entry contains information about the file.

frequency
> The frequency with which the problem occurs. The values are classified in the following way:
> - Less than 3 is Low
> - Less than 9 and more than 3 is Medium
> - 9 or more is High

IBM Blueworks Live sets the values in the following way:
- 1 for Low
- 3 for Medium
- 9 for High

hideOnMap
Indicates whether the user chose to hide the item from the map view in Blueworks Live.

inputs The array with the IDs of any inputs linked to the activity. Each entry in the array is an object with a single `val` property that holds the ID of the `INPUTOUTPUT` item.

isAuthor
This boolean access control property indicates whether the user who makes the request has author access to the process.

isRead
The value of `true` is set if the comment was read.

items The `items` map that has one entry for each item. The `key` is the item ID, and the `value` is the object with the item's properties.

itemType
The type of item.

Item type value	Description	Item type properties
ACTIVITY	Any item, including milestones, is an activity on the map or the diagram.	The `majorType` property distinguishes between different `subtypes` of an activity.
ACTOR	A person or a role. Actors are used for participants, business owners, and experts.	`itemType` and `Name`
COMMENT	A comment	The `name` property of the comment is the comment text.
FILEITEM	A file	No properties. Information about the file is in the `files` map. Look at the documentation for the `fileItems` property.
INPUTOUTPUT	Something that can be used as an input or an output.	`itemType` and `Name`
PROBLEM	A problem	`itemType` and `Name`
PROCESS	A process	`itemType` and `Name`
PROJECT	A project	`itemType` and `Name`
SYSTEM	A system	`itemType` and `Name`
SUPPLIERCUSTOMER	An item that can be used as a supplier or a customer	`itemType` and `Name`

majorType
The type of activity that the item represents. This property is present only for `ACTIVITY` items.

Activity type	Description
`activity`	Normal activity
`endEvent`	End event on the diagram
`gateway`	Gateway on the diagram
`intermediateEvent`	Intermediate event on the diagram
`linkedprocess`	Link to another process
`milestone`	Milestone
`startEvent`	Start event on the diagram
`subprocess`	Subprocess

name The name of the item.
- For a `comment` item, the name is the actual comment text.
- For a `process` item, the name of the process is in the name of the root activity in the `rootActivityId` property.

outputs
The array with the IDs of any outputs linked to the activity. Each entry in the array is an object with a single `val` property that holds the ID of the `INPUTOUTPUT` item.

participants
The value is the ID of a single participant `ACTOR` item associated with an activity.

problems
The array of links to `PROBLEM` items. Each entry in the array is an object with a `val` property, which is the ID of the linked `problem` item and optional `frequency` and `severity` properties.

process
The property of a comment with the ID of the process that the comment is associated with.

processId
The ID of the process. The `items` map contains an entry for this ID.

project
The property of the `PROCESS` item with the ID of the `PROJECT` item that represents the project that the process is in.

risk The amount of risk the activity introduces to the process. The risk is represented with one of the following numeric values.

Risk value	Description
1	Controlled
2	Low
3	Medium

Risk value	Description
4	High

rootActivityID

The ID of the top-level activity in the process. This property is not displayed on the map or diagram. Its children are the milestones.

severity

The severity of a problem. The values are classified in the following way:

- Less than 3 is Low
- More than 3 and less than 9 is Medium
- 9 or more is High

Blueworks Live sets the values in the following way:

- 1 for Low
- 3 for Medium
- 9 for High

size The property of an item in the files array holding the size of the file in bytes.

suppliers

The array that contains the IDs of any suppliers associated with the activity. Each entry in the array is an object with a single `val` property that holds the ID of the `SUPPLIERCUSTOMER` item.

systems

The array of links to `SYSTEM` items. Each entry in the array is an object with a single `val` property that holds the ID of the `SYSTEM` item.

type The property of the item in the files array that defines the type of the file.

Type value	Description
1	Generic document. Used if the file doesn't fall under one of the other values.
2	Microsoft Excel file
3	Image
4	PDF document
5	Microsoft PowerPoint document
6	Microsoft Project document
7	Microsoft Visio document
8	Microsoft Word document

val The ID of another item.

valueAdd

One of the following numeric values represents whether the activity adds value to the process.

Numeric value	Description
1	Yes
2	No
3	Required

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.
403	The user does not have access to the process.

Update favorites

Use this resource to update a list of favorite process applications for the user.

Method summary

HTTP Method	Path	Description
POST	/scr/api/UpdateFavorites	Updates a list of favorite process applications for the user.

POST /scr/api/UpdateFavorites

Description

Use this method to update a list of favorite process applications for the user. The list is saved in order provided by the caller. The list can be retrieved as part of the AppList call.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
favorites	Query	The list of item IDs for the user's favorite process applications. Only process application IDs are allowed for version 20110917.	Yes	JSON array of numbers

Name	Location	Description	Required	Type
version	Query	The version of the requested API. The only allowed value is 20110917.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

The `favorites` parameter

>**Example**
>
>```
>{
>    "favorites":[
>        {
>            "id":"2c0007"
>        },
>        {
>            "id":"905be"
>        }
>    ]
>}
>```

favorites
The list of favorite IDs.

favorites.id
The ID of an item that is a part of the favorites list.

Response

Example input
Update a list of favorite process applications with *80038* item ID:
/scr/api/UpdateFavorites?version=20110917&favorites={\
"favorites\":[{\"id\":\"5f500e1092496\"},{\"id\":\
"5f500e10924ec\"}]}

```
curl -i --location-trusted -g --user admin:password -H "Version:1.0" -H "Content-T
"https://www.blueworkslive.com/scr/api/UpdateFavorites?version=20110917&favorites=
```

Example output
```
{
    "favorites":[
        {
            "id":"5f500e1092496"
        },
        {
            "id":"5f500e10924ec"
        }
    ],
    "version":"20110917"
}
```

Response properties

version
The version of the API used to create a response.

favorites
The list of favorite IDs.

favorites.id
The ID of an item that is a part of the favorites list.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Application detail

Use this resource to retrieve information about an application definition.

Method summary

HTTP Method	Path	Description
GET	/scr/api/AppDetail	Retrieves information about application definition.

GET /scr/api/AppDetail

Description

Use this method to retrieve information about an application definition. This resource can form the basis of a call to /scr/api/LaunchInstance to launch a new instance of the application.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
`appId`	Query	The ID of the requested process application.	Yes	String
`version`	Query	The version of the requested API. The only allowed value is `20110917`.	Yes	String
`account`	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/AppDetail?version=20110917&appId=44eaaadd6d
```

Example output

```
{
  "app": {
    "id": "44eaaadd6d",
    "mailOnCompletion": false,
    "instructions": null,
    "subjectTitle": null,
    "processId": "44eaaadd69",
    "name": "ProcApp",
    "attachments": [],
    "tasks": [
      {
        "id": "44eaaade02",
        "approvalStep": false,
        "name": "",
        "assignedTo": null
      }
    ],
    "type": "workflow",
    "detailsTitle": null
  },
  "users": [],
  "version": "20110917"
}
```

Response properties

> **version**
>> The version of the API used to create a response.
>
> **app.name**
>> The name of the application.
>
> **app.id** The ID of the application.
>
> **app.processId**
>> The ID of the process that the application is related to. Use

this process ID together with the attachment ID to retrieve files attached to this application.

app.instructions
>The instructions on creating an instance of an application. Text length limit is 200 characters. The default value is `null`.

app.subjectTitle
>The title for edit area in which the user entered a subject for the instance. The default value is `null`.

app.detailsTitle
>The title for edit area where the user entered details for the instance. The default value is `null`.

app.type
>The application type has two possible values, `workflow` or `checklist`.

app.mailOnCompletion
>If `true`, then upon completion, an email is sent to the user who created the instance.

app.tasks
>The array of tasks associated with the instance.

app.tasks.name
>The name of the task.

app.tasks.id
>The ID of the task.

app.tasks.assignedTo
>The ID of the user whom the task is assigned to. An object with this ID appears in the user's array.

app.tasks.approvalStep
>Indicates whether this task is an approval step. This is only valid for instances of work flow type.

app.attachments
>The array of attachments associated with this instance or the application that it was started from.

app.attachments.name
>The name of the attachment.

app.attachments.id
>The ID of the attachment.

app.attachments.uploadedBy
>The ID of the user who uploaded the attachment. An object with this ID appears in the user's array.

app.attachments.uploadedOn
>The date and time when the attachment was uploaded. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

app.attachments.size
>The size of the attachment in bytes.

app.attachments.deletedOn

The date and time when the attachment was deleted. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC). This field is present only if the file was deleted from the account through the admin pages. In this case, the user is unable to download the file using `FileDownload`.

app.attachments.deletedBy

The name of the user who deleted the attachment. This field is present only if the file is deleted from the account through the admin pages.

users   The array of the users whose IDs are referenced from other locations in the JSON.

users.name

The full name of the user.

users.id

The ID of the user.

users.avatarId

The ID of the avatar set by the user. This is only present if the user set an avatar.

work   The work object.

work.name

The name displayed for the work instance.

work.id

The ID of the work instance.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Search

Use this resource to search for text across IBM Blueworks Live artifacts.

Method summary

HTTP Method	Path	Description
POST	/scr/api/Search	Searches for text across IBM Blueworks Live artifacts.

POST /scr/api/Search

Description

Use this method to search for text across IBM Blueworks Live artifacts.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must have at least participant access to the process.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
`returnFields`	Query	The list of fields to return. Valid values are `output`, `business_owner`, `problem`, `input`, `customer`, `supplier`, `system`, `expert`, `participant`, `external_participant`, `tag`, `cost`, `work_time`, `wait_time`, `decision_expert`, `kpis`, `business_motivation`, `decision_how_often_made`, `decision_volume`, `decision_how_often_updated`, `decision_frequency`, `decision_how_much_time`, `decision_latency`, `sources`, `decision_input`, `decision_output`, `decision_business_owner`, and the exact name of any custom property. Pass in the value of * to return all properties; the case does not matter in this instance. **Note:** The `activity_name`, `milestone_name`, `process_name`, `space_name`, `decision_name`, `sub_process_name`, `decision_diagram_name`, `decision_item_name`, and `event_name` are always returned if they match.	Yes	Comma-separated list of text strings

Name	Location	Description	Required	Type
searchFieldName	Query	The single field to search on. Valid values are `activity_name`, `milestone_name`, `process_name`, `space_name`, `output`, `business_owner`, `problem`, `input`, `customer`, `supplier`, `system`, `expert`, `participant`, `external_participant`, `tag`, `cost`, `work_time`, `wait_time`, `decision_expert`, `kpis`, `business_motivation`, `decision_how_often_made`, `decision_volume`, `decision_how_often_updated`, `decision_frequency`, `decision_how_much_time`, `decision_latency`, `sources`, `decision_input`, `decision_output`, `decision_business_owner`, `decision_name`, `event_name`, `attached_event_name`, `sub_process_name`, and the exact name of any custom property. This field is not case sensitive. **Note:** Searching for `activity_name` searches for `activity_name`, `sub_process_name`, `decision_name`, and `event_name`.	Yes	String

Name	Location	Description	Required	Type
searchValue	Query	The value to search for. When the **searchFieldName** is decision_volume, decision_frequency, or decision_latency, search for one of the values: • Low • Medium • High The case does not matter.	Yes	String
version	Query	The version of the requested API. Valid values are 20111217, 20120130, or 20150930.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
`accountId`	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
`includeArchived`	Query	The default value is `false`. Set the value to `true` to include archived data in the search.	No	Boolean
`topSpacesOnly`	Query	Whether only artifacts belonging to top-level spaces should be included in the output. If the value is `true`, only artifacts belonging to top-level spaces are shown; if it is `false`, all artifacts are shown. This filter is only useful when the `searchFieldName` is either `space_name` or `tag`; searching by any other field excludes the child spaces by default.	No	Boolean

Response

The data is returned as a list of spaces:

- Each `space` contains lists of `processes`.
- Each `process` a list of `milestones`.
- Each `milestone` a list of `activities`.

Every data member, except spaces, can contain the specified properties in the **returnFields** parameter.

Example input

/scr/api/Search?version=20111217&searchValue=Initial%20Hire&searchFieldName=input&re

/scr/api/Search?version=20111217&searchValue=Hiring%20-%20Onboarding&searchFieldName=

The following example shows an account with a custom text field labeled **Countries**:

/scr/api/Search?version=20111217&searchValue=canada&searchFieldName=countries&return

Example output

