



Synology Download Station

Official API



Synology Inc.
© 2013 Synology Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Synology Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Synology's copyright notice.

The Synology logo is a trademark of Synology Inc.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Synology retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Synology-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Synology is not responsible for typographical errors.

Synology Inc.
3F-3, No. 106, Chang-An W.
Rd. Taipei 103, Taiwan

Synology and the Synology logo are trademarks of Synology Inc., registered in the United States and other countries.

Marvell is registered trademarks of Marvell Semiconductor, Inc. or its subsidiaries in the United States and other countries.

Freescale is registered trademarks of Freescale Semiconductor, Inc. or its subsidiaries in the United

States and other countries.

Other products and company names mentioned herein are trademarks of their respective holders.

Even though Synology has reviewed this document, SYNOLOGY MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY. IN NO EVENT WILL SYNOLOGY BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Synology dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Table of Contents

Chapter 1: Introduction

Chapter 2: Get Started

API Workflow	5
Making Requests	6
Parsing Response	7
Common Error Code	8
Working Example.....	8

Chapter 3: Base API

API List	12
SYNO.API.Info	12
SYNO.API.Auth.....	14

Chapter 4: Download Station API

API List	16
SYNO.DownloadStation.Info	17
SYNO.DownloadStation.Schedule.....	20
SYNO.DownloadStation.Task.....	21
SYNO.DownloadStation.Statistic.....	30
SYNO.DownloadStation.RSS.Site	31
SYNO.DownloadStation.RSS.Feed	33

Appendix A: Download Task Status

Appendix B: Values for Details of Erroneous Tasks

Introduction

Welcome to the developer documentation for Synology's Download Station APIs. This official developer's guide allows you to build upon and extend your applications based on the APIs of Download Station, one of the most popular packages on Synology DiskStation, and interact with Download Station via HTTP request and response.

This document introduces the structure of Download Station APIs and details of API specifications. "Chapter 2: Get Started" describes the basic guidelines on how to use the APIs, which is suggested to read through before you jump right into the API spec. "Chapter 3: Base API" and "Chapter 4: Download Station API" list all available APIs and its detailed information.

Get Started

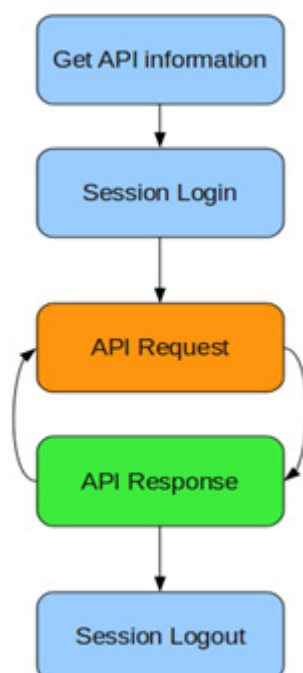
Before making use of Download Station APIs to develop your own applications, you need to have basic understanding of API concepts and API procedures.

This chapter explains how to execute and complete API process with five sections as follows whose explicit explanations are provided in order:

- **API Workflow:** Briefly introduces how to work with Download Station APIs.
- **Making Requests:** Elaborates how to construct API requests.
- **Parsing Response:** Describes how to parse response data.
- **Common Error Code:** Lists all common error codes that might be returned from all Download Station APIs.
- **Working Example:** Provides an example from a login session to request download task information.

API Workflow

The following introduces the four-step easy-to-follow workflow which guides you through the procedures of how to make your application interact with the Download Station APIs. The workflow is illustrated as a diagram bellow.



Step 1: Get API Information

First of all, your application needs to get API information from the target DiskStation to know which APIs are available for use on the target DiskStation. This information can be accessed simply through a request to query.cgi from the SYNO.API.Info API. The information provided in the response contains

available API name, API cgi path and the API version information. Once you have all the information at hand, your application can make further requests to all available APIs.

Step 2: Session Login

In order to make your application interact with Download Station, the application needs to log in a session with a DSM account and password. The login process is simply making a request to the SYNO.API.Auth API.

Step 3: Download API Requests

Once the account is successfully logged in, the application can start to make requests to all available Download APIs. In the next section “Making Requests”, instructions on how to form a valid API request and how to decode response information will be given.

Step 4: Session Logout

After finished with the steps above, the application can end the working session by making another request to SYNO.API.Auth API with the logout method.

Making Requests

There are five basic elements for constructing a valid request to any of the APIs.

