

TREX

Generated by Doxygen 1.8.17

1 TREX	1
1.1 Dependencies	1
1.2 Compilation	1
1.3 Doxygen	1
1.3.1 for HTML documentation:	1
1.3.2 for PDF documentation:	1
1.4 Usage	2
2 Third Party Dependencies	3
2.1 clipp - command line interfaces for modern C++	3
2.2 Licenses	3
2.2.1 MIT	3
3 Hierarchical Index	5
3.1 Class Hierarchy	5
4 Class Index	7
4.1 Class List	7
5 File Index	9
5.1 File List	9
6 Class Documentation	11
6.1 CBatch Class Reference	11
6.1.1 Detailed Description	11
6.2 CBatchSlurm Class Reference	12
6.2.1 Detailed Description	13
6.2.2 Member Function Documentation	13
6.2.2.1 get_jobs()	13
6.2.2.2 get_node_states()	13
6.2.2.3 get_nodes()	14
6.2.2.4 get_queues()	14
6.2.2.5 login()	15
6.2.2.6 logout()	15
6.2.2.7 set_node_state()	15
6.3 CXCat Class Reference	16
6.3.1 Member Function Documentation	16
6.3.1.1 get_bootstate()	17
6.3.1.2 get_group()	17
6.3.1.3 get_group_members()	17
6.3.1.4 get_group_names()	18
6.3.1.5 get_nodes()	18
6.3.1.6 get_os_image_names()	19
6.3.1.7 get_os_images()	19

6.3.1.8 login()	19
6.3.1.9 logout()	20
6.3.1.10 reboot_nodes()	20
6.3.1.11 set_group_attributes()	20
6.3.1.12 set_node_attributes()	21
6.3.1.13 set_os_image()	21
6.4 HttpSession Class Reference	22
6.4.1 Constructor & Destructor Documentation	22
6.4.1.1 HttpSession()	22
6.4.1.2 ~HttpSession()	23
6.4.2 Member Function Documentation	23
6.4.2.1 get_access_token()	23
6.4.2.2 login()	23
6.4.2.3 logout()	24
6.4.2.4 set_date_parse_descr()	24
6.4.2.5 set_login_path()	24
6.4.2.6 set_logout_path()	24
6.4.2.7 set_token_type()	25
6.4.2.8 ssl_verify()	25
6.5 utils::loginData Struct Reference	25
6.6 RestClient Class Reference	26
6.6.1 Constructor & Destructor Documentation	26
6.6.1.1 RestClient()	26
6.6.1.2 ~RestClient()	27
6.6.2 Member Function Documentation	27
6.6.2.1 del()	27
6.6.2.2 get()	27
6.6.2.3 get_last_http_code()	28
6.6.2.4 get_last_request_time()	28
6.6.2.5 get_last_url()	28
6.6.2.6 login()	28
6.6.2.7 patch()	29
6.6.2.8 post()	29
6.6.2.9 put()	29
6.6.2.10 set_host_config()	29
6.6.2.11 set_user_credentials()	30
6.6.2.12 ssl_verify()	30
6.6.2.13 useragent()	30
6.7 SessionToken Class Reference	31
6.7.1 Member Function Documentation	32
6.7.1.1 get_access_token()	32
6.7.1.2 get_expire_date()	32