```
{
 "spaces":[
  {
   "name":"Process Examples",
   "id":"90001",
   "processes":[
    {
     "name":"Hiring - Onboarding",
     "id":"9000b",
     "milestones":[
      {
       "name":"Select Candidate",
       "id":"90062",
       "outputs":[
        {
         "name":"Background Check Forms",
         "id":"9015b"
        }
       ],
       "business owners":[
        {
         "name":"Hiring Manager",
         "id":"90047"
        }
       ],
       "customers":[
        {
         "name":"Candidate",
         "id":"90055"
        }
       ],
       "inputs":[
        {
         "name":"Initial Hire Decision",
         "id":"90463"
        }
       ],
       "suppliers":[
        {
         "name":"Recruiter",
         "id":"9045e"
        },
        {
         "name":"Hiring Manager",
         "id":"90059"
        }
       ]
      }
     ]
    }
   ]
  }
 ]
}
```

Response properties

Properties that have no value are not present.

spaces An array that defines a `space`. Each space can have a list of `processes`.

processes

An array that defines a `process`. Each `process` can have a list of `milestones` and searched for properties.

decisions

An array that defines a `decision`. Each `decision` can have a list of `sub-decisions` and searched for properties.

sub-decisions

An array that defines a `sub-decision`. Each `sub-decision` can have searched for properties.

milestones

An array that defines a `milestone`. Each milestone can have a list of `activities` and searched for properties.

activities

An array that defines an `activity`. Each `activity` can have a list of searched for properties.

businessOwners

An array that contains the IDs of any business owners who are linked to the `activity` or `decision`. Each entry is a `name`, `id` pair.

customers

An array that contains the IDs of any customers who are associated with the `activity`. Each entry is a `name`, `id` pair.

experts

An array that contains the IDs of any experts who are linked to the `activity` or `decision`. Each entry is a `name`, `id` pair.

inputs An array that contains the IDs of any inputs that are linked to the `activity` or `decision`. Each entry is a `name`, `id` pair.

name The name of the item.

id The ID of the item. This ID is not present for the number properties.

outputs

An array that contains the IDs of any outputs that are linked to the `activity` or `decision`. Each entry is a `name`, `id` pair.

participants

An array that contains the IDs of any participants who are linked to the `activity`. Each entry is a `name`, `id` pair.

externalParticipant

An array that contains the ID of an external participant who are linked to a message event `activity`. There is only one entry, and it contains a `name`, `id` pair.

problems

    An array that contains the IDs of any problems that are linked to the `activity`. Each entry is a `name`, `id` pair.

suppliers

    An array that contains the IDs of any suppliers who are associated with the `activity`. Each entry is a `name`, `id` pair.

systems

    An array that contains the IDs of any systems that are associated with the `activity`. Each entry is a `name`, `id` pair.

linkedProcessId

    The process ID of the process that the `activity` links to. This ID is returned in versions 20111217 and 20120130 of the API. If this ID isn't present, then the `activity` does not link to a process. If the activity is linked to more than one process, only the ID of the first process is returned.

linkedProcessIds

    A list of process IDs for the processes that the activity links to. This list is returned in version 20150930 of the API. If this list isn't present, then the activity does not link to any processes.

linkedActivityId

    The activity ID of the other activity that the `activity` is linked to. If this ID isn't present, then the `activity` is not linked to an activity.

kpis   An array that contains the IDs of any key performance indicators that are linked to the `decision`. Each entry is a `name`, `id` pair.

business_motivation

    The value of the business motivation property that is linked to the `decision`. The entry has a `name` tag.

decision_how_often_made

    The value of how often the decision is made that is linked to the `decision`. The entry has a `name` tag.

decision_volume

    The value of the decision volume property that is linked to the `decision`. The entry has a `name` tag.

decision_how_often_updated

    The value of how often the decision is updated that is linked to the `decision`. The entry has a `name` tag.

decision_frequency

    The value of the decision frequency property that is linked to the `decision`. The entry has a `name` tag.

decision_how_much_time

    The value of how much time the decision takes that is linked to the `decision`. The entry has a `name` tag.

decision_latency

    The value of the decision latency property that is linked to the `decision`. The entry has a `name` tag.

sources

 The value of the sources property that is linked to the `decision`. The entry has a `name` tag.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Revision

Use this resource to query or create revisions of a process, policy, or decision.

Method summary

HTTP Method	Path	Description
GET	/scr/api/revision/ [process|policy|decision]/ {artifact_id}	Retrieves a list of revisions for a process, policy, or decision.
POST	/scr/api/revision/ [process|policy|decision]/ {artifact_id}	Creates a revision for the current tip of a process, policy, or decision.

GET /scr/api/revision/[process|policy|decision]/{artifact_id}

Description

 Use this method to retrieve a list of revisions for a process, policy, or decision.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an editor for the process, policy, or decision.
Rate limited	No

Parameters

Name	Location	Description	Required	Type
`artifact_id`	Path	The ID of the artifact for which the revisions are returned.	Yes	String

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

Get all revisions for a process with the ID d001:

/scr/api/revision/process/130024

```
curl -i --user admin:password "https://www.blueworkslive.com/scr/api/revision/proces
```

Example output

Get all revisions for a process with the ID 130024:

```
HTTP/1.1 200 OK
{
  "creationTime": "2013-01-14T07:22:03-0500",
  "id": "130024",
  "lastModified": "2013-01-14T07:51:22-0500",
  "revisions": [
      {
          "id": "130089",
          "label": "Second Snapshot",
          "time": "2013-01-14T07:51:22-0500"
```

```
        },
        {
          "id": "130065",
          "label": "First Snapshot",
          "time": "2013-01-14T07:48:00-0500"
        }
      ],
      "tip": "130089"
    }
```

Response properties

    creationTime
        The time when the process or the policy was created.

    id   The ID of the process, policy, or decision. This ID is the **artifact_id** that is provided in the request URI. The **artifact_id** becomes the ID for each revision.

    label The label that was given to the revision when a snapshot or revision was created.

    lastModified
        The time when the process, policy, or decision was last modified.

    revisions
        The list of revisions of the process, policy, or decision. This list is organized in reverse chronological order.

    time The time when the revision was created.

    tip   The current tip of the process. It determines whether the latest revision is the same as the tip; in the previous example output, they are the same.

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400	The URI syntax is incorrect, or the artifact with the provided ID does not exist.
401	The user is not authorized to make the request.
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.

POST /scr/api/revision/[process|policy|decision]/{artifact_id}

Description
    Use this method to create a new revision for the current tip of a process, policy, or decision.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an editor for the process, policy, or decision.
Rate limited	No

Parameters

Name	Location	Description	Required	Type
artifact_id	Path	The ID of the artifact for which the revisions are returned.	Yes	String
account	Query	The name of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String
accountId	Query	The ID of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String
label	Query	This label for the revision must be URI-encoded.	Yes	String

Response

> **Example input**
> Create a revision for the tip of process *130054* named *Third Snapshot*: /scr/api/revision/process/130024?label=Third %20Snapshot

```
curl -i --user admin:password -X POST
"https://www.blueworkslive.com/scr/api/revision/process/130024?label=Third%20Snaps
```

Example output

Create a revision for the tip of process 130054 named `Third Snapshot`:

```
HTTP/1.1 200 OK
Content-Length: 180
{
    "id": "1300a1",
    "label": "Third Snapshot",
    "time": "2013-01-14T08:08:22-0500"
}
```

Response properties

id   The ID of the created revision.

label   The label of the created revision.

time   The time when the revision was created.

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400	The URI syntax is incorrect, or the artifact with the provided ID does not exist.
401	The user is not authorized to make the request.
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.

Blueprints, policy models, and glossary values

Query blueprints, policy models, and glossary values that meet specified criteria.

Blueprints

Use this resource to retrieve a list of blueprints that meet specified criteria.

Method summary

HTTP Method	Path	Description
GET	/bwl/blueprints	Returns a list of blueprints that meet specified criteria. The criteria are expressed using zero or more filter parameters (`space-id`, `space-name`, `archived-state`) and exactly one search parameter (`name` or `tag`). Only blueprints that you have access to will be returned.

GET /blueprints

Description

Use this method to return a list of blueprints that meet specified criteria.

Parameters

Name	Location	Description	Required	Type
`X-IBM-API-Version`	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is `1.0.0`.	No	String

Name	Location	Description	Required	Type
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
`X-IBM-BWL-Account-Name`	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String
`space-id`	Query	The identifier of the space to search within. If not specified, the search is account wide. If this space contains child spaces, the child spaces will also be searched. If both a space-id and space-name are specified, the space-id takes precedence.	No	String

Name	Location	Description	Required	Type
`space-name`	Query	The name of the space to search within. If not specified, the search is account wide. If both a space-id and space-name are specified, the space-id takes precedence.	No	String
`archived-state`	Query	Filters the results based on the archived state of the artifact. If not specified, 'active' is assumed. The only allowed values are `active` or `archived`.	No	String
`name`	Query	If specified, only blueprints with this exact name are returned. Either a name or a tag must be specified, but not both.	No	String
`tag`	Query	If specified, only blueprints with this exact tag are returned. Either name or tag must be specified, but not both.	No	String

Response

Example input

LIST all blueprints named MyBlueprint within the space named MySpace, in an account that you have access to:

```
https://www.blueworkslive.com/bwl/blueprints?space-name=MySpace&name=MyBlueprint
```

Example output

LIST all blueprints named MyBlueprint within the space named MySpace, in an account that you have access to:

```
HTTP/1.1 200 OK
[
  {
    "id": "250012",
    "name": "MyBlueprint",
    "type": "blueprint",
    "space-ids": [
```

```
          "250006"
        ],
        "space-names": [
          "MySpace"
        ],
        "archived-state": "active",
        "published-state": "unpublished"
      }
    ]
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.
2. Open a web browser, and navigate to http://editor.swagger.io.
3. Import RestApis.yaml using the **File** > **Import File** option.

Blueprints/{blueprint-id}

Use this resource to retrieve information about a specific blueprint.

Method summary

HTTP Method	Path	Description
GET	/bwl/blueprints/{blueprint-id}	Returns information about where a specific blueprint is used. You must have access to this blueprint.

GET /blueprints/{blueprint-id}

Description

Use this method to return the details of a specific blueprint.

Parameters

Name	Location	Description	Required	Type
blueprint-id	Path	The identifier of the blueprint to retrieve.	Yes	String

Name	Location	Description	Required	Type
`X-IBM-API-Version`	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is `1.0.0`.	No	String
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
`X-IBM-BWL-Account-Name`	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Response

Example input

GET the details of a specific blueprint:

```
https://www.blueworkslive.com/bwl/blueprints/250012
```

Example output

GET the details of a specific blueprint:

```
{
  "name": "MyBlueprint",
  "id": "250012",
  "type": "blueprint",
  "tags": [

  ],
  "milestones": [
    {
      "name": "Milestone 1",
      "id": "250016",
      "type": "milestone",
      "activities": [
        {
```

```
                "name": "Activity",
                "id": "25001a",
                "type": "activity",
                "sub-type": "normal"
              }
            ]
          },
          {
            "name": "Milestone 2",
            "id": "250018",
            "type": "milestone"
          }
        ],
        "space-ids": [
          "250006"
        ],
        "space-names": [
          "MySpace"
        ],
        "archived-state": "active",
        "published-state": "unpublished"
      }
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.
403	Access is forbidden because you have insufficient permissions for the specified blueprint.
404	The blueprint-id parameter identifies a blueprint that does not exist in this account.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.
2. Open a web browser, and navigate to http://editor.swagger.io.
3. Import RestApis.yaml using the **File** > **Import File** option.

Blueprints/{blueprint-id}/usage

Use this resource to retrieve information about where a specific blueprint is used.

Method summary

HTTP Method	Path	Description
GET	/bwl/blueprints/{blueprint-id}/usage	Returns information about where a specific blueprint is used. You must have access to this blueprint.

GET /blueprints/{blueprint-id}/usage

Description

Use this method to return the details of where a specific blueprint is used.

Parameters

Name	Location	Description	Required	Type
`blueprint-id`	Path	The identifier of the blueprint to retrieve.	Yes	String
`X-IBM-API-Version`	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is `1.0.0`.	No	String

Name	Location	Description	Required	Type
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
X-IBM-BWL-Account-Name	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Response

Example input

GET the details of where a specific blueprint is used:

```
https://www.blueworkslive.com/bwl/blueprints/250012/usage
```

Example output

GET the details of where a specific blueprint is used:

```
{
  "target-artifact-name": "MyBlueprint",
  "target-artifact-id": "250012",
  "usage-count": 1,
  "used-by-artifacts": [
    {
      "id": "250036",
      "name": "AnotherBlueprint",
      "type": "blueprint",
      "space-ids": [
        "250006"
      ],
      "space-names": [
        "MySpace"
```

```
        ],
        "archived-state": "active",
        "published-state": "unpublished",
        "blueprint-items": [
          {
            "type": "activity",
            "id": "25003e",
            "name": "Linked to MyBlueprint"
          }
        ]
      }
    ]
  }
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.
403	Access is forbidden because you have insufficient permissions for the specified blueprint.
404	The blueprint-id parameter identifies a blueprint that does not exist in this account.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.
2. Open a web browser, and navigate to http://editor.swagger.io.
3. Import RestApis.yaml using the **File** > **Import File** option.

Policies

Use this resource to retrieve a list of policies that meet specified criteria.

Method summary

HTTP Method	Path	Description
GET	/bwl/policies	Returns a list of policy models that meet specified criteria. Only policies that you have access to are returned.

GET /policies

Description

Use this method to return a list of policies that meet specified criteria.

Parameters

Name	Location	Description	Required	Type
`X-IBM-API-Version`	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is `1.0.0`.	No	String
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
X-IBM-BWL-Account-Name	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String
space-id	Query	The identifier of the space to search within. If not specified, the search is account wide. If this space contains child spaces, the child spaces will also be searched. If both a space-id and space-name are specified, the space-id takes precedence.	No	String

Name	Location	Description	Required	Type
`space-name`	Query	The name of the space to search within. If not specified, the search is account wide. If both a space-id and space-name are specified, the space-id takes precedence.	No	String
`archived-state`	Query	Filters the results based on the archived state of the artifact. If not specified, 'active' is assumed. The only allowed values are `active` or `archived`.	No	String
`name`	Query	If specified, only policy models with this exact name are returned. Either a name or a tag must be specified, but not both.	No	String
`tag`	Query	If specified, only policy models with this exact tag are returned. Either name or tag must be specified, but not both.	No	String

Response

Example input

LIST all policies named MyPolicy in an account that you have access to:

```
https://www.blueworkslive.com/bwl/policies?name=MyPolicy
```

Example output

LIST all policies named MyPolicy in an account that you have access to:

```
HTTP/1.1 200 OK
[
  {
    "id": "250062",
    "name": "MyPolicy",
    "type": "policy",
```

```
          "space-ids": [
            "250006"
          ],
          "space-names": [
            "MySpace"
          ],
          "archived-state": "active"
        },
        {
          "id": "250093",
          "name": "MyPolicy",
          "type": "policy",
          "space-ids": [
            "250087"
          ],
          "space-names": [
            "MyOtherSpace"
          ],
          "archived-state": "active"
        }
      ]
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.
2. Open a web browser, and navigate to http://editor.swagger.io.
3. Import RestApis.yaml using the **File** > **Import File** option.

Policies/{policy-id}

Use this resource to retrieve the details of a specific policy model.

Method summary

HTTP Method	Path	Description
GET	/bwl/policies/{policy-id}	Returns the details of a specific policy model. You must have access to this policy model.

GET /policies/{policy-id}

Description

Use this method to return the details of a specific policy model.

Parameters

Name	Location	Description	Required	Type
policy-id	Path	The identifier of the policy model to retrieve.	Yes	String
X-IBM-API-Version	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is 1.0.0.	No	String

Name	Location	Description	Required	Type
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
X-IBM-BWL-Account-Name	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Response

Example input

GET the details of a specific policy:

```
https://www.blueworkslive.com/bwl/policies/250062
```

Example output

GET the details of a specific policy:

```
{
  "name": "MyPolicy",
  "id": "250062",
  "type": "policy",
  "tags": [

  ],
  "description": "A policy description.:lt;br \/&gt;",
  "space-ids": [
    "250006"
  ],
  "space-names": [
    "MySpace"
  ],
```

```
              "archived-state": "active",
              "file-references": [
                {
                  "name": "Policy specs.doc",
                  "id": "250073"
                }
              ],
              "hyperlink-references": [

              ]
          }
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.
403	Access is forbidden because you have insufficient permissions for the specified policy.
404	The policy-id parameter identifies a policy that does not exist in this account.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.