- **API name:** Name of the API requested
- **version:** Version of the API requested
- **path:** cgi path of the API. The path information can be retrieved by requesting SYNO.API.Info
- **method:** Method requested
- **_sid:** Authorized session ID. Each API request should pass a sid value via either HTTP GET/POST “_sid” argument or “id” value in cookie.

and the syntax for the request is:

```
GET
/webapi/<CGI_PATH>?api=<API_NAME>&version=<VERSION>&method=<METHOD>[&<PARAMS>][&_sid=<SID>]
```

in which <PARAMS> represents the parameters for the requested method which is optional.

Here's an example. If you want to make a request to the SYNO.API.Info API version 1 with the query method on your DiskStation whose address is http://myds.com:port (default port for http is 5000 and 5001 for https) for the list of all available API methods, the corresponding parameters are:

API name: SYNO.API.Info
version: 1
path: query.cgi
method: query
params: query=all

and the request will look like this:

```
http://myds.com:port/webapi/query.cgi?api=SYNO.API.Info&version=1&method=query&query=all
```

Note that an API's path and supported version information can be acquired by sending a request to SYNO.API.Info. The only API with a fixed location is SYNO.API.Info itself so that you can always request SYNO.API.Info with /webapi/query.cgi.

Parsing Response

All API responses are encoded in the JSON format, and the JSON response contains elements as follows:

Key	Value	Description
success	true/false	"true" when the request finishes successfully, "false" when the request fails with an error code.
data	<object>	The data object contains all the response information described in each method.
error	Error code	An integer number error code will be returned when a request fails. There are two kinds of error codes: <i>common error codes</i> are error codes that are shared between all APIs; <i>API error codes</i> are API specific error codes that are listed under the corresponding API spec.

Following are examples for a failed request and successful request:

Example: Response to a request with invalid parameters

```
{
  "success":false,
  "error":101
}
```

Example: Response to a request for the package info of Download Station

```
{
  "success":true,
  "data": {
    "is_manager":true,
    "version":2269,
    "version_string":"3.2-2269"
  }
}
```

Note that to demonstrate examples with clarity, only the data object is included in the response examples given in the following sections.

Common Error Code

The codes listed below are general error codes for all Download Station APIs.

Code	Description
100	Unknown error
101	Invalid parameter
102	The requested API does not exist
103	The requested method does not exist
104	The requested version does not support the functionality
105	The logged in session does not have permission
106	Session timeout
107	Session interrupted by duplicate login

Working Example

The following demonstrates a working example for requesting a download task list from DiskStation. To implement the example, simply replace the DiskStation address used in the example with your DiskStation address and paste the URL to a browser. Then the JSON response will show up in a response page. Furthermore, you can find an online JSON parser to make the response more readable for your experiments.

Step 1: Get API Information

In order to make API requests, you should first request SYNO.API.Info to get the SYNO.API.Auth API info for session login and SYNO.DownloadStation.Task API info for download task list.

Request

```
http://myds.com:5000/webapi/query.cgi?api=SYNO.API.Info&version=1&method=query&query=SYNO.API.Auth,SYNO.DownloadStation.Task
```

Response

```
{
  "data": {
    "SYNO.API.Auth": {
      "path": "auth.cgi",
      "minVersion": 1,
      "maxVersion": 2
    },
    "SYNO.DownloadStation.Task": {
      "path": "DownloadStation/task.cgi",
      "minVersion": 1,
      "maxVersion": 1
    }
  },
  "success": true
}
```


Step 2: Session Login

After the SYNO.API.Auth path and supported version info are returned, you can log in a session by requesting Auth API version 1 located at /webapi/auth.cgi.

Request

```
http://myds.com:5000/webapi/auth.cgi?api=SYNO.API.Auth&version=2&method=login&account=admin&passwd=12345&session=DownloadStation&format=cookie
```

Response

```
{
  "data": {
    "sid": "ohOCjwhHhwghw"
  },
  "success": true
}
```

Step 3: Request Download API

After a session is logged in, you can continue to call the list method in SYNO.DownloadStation.Task for download task list. The cgi path and version are provided in the response of Step 1, and the list of all tasks can be requested by excluding the offset and limit parameters.

Request

```
http://myds.com:5000/webapi/DownloadStation/task.cgi?api=SYNO.DownloadStation.Task&version=1&method=list
```

Response

```
{
  "data": {
    "total": 3,
    "offset": 0,
    "tasks": [ {
      "id": "dbid_001",
      "type": "bt",
      "username": "admin",
      "title": "File 1",
      "size": "123456",
      "status": "downloading",
      "status_extra": null
    }, {
      "id": "dbid_002",
      "type": "http",
      "username": "bbb",
      "title": "File 2",
      "size": "654321",
      "status": "waiting",
      "status_extra": null
    }
  ]
}
```

```

    }, {
        "id": "dbid_003",
        "type": "ftp",
        "username": "user1",
        "title": "File 3",
        "size": "654321",
        "status": "finished",
        "status_extra": null
    }
  ],
  "success": true
}

```

From the response list, it can be observed that there are three tasks in Download Station. Let's say you are interested in the task whose task ID is `dbid_001` and want to know more details about it, so you make another request to the `getinfo` method. In this request, you will need to add the parameter `additional=detail,transfer` for the method to request detailed objects and transfer them in response.