6.7.1.3 get_expire_time()	32
6.7.1.4 get_refresh_token()	32
6.7.1.5 get_token_type()	33
6.7.1.6 read_token()	33
6.7.1.7 set_date_parse_descr()	33
6.7.1.8 set_keys_access_token()	33
6.7.1.9 set_keys_by_token_type()	35
6.7.1.10 set_keys_expire_date()	35
6.7.1.11 set_keys_expire_time()	35
6.7.1.12 set_keys_refresh_token()	36
6.7.1.13 set_keys_token_type()	36
6.7.1.14 token_expired()	36
6.7.1.15 token_time_to_expire()	36
6.8 SessionTokenTypes Struct Reference	37
7 File Documentation	39
7.1 /home/ntippman/projects/trex/src/batchExchange/CBatch.h File Reference	39
7.1.1 Detailed Description	40
7.2 /home/ntippman/projects/trex/src/batchExchange/CBatchSlurm.cpp File Reference	40
7.2.1 Detailed Description	41
7.3 /home/ntippman/projects/trex/src/batchExchange/CBatchSlurm.h File Reference	41
7.3.1 Detailed Description	42
7.4 /home/ntippman/projects/trex/src/batchExchange/CXCat.cpp File Reference	42
7.4.1 Detailed Description	42
7.5 /home/ntippman/projects/trex/src/batchExchange/CXCat.h File Reference	42
7.5.1 Detailed Description	43
7.6 /home/ntippman/projects/trex/src/batchExchange/main.cpp File Reference	44
7.6.1 Detailed Description	44
7.6.2 Function Documentation	45
7.6.2.1 sigHandler()	45
7.7 /home/ntippman/projects/trex/src/batchExchange/utls.cpp File Reference	45
7.7.1 Detailed Description	45
7.8 /home/ntippman/projects/trex/src/batchExchange/utls.h File Reference	46
7.8.1 Detailed Description	47
7.8.2 Function Documentation	47
7.8.2.1 decode_brace()	47
7.8.2.2 ends_with()	48
7.8.2.3 is_number()	48
7.8.2.4 starts_with()	49
7.8.2.5 str_extract_regex_occurrences()	49
7.8.2.6 str_match_any_wildcard()	49
7.8.2.7 str_match_wildcard()	50

7.8.2.8 <code>str_split()</code>	50
7.8.2.9 <code>vector_contains()</code>	50

Index	53
--------------	-----------

Chapter 1

TREX

Command line tool for provisioning via xCat in compliance with the batch scheduler

1.1 Dependencies

- libcurl
- rapidjson

1.2 Compilation

```
./cmakeHelper.sh  
cd cbuild  
make -j $(nproc)
```

1.3 Doxygen

```
cd doc && doxygen Doxyfile
```

1.3.1 for HTML documentation:

```
doc/build/html/index.html
```

1.3.2 for PDF documentation:

```
make -C build/latex
```

generates doc/build/latex/refman.pdf

1.4 Usage

USAGE:

```
./TREX deploy <nodes> [--group] [--image <image>] [--prescripts <prescripts>]
    [--postbootscripts <postbootscripts>] [--postscripts <postscripts>] [--provmethod
    <provmethod>] [-h] [--json] [-b (slurm|pbs)] [-l <path>]
./TREX nodes [<nodes>] [-h] [--json] [-b (slurm|pbs)] [-l <path>]
./TREX state [<nodes>] [--state <state>] [--reason <reason>] [-h] [--json] [-b (slurm|pbs)]
    [-l <path>]
./TREX jobs [<jobIDs>] [-h] [--json] [-b (slurm|pbs)] [-l <path>]
./TREX queues [<queues>] [-h] [--json] [-b (slurm|pbs)] [-l <path>]
./TREX images [<images>] [-h] [--json] [-b (slurm|pbs)] [-l <path>]
./TREX bootstate [<nodes>] [-h] [--json] [-b (slurm|pbs)] [-l <path>]
./TREX reboot <nodes> [-h] [--json] [-b (slurm|pbs)] [-l <path>]
```

PARAMETERS:

COMMANDS

```
deploy <nodes> --group [--image <image>] [--prescripts <prescripts>] [--postbootscripts
<postbootscripts>] [--postscripts <postscripts>] [--provmethod <provmethod>]
    Deploy <image> on <nodes/groups>
nodes <nodes>
    Get node information [of <nodes>]
state <nodes> [--state <state>] [--reason <reason>]
    Get/Set state [of <nodes>]
jobs <jobIDs>
    Get job info [of <jobIDs>]
queues <queues>
    Get queue information [of <queues>]
images <images>
    Get information for available images [<images>]
bootstate <nodes>
    Get bootstate [of <nodes>]
reboot <nodes>
    Reboot <nodes>
```

OPTIONS

```
-h, --help
    Shows this help message
--json Output as json
-b, --batch (slurm|pbs)
    Batch System
-l, --loginFile <path>
    Path for login data
```


Chapter 2

Third Party Dependencies

2.1 clipp - command line interfaces for modern C++

<https://github.com/muellan/clipp>; Copyright (c) 2017 André Müller; foss@andremueller-online.de

License: [MIT](#)

2.2 Licenses

2.2.1 MIT

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

CBatch	11
CBatchSlurm	12
CXCat	16
HttpSession	22
utils::loginData	25
RestClient	26
SessionToken	31
SessionTokenTypes	37