2. Open a web browser, and navigate to http://editor.swagger.io.

3. Import RestApis.yaml using the **File** > **Import File** option.

Policies/{policy-id}/usage

Use this resource to retrieve the details of where a specific policy model is used.

Method summary

HTTP Method	Path	Description
GET	/bwl/policies/{policy-id}/ usage	Returns information about where a specific policy model is used. You must have access to this policy model.

GET /policies/{policy-id}/usage

Description

Use this method to return the details of where a specific policy model is used.

Parameters

Name	Location	Description	Required	Type
`policy-id`	Path	The identifier of the policy model to retrieve.	Yes	String
`X-IBM-API-Version`	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is `1.0.0`.	No	String
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
X-IBM-BWL-Account-Name	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Response

Example input

GET the details of where a specific policy is used:

```
https://www.blueworkslive.com/bwl/policies/250062/usage
```

Example output

GET the details of where a specific policy is used:

```
{
  "target-artifact-name": "MyPolicy",
  "target-artifact-id": "250062",
  "usage-count": 2,
  "used-by-artifacts": [
    {
      "id": "250036",
      "name": "AnotherBlueprint",
      "type": "blueprint",
      "space-ids": [
        "250006"
      ],
      "space-names": [
        "MySpace"
```

```
                            ],
                            "archived-state": "active",
                            "published-state": "unpublished",
                            "blueprint-items": [
                                {
                                    "type": "activity",
                                    "id": "250038",
                                    "name": "AnotherBlueprint"
                                }
                            ]
                        },
                        {
                            "id": "2500b1",
                            "name": "MyDecision",
                            "type": "decision",
                            "space-ids": [
                                "250006"
                            ],
                            "space-names": [
                                "MySpace"
                            ],
                            "archived-state": "active",
                            "published-state": "unpublished",
                            "decision-items": [
                                {
                                    "type": "sub-decision",
                                    "id": "2500b3",
                                    "name": "MyDecision"
                                }
                            ]
                        }
                    ]
                }
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.
403	Access is forbidden because you have insufficient permissions for the specified policy.
404	The policy-id parameter identifies a policy that does not exist in this account.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.
2. Open a web browser, and navigate to http://editor.swagger.io.
3. Import RestApis.yaml using the **File** > **Import File** option.

Glossary-values

Use this resource to retrieve a list of glossary values that meet specified criteria.

Method summary

HTTP Method	Path	Description
GET	/bwl/glossary-values	Returns a list of glossary values that match specified criteria. The criteria might consist of consist of a category, or a category plus name, or neither. If neither name nor category is specified, all visible glossary values are returned.

GET /glossary-values

Description

Use this method to return a list of glossary values that meet specified criteria.

Parameters

Name	Location	Description	Required	Type
X-IBM-API-Version	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is 1.0.0.	No	String

Name	Location	Description	Required	Type
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
X-IBM-BWL-Account-Name	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String
name	Query	If specified, only values with this exact name are returned. If a name is specified, a category must be provided.	No	String

Name	Location	Description	Required	Type
category	Query	If specified, only values in this category are returned. A category could consist of one of the predefined categories listed, or it could be the name of a custom text property that is configured on the account. The category must be enabled for the account. Possible values are • participant-businessowner-expert • externalparticipant • system • supplier-customer • input-output • problem • kpi • goal • tag	No	String

Response

Example input

LIST all glossary values in the KPI category, in an account that you have access to:

```
https://www.blueworkslive.com/bwl/glossary-values?category=kpi
```

Example output

LIST all glossary values in the KPI category, in an account that you have access to:

```
HTTP/1.1 200 OK
[
  {
    "id": "2500e2",
    "name": "myKPI2",
    "category": "kpi",
    "visibility": "always-visible",
    "added-by-user": {
      "full-name": "Jane Doe",
      "email-address": "jdoe@myorg.com"
```

```
            },
            "added-by-date": "2016-08-09T23:39:30.149+0000"
          },
          {
            "id": "2500d8",
            "name": "myKPI1",
            "category": "kpi",
            "visibility": "always-visible",
            "added-by-user": {
              "full-name": "Jane Doe",
              "email-address": "jdoe@myorg.com"
            },
            "added-by-date": "2016-08-09T23:39:25.394+0000"
          }
        ]
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.
2. Open a web browser, and navigate to http://editor.swagger.io.
3. Import RestApis.yaml using the **File** > **Import File** option.

Glossary-values/{value-id}/usage

Use this resource to retrieve information for where a specified glossary value is used.

Method summary

HTTP Method	Path	Description
GET	/bwl/glossary-values/{value-id}/usage	Returns usage information for a specific glossary value.

GET /glossary-values/{value-id}/usage

Description

Use this method to return usage information for a specified glossary value.

Parameters

Name	Location	Description	Required	Type
`value-id`	Path	The identifier of the glossary value of interest.	Yes	String
`X-IBM-API-Version`	Header	The version of this API. If you are programming against the API, you MUST include the version. If omitted, the latest version of the API is used, which might be incompatible with prior versions. The current value is `1.0.0`.	No	String
`X-IBM-BWL-Account-Id`	Header	The identifier of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Name	Location	Description	Required	Type
X-IBM-BWL-Account-Name	Header	The name of the account to access. You must have access to this account. If you have access to multiple accounts, you must identify the specific account to access. If you are programming against the API, you are strongly encouraged to include either the account ID or the account name header, even if you don't currently belong to multiple accounts. If both the account ID and account name are provided, the account ID takes precedence. This header is not used if you instead send a JSESSIONID cookie generated from a previous successful call.	No	String

Response

Example input

GET the details of where a specific glossary value is used:

```
https://www.blueworkslive.com/bwl/glossary-values/2500d8/usage
```

Example output

GET the details of where a specific glossary value is used:

```
{
  "target-artifact-name": "myKPI1",
  "target-artifact-id": "2500d8",
  "usage-count": 1,
  "used-by-artifacts": [
    {
      "id": "2500b1",
      "name": "MyDecision",
      "type": "decision",
      "space-ids": [
        "250006"
      ],
      "space-names": [
        "MySpace"
```

```
                        ],
                        "archived-state": "active",
                        "published-state": "unpublished",
                        "decision-items": [
                          {
                            "type": "sub-decision",
                            "id": "2500b3",
                            "name": "MyDecision"
                          }
                        ]
                      }
                    ]
                  }
```

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	You did not pass authentication. This response could appear for the following reasons: • An invalid user name and/or password was provided. • You belong to multiple accounts and an account was not specified in the request.
403	Access is forbidden because you have insufficient permissions for the specified glossary value.
404	The value-id parameter identifies a glossary value that does not exist in this account.

More details

To get more details on the operations and response values, use a Swagger Editor to view the APIs:

1. Download the RestApis.yaml file.
2. Open a web browser, and navigate to http://editor.swagger.io.
3. Import RestApis.yaml using the **File** > **Import File** option.

File management and export

Manage files that are uploaded to your account and export artifact and space details using the Blueworks Live file management and export APIs.

File download

Use this resource to download a file that is attached to a process, an instance, or a comment.

Method summary

HTTP Method	Path	Description
GET	/scr/api/FileDownload	Downloads a file that is attached to a process, an instance, or a comment.

GET/scr/api/FileDownload

Description

Use this method to download a file that is attached to a process, an instance, or a comment.

Resource information

Requirements	Description
Response format	Raw data for the file in the same format it was uploaded in.
Requires authentication	Yes. The user must have at least access to the process, instance, or post to which the file is attached. If a file is attached to a post that is not posted yet or to an instance that is not started yet, the file can be downloaded only by the user who uploaded it.
Rate limited	Not yet

Parameters

The user must provide one of the following parameters to specify an object that the file is attached to:

- **processId**
- **instanceId**
- **postId**

If the file is attached to a post that is not posted yet or an instance that is not started yet, then the user is not required to provide the **postId** or **instanceId**.

Name	Location	Description	Required	Type
fileItemId	Query	The ID of the file item to download. This parameter is returned as a property of the response from `FileUpload`, `ProcessData`, `WorkDetail`, or `AppDetail` responses. For `FileUpload`, `WorkDetail`, and `AppDetail`, the value to be used is in the `id` field. For `ProcessData`, the value corresponds to the key in the `files` dictionary.	Yes	String

Name	Location	Description	Required	Type
processId	Query	The ID of the process that contains an activity with the file attachment.	Yes, if the downloaded file is attached to a process.	String
instanceId	Query	The ID of the instance that the file is attached to.	Yes, if the downloaded file is attached to an instance.	String
postId	Query	The ID of the post that the file is attached to.	Yes, if the downloaded file is attached to a post.	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Response

The data from the file.

Example input

```
https://www.blueworkslive.com/scr/api/FileDownload?fileItemId=9042b
```

File data

Response properties
None

Response messages

HTTP code	Reason
400	Bad request. See the response body for more information.
401	The user is not authorized to make the request.
403	The user does not have access to the process, instance, or post that the file is attached to.

File upload

Use this resource to add file attachments to process activities or work items.

Method summary

HTTP Method	Path	Description
POST	/scr/api/FileUpload	Adds file attachments to process activities or work items.

POST /scr/api/FileUpload

Description
Use this method to add file attachments to process activities or work items.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must have author access to the process that the file is attached to.
Rate limited	Not yet

Parameters
The entries for `locationField`, `ancestorIdField`, `parentIdField`, and `fileData` must be passed in standard multipart and multiform data format.

Name	Location	Description	Required	Type
`X-IBM-API-Version`	Header	The version of the API to use. The value must be `20160630`.	No	String

Name	Location	Description	Required	Type
`locationField`	Query	Depending on the object that the file is attached to, possible values are `activity`, `app`, `decision`, `instance`, `policy`, or `post`.	Yes	String
`ancestorIdField`	Query	The meaning depends on **locationField** value: • The process blueprint ID for an `activity` • The process application ID for an `app` • The decision diagram ID for a `decision` • The work ID of an existing instance for an `instance` • For all other **locationField** values, this parameter does not need to be provided	Yes	String

Name	Location	Description	Required	Type
parentIdField	Query	The meaning depends on **locationField** value: • The process ID for an activity • The application ID for an app • The decision ID for a decision • The work ID of an existing instance for an instance • The policy ID for a policy • This value does not have to be provided for post	Yes	String
fileData	Query	The file data.	Yes	RFC 1867 compliant multipart/form-data stream
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example output

```
{
    "size":"3.0",
    "parentId":"",
    "uploadUserId":"7f0002",
    "fileItemId":"123b",
    "name":"testFileUpload",
    "ancestorId":"",
    "uploadDate":"1441217698308",
    "status":"SUCCESS",
    "isPublic":"false",
    "location":"instance",
    "uploadUserName":"admin",
    "version":"20160630",
    "type":"1",
    "id":"5f501e134978a"
}
```

Response properties

uploadUserId
> The ID of the user who uploaded the file.

location
> The location that the file was uploaded to.

status The string value that indicates the result of the upload attempt. The possible values include:

> **SUCCESS**
> > The operation was successful.
>
> **FAILED**
> > The upload failed.
>
> **FAILED_DELETED**
> > The upload failed. Verify the file does not contain a virus, and try again.
>
> **SIZE** The upload failed because the file was larger than 50 MB.

SPACE

The upload failed because the account did not have enough space available.

type The type of file or document with the associated numeric value.

Type value	Description
1	Generic document. Used if the file doesn't fall under one of the other values.
2	Microsoft Excel file
3	Image
4	PDF document
5	Microsoft PowerPoint document
6	Microsoft Project document
7	Microsoft Visio document
8	Microsoft Word document
9	Text file

version

The version of the API that was used to create the response.

size The size of the uploaded file in bytes.

id The ID of the uploaded profile. Use this ID with the `FileDownload` service to download the file.

parentId

The ID of the parent item that the uploaded file is attached to.

name The name of the uploaded file.

ancestorId

The ID of the ancestor that the uploaded file belongs to.

uploadDate

The date the file was uploaded. The value is expressed in milliseconds since epoch, which is 1 January 1970 at midnight Coordinated Universal Time (UTC).

uploadUserName

The name of the user who uploaded the file.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.
405	The specified method is not allowed. Select POST method.

List files

Use this resource to retrieve a list of all or a range of files for an account.

Method summary

HTTP Method	Path	Description
GET	/scr/api/ListFiles	Retrieves a list of all or a range of files for the account.

GET /scr/api/ListFiles

Description

Use this method to retrieve a list of all or a range of files for the account.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an administrator to access the list file API.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
`accountId`	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
`from`	Query	Lists all files uploaded after this date. The date is represented with the following ISO 8601 date formats: • The long form is 2012-08-13T15:44:31.571-05:00. • The short form is 2012-08-13. The short form dates are assumed to be midnight Central Standard Time (CST) of the date specified. Therefore, 2012-07-25 is the same as 2012-07-25T00:00:00.000-05:00.	No	ISO 8601 date format

Response

Example input

```
https://blueworkslive.com/scr/api/ListFiles?from=2012-08-13T15:37:01.887-05:00
```

Example output

```
{
  "files":[
    {
```

```
                    "fileId":"b0050",
                    "uploadUserId":"bob@foo.com",
                    "uploadDate":"2012-08-13T15:37:08.770-05:00",
                    "fileSize":591728,
                    "fileName":"Earth.tif",
                    "attachedToId":"b000f",
                    "attachedToType":"process"
                }
            ]
        }
```

Response properties

fileId The unique ID of the file. Use this property to call the `FileDownload`.

uploadUserId
The login ID or email address of the user who uploaded the file.

uploadDate
The date when the file was uploaded, expressed in ISO 8601 format.

fileSize
The file size in bytes.

fileName
The name of the file.

attachedToId
The ID of the process, post, or other item that the file is attached to.

attachedToType
The process, post, or other type of object that the file is attached to.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.