Request

```

http://myds.com:5000/webapi/DownloadStation/task.cgi?api=SYNO.DownloadStation.Task&
version=1&method=getinfo&id=dbid_001&additional=detail,transfer

```

Response

```

{
  "data": {
    "tasks": [ {
      "id": "dbid_001",
      "type": "bt",
      "username": "admin",
      "title": "File 1",
      "size": "123456",
      "status": "downloading",
      "status_extra": null,
      "additional": {
        "detail": {
          "connected_leechers": 0,
          "connected_seeders": 0,
          "create_time": "1341210005",
          "destination": "Download",
          "priority": "auto",
          "total_peers": 0,
          "uri": "http://mp3.com/mix.torrent"
        },
        "transfer": {

```

```

        "size_downloaded": "54642",
        "size_uploaded": "435",
        "speed_download": "2605",
        "speed_upload": "0"
    }
}
}],
{
    "success": true
}

```

Step 4: Session Logout

When you are finished with the procedures or if you want to switch to another user, you should log out the current session. The session will be ended by calling the logout method in SYNO.API.Auth.

Example:

```

http://myds.com:5000/webapi/auth.cgi?api=SYNO.API.Auth&version=1&method=logout&session=DownloadStation

```

Base API

API List

The following table is the overview of two fundamental APIs defined in this chapter:

API Name	Description
SYNO.API.Info	Provides available API info.
SYNO.API.Auth	Performs session login and logout.

SYNO.API.Info

Overview

Availability: Since DSM 4.0

Version: 1

Method

Query

Request

Parameter	Description	Availability
query	API names concatenated by "," or use "ALL" to get all supported APIs.	1 and later

Example:

```
GET /webapi/query.cgi? api=SYNO.API.Info&version=1&method=query&query=ALL
```

Response

Contains API description objects.

Example:

```
{
  "SYNO.API.Info": {
    "path": "query.cgi",
    "minVersion": 1,
    "maxVersion": 1
  },
  "SYNO.API.Auth": {
    "path": "auth.cgi",
    "minVersion": 1,
    "maxVersion": 2
  },
}
```

```

"SYNO.DownloadStation.Schedule": {
    "path": "DownloadStation/schedule.cgi",
    "minVersion": 1,
    "maxVersion": 1
},
"SYNO.DownloadStation.Task": {
    "path": "DownloadStation/task.cgi",
    "minVersion": 1,
    "maxVersion": 1
},
"SYNO.DownloadStation.RSS.Site": {
    "path": "DownloadStation/RSSsite.cgi",
    "minVersion": 1,
    "maxVersion": 1
},
"SYNO.DownloadStation.RSS.Feed": {
    "path": "DownloadStation/RSSfeed.cgi",
    "minVersion": 1,
    "maxVersion": 1
},
"SYNO.DownloadStation.Statistic": {
    "path": "DownloadStation/statistic.cgi",
    "minVersion": 1,
    "maxVersion": 1
}
}

```

Response Objects

Here are API description objects:

Parameter	Description	Availability
key	API name	1 and later
path	API cgi path	1 and later
minVersion	Minimum API version supported	1 and later
maxVersion	Maximum API version supported	1 and later

API Error Code

No specific API error codes.

SYNO.API.Auth

Overview

Availability: Since DSM 4.0

Version: 2 (Since DSM 4.1)

Method

Login

Request

Parameter	Description	Availability
account	Login account name	1 and later
passwd	Login account password	1 and later
session	Login session name	1 and later
format	Returned format of session ID. Following are the two possible options and the default value is <code>cookie</code> . <code>cookie</code> : The login session ID will be set to cookie. <code>sid</code> : The login sid will only be returned as response json data and the cookie will not be set.	2 and later
otp_code	This option is not required to log into Download Station sessions currently. However, please note that DSM 4.2 and later includes a 2-step verification option. If enabled, the user requires a verification code to log into DSM sessions.	2 and later

Example:

```
GET /webapi/auth.cgi?
api=SYNO.API.Auth&version=2&method=login&account=admin&passwd=12345&session=
DownloadStation&format=sid
```

Response

Parameter	Description	Availability
sid	Authorized session ID. When the user log in with <code>format=sid</code> , cookie will not be set and each API request should provide a request parameter <code>sid=<sid></code> along with other parameters.	2 and later

Example:

```
{
  sid: "ohOCjwhHhwghw"
}
```

Logout

Request

Parameter	Description	Availability
session	Session name to be logged out	1 and later

Example:

```
GET /webapi/auth.cgi?  
api=SYNO.API.Auth&version=1&method=logout&session=DownloadStation
```

Response

No specific response. It returns an empty success response if completed without error.