Chapter 4

Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

CBatch		
Interface	11
CBatchSlurm		
Slurm Class	12
CXCat	16
HttpSession	22
utils::loginData	25
RestClient	26
SessionToken	31
SessionTokenTypes	37

Chapter 5

File Index

5.1 File List

Here is a list of all documented files with brief descriptions:

/home/ntippman/projects/trex/src/batchExchange/ base64.h	??
/home/ntippman/projects/trex/src/batchExchange/ CBatch.h	
Definitions for CBatch.h	39
/home/ntippman/projects/trex/src/batchExchange/ CBatchSlurm.cpp	
CBatch Slurm implementation	40
/home/ntippman/projects/trex/src/batchExchange/ CBatchSlurm.h	
Header for CBatchSlurm.cpp	41
/home/ntippman/projects/trex/src/batchExchange/ curlHelper.h	??
/home/ntippman/projects/trex/src/batchExchange/ CXCat.cpp	
CXCat implementation	42
/home/ntippman/projects/trex/src/batchExchange/ CXCat.h	
Header for CXCat	42
/home/ntippman/projects/trex/src/batchExchange/ httpSession.h	??
/home/ntippman/projects/trex/src/batchExchange/ main.cpp	
CLI	44
/home/ntippman/projects/trex/src/batchExchange/ restClient.h	??
/home/ntippman/projects/trex/src/batchExchange/ sessionTokenTypes.h	??
/home/ntippman/projects/trex/src/batchExchange/ utils.cpp	
Collection of helper functions	45
/home/ntippman/projects/trex/src/batchExchange/ utils.h	
Header for utils.cpp	46

Chapter 6

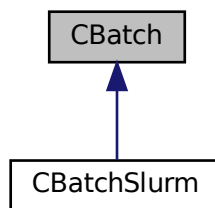
Class Documentation

6.1 CBatch Class Reference

Interface.

```
#include <CBatch.h>
```

Inheritance diagram for CBatch:



Protected Member Functions

- virtual int **get_jobs** (const std::vector< std::string > &, std::string &)=0
- virtual int **get_nodes** (const std::vector< std::string > &, std::string &)=0
- virtual int **get_queues** (const std::vector< std::string > &, std::string &)=0
- virtual int **get_node_states** (const std::vector< std::string > &, std::string &)=0
- virtual int **set_node_state** (const std::vector< std::string > &, std::string, std::string)=0

6.1.1 Detailed Description

Interface.

The documentation for this class was generated from the following files:

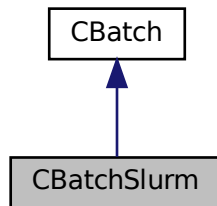
- /home/ntippman/projects/trex/src/batchExchange/[CBatch.h](#)
- /home/ntippman/projects/trex/src/batchExchange/CBatch.cpp

6.2 CBatchSlurm Class Reference

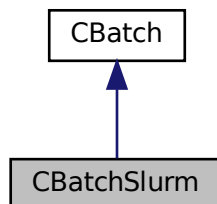
Slurm Class.

```
#include <CBatchSlurm.h>
```

Inheritance diagram for CBatchSlurm:



Collaboration diagram for CBatchSlurm:



Public Member Functions

- [CBatchSlurm](#) (std::string, std::string, std::string, std::string, bool)
Constructor.
- virtual [~CBatchSlurm](#) ()
Destructor.
- int [login](#) ()
Login session.
- int [logout](#) ()
Logout session.
- int [get_jobs](#) (const std::vector< std::string > &, std::string &)
Get job information.
- int [get_nodes](#) (const std::vector< std::string > &, std::string &)

Get node information.

- int [get_queues](#) (const std::vector< std::string > &, std::string &)

Get queue information.

- int [get_node_states](#) (const std::vector< std::string > &, std::string &)

Get node states.

- int [set_node_state](#) (const std::vector< std::string > &, std::string, std::string)

Set node information.

- int **drain_nodes** (std::vector< std::string > &, const std::string &)
- int **drained** (std::vector< std::string > &, unsigned int &)

Additional Inherited Members

6.2.1 Detailed Description

Slurm Class.

6.2.2 Member Function Documentation

6.2.2.1 [get_jobs\(\)](#)

```
int CBatchSlurm::get_jobs (
    const std::vector< std::string > & filter,
    std::string & output ) [virtual]
```

Get job information.