Print diagram

Use this resource to produce a PDF file that contains a process diagram.

Method summary

HTTP Method	Path	Description
GET	/scr/api/PrintDiagram	Produces a PDF file that contains a process diagram.

GET /scr/api/PrintDiagram

Description
Use this method to produce a PDF file that contains a process diagram.

Resource information

Requirements	Description
Response format	PDF
Requires authentication	Yes. The user must have at least participant access to the process that is printed.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
processId	Query	The ID of the process to print.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

The PDF view of the diagram

Example input

```
https://www.blueworkslive.com/scr/api/PrintDiagram?processId=80012
```

Example output
PDF file

Response properties
None

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.
403	The user does not have access to the process that is printed.

BPMN export

Use this resource to export IBM Blueworks Live process diagrams to BPMN 2.0 format.

Method summary

HTTP Method	Path	Description
GET	/scr/api/BpmnExport/ [ZIP_FILE_NAME.zip]	Exports IBM Blueworks Live process diagrams to BPMN 2.0 format.

GET /scr/api/BpmnExport/[ZIP_FILE_NAME.zip]

Description
Use this method to export IBM Blueworks Live process diagrams to BPMN 2.0 format.

Resource information

Requirements	Description
Response format	ZIP
Requires authentication	Yes. The user who makes the request must be an administrative editor.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
from	Query	The date is in the `YYYY-MM-DD` format. The user can use this parameter to filter out processes that were modified before this date. This date is in the timezone set by the user preferences. If the user timezone isn't set, the API uses the default server timezone.	No	String

Response

Data from the compressed file.

Example input

This example demonstrates an export from `MyAccount` of all active processes that have diagrams and that were modified on or after 1 December 2011.

```
http://www.blueworkslive.com/scr/api/BpmnExport/Backup.zip?account=MyAccount&from=20
```

Example output

ZIP file with process diagrams.

Response properties

None.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.
404	The specified process diagrams were not found.

Excel export

Use this resource to export an Excel representation of a space, process, or decision.

Method summary

HTTP Method	Path	Description
GET	/scr/api/ExcelExport	Retrieves an Excel representation of space, process, or decision.

GET /scr/api/ExcelExport

Description

Use this method to export an Excel representation of a space, process, or decision.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must have at least participant access to the item.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
decisionID	Query	The ID of the decision to export. Only one of **spaceID**, **processID**, and **decisionID** must be specified.	No	String
processID	Query	The ID of the process to export. Only one of **spaceID**, **processID**, and **decisionID** must be specified.	No	String

Name	Location	Description	Required	Type
spaceID	Query	The ID of the space to export. Only one of **spaceID**, **processID**, and **decisionID** must be specified.	No	String

Response
> The Excel export for the space, process, or decision that was exported.

Example input

> This example shows how to use the ExcelExport and Search APIs to export a space and its subspaces given a space name.
>
> The sample uses the json4j library from the Apache Wink project to decode the JSON responses sent back by the API.
>
> 1. Download the java file SpaceExcelExportExample.java.
> 2. Download the jar json4j.
> 3. Compile the sample.
>   - For Windows operating systems:
>      ```
>      javac -cp .;wink-json4j-1.3.0.jar SpaceExcelExportExample.java
>      ```
>   - For Linux operating systems:
>      ```
>      javac -cp ./wink-json4j-1.3.0.jar SpaceExcelExportExample.java
>      ```
> 4. Run it.
>
>   Then the system downloads an Excel file for each of the spaces returned from search for "Process Examples" and puts each file in \Temp\.
>
>   The username, password, and the optional parameter account are the user name, password, and account name of the Blueworks Live user.
>   - For Windows operating systems:
>      ```
>      java -cp .;wink-json4j-1.3.0.jar SpaceExcelExportExample "Process Examples" \Te
>      ```
>   - For Linux operating systems:
>      ```
>      java -cp .:./wink-json4j-1.3.0.jar SpaceExcelExportExample "Process Examples" \
>      ```

Example output
> Excel file

Response properties
> None

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.
403	The user does not have access to the process. The process requires at least participant access.

Export process

Use this resource to export a specific revision of a process in BPMN 2.0 format.

Method summary

HTTP Method	Path	Description
GET	/scr/api/export/process/ {process_id}/{revision_id}	Exports a specific revision of a process in BPMN 2.0 format.

GET /scr/api/export/process/{process_id}/{revision_id}

Description

Use this method to export a specific revision of a process in BPMN 2.0 format.

Resource information

Requirements	Description
Response format	ZIP
Requires authentication	Yes. The user who requests the process export must be an editor for that process.
Rate limited	No

Parameters

Name	Location	Description	Required	Type
process_id	Path	The ID of the exported process.	Yes	String
revision_id	Path	The ID of the exported revision.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

The data from the process in BPMN 2.0 format including all of the linked processes for the requested process.

Example input

The following example shows how to export a revision with **revision_id** value 1300a1 and the **process_id** value 130024.

`http://www.blueworkslive.com/scr/api/export/process/130024/1300a1`

Example output

File data

Response properties

None

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.
403	The user does not have access to the process. The process requires at least editor access.

User management

Perform user management tasks and query user data using the Blueworks Live user management APIs.

Authentication

Use this resource to authenticate a user account.

Method summary

HTTP Method	Path	Description
GET	/api/Auth	Authenticates a user account.

GET /api/Auth

Description

     Use this method to authenticate a user account.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The provided authentication is checked by a call.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
`accountId`	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
`version`	Query	The version of the requested API. The response varies depending on the requested version. Two possible values are 20091212 and 20110917. If the version parameter is not provided or if the value is not recognized, then the results for the 20091212 version are returned.	No. This parameter might become required in the future.	String

Responses for version `20091212`

Example input

```
https://www.blueworkslive.com/api/Auth?version=20091212
```

Example output

```
{
  "result": "authenticated",
  "accounts": ["user_account1", "user_account2"],
  "selectedAccount": "user_account1",
  "selectedUserId": "23ed4a3"
}
{
  "result": "trial_expired",
  "accounts": ["user_account1", "user_account2"],
  "selectedAccount": "user_account1",
  "selectedUserId": "23ed4a3",
  "mainAdminEmail": "admin_email@website.com",
}
```

```
{
  "result": "free_or_full",
  "accounts": ["user_account1", "user_account2"],
  "selectedAccount": "user_account1",
  "selectedUserId": "65f6a",
  "accountType": "TRIAL"
}
{
  "result": "multiaccount",
  "accounts": ["user_account1", "user_account2"],
  "selectedAccount": null,
  "selectedUserId": null,
  "accountType": "TRIAL"
}
```

Response properties

result Contains the following possible values:

authenticated
> The user is fully authenticated and permitted to use the application.

notauthenticated
> The provided user name or password is incorrect.

multiaccount
> The provided user name belongs to multiple accounts. The user must select a specific account to log in with.

showeula
> The user must accept the terms of use before proceeding.

inactiveaccount
> The account that the user attempted to log in to is inactive.

disabled
> The provided user login is disabled.

invalidpassword
> The provided password was correct, but the administrator invalidated it.

expiredpassword
> The password for the user expired automatically.

lockedout
> The user is locked out of the account until it is explicitly unlocked.

lockedout_temporarily
> The user is locked out of the account temporarily until it is explicitly unlocked, or until the lockout period expires.

trial_expired
> The trial period for the account expired and the user is no longer allowed to log on. The account administrator, who can be reached at

`mainAdminEmail`, must choose to upgrade the user to a full (paid) account or to downgrade to a free account.

free_or_full
> The user account expired, and the user must choose whether to upgrade to the full (paid) status or to downgrade to the free status.

ip_address_not_allowed
> The account does not allow to log in from the client's IP address.

api_disabled
> The APIs are disabled for the provided account.

noncompliantpassword
> The password was invalidated because the password policy for the account changed.

viewer_not_allowed
> A user with a viewer license tried to authenticate with an account that had viewer licenses disabled.

accounts
> Optional parameter. The array with names and IDs of accounts for which the user provided correct user name and password. If the provided user name and password don't match any account, then this property is not provided.

selectedAccount
> The name of account provided by the **account** or **accountId** parameter or the ID of a single account that the user was able to log in to. If the **account** or the **accountId** parameter is not provided, and a single account cannot be automatically selected, the ID of the account has `null` value.

selectedUserId
> The ID of the user.

token The value used for a single sign-on; it is not used by the API.

mainAdminEmail
> Optional parameter. It is provided only if the result value is `trial_expired`.
>
> The email address of the main administrator for the account. The account administrator must choose whether to upgrade to a full (paid) account or downgrade to a free account.

accountType
> Optional parameter. The current type of account. It is provided only if the result value is `free_or_full`.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Response for version **20110917**

Example input

```
https://www.blueworkslive.com/api/Auth?version=20110917
```

Example output

```
{
  "result": "authenticated",
  "accounts": [
    {
      "name": "user_account1",
      "id":"42563a3"
    },
    {
      "name": "user_account2",
      "id": "45627bc"
    }
  ],
  "selectedAccountId": "42563a3",
  "selectedUserId": "23ed4a3"
  "selectedUserLicense":
    {
      "name": "Editor",
      "id": "1"
    }
}
{
  "result": "trial_expired",
  "accounts": [
    {
      "name": "user_account1",
      "id":"42563a3"
    },
    {
      "name": "user_account2",
      "id": "45627bc"
    }
  ],
  "selectedAccountId": "42563a3",
  "selectedUserId": "23ed4a3",
  "selectedUserLicense":
    {
      "name": "Editor",
      "id": "1"
    },
  "mainAdminEmail": "admin_email@website.com",
}
{
  "result": "free_or_full",
  "accounts": [
    {
      "name": "user_account1",
      "id":"42563a3"
```

```
    },
    {
        "name": "user_account2",
        "id": "45627bc"
    }
],
"selectedAccountId": "42563a3",
"selectedUserId": "65f6a",
"selectedUserLicense":
    {
        "name": "Editor",
        "id": "1"
    },
"accountType": "TRIAL"
}
{
    "result": "multiaccount",
    "accounts": [
        {
            "name": "user_account1",
            "id":"42563a3"
        },
        {
            "name": "user_account2",
            "id": "45627bc"
        }
    ],
    "selectedAccountId": null,
    "selectedUserId": null,
    "selectedUserLicense": null,
    "accountType": "TRIAL"
}
```

Response properties

result   Contains the following possible values:

authenticated
> The user is fully authenticated and permitted to use the application.

notauthenticated
> The provided user name or password is incorrect.

multiaccount
> The provided user name belongs to multiple accounts. The user must select a specific account to log in with.

showeula
> The user must accept the terms of use before proceeding.

inactiveaccount
> The account that the user attempted to log in to is inactive.

disabled
> The provided user login is disabled.

invalidpassword
> The provided password was correct, but the administrator invalidated it.

expiredpassword
> The password for the user expired automatically.

lockedout

The user is locked out of the account until it is explicitly unlocked.

lockedout_temporarily

The user is locked out of the account temporarily until it is explicitly unlocked, or until the lockout period expires.

trial_expired

The trial period for the account expired and the user is no longer allowed to log on. The account administrator, who can be reached at `mainAdminEmail`, must choose to upgrade the user to a full (paid) account or to downgrade to a free account.

free_or_full

The user account expired, and the user must choose whether to upgrade to a full (paid) status or to downgrade to a free status.

ip_address_not_allowed

The account does not allow to log in from the client's IP address.

api_disabled

The APIs are disabled for the provided account.

noncompliantpassword

The password was invalidated because password policy for the account changed.

viewer_not_allowed

A user with the viewer license tried to authenticate with an account that had viewer licenses disabled.

accounts

Optional parameter. The array with names and IDs of accounts for which the user provided correct user name and password. If the provided user name and password don't match any account, then this property is not provided.

accounts.name

The name of the account.

accounts.id

The ID of the account.

selectedAccountId

The ID of the account provided by the **account** or **accountId** parameter or the ID of a single account that the user was able to log in to. If the **account** or the **accountId** parameter is not provided, and a single account cannot be automatically selected, the ID of the account has `null` value.

selectedUserId

The ID of the user, which has `null` value if authentication result is not `authenticated`.

selectedUserLicense
The license of the user, which has `null` value if authentication result is not `authenticated`.

selectedUserLicense.name
The license name has one of the following four values:

- `Editor`
- `Contributor`
- `Community`
- `Viewer`

selectedUserLicense.id
The license ID has one of the following four values: 1, 2, 3, 4. Each ID value corresponds to one of the following license types:

- 1 - Editor
- 2 - Contributor
- 3 - Community
- 4 - Viewer

token The value used for a single sign-on; it is not used by the API.

mainAdminEmail
Optional parameter. It is provided only if the result value is `trial_expired`.

The email address of the main administrator for the account. The account administrator must choose whether to upgrade to a full (paid) account or downgrade to a free account.

accountType
Optional parameter. This is the current type of the account. It is provided only if the result value is `free_or_full`.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

User list

Use this resource to return a list of users in the authenticated user account.

Method summary

HTTP Method	Path	Description
GET	/scr/api/UserList	Returns a list of users in the authenticated user account.

GET /scr/api/UserList

Description

Use this method to return a list of users in the authenticated user account.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user making the request must be an administrator.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
version	Query	The version of the requested API. The response might vary depending on the requested version. Two possible values are 20091212 and 20110917. If a parameter is not provided, then the results for the 20091212 version are returned.	No. This parameter might become required in the future.	String

Response for version 20091212

Example input

```
https://www.blueworkslive.com/scr/api/UserList
```

Example output

```
[
  {
    "name":"user name 1",
    "email":"user1_email@website.com",
    "role":"Administrator",
    "date":"2011\/07\/14 09:39:10 -05:00",
    "archived":false,
    "invited":false,
    "licensed":true,
    "locked":false
  },
  {
    "name":"user name 2",
    "email":"user2_email@website.com",
    "role":"Participant",
    "date":"",
    "archived":false,
    "invited":false,
    "licensed":true,
    "locked":false
  },
  {
    "name":"user name 3",
    "email":"user3_email@website.com",
    "role":"Author",
    "date":"2011\/07\/14 09:25:43 -05:00",
    "archived":false,
    "invited":false,
    "licensed":true,
    "locked":false
  }
]
```

Response properties

name The full name of the user.

email The email address of the user.

role Contains one of the following possible values:

- `Administrator` if the user is an administrator
- `Author` if the user is an editor
- `Participant` for all other roles

date The value depends on values of the `archived` and `invited` properties.
- If `archived` is `true`, then it's the date when the user was archived.
- If `invited` is `true`, then it's the date when the user was invited.
- In all other instances, it's the date of the user's most recent login.

archived

Contains one of the following possible values:
- `true` if the user is archived
- `false` if the user is not archived

invited

Contains one of the following possible values:
- `true` if the user is invited
- `false` if the user is not invited

licensed

Contains one of the following possible values:
- `true` if the user is licensed, that is the user is counted in the license used counts
- `false` if the user is not counted in the license used counts

locked

The value is `true` if the user is locked out, that is the user can't log in at the moment. Otherwise, the value is `false`.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.