API Error Code

Code	Description
400	No such account or incorrect password
401	Account disabled
402	Permission denied
403	2-step verification code required
404	Failed to authenticate 2-step verification code

Download Station API

API List

The following table is the overview of all Download Station APIs defined in this chapter. All Download Station APIs are required to log in with SYNO.API.Auth with session=DownloadStation.

API Name	Description
SYNO.DownloadStation.Info	Provides Download Station info and settings. Sets Download Station settings.
SYNO.DownloadStation.Schedule	Provides advanced schedule settings. Sets advanced schedule settings.
SYNO.DownloadStation.Task	Provides task listing and detailed task information. Performs task actions: create, delete, resume, pause.
SYNO.DownloadStation.Statistic	Provides total download/upload statistics.
SYNO.DownloadStation.RSS.Site	Provides RSS site listing. Refreshes RSS site.
SYNO.DownloadStation.RSS.Feed	Provides RSS feed listing.

SYNO.DownloadStation.Info

Overview

Availability: Since Download Station 3.2-2258

Version: 1

Method

GetInfo

Request

No parameters required.

Example:

```
GET /webapi/DownloadStation/info.cgi?
api=SYNO.DownloadStation.Info&version=1&method=getinfo
```

Response

Parameter	Type	Description	Availability
version	int	Build number of Download Station	1 and later
version_string	string	Full version string of Download Station	1 and later
is_manager	bool	If the logged in user is manager	1 and later

Example:

```
{
    "is_manager":true,
    "version":2269,
    "version_string":"3.2-2269"
}
```

GetConfig

Request

No parameters required.

Example:

```
GET /webapi/DownloadStation/info.cgi?
api=SYNO.DownloadStation.Info&version=1&method=getconfig
```

Response

Parameter	Description	Availability	Privilege
bt_max_download	Max BT download speed in KB/s ("0" means unlimited)	1 and later	admin only
bt_max_upload	Max BT upload speed in KB/s ("0" means unlimited)	1 and later	admin only
emule_max_download	Max eMule download speed in KB/s ("0" means unlimited)	1 and later	admin only
emule_max_upload	Max eMule upload speed in KB/s ("0" means unlimited)	1 and later	admin only
nzb_max_download	Max NZB download speed in KB/s ("0" means unlimited)	1 and later	admin only
http_max_download	Max HTTP download speed in KB/s ("0" means unlimited). For more info, please see <i>Limitations</i>	1 and later	admin only
ftp_max_download	Max FTP download speed in KB/s ("0" means unlimited). For more info, please see <i>Limitations</i> .	1 and later	admin only
emule_enabled	If eMule service is enabled	1 and later	admin only
unzip_service_enabled	If Auto unzip service is enabled for users except admin or administrators group	1 and later	admin only

Example:

```
{
  "bt_max_download":13,
  "bt_max_upload":11,
  "emule_enabled":true,
  "emule_max_download":13,
  "emule_max_upload":10,
  "ftp_max_download":0,
  "http_max_download":0,
  "nzb_max_download":10,
  "unzip_service_enabled":true
}
```

SetServerConfig

Request

Parameter	Description	Availability	Privilege
bt_max_download	Max BT download speed in KB/s ("0" means unlimited)	1 and later	admin only
bt_max_upload	Max BT upload speed in KB/s ("0" means unlimited)	1 and later	admin only
emule_max_download	Max eMule download speed in KB/s ("0" means unlimited)	1 and later	admin only
emule_max_upload	Max eMule upload speed in KB/s ("0" means unlimited)	1 and later	admin only
nzb_max_download	Max NZB download speed in KB/s ("0" means unlimited)	1 and later	admin only
http_max_download	Max HTTP download speed in KB/s ("0" means unlimited). For more info, please see <i>Limitations</i> .	1 and later	admin only
ftp_max_download	Max FTP download speed in KB/s ("0" means unlimited). For more info, please see <i>Limitations</i> .	1 and later	admin only
emule_enabled	If eMule service is enabled	1 and later	admin only
unzip_service_enabled	If Auto unzip service is enabled for users except admin or administrators group	1 and later	admin only

Example:

```
GET /webapi/DownloadStation/info.cgi?
api=SYNO.DownloadStation.Info&version=1&method=setserverconfig&bt_max_downlo
ad=10&emule_enabled=true
```

Response

No specific response. It returns an empty success response if completed without error.