Parameters

<i>jobs</i>	Job Id(s)
<i>output</i>	output

Returns

- 0 Success
- 1 Error

Implements [CBatch](#).

6.2.2.2 [get_node_states\(\)](#)

```
int CBatchSlurm::get_node_states (
    const std::vector< std::string > & filter,
    std::string & output ) [virtual]
```

Get node states.

Parameters

<i>filter</i>	Nodes(s)
<i>output</i>	output

Returns

0 Success

1 Error

Implements [CBatch](#).**6.2.2.3 get_nodes()**

```
int CBatchSlurm::get_nodes (
    const std::vector< std::string > & filter,
    std::string & output ) [virtual]
```

Get node information.

Parameters

<i>filter</i>	Nodes(s)
<i>output</i>	output

Returns

0 Success

1 Error

Implements [CBatch](#).**6.2.2.4 get_queues()**

```
int CBatchSlurm::get_queues (
    const std::vector< std::string > & filter,
    std::string & output ) [virtual]
```

Get queue information.

Parameters

<i>filter</i>	Queue(s)
<i>output</i>	output

Returns

- 0 Success
- 1 Error

Implements [CBatch](#).

6.2.2.5 login()

```
int CBatchSlurm::login ( )
```

Login session.

Returns

- 0 Success
- 1 Error

6.2.2.6 logout()

```
int CBatchSlurm::logout ( )
```

Logout session.

Returns

- 0 Success
- 1 Error

6.2.2.7 set_node_state()

```
int CBatchSlurm::set_node_state (
    const std::vector< std::string > & nodes,
    std::string state,
    std::string reason ) [virtual]
```

Set node information.

Parameters

<i>queues</i>	nodes list of nodes
<i>state</i>	State to be set
<i>reason</i>	Reason for state change

Returns

- 0 Success
- 1 Error

Implements [CBatch](#).

The documentation for this class was generated from the following files:

- [/home/ntippman/projects/trex/src/batchExchange/CBatchSlurm.h](#)
- [/home/ntippman/projects/trex/src/batchExchange/CBatchSlurm.cpp](#)

6.3 CXCat Class Reference

Public Member Functions

- [CXCat](#) (std::string, std::string, std::string, std::string, bool)
Constructor.
- [~CXCat](#) ()
Destructor.
- int [login](#) ()
Login session.
- int [logout](#) ()
Logout session.
- int [get_os_images](#) (const std::vector< std::string > &, std::string &)
Get os images.
- int [get_os_image_names](#) (std::vector< std::string > &)
Get names of os images.
- int [get_bootstate](#) (const std::vector< std::string > &, std::string &)
Get os images.
- int [get_nodes](#) (std::string &)
Get information of all available nodes.
- int [set_os_image](#) (const std::vector< std::string > &, std::string)
Set os image.
- int [reboot_nodes](#) (const std::vector< std::string > &)
Reboot nodes.
- int [set_group_attributes](#) (std::string, const std::string &)
Set attributes of group.
- int [set_node_attributes](#) (const std::vector< std::string > &, const std::string &)
Set attributes of nodes.
- int [get_group_members](#) (std::string group, std::vector< std::string > &output)
Get all members of group.
- int [get_group_names](#) (std::vector< std::string > &)
Get names of all groups.
- int [get_group](#) (std::string, std::string &)
Get attributes of group.

6.3.1 Member Function Documentation

6.3.1.1 get_bootstate()

```
int CXCat::get_bootstate (
    const std::vector< std::string > & filter,
    std::string & output )
```

Get os images.

Parameters

<i>filter</i>	filter
<i>output</i>	output

Returns

0 Success

1 Error

6.3.1.2 get_group()

```
int CXCat::get_group (
    std::string group,
    std::string & output )
```

Get attributes of group.

Parameters

<i>group</i>	group
<i>output</i>	output

Returns

0 Success

1 Error

6.3.1.3 get_group_members()

```
int CXCat::get_group_members (
    std::string group,
    std::vector< std::string > & output )
```

Get all members of group.

Parameters

<i>group</i>	group
<i>output</i>	output

Returns

0 Success

1 Error

6.3.1.4 get_group_names()

```
int CXCat::get_group_names (
    std::vector< std::string > & output )
```

Get names of all groups.