Response for version `20110917`

Example input

```
https://www.blueworkslive.com/scr/api/UserList?version=20110917
```

Example output

```
{
  "version":"20110917",
  "users":[
    {
      "name":"user name 1",
      "id":"7000f",
      "email":"user1_email@website.com",
      "avatarId":"70010",
      "license":"Editor",
```

```
            "admin":true,
            "date":1310654350393,
            "archived":false,
            "invited":false,
            "licensed":true,
            "locked":false
          },
          {
            "name":"user name 2",
            "id":"70022",
            "email":"user2_email@website.com",
            "license":"Community",
            "admin":false,
            "archived":false,
            "invited":false,
            "licensed":true,
            "locked":false
          },
          {
            "name":"user name 3",
            "id":"70021",
            "email":"user3_email@website.com",
            "license":"Contributor",
            "admin":false,
            "archived":false,
            "invited":false,
            "licensed":true,
            "locked":false
          },
          {
            "name":"user name 4",
            "id":"70020",
            "email":"user4_email@website.com",
            "license":"Viewer",
            "admin":false,
            "date":1310653543628,
            "archived":false,
            "invited":false,
            "licensed":true,
            "locked":false
          }
        ]
      }
```

Response properties

version
> The version of the API used to create the response.

users The array of user objects.

user.name
> The full name of the user.

user.id
> The ID of the user.

user.email
> The email address of the user.

user.avatarId
> The ID of the user's avatar.

user.license
> Contains one of the following possible values:
> - Editor
> - Contributor

- Community
- Viewer

user.admin

Contains one of the following possible values:

- `true` if the user is an administrator
- `false` if the user is not an administrator

user.date

This property depends on the values of the `archived` and `invited` properties.

- If `archived` is `true`, then it's the date when the user was archived.
- If `invited` is `true`, then it's the date when the user was invited.
- In all other instances, it's the date of the user's most recent login.
- If all the values are null, then the date property is not present.

The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

user.archived

Contains one of the following possible values:

- `true` if the user is archived
- `false` if the user is not archived

user.invited

Contains one of the following possible values:

- `true` if the user is invited
- `false` if the user is not invited

user.licensed

Contains one of the following possible values:

- `true` if the user is licensed, that is the user is counted in the license used counts
- `false` if the user is not counted in the license used counts

user.locked

The value is `true` if the user is locked out, that is the user can't log in at the moment. Otherwise, the value is `false`.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.

Avatar

Use this resource to retrieve an avatar corresponding to the **avatarId** parameter or a default avatar, if the **avatarId** parameter is not found or not provided.

Method summary

HTTP Method	Path	Description
GET	/scr/api/Avatar	Retrieves an avatar corresponding to the **avatarId** parameter or a default avatar, if the **avatarId** parameter is not found or not provided.

GET /scr/api/Avatar

Description

Use this method to retrieve an avatar corresponding to the **avatarId** parameter or a default avatar, if the **avatarId** parameter is not found or not provided.

Resource information

Requirements	Description
Response format	Image data
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
avatarId	Query	The ID of the avatar to download. A default avatar image is returned in the following cases: • When the **avatarId** is not provided • When an **avatarId** is provided, but an avatar matching the ID is not found	No	String

Name	Location	Description	Required	Type
location	Query	Indicates the location where the image is displayed in IBM Blueworks Live. Each location returns an avatar in a different image size. If the location value is not provided, then the image is returned in the default size of 85 x 85 pixels. The possible values include: • chat (20 x 20 pixels) • comments (30 x 30 pixels) • news (40 x 40 pixels) • settings (85 x 85 pixels)	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/Avatar?avatarId=a0777
```

Example output

Image data for the avatar.

Response properties

None

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.

User provisioning

Use this resource to complete simple queries and user provisioning and deprovisioning requests. The user must be an account administrator.

Method summary

HTTP Method	Path	Description
GET	/scr/api/provision/user/	Retrieves details about a specific user account and lists all users in the account as a JSON array.

HTTP Method	Path	Description
PUT	/scr/api/provision/user/	Adds a user or updates the user's membership in groups.
DELETE	/scr/api/provision/user/	Archives each user individually.

GET /scr/api/provision/user/

Description

Use this method to retrieve details about a specific user account and list all account users in a JSON array.

The user provisioning service provides the following forms of a query method:

- The base URI behaves as a collection URI and retrieves a list of users in the account as a JSON array.
- The base URI with the user's email address behaves as an element URI and retrieves specific details about the account that is specified by the email address.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an account administrator.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
version	Header	The requested API version is 1.0.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

LIST all users in the account: /scr/api/provision/user/

```
curl -i --user admin:password -H "Version:1.0" "https://www.blueworkslive.com/scr/ap
```

GET the details of a specific user: /scr/api/provision/user/{*email address*}

```
curl -i --user admin:password -H "Version:1.0" "https://www.blueworkslive.com/scr/ap
```

Note: The email address must be properly encoded.

Example output

LIST all users in the account:

```
HTTP/1.1 200 OK
{"users":["admin@us.ibm.com","user_id@us.ibm.com","test@us.ibm.com"]}
```

GET the details of a specific user:

```
HTTP/1.1 200 OK
Content-Length: 108
{"username":"user_id@us.ibm.com","fullname":"User Name","license":"EDITOR","admin":fa
```

Response properties

username
> The email address of the provisioned user.

fullname
> The full name of the provisioned user. The default value is an empty string.

license
> The license level of the provisioned user. The default value is EDITOR

admin The value is true if the user is an administrator. Otherwise, the value is false.

active The value is `true` if the user is active. Otherwise, the value is `false`.

usergroups

The list of groups that the user is a part of. This property is returned only if the user is a part of at least one group.

usergroups.groupname

The name of a group that the user is a part of.

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400 with JSON payload	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.
404	The specified GET user not found.

PUT /scr/api/provision/user/

Description

Use this method to add a user and update the user's membership in groups.

• The base URI behaves as an add method. With the add method, when a previously archived user is added, the account of that user is reactivated.

• The base URI with the user name behaves as an update method. The update method is used to update the user's membership in one or multiple groups. If the user name is non-existent, then a user is created with default attributes, as listed below.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an account administrator.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
version	Header	The requested API version is `1.0`.	Yes	String

Name	Location	Description	Required	Type
`content-type`	Header	The value must be `application/json`.	Yes	String
`account`	Query	The name of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String
`accountId`	Query	The ID of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String
`JSON Request username`	JSON Body	The email address of the user to be provisioned.	Yes, when adding a new user.	String
`JSON Request fullname`	JSON Body	The full name of the user to be provisioned. A blank full name is used if none is provided.	Yes	String
`JSON Request license`	JSON Body	The license level of the user. The possible values are `VIEWER`, `COMMUNITY`, `CONTRIBUTOR`, or `EDITOR`. The license is set to default `EDITOR` level if none is provided.	Yes	String

Name	Location	Description	Required	Type
JSON Request admin	JSON Body	The parameter value, `true` or `false`, indicates whether the provisioned user must be an administrator. The default setting is `false`.	Yes	Boolean
JSON Request usergroups	JSON Body	The JSON array of group names. Add user to the list of usergroups. The user is not added to the groups if the list is empty.	Yes, for updating.	JSON array of strings

Response

Example input

Provision a specific user: /scr/api/provision/user/

Provision a user with only the required **username** parameter in the JSON Body:

```
curl -i --user admin:password -H "Version:1.0" -H "Content-Type:application/json"
        "{\"username\":\"user1@us.ibm.com\"}" "https://www.blueworkslive.com/s
```

Provision a user with all parameters in the JSON Body:

```
curl -i --user admin:password -H "Version:1.0" -H "Content-Type:application/json"
        --data "{\"username\":\"john.smith@us.ibm.com\",\"fullname\":\"John Sm
        \"usergroups\":[{\"groupname\":\"group1\"},{\"groupname\":\"group2\"}]
```

Add an existing user to a group: /scr/api/provision/user/{*email address*}

Note: The email address must be properly encoded.

Add a user to groups:

```
curl -i --user admin:password -H "Version:1.0" -H "Content-Type:application/json"
        --data "{\"usergroups\":[{\"groupname\":\"group1\"},{\"groupname\":\"g
```

Add a user to an invalid group:

```
curl -i --user admin:password -H "Version:1.0" -H "Content-Type:application/json"
        --data "{\"usergroups\":[{\"groupname\":\"invalid\"}]}" "https://www.b
```

Adding multiple users

A sample MS-DOS script is used to exercise the PUT request multiple times with values from a CSV file. The script maps token to each column in the CSV file. More robust scripting with Perl, Python, or another language can also be used.

```
for /f "tokens=1,2,3,4,5,6* delims=," %%a in (usersdetail.csv)
"{\"username\":\"%%a\", \"fullname\":\"%%b\",\"license\":\"%%c
"http://localhost:8080/scr/api/provision/user/"
```

The `usersdetail.csv` is then in the same directory as the batch script with the following sample content:

```
abc@ibm.com,john smith,editor,true, groupA, groupB
bac@ibm.com,smith john,community,false,group1, groupB
```

Note: The token is manually mapped to columns in the CSV file; therefore, it is best if all provisioned users belong to the same number of groups.

Example output

Provision a user with only the required **username** parameter in the JSON Body:

```
HTTP/1.1 200 OK
Content-Length: 107
{"username":"user1@us.ibm.com","fullname":"","license":"EDITOR","admin":false,"activ
```

Provision a user with all parameters in the JSON Body:

```
HTTP/1.1 200 OK
Content-Length: 155
{"username":"john.smith@us.ibm.com","fullname":"John Smith","license":"EDITOR","admin
"usergroups":[{"groupname":"group1"},{"groupname":"group2"}]}
```

Add a duplicate user:

```
HTTP/1.1 400 Bad Request
Content-Length: 71
{"message":"The provided username is already a member of this account"}
```

Add a user to groups:

```
HTTP/1.1 200
OK Content-Length: 107
{"username":"user1@us.ibm.com","fullname":"User One","license":"EDITOR","admin":true
```

Add a user to an invalid group:

```
HTTP/1.1 400 Bad Request
Content-Length: 79
{"message":"One of the provided groupname is either archived or is not valid."}
```

Response properties

username
> The email address of the provisioned user.

fullname
> The full name of the provisioned user. The default value is an empty string.

license
> The license level of the provisioned user. The default value is EDITOR.

admin The value is `true` if the user is an administrator. Otherwise, the value is `false`.

active The value is `true` if the user is active. Otherwise, the value is `false`.

usergroups
> The list of groups that the user is a part of. This is returned only if the user is a part of at least one group.

The name of the group that the user is a part of.

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400	Any other errors with JSON message. One of the following request problems might cause this message: • The specified user already exists • There is an issue with the format or the licensing .
401	The user is not authorized to make the request.
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.
404	The specified user not found.

DELETE /scr/api/provision/user/

Description

Use this method to archive each user individually. The user archiving service provides a single form of the DELETE method, so there is no support for collections. This method can be called repeatedly to delete multiple users.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an account administrator.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
`account`	Query	The name of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String
`accountId`	Query	The ID of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String
`version`	Header	The requested API version is 1.0.	Yes	String

Response

Example input

Delete a specific user in the account: `/scr/api/provision/user/`
`{email address}`

```
curl -i --user admin:password -H "Version:1.0" -X DELETE "https://www.blueworkslive.d
```

Note: The email address must be properly encoded.

Example output

Delete a user:

```
HTTP/1.1 200 OK
Content-Length: 21
{"message":"Success"}
```

Delete a non-existent user:

```
HTTP/1.1 404 Not Found
Content-Length: 0
```

Response properties
None

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400	Any other errors with JSON message. Check the JSON message element for details.
401	The user is not authorized to make the request.
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.
404	The requested user not found.

Manage group

Use this resource to complete simple queries, modify and archive group requests. The user must be an account administrator.

Method summary

HTTP Method	Path	Description
GET	/scr/api/ManageGroup	Retrieves a JSON array of all groups in the account and extracts details about a specific group.
PUT	/scr/api/ManageGroup	Adds and updates a group.
DELETE	/scr/api/ManageGroup	Archives each group individually.

GET /scr/api/ManageGroup

Description
Use this method to retrieve a JSON array of all groups in the account or to extract details about a specific group.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an account administrator. To archive a group, the user must also have the MANAGE permission.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
version	Header	The version of the requested API is 1.0	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

LIST all groups in the account: /scr/api/ManageGroup

```
curl -i --user admin:password -H "Version:1.0" "https://www.blueworkslive.com/scr/a
```

GET the details of a specific group: /scr/api/ManageGroup/
{*groupname*}

```
curl -i --user admin:password -H "Version:1.0" "https://www.blueworkslive.com/scr/ap
```

Note: The *groupname* must be properly encoded.

Example output

LIST all groups in the account:

```
                        HTTP/1.1 200 OK
                        Content-Length: 122
                        {"usergroups":["group admin","group1","group2","group3","group7","group18","group1
```

GET the details of a specific group:

```
                        HTTP/1.1 200 OK
                        Content-Length: 149
                        {"groupname":"group admin","description":"This is a custom group for admins","user
                                    "groups":["group2","group3"]}
```

Response properties

groupname
: The name of the group.

description
: The description of the group.

users The list of users who are members of the group.

groups
: The list of subgroups within the group.

usergroups
: The list of groups in the account.

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400 with JSON payload	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.
404	The specified GET group not found.

PUT /scr/api/ManageGroup

Description
: Use this method to add and update a group. Adding and updating a group is possible with the following methods:

  • The base URI behaves as an add method. The user who creates a new group becomes a member of the group and gets the MANAGE permission.

  • The base URI with the group name behaves as an update method.

  A list of users or groups can be added as a member of the group with either one of the methods. A parent user group cannot be added as a member of its child group.

Resource information

Requirements	Description
Response format	JSON

Requirements	Description
Requires authentication	Yes. The user must be an account administrator. To archive a group, the user must also have the MANAGE permission.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
version	Header	The version of the requested API is 1.0	Yes	String
content-type	Header	Must be application/ json	Yes	String
group_name	Path	If a group name is provided in the path, the command updates the specified group name with data from the request body.	No	String
account	Query	The name of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String
accountId	Query	The ID of the account to authenticate against. If an account is not provided, then all accounts that the user is a member of are checked. And if there is a single match for the supplied password, then that account is used.	No	String

Name	Location	Description	Required	Type
JSON Request group name	JSON Body	Defines a new added group.	Yes	String
JSON Request description	JSON Body	Blank description is used if none is provided.	No	String
JSON Request users	JSON Body	The array of users to be added as members of the group.	Yes	JSON array of strings
JSON Request groups	JSON Body	The array of groups to be added as members of the group.	Yes	JSON array of strings

Response

Example input

Add a new group: /scr/api/ManageGroup

Add a group with just the required **group name** parameter in the JSON Body:

```
curl -i --user admin:password -H "Version:1.0" -H "Content-Type:application/json"
            --data "{\"groupname\":\"group admin\"}" "https://www.blueworkslive.cc
```

Add a group with a list of users and groups as members in the JSON Body:

```
            curl -i --user admin:password -H "Version:1.0" -H "Content-Type:applic
            --data "{\"groupname\":\"group 10\", \"description\":\"group 10 descri
            "https://www.blueworkslive.com/scr/api/ManageGroup/"
```

Add an existing user to the group: /scr/api/ManageGroup/
{*groupname*}

Note: The *groupname* must be properly encoded.

Update the group with a list of users or groups:

```
            curl -i --user admin:password -H "Version:1.0" -H "Content-Type:applic
            --data "{\"users\":[\"admin2\", \"user2\"], \"groups\":[\"group3\"]}"
```

Update the group with an invalid user in the participant list:

```
            curl -i --user admin:password -H "Version:1.0" -H "Content-Type:applic
            --data "{\"users\":[\"invalidUser\",\"user 10\"]}" "https://www.bluewc
```

Example output

Add a group with the required group name parameter in the JSON Body:

```
HTTP/1.1 200 OK
Content-Length: 68
            {"groupname":"group admin","description":"","users":["admin"]}
HTTP/1.1 200 OK
Content-Length: 180
            {"groupname":"group 10","description":"group 10 description","users":[
```

Add a group with a list of users and groups as members in the
JSON Body:

```
HTTP/1.1 200 OK
Content-Length: 155
          {"groupname":"group 10","description":"group 10 description","users":["adm
```

Update the group with a list of users or groups:

Add duplicate group:

```
HTTP/1.1 400 Bad Request
Content-Length: 71
          {"message":"The provided group name already exist in this account"}
```

Update the group with an invalid user in the participant list:

```
HTTP/1.1 400 Bad Request
Content-Length: 100
          {"message":"One of the provided user name is not valid. Ensure the user na
```

Response properties

groupname
      The name of the group.

description
      The description of the group.

users   The list of users who are members of the group.

groups
      The list of groups that are members within the group.

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400	Any other errors with JSON message. One of the following request problems might cause this message: • The specified user already exists. • There is an issue with the format or the licensing.
401	The user is not authorized to make the request.
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.
404	The specified group not found.

DELETE /scr/api/ManageGroup

Description
Use this method to archive each group individually. This group archiving
method provides a single form of the DELETE method, and there is no
support for collections. To delete multiple groups, call this method
repeatedly. The user must have the MANAGE permission.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes. The user must be an account administrator. To archive a group, the user must also have the MANAGE permission.
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
version	Header	The version of the requested API is 1.0	Yes	String

Response

Example input

Delete a specific group in the account: /scr/api/ManageGroup/ {*groupname*}

```
curl -i --user admin:password -H "Version:1.0" -X DELETE "https://www.blueworksliv
```

Note: The *groupname* must be properly encoded.

Example output

Delete group:

```
HTTP/1.1 200 OK
Content-Length: 21
            {"message":"Success"}
```

Delete a non-existent group:

```
HTTP/1.1 404 Not Found
Content-Length: 0
```

Delete a group where user does not have the MANAGE permission:

```
HTTP/1.1 400 Bad Request
Content-Length: 71
            {"message":"User attempted to modify group without proper permission."
```

Response properties
None

Response messages

HTTP code	Reason
200 with JSON payload	The request was completed successfully.
400	Any other errors with JSON message. Check the JSON message element for details.
401	The user is not authorized to make the request.

HTTP code	Reason
403	The access is forbidden. This message might appear for one of the following reasons: • The specified credentials are invalid. • You are not an editor for this process. • The administrator did not accept the Terms and Conditions of service.
404	The requested group not found.

Work management

Query data about in-flight and completed tasks, launch new work, and complete or comment on existing work using the Blueworks Live work management APIs.

Task list

Use this resource to get a list of open tasks assigned to the current user and a list of the user's completed tasks.

Method summary

HTTP Method	Path	Description
GET	/scr/api/TaskList	Gets a list of open tasks assigned to the current user and a list of the user's completed tasks.

GET /scr/api/TaskList

Description

Use this method to get a list of open tasks assigned to the current user and a list of the user's completed tasks.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
type	Query	The type of list to return. The following values are supported: `assigned` or `completed`. If the value is not supported, the 400 response code is returned.	Yes	String

Name	Location	Description	Required	Type
version	Query	The version of the requested API. The only allowed value is 20110917.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
limit	Query	The maximum number of tasks to return. If this parameter is not provided, the default value of 100 is used.	No	Integer

Name	Location	Description	Required	Type
startFrom	Query	If this parameter is provided, then the returned list starts with the task at this index in the list (zero based). Use this parameter if the number of available tasks is higher than the number of the returned tasks in an earlier call limit.	No	Integer

Response

Example input

```
https://www.blueworkslive.com/scr/api/TaskList?type=completed&version=20110917
```

Example output

```
{
        "work":[
          {
             "name":"ProcessApp 1 - Multiple process apps.",
             "id":"5f501e1345eda"
          },
          {
             "name":"ProcessApp 2 - Multiple process apps.",
             "id":"5f500e134a57f"
          },
        ]
}
```

Response properties

version
> The version of the API that was used to create a response.

more Contains one of the following possible values:
- `true` if there are more tasks on the server
- `false` if the task entry includes all of the tasks available

type The type of tasks in the list. The possible values are `assigned` or `completed`. These values match the one that can be passed for the type parameter.

tasks The array of tasks. The tasks are listed in reverse chronological order of `dueDate`. Items with the same `dueDate` are sorted by instance ID and task ID.

task.name
> The name of the task.

task.id
> The ID of the task.

task.workId
> The ID of the associated work. This property matches the

ID property of one of the entries in the work array. More than one task can reference the same work entry.

task.dueDate
Optional parameter. The due date for the task. If the task has no due date, this field is not present. The due date is listed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work The array of work items. The work items are in no particular order. There is an entry in the array for every unique value of workId in the tasks array.

work.name
The name that was given to the work instance when it was started.

work.id
The ID of the work instance.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Task list count

Use this resource to get the number of tasks assigned to or completed by the user.

Method summary

HTTP Method	Path	Description
GET	/scr/api/TaskListCount	Gets the number of tasks that are assigned to or completed by the user.

GET /scr/api/TaskListCount

Description
Use this method to get the number of tasks assigned to or completed by the user.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
type	Query	The type of tasks to be counted. The supported values are `assigned` or `completed`. If the value is not supported, the response code of 400 is returned.	Yes	String
version	Query	The version of the requested API. The only allowed value is `20110917`.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/TaskListCount?type=completed&version=20110917
```

Example output

```
{
    "count":17
}
```

Response properties

count   The total number of tasks of a specific type that are owned by the user.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Work list

Use this resource to retrieve a list of open or closed work items that were started by the user.

Method summary

HTTP Method	Path	Description
GET	/scr/api/WorkList	Retrieves a list of open or closed work items that were started by the user.

GET /scr/api/WorkList

Description

Use this method to retrieve a list of open or closed work items that were started by the user.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
type	Query	The type of list to return with the following possible values: • `open` • `closed` If the value is not supported or if no value is provided, then the 400 response code is returned.	Yes	String
version	Query	The version of the requested API. The only allowed value is `20110917`.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
limit	Query	If this parameter is provided, then it sets the maximum number of tasks to return. Otherwise, the default value of 100 is used.	No	Integer
startFrom	Query	If this parameter is provided, then the returned list starts with the task at this index in the list (zero based). Use this parameter if the number of available tasks is higher than the number of the returned tasks in an earlier call.	No	Integer

Response

Example input

```
https://www.blueworkslive.com/scr/api/WorkList?type=closed&version=20110917
```

Example output

```
{
    "work": [],
    "more": false,
    "apps": [],
    "type": "closed",
    "version": "20110917"
}
```

Response properties

more   Contains one of the following possible values:
- `true` if there are more work instances on the server.
- `false` if the work entry includes all of the instances available.

type   The type of work instances included in the list with the following possible values:
- `open`
- `closed`

These values match the possible values for the **type** parameter passed to the request.

work   The array of work instances listed by `dueDate` in reverse chronological order. Items with the same `dueDate` are sorted by instance ID.

work.name
   The name given to the work instance when it was started.

work.id
> The ID of the work instance that the task is a part of.

work.status
> The status of the work instance contains one of the following possible values:
> - `Active`
> - `Completed`
> - `Cancelled`
> - `Overdue`

work.remainingTasks
> The number of tasks to be completed for the specific work instance.

work.dueDate
> Optional property. The due date for the task that has the smallest `dueDate` in the instance, that is the task that is due first. If the tasks in the instance don't have due dates, then this field is not present. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work.appId
> The ID of the related entry in applications array.

apps The array of application instances that list one entry for each unique `appId` value in the work array.

app.id The ID of application that the work instance was started from.

app.type
> The type of application contains one of the following possible values:
> - `Workflow`
> - `Checklist`

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	The required **type** parameter specification must be provided.
401	The user is not authorized to make the request.

Application list

Use this resource to retrieve a list of process applications for which the user is able to start instances.

Method summary

HTTP Method	Path	Description
GET	/scr/api/AppList	Retrieves a list of process applications that the user is able to start instances of.

GET /scr/api/AppList

Description

Use this method to retrieve a list of process applications for which the user is able to start instances.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
version	Query	The version of the requested API. The only allowed value is 20110917.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Name	Location	Description	Required	Type
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

```
https://ibm.blueworkslive.com/scr/api/AppList?version=20110917
```

Example output

```
{
  "more": false,
  "apps": [
    {
      "id": "31ea912ac7",
      "new": true,
      "processId": "31ea912ac3",
      "name": "develop process diagram",
      "type": "workflow"
    },
    {
      "id": "9bea79df89",
      "new": true,
      "processId": "9bea79df85",
      "name": "any process app",
      "type": "checklist"
    },
    {
      "id": "1000062ea88cacf",
      "new": true,
      "processId": "1000062ea88cacb",
      "name": "test due date in email",
      "type": "checklist"
    },
    {
      "id": "10000d6400c755d",
      "new": true,
      "processId": "10000d6400c7558",
      "name": "test notification email",
      "type": "workflow"
    },
    {
      "id": "1000062ea88f4cc",
      "new": true,
      "processId": "1000062ea88f4c8",
      "name": "test date on task",
```

```
            "type": "workflow"
          },
          {
            "id": "3f37005d",
            "new": true,
            "processId": "3f370019",
            "name": "Submit Process for Review",
            "type": "workflow"
          },
          {
            "id": "100004e3fe76b25",
            "new": true,
            "processId": "100004e3fe76b20",
            "name": "testing 2",
            "type": "workflow"
          },
          {
            "id": "400004cea8bdab9",
            "new": true,
            "processId": "400004cea8bd626",
            "name": "test due date in email",
            "type": "checklist"
          },
          {
            "id": "400004cea8bdacb",
            "new": true,
            "processId": "400004cea8bd625",
            "name": "test date on task",
            "type": "workflow"
          }
      ],
      "favorites": [],
      "version": "20110917"
    }
```

Response properties

version

The version of the API used to create the response.

more The value is `true` if there are more application definition instances on the server, and `false` if the application entry includes all of the definitions available.

apps The array of application instances listed in alphabetical order by name.

app.name

The name of the application that the work instance was started from.

app.id The ID of the application that the work instance was started from.

app.processId

The ID of the process that is related to this application. Use this property to match news feed entries to an application.

app.type

The type of application with two possible values, `workflow` or `checklist`.

app.new

This property is present only if the value is `true`. It indicates a process application that the user did not see before.

favorites

The array of applications that the user selected as favorites. The order matches the one provided in the previous call to `UpdateFavorites`. This list might include application IDs that are not returned in application array. The application ID is not returned in application array when a user tags an application as favorite, but later the application becomes unshared. The ID is included in the favorites list, so the caller can access it by including it as part of a call to `UpdateFavorites`.

favorites.id

The ID of the favorite process application. An object with this ID appears in the application array.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

User list for work

Use this resource to return a list of the users to whom the tasks can be assigned.

Method summary

HTTP Method	Path	Description
GET	/scr/api/UserListForWork	Returns a list of the users to whom the tasks can be assigned.