API Error Code

No specific API error codes.

Limitations

1. Currently `http_max_download` and `ftp_max_download` share the same config value. When both parameters are requested to be set at the same time, the requested `ftp_max_download` rate will overwrite the requested `http_max_download` rate.

2. Considering identical rates may be simultaneously used by the other packages other than Download Station, the HTTP and FTP max download rates will not affect tasks whose current status is "downloading". New rates will only be applied to those newly added/resumed HTTP/FTP tasks.

SYNO.DownloadStation.Schedule

Overview

Availability: Since Download Station 3.2-2258

Version: 1

Method

GetConfig

Request

No parameters required.

Example:

```
GET /webapi/DownloadStation/schedule.cgi?
api=SYNO.DownloadStation.Schedule&version=1&method=getconfig
```

Response

Parameter	Description	Availability	Privilege
enabled	If download schedule is enabled	1 and later	admin only
emule_enabled	If eMule download schedule is enabled	1 and later	admin only

Example:

```
{
    "enabled":true,
    "emule_enabled":false
}
```

SetConfig

Request

Parameter	Description	Availability	Privilege
enabled	If download schedule is enabled	1 and later	admin only
emule_enabled	If eMule download schedule is enabled	1 and later	admin only

Example:

```
GET /webapi/DownloadStation/schedule.cgi?
api=SYNO.DownloadStation.Schedule&version=1&method=setconfig&enabled=true
```

Response

No specific response. It returns an empty success response if completed without error.

API Error Code

No specific API error codes.

SYNO.DownloadStation.Task

Overview

Availability: Since Download Station 3.2-2258

Version: 1

Method

List

Request

Parameter	Description	Availability
offset	Optional. Beginning task on the requested record. Default to "0".	1 and later
limit	Optional. Number of records requested: "-1" means to list all tasks. Default to "-1".	1 and later
additional	Optional. Additional requested info, separated by ",". When an additional option is requested, objects will be provided in the specified additional option. Possible options include: <ul style="list-style-type: none">■ detail: returns Task_Detail object■ transfer: returns Task_Transfer object■ file: returns Task_File object (BT only)■ tracker: returns Task_Tracker object (BT only)■ peer: returns Task_Peer object (BT only)	1 and later

Example:

```
GET /webapi/DownloadStation/task.cgi?
api=SYNO.DownloadStation.Task&version=1&method=list&additional=detail,file
```

Response

Parameter	Type	Description	Availability
total	int	Total number of records	1 and later
offset	int	Requests offset	1 and later
tasks	Array	Array of task objects	1 and later

Example:

```
{
  "total":12,
  "offset":10,
  "tasks": [{
    "id":"dbid_001",
    "type":"bt",
    "username":"admin",
    "title":"TOP 100 MIX",
    "size":"9427312332",
```

```

        "status": "downloading",
        "status_extra": null,
        "additional": {
            "detail": {
                "connected_leechers": 0,
                "connected_seeders": 0,
                "create_time": "1341210005",
                "destination": "Download",
                "priority": "auto",
                "total_peers": 0,
                "uri": "http://mp3.com/mix.torrent"
            },
            "file": [{
                "filename": "mix001.mp3",
                "priority": "normal",
                "size": "41835",
                "size_downloaded": "0"
            }, {
                "filename": "mix002.mp3",
                "priority": "normal",
                "size": "31689",
                "size_downloaded": "0"
            }]
        }
    }, {
        "id": "dbid_002",
        "type": "http",
        "username": "user1",
        "title": "short clip",
        "size": "112092412",
        "status": "finished",
        "status_extra": null,
        "additional": {
            "detail": {
                "connected_leechers": 0,
                "connected_seeders": 0,
                "create_time": "1356214565",
                "destination": "Download",
                "priority": "auto",
                "total_peers": 0,
                "uri": "http://mymovies.com/mv.avi"
            }
        }
    }
}

```

```
}]
}
```

GetInfo

Request

Parameter	Description	Availability
id	Task IDs, separated by ",".	1 and later
additional	<p>Optional. Additional requested info, separated by ",". When an additional option is requested, objects will be provided in the specified additional option.</p> <p>Possible options include:</p> <ul style="list-style-type: none"> ■ detail: returns <code>Task_Detail</code> object ■ transfer: returns <code>Task_Transfer</code> object ■ file: returns <code>Task_File</code> object (BT only) ■ tracker: returns <code>Task_Tracker</code> object (BT only) ■ peer: returns <code>Task_Peer</code> object (BT only) 	1 and later

Example:

```
GET /webapi/DownloadStation/task.cgi?
api=SYNO.DownloadStation.Task&version=1&method=getinfo&id=dbid_001,dbid_002
&additional=detail
```