Parameters

<i>output</i>	output
---------------	--------

Returns

0 Success

1 Error

6.3.1.5 get_nodes()

```
int CXCat::get_nodes (
    std::string & output )
```

Get information of all available nodes.

Parameters

<i>output</i>	output
---------------	--------

Returns

0 Success

1 Error

6.3.1.6 get_os_image_names()

```
int CXCat::get_os_image_names (
    std::vector< std::string > & output )
```

Get names of os images.

Parameters

<i>output</i>	output
---------------	--------

Returns

0 Success

1 Error

6.3.1.7 get_os_images()

```
int CXCat::get_os_images (
    const std::vector< std::string > & filter,
    std::string & output )
```

Get os images.

Parameters

<i>filter</i>	filter
<i>output</i>	output

Returns

0 Success

1 Error

6.3.1.8 login()

```
int CXCat::login ( )
```

Login session.

Returns

0 Success

1 Error

6.3.1.9 logout()

```
int CXCat::logout ( )
```

Logout session.

Returns

0 Success

1 Error

6.3.1.10 reboot_nodes()

```
int CXCat::reboot_nodes (
    const std::vector< std::string > & filter )
```

Reboot nodes.

Parameters

<i>filter</i>	filter
---------------	--------

Returns

0 Success

1 Error

6.3.1.11 set_group_attributes()

```
int CXCat::set_group_attributes (
    std::string group,
    const std::string & attributes )
```

Set attributes of group.

Parameters

<i>group</i>	name of group
<i>attributes</i>	json attributes

Returns

0 Success

1 Error

6.3.1.12 set_node_attributes()

```
int CXCat::set_node_attributes (
    const std::vector< std::string > & nodes,
    const std::string & attributes )
```

Set attributes of nodes.

Parameters

<i>nodes</i>	list of nodes
<i>attributes</i>	json attributes

Returns

- 0 Success
- 1 Error

6.3.1.13 set_os_image()

```
int CXCat::set_os_image (
    const std::vector< std::string > & filter,
    std::string osImage )
```

Set os image.

Parameters

<i>filter</i>	filter
<i>output</i>	output

Returns

- 0 Success
- 1 Error

The documentation for this class was generated from the following files:

- [/home/ntippman/projects/trex/src/batchExchange/CXCat.h](#)
- [/home/ntippman/projects/trex/src/batchExchange/CXCat.cpp](#)

6.4 HttpSession Class Reference

Public Member Functions

- [HttpSession](#) (const std::string &usernameIn, const std::string &passwordIn, const std::string &server↵AddressIn, const std::string &serverPortIn)
Constructor.
- [~HttpSession](#) ()
Destructor.
- void [ssl_verify](#) (bool verify)
Simple setter if ssl certificate should be checked or not.
- int [set_token_type](#) (int tokenTypeIn, const [SessionTokenTypes](#) *sessionTokenTypes=nullptr)
Sets keys to extract session token more easily.
- void [set_date_parse_descr](#) (const std::string formatDesc)
Setter for data format.
- void [set_login_path](#) (const std::string loginPathIn)
Defines the URL path for login.
- void [set_logout_path](#) (const std::string logoutPathIn)
Defines the URL path for logout.
- std::string [get_access_token](#) ()
Get access token.
- int [login](#) ()
Gets authentication token.
- int [logout](#) ()
Frees authentication token.

6.4.1 Constructor & Destructor Documentation

6.4.1.1 HttpSession()

```
HttpSession::HttpSession (
    const std::string & usernameIn,
    const std::string & passwordIn,
    const std::string & serverAddressIn,
    const std::string & serverPortIn )
```

Constructor.

Constructor with necessary parameters for function. Initililizes libcurl handle too.

Parameters

<i>usernameIn</i>	login name of user
<i>passwordIn</i>	login password of user
<i>server↵AddressIn</i>	url of server
<i>serverPortIn</i>	port of server

6.4.1.2 ~HttpSession()

```
HttpSession::~~HttpSession ( )
```

Destructor.

Frees session if not done yet and libcurl ressources.

6.4.2 Member Function Documentation

6.4.2.1 get_access_token()

```
std::string HttpSession::get_access_token ( )
```

Get access token.

Returns

the access token itself

6.4.2.2 login()

```
int HttpSession::login ( )
```

Gets authentication token.