GET /scr/api/UserListForWork

Description

Use this method to return a list of the users to whom the tasks can be assigned.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/UserListForWork
```

Example output

```
{
  "users":[
    {
      "name":"user name 1",
      "id":"7000f",
      "avatarId":"70010",
      "license":"Editor"
    },
    {
      "name":"user name 2",
      "id":"70022",
      "license":"Contributor"
    },
    {
      "name":"user name 3",
```

```
                                "id":"70020",
                                "license":"Editor",
                                "avatarId":"7612738"
                        }
                ]
        }
```

Response properties

users   The array of user objects.

user.name
    The full name of the user.

user.id
    The ID of the user.

user.avatarId
    The avatar ID of the user. This field is omitted if the user
    doesn't have an avatar.

user.license
    Contains one of the following possible values:

    - `Editor` if the user is an editor
    - `Contributor` if the user is a contributor

    The `Community` and the `Viewer` users are not returned
    because they can't get tasks assigned to them.

Response messages

HTTP code	Reason
200	The request was completed successfully.
401	The user is not authorized to make the request.

Launch instance

Use this resource to start a new work instance from an application definition.

Method summary

HTTP Method	Path	Description
POST	/scr/api/LaunchInstance	Starts a new work instance from an application definition.

POST /scr/api/LaunchInstance

Description
    Use this method to start a new work instance from an application
    definition.

Resource information

Requirements	Description
Response format	JSON

Requirements	Description
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
`instance`	Query	The definition for the instance to create. The instance format is shown in the example.	Yes	See Instance parameters below
`version`	Query	The version of the requested API. The only allowed value is `20110917`.	Yes	String
`account`	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
`accountId`	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Instance parameters

Name	Location	Description	Required	Type
`app.subject`	Query	The subject for the new work instance.	Yes	String
`app.id`	Query	The ID of the application.	Yes	String
`app.details`	Query	The details for the new work instance.	No	String
`app.mailOnCompletion`	Query	The notification email setting. Set to `true` to send a notification email when the work item is completed, or set to `false` to omit sending an email.	No	Boolean
`app.timeZoneId`	Query	The timezone to start the instance in. If the timezone is not provided, then the value from the user's current preferences is used. If user's timezone preference is also not set, then the server default is used.	No	String
`app.tasks`	Query	The array of tasks associated with the instance. This task can be either an existing one from an `AppDetails` response, in which case the ID property must be provided, or a new task without an ID parameter.	Yes	JSON array of tasks
`app.tasks.name`	Query	The name of the task.	Yes	String

Name	Location	Description	Required	Type
`app.tasks.id`	Query	The ID of the task. You must provide the ID for a task that is a part of the application definition. You don't need to provide the ID for a new task.	No	String
`app.tasks.assignedTo`	Query	The ID of the user to whom the task is assigned.	Yes	String
`app.tasks.dueDateExpression`	Query	The expression that specifies the due date must be in the following format: `yyyy/mm/dd`.	No	String
`app.tasks.approvalStep`	Query	Set the value to `true` for an approval step, and omit the parameter or set the value to `false` for a step that is not an approval.	No	Boolean

Name	Location	Description	Required	Type
app.attachments	Query	The array of attachments to associate with this instance. These attachments can be either originally attached to the application definition, or they might be newly created just for this instance. Each attachment is identified by its ID; the ID information can be retrieved from the `AppDetails` attachments array or returned from a file upload call. Provide an empty array if you are not attaching files.	Yes	JSON array of attachments
app.attachments.id	Query	The ID of the attachment.	Yes, if an attachment is provided	String

Response

Example input

```
curl -i --location-trusted -g --user admin:password -H "Version:1.0" -H "Content-Typ
"https://www.blueworkslive.com/scr/api/LaunchInstance?version=20110917&instance={\"ap
```

Example output

```
{
    "work":{
        "runtimeVersion":[
            "ede1c9df-61be-4774-99ab-196ec6742d57",
            "5f501e134ae25",
            1
        ],
        "details":"my sample details",
        "canManageWork":true,
        "name":"ProcessApp with multiple tasks A - New for test.",
        "subject":"New for test.",
        "tasks":[
            {
                "status":"NotStarted",
                "assignedTo":"7f0002",
                "name":"task 1",
                "id":"5f501e134ae2f",
                "approvalStep":false
            }
        ],
```

```
        "attachments":[

        ],
        "status":"Active",
        "comments":[

        ],
        "startedAt":1441218212876,
        "startedBy":"7f0002",
        "id":"5f501e134ae25",
        "appId":"5f500e12a00ee"
    },
    "app":{
        "detailsTitle":null,
        "type":"workflow",
        "name":"ProcessApp with multiple tasks A",
        "id":"5f500e12a00ee",
        "subjectTitle":null
    },
    "version":"20110917",
    "users":[
        {
            "avatarId":"5f500e1260203",
            "name":"admin",
            "id":"7f0002"
        }
    ]
}
```

Response properties

version
> The version of the API used to create the response.

work The work object.

work.runtimeVersion
> The array that identifies the version of the work instance.
> Treat this value as an opaque data item. The
> `work.runtimeVersion` is required for the WorkAction.

work.name
> The name to display for the work instance.

work.id
> The ID of the work instance.

work.subject
> The subject entered by the user who created the instance.

work.details
> The details entered by the user who created the instance.

work.status
> The status of work includes one of the following values:
> - `Active`
> - `Completed`
> - `Cancelled`
> - `Overdue`

work.startedBy
> The ID of the user who created the instance. An object with
> this ID appears in the user's array.

work.startedAt

The date and time when the instance was started. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work.completedBy

The ID of the user who completed or canceled the instance. Only present if the status is `Completed` or `Cancelled`. An object with this ID appears in the user's array.

work.completedAt

The date and time when the instance was completed or canceled. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC). Only present if the status is `Completed` or `Cancelled`.

work.appId

The ID of application from which the instance was started. This application appears in the application object.

work.canManageWork

Indicates whether the user has permissions to cancel work, change subject, or reassign tasks.

work.tasks

The array of tasks associated with the instance.

work.tasks.name

The name of the task.

work.tasks.id

The ID of the task.

work.tasks.assignedTo

The ID of user to whom the task is assigned. An object with this ID appears in the user's array.

work.tasks.actioner

The ID of the user who is not the task owner and who created the last change in the task status. An object with this ID appears in the user's array. When a task is reassigned, the `work.tasks.actioner` refers to the ID of the user who did the reassigning, while the `assignedTo` user is the user to whom the task was originally assigned.

work.tasks.dueDate

The date when the task is due. This property is present only if a `dueDate` is set for the task.

work.tasks.status

Contains one of the following possible values:

- `Pending`
- `NotStarted`
- `InProgress`
- `Completed`
- `Reassigned`

work.tasks.reassignmentReason

This property is present only if the status is `Reassigned`. Contains one of the following possible values:

- `TaskReassigned`
- `OwnerDowngraded`
- `OwnerArchived`

work.tasks.approvalStep

The value is either `true` or `false`.

work.tasks.isApproved

If the task is an approval step, that is `approvalStep` is `true`, and the task is completed, that is the `status` value is `Completed`, then `isApproved` is present, and the value is `true` or `false`.

work.attachments

The array of attachments associated with the instance or the application it was started from.

work.attachments.name

The name of the attachment.

work.attachments.id

The ID of the attachment.

work.attachments.uploadedBy

The ID of the user who uploaded the attachment. An object with this ID appears in the users array.

work.attachments.uploadedOn

The date and time when the attachment was uploaded. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work.attachments.size

The size of the attachment in bytes.

work.attachments.type

The type of attachment file with the associated numeric value.

Attachment type	Description
1	Generic document. Used if the file doesn't fall under one of the other values.
2	Microsoft Excel file
3	Image
4	PDF document
5	Microsoft PowerPoint document
6	Microsoft Project document
7	Microsoft Visio document
8	Microsoft Word document

work.attachments.canDelete

Indicates whether the user has permissions to delete the attachment.

work.attachments.deletedOn

The date and time when the attachment was deleted. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC). This field is present only if the file was deleted from the account through the admin pages. In this case, you can't download the file using `FileDownload`.

work.attachments.deletedBy

The name of the user who deleted the attachment. This field is present only if the file was deleted from the account through the admin pages.

work.comments

The array of comments on the instance.

work.comments.from

The ID of the user who created the comment. An object with this ID appears in the user's array.

work.comments.id

The ID of the comment.

work.comments.commentDate

The date when the comment was created.

work.comments.text

The text of the comment.

work.comments.canDelete

Indicates whether the user can delete the comment.

work.comments.replies

The list of replies to the parent comment.

app   The information about the application that the instance was started from.

app.name

The name of the application that the work instance was started from.

app.id The ID of the application from that the work instance was started from.

app.subjectTitle

The title for edit area where the user entered a subject for the instance.

app.detailsTitle

The title for edit area where the user entered details for the instance.

app.type

The type of the application that contains one of the following possible values:
- `workflow`
- `checklist`

users The array of users whose IDs are referenced from other locations in the JSON.

users.name
>>> The full name of the user.

users.id
>>> The ID of the user.

users.avatarId
>>> The ID of the avatar the user set. This property is present only if the user set an avatar.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Work action

Use this resource to complete an action in a work item.

Method summary

HTTP Method	Path	Description
POST	/scr/api/WorkAction	Completes an action in a work item.

POST /scr/api/WorkAction

Description
>>> Use this method to complete an action in a work item.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
`action`	Query	The information for the action to complete. The format for this parameter is shown in the example. Depending on the requested action type, some of the fields are required.	Yes	See Action parameters below
`version`	Query	The version of the requested API. The only allowed value is `20110917`.	Yes	String
`account`	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
`accountId`	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Action parameters

Name	Description	Data type
`action.subject`	The subject for a new work instance.	String
`action.runtimeVersion`	The array of values provided by the `WorkDetail`.	String
`action.workId`	The ID of the work item to complete the action on.	String
`action.appId`	The ID of the application this work relates to.	String
`action.timeZoneId`	The timezone to start the instance in. If the timezone isn't provided, then the value from the user's current preferences is used. If user's timezone preference is also not set, then the server default is used.	String

Name	Description	Data type
`action.type`	The type of action to complete. This parameter determines what extra fields are required. The following action types are available: **COMPLETE** Complete the current task. **APPROVE** Approve current task. Only valid if the task is an approval step. Requires the `action.task.id` parameter. **REJECT** Reject current task. Only valid if the task is an approval step. Requires the `action.task.id` parameter. **ADD_TASK** Complete current task and add a new one. Requires the `action.task.id`, `action.task.assignedTo`, `action.task.name`, `action.task.approvalStep` (optional), and `action.task.dueDateExpression` (optional) parameters. **REASSIGN** Reassign task. Requires the `action.task.id` and `action.task.assignedTo` parameters. **CHANGE_SUBJECT** Change work subject. Requires the `action.subject` parameter. **CANCEL_INSTANCE** Cancel work item. **REMOVE_ATTACHMENT** Remove attachment from the instance. Requires the `action.attachmentId` parameter.	String

Name	Description	Data type
`action.subject`	Required by the CHANGE_SUBJECT action. New subject for the work item.	String
`action.attachmentId`	Required by the REMOVE_ATTACHMENT action. The ID of the attachment to remove from the instance.	String
`action.task.id`	Required by the ADD_TASK, REASSIGN, APPROVE, COMPLETE, and REJECT actions. The ID of the task that the action is completed on. If an ADD_TASK action is completed, then this task ID is completed and a new one is created.	String
`action.task.assignedTo`	Required by the REASSIGN and ADD_TASK actions. The ID of the user to whom this task is assigned.	String
`action.task.name`	Required by the ADD_TASK action. The name of the task.	String
`action.task.approvalStep`	Optional for the ADD_TASK action, and only needed for work items of workflow type. Indicates whether the task is an approval step. It is assumed to have a value of false if the parameter is not found.	Boolean
`action.task.dueDateExpression`	Optional for the ADD_TASK action. Indicates the due date for this task, which must be in the YYYY/MM/DD format.	String

Response

Example input

```
curl -i --location-trusted -g --user admin:password -H "Version:1.0" -H "Content-T
"https://www.blueworkslive.com/scr/api/WorkAction?version=20110917&action={\"actic
```

Action parameter examples

```
{
    "action": {
      "runtimeVersion":[
        "037ee4bb-30c8-40e7-b083-0946e750f889",
        "b0059",
        0
      ],
      "workId":"b0059",
      "appId":"87654",
      "type":"CANCEL_INSTANCE",
      "timeZoneId":"America/Los_Angeles"
  }
}
{
    "action": {
      "runtimeVersion":[
        "037ee4bb-30c8-40e7-b083-0946e750f889",
```

```
            "b0059",
             0
          ],
          "attachmentId":"32345",
          "workId":"b0059",
          "appId":"87654",
          "type":"REMOVE_ATTACHMENT",
          "timeZoneId":"America/Los_Angeles"
   }
{
    "action": {
       "runtimeVersion":[
          "037ee4bb-30c8-40e7-b083-0946e750f889",
          "b0059",
           0
        ],
        "workId":"b0059",
        "appId":"453c78",
        "type":"APPROVE", /* COMPLETE or REJECT */
        "timeZoneId":"America/Los_Angeles",
        "task": {
          "id":"947593"
        }
   }
{
    "action": {
       "runtimeVersion":[
          "037ee4bb-30c8-40e7-b083-0946e750f889",
          "b0059",
           0
        ],
        "workId":"b0059",
        "appId":"98765",
        "type":"ADD_TASK",
        "timeZoneId":"America/Los_Angeles",
        "task": {
          "id":"", /* Id of existing task to close and replace with this one */
          "assignedTo":"94837",
          "name":"New task",
          "dueDateExpression":"2011/09/12",
          "approvalStep":true
        }
   }
{
    "action": {
       "runtimeVersion":[
          "037ee4bb-30c8-40e7-b083-0946e750f889",
          "b0059",
           0
        ],
        "workId":"b0059",
        "appId":"c67890",
        "type":"REASSIGN",
        "timeZoneId":"America/Los_Angeles",
        "task": {
          "id":"97c456",
          "assignedTo":"8654a"
        }
   }
{
    "action": {
       "runtimeVersion":[
          "037ee4bb-30c8-40e7-b083-0946e750f889",
          "b0059",
           0
        ],
```

```
                                "workId":"b0059",
                                "appId":"9a3456",
                                "type":"CHANGE_SUBJECT",
                                "timeZoneId":"America/Los_Angeles",
                                "subject":"New subject for this work."
                        }
```

Example output

```
*same as WorkDetail*

{
    "work":{
        "runtimeVersion":[
            "ede1c9df-61be-4774-99ab-196ec6742d57",
            "5f501e134c6bf",
            2
        ],
        "details":"my sample details",
        "canManageWork":true,
        "name":"ProcessApp 1 approval task - New for test.",
        "subject":"New for test.",
        "tasks":[
            {
                "isApproved":false,
                "status":"Completed",
                "assignedTo":"7f0002",
                "name":"toBeApproved\/Rejected",
                "id":"5f501e134c6ce",
                "approvalStep":true
            },
            {