Response

Parameter	Type	Description	Availability
tasks	Array	Array of Task objects.	1 and later

Example:

```
{
  "tasks": [{
    "id": "dbid_001",
    "type": "bt",
    "username": "admin",
    "title": "TOP 100 MIX",
    "size": "9427332",
    "status": "downloading",
    "status_extra": null,
    "additional": {
      "detail": {
        "connected_leechers": 0,
        "connected_seeders": 0,
        "create_time": "1341210005",
        "destination": "Download",
        "priority": "auto",
        "total_peers": 0,
```

```

        "uri":"http://mp3.com/mix.torrent"
    }
}
}, {
    "id":"dbid_002",
    "type":"http",
    "username":"user1",
    "title":"short clip",
    "size":"112092412",
    "status":"finished",
    "status_extra":null,
    "additional": {
        "detail": {
            "connected_leechers":0,
            "connected_seeders":0,
            "create_time":"1356214565",
            "destination":"Download",
            "priority":"auto",
            "total_peers":0,
            "uri":"http://mymovies.com/mv.avi"
        }
    }
}
}]
}

```

Create

Request

Parameter	Description	Availability
uri	Optional. Accepts HTTP/FTP/magnet/ED2K links, separated by ",".	1 and later
file	Optional. File uploading from client. For more info, please see <i>Limitations</i> on page 29.	1 and later
username	Optional. Login username	1 and later
password	Optional. Login password	1 and later
unzip_password	Optional. Password for unzipping download tasks	1 and later

Example:

```

POST /webapi/DownloadStation/task.cgi

api=SYNO.DownloadStation.Task&version=1&method=create&uri=ftps://192.0.0.1:2
1/test/test.zip&username=admin&password=123

```

Response

No specific response. It returns an empty success response if completed without error.

Delete

Request

Parameter	Description	Availability
id	Task IDs to be deleted, separated by ",".	1 and later
force_complete	Delete tasks and force to move uncompleted download files to the destination.	1 and later

Example:

```
GET /webapi/DownloadStation/task.cgi?
api=SYNO.DownloadStation.Task&version=1&method=delete&id=dbid_001,dbid_002
&force_complete=true
```

Response

Response is an array of response objects with following parameters.

Parameter	Description	Availability
id	Task IDs	1 and later
error	Action result. Error=0 for success.	1 and later

Example:

```
[{
    "error":405,
    "id":"dbid_001"
},{
    "error":0,"
    id":"dbid_002"
}]
```

Pause

Request

Parameter	Description	Availability
id	Task IDs to be paused, separated by ",".	1 and later

Example:

```
GET /webapi/DownloadStation/task.cgi?
api=SYNO.DownloadStation.Task&version=1&method=pause&id=dbid_001,dbid_002
```

Response

Response is an array of response objects with following parameters.

Parameter	Description	Availability
id	Task IDs	1 and later
error	Action result. Error=0 for success.	1 and later

Example:

```
[{
  "error":405,
  "id":"dbid_001"
},{
  "error":0,"
  id":"dbid_002"
}]
```

Resume

Request

Parameter	Description	Availability
id	Task IDs to be resumed, separated by ",".	1 and later

Example:

```
GET /webapi/DownloadStation/task.cgi?
api=SYNO.DownloadStation.Task&version=1&method=resume&id=dbid_001,dbid_002
```

Response

Response is an array of response objects with following parameters.

Parameter	Description	Availability
id	Task IDs	1 and later
error	Action result. Error=0 for success.	1 and later

Example:

```
[{
  "error":405,
  "id":"dbid_001"
},{
  "error":0,"
  id":"dbid_002"
}]
```

Response Objects

<Task Object> definition:

Member	Type	Description	Availability
id	string	Task ID	1 and later
type	string	Possible types: BT, NZB, http, ftp, eMule	1 and later
username	int	Task owner	1 and later
title	string	Task title	1 and later
size	string	Task size in bytes	1 and later
status	string	Current task status. Possible status values are listed in Appendix A: Download Task Status or Appendix B: Values for Details of Erroneous Task.	1 and later
status_extra	<object>	Status_Extra object which provides extra information about task status.	1 and later
additional	<object>	Optional. Additional object	1 and later

<Additional Object> definition:

Member	Type	Description	Availability
detail	<object>	A Task_Detail object	1 and later
transfer	<object>	A Task_Transfer object	1 and later
file	<object>	Array of Task_File objects	1 and later
tracker	<object>	Array of Task_Tracker objects	1 and later
peer	<object>	Array of Task_Peer objects	1 and later

<Status_Extra Object> definition:

Member	Type	Description	Availability
error_detail	string	Available when status=error, providing error info. Possible error_detail values are listed in Appendix B: Values for Details of Erroneous Task.	1 and later
unzip_progress	int	Available when status=extracting, ranging from 0 to 100.	1 and later

<Task_Detail Object> definition:

Member	Type	Description	Availability
destination	string	Download destination	1 and later
uri	string	Task uri: HTTP/FTP/BT/Magnet/ED2K links	1 and later
create_time	Time stamp	Task created time. For more information, please see Limitations.	1 and later
priority	string	Task priority. Possible values are: "auto" , " low" , " normal" , " high" .	1 and later
total_peers	int	For BT: total peers For eMule: total source	
connected_seeders	int	For BT: connected seeders For eMule: transfer source	1 and later
connected_leechers	int	For BT: connected leechers For eMule: 0	1 and later

<Task_Transfer Object> definition:

Member	Type	Description	Availability
size_downloaded	string	Task downloaded size in bytes	1 and later
size_uploaded	string	Task uploaded size in bytes	1 and later
speed_download	int	Task download speed: byte/s	1 and later
speed_upload	int	Task upload speed: byte/s	1 and later

<Task_File Objects> definition:

Member	Type	Description	Availability
filename	string	File name	1 and later
size	string	File size in bytes	1 and later
size_downloaded	string	Downloaded file size in bytes	1 and later
priority	string	Possible priority: " skip" , " low" , " high" , "normal"	1 and later

<Task_Tracker Object> definition:

Member	Type	Description	Availability
url	string	Tracker url	1 and later
status	string	Tracker status	1 and later
update_timer	int	Next update timer	1 and later
seeds	int	Number of seeds	1 and later
peers	int	Number of peers	1 and later

<Task_Peer Object> definition:

Member	Type	Description	Availability
address	string	Peer address	1 and later
agent	string	Peer client name	1 and later
progress	float	Peer progress	1 and later
speed_download	int	Peer download speed: byte/s	1 and later
speed_upload	int	Peer upload speed: byte/s	1 and later

API Error Code

Code	Description
400	File upload failed
401	Max number of tasks reached
402	Destination denied
403	Destination does not exist
404	Invalid task id
405	Invalid task action
406	No default destination

Limitations

- 1 `create_time` is currently not supported by eMule tasks (`create_time` = null for eMule tasks).
- 2 Due to multipart upload limitations, creating tasks by uploading files should adhere to one of the following implementations:
 - a Set the upload files as the only POST request method data, and set other files as GET parameters.
 - b Set all the parameters as POST data, and the upload file should implement LAST parameters.

SYNO.DownloadStation.Statistic

Overview

Availability: Since Download Station 3.2-2258

Version: 1

Method

GetInfo

Request

No parameters required.

Example:

```
GET /webapi/DownloadStation/statistic.cgi?
api=SYNO.DownloadStation.Statistic&version=1&method=getinfo
```

Response

Parameter	Type	Description	Availability
speed_download	int	Total download speed except for eMule: byte/s	1 and later
speed_upload	int	Total upload speed except for eMule: byte/s	1 and later
emule_speed_download	int	Total eMule download speed: byte/s	1 and later
emule_speed_upload	int	Total eMule upload speed: byte/s	1 and later

Example:

```
{
    "speed_download":1096013,
    "speed_upload":321,
    "emule_speed_download":81731,
    "emule_speed_upload":464,
}
```

API Error Code

No specific API error codes.

SYNO.DownloadStation.RSS.Site

Overview

Availability: Since Download Station 3.2-2258

Version: 1

Method

List

Request

Parameter	Description	Availability
offset	Optional. Beginning task on the requested record. Default to "0".	1 and later
limit	Optional. Number of records requested: "-1" means to list all tasks. Default to "-1".	1 and later

Example:

```
GET /webapi/DownloadStation/RSSsite.cgi?
api=SYNO.DownloadStation.RSS.Site&version=1&method=list&offset=10&limit=10
```

Response

Parameter	Type	Description	Availability
total	int	Total number of records	1 and later
offset	int	Requests offset	1 and later
sites	array	Array of site objects	1 and later

Example:

```
{
  "offset":10,
  "total":12,
  "site":[{
    "id":6,
    "is Updating":true,
    "last_update":1340240217,
    "title":"Fenopy rss",
    "url":"http://fenopy.eu/rss.xml?row=50&cat=3&type=2",
    "username":"admin"
  }]
}
```

Refresh

Request

Parameter	Description	Availability
id	RSS site IDs to be refreshed. Separate IDs by "," or ID=ALL for all sites	1 and later

Example:

```
GET /webapi/DownloadStation/RSSsite.cgi?  
api=SYNO.DownloadStation.RSS.Site&version=1&method=refresh&id=1,2
```

Response

No specific response. It returns an empty success response if completed without error.