Returns

0-100 error codes from libcurl >100 http error codes -1 no libcurl handle -2 no username or password -3 no host address or port -4 could not receive access token -5 already logged in -10 could'n extract token -100 unknown session type

Parameters

<i>login</i> ↔ <i>PathIn</i>	REST path to login endpoint (leading / is needed e.g. "/token")
---------------------------------	---

6.4.2.3 logout()

```
int HttpSession::logout ( )
```

Frees authentication token.

Returns

0-100 error codes from libcurl >100 http error codes

6.4.2.4 set_date_parse_descr()

```
void HttpSession::set_date_parse_descr (
    const std::string formatDesc )
```

Setter for data format.

Must have a format like it is described in strftime. <https://www.cplusplus.com/reference/ctime/strftime/>

Parameters

<i>formatDesc</i>	strftime like date description
-------------------	--------------------------------

6.4.2.5 set_login_path()

```
void HttpSession::set_login_path (
    const std::string loginPathIn )
```

Defines the URL path for login.

Parameters

<i>login↔ PathIn</i>	URL path (leading / is needed)
--------------------------	--------------------------------

6.4.2.6 set_logout_path()

```
void HttpSession::set_logout_path (
    const std::string logoutPathIn )
```

Defines the URL path for logout.

Parameters

<i>logout↔ PathIn</i>	URL path (leading / is needed)
---------------------------	--------------------------------

6.4.2.7 set_token_type()

```
int HttpSession::set_token_type (
    int tokenTypeIn,
    const SessionTokenTypes * sessionTokenTypes = nullptr )
```

Sets keys to extract session token more easily.

Parameters

<i>tokenTypeIn</i>	defines which type of token should be used
<i>sessionTokenTypes</i>	struct with json keys for token attributes

6.4.2.8 ssl_verify()

```
void HttpSession::ssl_verify (
    bool verfiy )
```

Simple setter if ssl certificate should be checked or not.

Parameters

<i>verfiy</i>	true if ssl certificate should be checked
---------------	---

The documentation for this class was generated from the following files:

- /home/ntippman/projects/trex/src/batchExchange/httpSession.h
- /home/ntippman/projects/trex/src/batchExchange/httpSession.cpp

6.5 utils::loginData Struct Reference

Public Attributes

- std::string **username**
- std::string **password**
- std::string **host**
- std::string **port**

The documentation for this struct was generated from the following file:

- /home/ntippman/projects/trex/src/batchExchange/[utils.h](#)

6.6 RestClient Class Reference

Public Member Functions

- [RestClient](#) (const int)
Constructor.
- [~RestClient](#) ()
Destructor.
- int [login](#) ()
Loggs in if needed.
- int [logout](#) ()
Removes login tokens and frees sessions.
- void [set_user_credentials](#) (const std::string &, const std::string &)
Simple setter for username an password.
- void [set_host_config](#) (const std::string &, const std::string &)
Simple setter for server settings.
- void [ssl_verify](#) (bool)
Simple setter if ssl certificate should be checked or not.
- void [useragent](#) (const std::string &)
Simple setter user agent.
- int [get](#) (const std::string &, std::string &, std::string &)
Represents http get request.
- int [post](#) (const std::string &, const std::string &, std::string &, std::string &)
Represents http post request.
- int [del](#) (const std::string &, std::string &, std::string &)
Represents http delete request.
- int [patch](#) (const std::string &, const std::string &, std::string &, std::string &)
Represents http patch request.
- int [put](#) (const std::string &, const std::string &, std::string &, std::string &)
Represents http put request.
- int [call](#) (std::string, const std::string &, std::string &, const std::string &="")
- long [get_last_http_code](#) ()
Getter for http return code of last request.
- double [get_last_request_time](#) ()
Getter for execution time of last request.
- std::string [get_last_url](#) ()
Getter for url of last request.

6.6.1 Constructor & Destructor Documentation

6.6.1.1 RestClient()

```
RestClient::RestClient (
    const int authTypeIn )
```

Constructor.

Constructor with necessary parameters for function. Initililizes libcurl handle too.

Parameters

<i>authTypeIn</i>	type of http connection (e.g. no login, token based, ...)
-------------------	---

6.6.1.2 ~RestClient()

```
RestClient::~RestClient ( )
```

Destructor.

Frees session if not done yet and libcurl ressources.