                "status":"NotStarted",
                "dueDate":2554358399278,
                "assignedTo":"7f0002",
                "name":"New task.",
                "id":"5f501e134c6e2",
                "approvalStep":false
            }
        ],
        "attachments":[
            {
                "size":3.0,
                "uploadedBy":"7f0002",
                "type":9,
                "name":"foo.txt",
                "id":"5f500e1349358",
                "canDelete":true,
                "uploadedOn":1441206442048
            }
        ],
        "status":"Active",
        "comments":[

        ],
        "startedAt":1441219222958,
        "startedBy":"7f0002",
        "id":"5f501e134c6bf",
        "appId":"5f500e1349338"
    },
    "app":{
        "detailsTitle":null,
        "type":"workflow",
        "name":"ProcessApp 1 approval task",
        "id":"5f500e1349338",
        "subjectTitle":null
    },
    "version":"20110917",
```

```
        "users":[
            {
                "avatarId":"5f500e1260203",
                "name":"admin",
                "id":"7f0002"
            }
        ]
    }
}
```

Response properties

version

The version of the API used to create the response.

work The work object.

work.runtimeVersion

The array that identifies the version of the work instance. Treat this value as an opaque data item. The `work.runtimeVersion` is required for the WorkAction.

work.name

The name to display for the work instance.

work.id

The ID of the work instance.

work.subject

The subject entered by the user who created the instance.

work.details

The details entered by the user who created the instance.

work.status

The status of work includes one of the following values:

- `Active`
- `Completed`
- `Cancelled`
- `Overdue`

work.startedBy

The ID of the user who created the instance. An object with this ID appears in the user's array.

work.startedAt

The date and time when the instance was started. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work.completedBy

The ID of the user who completed or canceled the instance. Only present if the status is `Completed` or `Cancelled`. An object with this ID appears in the user's array.

work.completedAt

The date and time when the instance was completed or canceled. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC). Only present if the status is `Completed` or `Cancelled`.

work.appId
> The ID of application from which the instance was started. This application appears in the application object.

work.canManageWork
> Indicates whether the user has permissions to cancel work, change subject, or reassign tasks.

work.tasks
> The array of tasks associated with the instance.

work.tasks.name
> The name of the task.

work.tasks.id
> The ID of the task.

work.tasks.assignedTo
> The ID of user to whom the task is assigned. An object with this ID appears in the user's array.

work.tasks.actioner
> The ID of the user who is not the task owner and who created the last change in the task status. An object with this ID appears in the user's array. When a task is reassigned, the `work.tasks.actioner` refers to the ID of the user who did the reassigning, while the `assignedTo` user is the user to whom the task was originally assigned.

work.tasks.dueDate
> The date when the task is due. This property is present only if a `dueDate` is set for the task.

work.tasks.status
> Contains one of the following possible values:
> - `Pending`
> - `NotStarted`
> - `InProgress`
> - `Completed`
> - `Reassigned`

work.tasks.reassignmentReason
> This property is present only if the status is `Reassigned`. Contains one of the following possible values:
> - `TaskReassigned`
> - `OwnerDowngraded`
> - `OwnerArchived`

work.tasks.approvalStep
> The value is either `true` or `false`.

work.tasks.isApproved
> If the task is an approval step, that is `approvalStep` is `true`, and the task is completed, that is the `status` value is `Completed`, then `isApproved` is present, and the value is `true` or `false`.

work.attachments
> The array of attachments associated with the instance or the application it was started from.

work.attachments.name
> The name of the attachment.

work.attachments.id
> The ID of the attachment.

work.attachments.uploadedBy
> The ID of the user who uploaded the attachment. An object with this ID appears in the users array.

work.attachments.uploadedOn
> The date and time when the attachment was uploaded. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work.attachments.size
> The size of the attachment in bytes.

work.attachments.type
> The type of attachment file with the associated numeric value.

Attachment type	Description
1	Generic document. Used if the file doesn't fall under one of the other values.
2	Microsoft Excel file
3	Image
4	PDF document
5	Microsoft PowerPoint document
6	Microsoft Project document
7	Microsoft Visio document
8	Microsoft Word document

work.attachments.canDelete
> Indicates whether the user has permissions to delete the attachment.

work.attachments.deletedOn
> The date and time when the attachment was deleted. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC). This field is present only if the file was deleted from the account through the admin pages. In this case, you can't download the file using `FileDownload`.

work.attachments.deletedBy
> The name of the user who deleted the attachment. This field is present only if the file was deleted from the account through the admin pages.

work.comments
> The array of comments on the instance.

work.comments.from
> The ID of the user who created the comment. An object with this ID appears in the user's array.

work.comments.id
> The ID of the comment.

work.comments.commentDate
> The date when the comment was created.

work.comments.text
> The text of the comment.

work.comments.canDelete
> Indicates whether the user can delete the comment.

work.comments.replies
> The list of replies to the parent comment.

app The information about the application that the instance was started from.

app.name
> The name of the application that the work instance was started from.

app.id The ID of the application from that the work instance was started from.

app.subjectTitle
> The title for edit area where the user entered a subject for the instance.

app.detailsTitle
> The title for edit area where the user entered details for the instance.

app.type
> The type of the application that contains one of the following possible values:
> - `workflow`
> - `checklist`

users The array of users whose IDs are referenced from other locations in the JSON.

users.name
> The full name of the user.

users.id
> The ID of the user.

users.avatarId
> The ID of the avatar the user set. This property is present only if the user set an avatar.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Work comment

Use this resource to add and remove comments in a work instance.

Method summary

HTTP Method	Path	Description
POST	/scr/api/WorkComment	Adds and removes comments in a work instance.

POST /scr/api/WorkComment

Description

Use this method to add and remove comments in a work instance. Use the Post comment call to add comments to activities in required blueprints.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
action	Query	The allowed values are add and remove.	Yes	String
workId	Query	The ID of the work item that the comment is added to.	Yes, for an add operation	String
parentCommentId	Query	Indicates whether the comment is a reply to another comment. The parentCommentId must match the comment in the provided workId and must also be a top-level comment (replies to replies are not allowed).	No, for an add operation	String
comment	Query	The text for the added comment. The length limit for the text is 200 characters.	Yes, for an add operation	String

Name	Location	Description	Required	Type
`commentId`	Query	The ID of the removed comment.	Yes, for a remove operation	String
`version`	Query	The version of the requested API. The only allowed value is 20110917.	Yes	String
`account`	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String
`accountId`	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a `JSESSIONID` cookie from a previous successful call.	No	String

Response

Example input

Add a comment *asdf* to the work item with *a0713* ID :
```
/scr/api/WorkComment?version=20110917&action=add&workId=a0713
&comment=asdf
```

```
curl -i --location-trusted -g --user admin:password -H "Version:1.0" -H "Content-T
"https://www.blueworkslive.com/scr/api/WorkComment?version=20110917&action=add&wor
```

Example output
```
{
    "version":"20110917",
    "action":"add",
    "commentId","234ac0"
}
```

Response properties

version
> The version of the API used to create a response.

action The action completed in the call.

commentId
> The ID of the added or removed comment.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.

Work detail

Use this resource to retrieve information about a particular piece of work.

Method summary

HTTP Method	Path	Description
GET	/scr/api/WorkDetail	Retrieves information about a particular piece of work.

GET /scr/api/WorkDetail

Description
> Use this method to retrieve information about a particular piece of work.

Resource information

Requirements	Description
Response format	JSON
Requires authentication	Yes
Rate limited	Not yet

Parameters

Name	Location	Description	Required	Type
version	Query	The version of the requested API. The only allowed value is 20110917.	Yes	String

Name	Location	Description	Required	Type
workId	Query	The ID of the piece of work you want to retrieve.	Yes	String
account	Query	The name of the account to access. Together with the user name and password that are provided by basic authentication, the account name is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String
accountId	Query	The ID of the account to access. Together with the user name and password that are provided by basic authentication, the account ID is used to check whether the access is allowed. Alternatively, use a JSESSIONID cookie from a previous successful call.	No	String

Response

Example input

```
https://www.blueworkslive.com/scr/api/WorkDetail?workId=a0713&version=20110917
```

Example output

```
{
    "work":{
        "runtimeVersion":[
            "ede1c9df-61be-4774-99ab-196ec6742d57",
            "5f501e134c6bf",
            2
        ],
        "details":"my sample details",
        "canManageWork":true,
        "name":"ProcessApp 1 approval task - New for test.",
```

```
            "subject":"New for test.",
            "tasks":[
                {
                    "isApproved":false,
                    "status":"Completed",
                    "assignedTo":"7f0002",
                    "name":"toBeApproved\/Rejected",
                    "id":"5f501e134c6ce",
                    "approvalStep":true
                },
                {
                    "status":"NotStarted",
                    "dueDate":2554358399278,
                    "assignedTo":"7f0002",
                    "name":"New task.",
                    "id":"5f501e134c6e2",
                    "approvalStep":false
                }
            ],
            "attachments":[
                {
                    "size":3.0,
                    "uploadedBy":"7f0002",
                    "type":9,
                    "name":"foo.txt",
                    "id":"5f500e1349358",
                    "canDelete":true,
                    "uploadedOn":1441206442048
                }
            ],
            "status":"Active",
            "comments":[

            ],
            "startedAt":1441219222958,
            "startedBy":"7f0002",
            "id":"5f501e134c6bf",
            "appId":"5f500e1349338"
        },
        "app":{
            "detailsTitle":null,
            "type":"workflow",
            "name":"ProcessApp 1 approval task",
            "id":"5f500e1349338",
            "subjectTitle":null
        },
        "version":"20110917",
        "users":[
            {
                "avatarId":"5f500e1260203",
                "name":"admin",
                "id":"7f0002"
            }
        ]
    }
```

Response properties

version
> The version of the API used to create the response.

work The work object.

work.runtimeVersion
> The array that identifies the version of the work instance.
> Treat this value as an opaque data item. The
> work.runtimeVersion is required for the WorkAction.

work.name
>The name to display for the work instance.

work.id
>The ID of the work instance.

work.subject
>The subject entered by the user who created the instance.

work.details
>The details entered by the user who created the instance.

work.status
>The status of work includes one of the following values:
>- `Active`
>- `Completed`
>- `Cancelled`
>- `Overdue`

work.startedBy
>The ID of the user who created the instance. An object with this ID appears in the user's array.

work.startedAt
>The date and time when the instance was started. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work.completedBy
>The ID of the user who completed or canceled the instance. Only present if the status is `Completed` or `Cancelled`. An object with this ID appears in the user's array.

work.completedAt
>The date and time when the instance was completed or canceled. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC). Only present if the status is `Completed` or `Cancelled`.

work.appId
>The ID of application from which the instance was started. This application appears in the application object.

work.canManageWork
>Indicates whether the user has permissions to cancel work, change subject, or reassign tasks.

work.tasks
>The array of tasks associated with the instance.

work.tasks.name
>The name of the task.

work.tasks.id
>The ID of the task.

work.tasks.assignedTo
>The ID of user to whom the task is assigned. An object with this ID appears in the user's array.

work.tasks.actioner

The ID of the user who is not the task owner and who created the last change in the task status. An object with this ID appears in the user's array. When a task is reassigned, the `work.tasks.actioner` refers to the ID of the user who did the reassigning, while the `assignedTo` user is the user to whom the task was originally assigned.

work.tasks.dueDate

The date when the task is due. This property is present only if a `dueDate` is set for the task.

work.tasks.status

Contains one of the following possible values:
- `Pending`
- `NotStarted`
- `InProgress`
- `Completed`
- `Reassigned`

work.tasks.reassignmentReason

This property is present only if the status is `Reassigned`. Contains one of the following possible values:
- `TaskReassigned`
- `OwnerDowngraded`
- `OwnerArchived`

work.tasks.approvalStep

The value is either `true` or `false`.

work.tasks.isApproved

If the task is an approval step, that is `approvalStep` is `true`, and the task is completed, that is the `status` value is `Completed`, then `isApproved` is present, and the value is `true` or `false`.

work.attachments

The array of attachments associated with the instance or the application it was started from.

work.attachments.name

The name of the attachment.

work.attachments.id

The ID of the attachment.

work.attachments.uploadedBy

The ID of the user who uploaded the attachment. An object with this ID appears in the users array.

work.attachments.uploadedOn

The date and time when the attachment was uploaded. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC).

work.attachments.size

The size of the attachment in bytes.

work.attachments.type

The type of attachment file with the associated numeric value.

Attachment type	Description
1	Generic document. Used if the file doesn't fall under one of the other values.
2	Microsoft Excel file
3	Image
4	PDF document
5	Microsoft PowerPoint document
6	Microsoft Project document
7	Microsoft Visio document
8	Microsoft Word document

work.attachments.canDelete

Indicates whether the user has permissions to delete the attachment.

work.attachments.deletedOn

The date and time when the attachment was deleted. The value is expressed in milliseconds since epoch, that is 1 January 1970 at midnight Coordinated Universal Time (UTC). This field is present only if the file was deleted from the account through the admin pages. In this case, you can't download the file using `FileDownload`.

work.attachments.deletedBy

The name of the user who deleted the attachment. This field is present only if the file was deleted from the account through the admin pages.

work.comments

The array of comments on the instance.

work.comments.from

The ID of the user who created the comment. An object with this ID appears in the user's array.

work.comments.id

The ID of the comment.

work.comments.commentDate

The date when the comment was created.

work.comments.text

The text of the comment.

work.comments.canDelete

Indicates whether the user can delete the comment.

work.comments.replies

The list of replies to the parent comment.

app   The information about the application that the instance was started from.

app.name
> The name of the application that the work instance was started from.

app.id The ID of the application from that the work instance was started from.

app.subjectTitle
> The title for edit area where the user entered a subject for the instance.

app.detailsTitle
> The title for edit area where the user entered details for the instance.

app.type
> The type of the application that contains one of the following possible values:
> - `workflow`
> - `checklist`

users The array of users whose IDs are referenced from other locations in the JSON.

users.name
> The full name of the user.

users.id
> The ID of the user.

users.avatarId
> The ID of the avatar the user set. This property is present only if the user set an avatar.

Response messages

HTTP code	Reason
200	The request was completed successfully.
400	Bad request. Some required parameters were missing or contained invalid values.
401	The user is not authorized to make the request.
404	The specified piece of work was not found.