Response Objects

<Site Object> definition:

Member	Type	Description	Availability
id	int	Site ID	1 and later
is_updating	bool	If the site is updating now	1 and later
title	string	Site title	1 and later
url	string	Site RSS URL	1 and later
last_update	time stamp	Last updated time	1 and later
username	string	Owner's username	1 and later

API Error Code

No specific API error codes.

SYNO.DownloadStation.RSS.Feed

Overview

Availability: Since Download Station 3.2-2258

Version: 1

Method

List

Request

Parameter	Description	Availability
id	RSS site IDs	1 and later
offset	Optional. Beginning task on the requested record. Default to "0".	1 and later
limit	Optional. Number of records requested: "-1" means to list all tasks. Default to "-1".	1 and later

Example:

```
GET /webapi/DownloadStation/RSSfeed.cgi?
api=SYNO.DownloadStation.RSS.Feed&version=1&method=list&offset=10&limit=10
```

Response

Parameter	Type	Description	Availability
total	int	Total number of records	1 and later
offset	int	Requests offset	1 and later
feeds	array	Array of feed objects	1 and later

Example:

```
{
  "total":50,
  "offset":10,
  "feeds":[{
    "download_uri":"http://download.com/abc.torrent",
    "external_link":"http://download.com/abc",
    "size":"792230407",
    "time":1340206828,
    "title":"ABC.x264 3LTON"
  },{
    "download_uri":"http://download.com/123.torrent",
    "external_link":"http://download.com/123",
    "size":"788730068",
    "time":1340206815,
    "title":"123.PADD0"
  }]
}
```

```
}
```

Response Objects

<Feed Object> definition:

Member	Type	Description	Availability
title	string	Feed title	1 and later
size	string	Feed size in bytes	1 and later
time	time stamp	Feed listed time	1 and later
download_uri	string	Link for download	1 and later
external_link	string	Page link	1 and later

API Error Code

No specific API error codes.

Download Task Status

Followings are the possible statuses for download tasks:
waiting
downloading
paused
finishing
finished
hash_checking
seeding
filehosting_waiting
extracting
error

Values for Details of Erroneous Tasks

Followings are the possible values for err_detail when errors occur to download tasks:
broken_link
destination_not_exist
destination_denied
disk_full
quota_reached
timeout
exceed_max_file_system_size
exceed_max_destination_size
exceed_max_temp_size
encrypted_name_too_long
name_too_long
torrent_duplicate
file_not_exist
required_premium_account
not_supported_type
ftp_encryption_not_supported_type
extract_failed
extract_failed_wrong_password
extract_failed_invalid_archive
extract_failed_quota_reached
extract_failed_disk_full
unknown
broken_link

Document Revision History

This table describes the changes to the Synology Download Station Official API document.

Date	Note
2012-07-16	Initial release
2012-07-25	<ol style="list-style-type: none">1. Update the Request example of “<i>Step 2: Session Login</i>”2. Add the “session” parameter and update the Login example of SYNO.API.Auth3. Revise descriptions for Chapter 3: API List
2012-07-30	<ol style="list-style-type: none">1. Modify the Response description of “<i>Step 2: Session Login</i>”2. Update Response and Example of Delete, Pause, and Resume Methods for SYNO.DOWNLOAD.Task3. Update the API Error Code for SYNO.DownloadStation.Task
2012-08-07	<ol style="list-style-type: none">1. Update the Response example of “<i>Step 1: Get API Information</i>”2. Update the Request and Response examples of “<i>Step 2: Session Login</i>”3. Fix a typo4. Update the example of “<i>Step 4: Session Logout</i>”5. Update the Response example of SYNO. API. Info6. Update SYNO. API. Auth
2012-08-09	<ol style="list-style-type: none">1. Add one more element for request making as well as update the request syntax2. Update the Request example of “<i>Step 2: Session Login</i>”3. Modify the description of the Response parameter for SYNO. API. Auth
2012-08-10	<ol style="list-style-type: none">1. Modify the method for SYNO.DownloadStation.Statistic
2012-10-09	<ol style="list-style-type: none">1. Modify object type of the member <code>id</code> for <Task Object>
2012-10-19	<ol style="list-style-type: none">1. Update port number for making requests to SYNO. API. Info
2013-01-07	<ol style="list-style-type: none">1. Updated for Download Station 3.3
2013-02-25	<ol style="list-style-type: none">1. Update API error code for Response Objects
2013-03-13	<ol style="list-style-type: none">1. Update the Response description of the Login method for SYNO. API. Auth