6.6.2 Member Function Documentation

6.6.2.1 del()

```
int RestClient::del (
    const std::string & restPath,
    std::string & response,
    std::string & header )
```

Represents http delete request.

Returns

0-100 error codes from libcurl >100 http error codes -1 no libcurl handle

6.6.2.2 get()

```
int RestClient::get (
    const std::string & restPath,
    std::string & response,
    std::string & header )
```

Represents http get request.

Returns

0-100 error codes from libcurl >100 http error codes -1 no libcurl handle

6.6.2.3 get_last_http_code()

```
long RestClient::get_last_http_code ( )
```

Getter for http return code of last request.

Returns

code of last request

6.6.2.4 get_last_request_time()

```
double RestClient::get_last_request_time ( )
```

Getter for execution time of last request.

Returns

execution time of last request

6.6.2.5 get_last_url()

```
std::string RestClient::get_last_url ( )
```

Getter for url of last request.

Returns

url of last request

6.6.2.6 login()

```
int RestClient::login ( )
```

Loggs in if needed.

Returns

0-100 error codes from libcurl >100 http error codes -1 no libcurl handle -2 no username or password -3 no host address or port -4 could not receive access token -5 already logged in -100 unknown authentication method

6.6.2.7 patch()

```
int RestClient::patch (
    const std::string & restPath,
    const std::string & postData,
    std::string & response,
    std::string & header )
```

Represents http patch request.

Returns

0-100 error codes from libcurl >100 http error codes -1 no libcurl handle

6.6.2.8 post()

```
int RestClient::post (
    const std::string & restPath,
    const std::string & postData,
    std::string & response,
    std::string & header )
```

Represents http post request.

Returns

0-100 error codes from libcurl >100 http error codes -1 no libcurl handle

6.6.2.9 put()

```
int RestClient::put (
    const std::string & restPath,
    const std::string & postData,
    std::string & response,
    std::string & header )
```

Represents http put request.

Returns

0-100 error codes from libcurl >100 http error codes -1 no libcurl handle

6.6.2.10 set_host_config()

```
void RestClient::set_host_config (
    const std::string & serverAddressIn,
    const std::string & serverPortIn )
```

Simple setter for server settings.

Parameters

<i>server↔ AddressIn</i>	server address or dns name
<i>serverPortIn</i>	port to communicate with (e.g. 80 for http, 443 for https)

6.6.2.11 set_user_credentials()

```
void RestClient::set_user_credentials (
    const std::string & usernameIn,
    const std::string & passwordIn )
```

Simple setter for username an password.

Parameters

<i>username↔ In</i>	username for login
<i>passwordIn</i>	password for login

6.6.2.12 ssl_verify()

```
void RestClient::ssl_verify (
    bool sslVerifyIn )
```

Simple setter if ssl certificate should be checked or not.

Parameters

<i>ssl↔ VerifyIn</i>	true if ssl certificate should be checked
--------------------------	---

6.6.2.13 useragent()

```
void RestClient::useragent (
    const std::string & useragent )
```

Simple setter user agent.

Sets the name how libcurl authenticated against the server.

Parameters

<i>useragent</i>	authentication name
------------------	---------------------

The documentation for this class was generated from the following files:

- /home/ntippman/projects/trex/src/batchExchange/restClient.h
- /home/ntippman/projects/trex/src/batchExchange/restClient.cpp

6.7 SessionToken Class Reference

Public Member Functions

- [SessionToken](#) ()
Constructor.
- [~SessionToken](#) ()
Destructor.
- void [set_keys_token_type](#) (const std::vector< std::string > &keysTokenTypeln)
Set token type.
- void [set_keys_access_token](#) (const std::vector< std::string > &keysAccessTokenln)
Set access token.
- void [set_keys_refresh_token](#) (const std::vector< std::string > &keysRefreshTokenln)
Set refresh token.
- void [set_keys_expire_time](#) (const std::vector< std::string > &keysExpireTimeln)
Set expire time.
- void [set_keys_expire_date](#) (const std::vector< std::string > &keysExpireDateIn)
Set expire date.
- void [set_keys_by_token_type](#) (const [SessionTokenTypes](#) token)
Sets keys to extract session token more easily.
- void [set_date_parse_descr](#) (std::string formatDesc)
Setter for data format.
- int [read_token](#) (const std::string &token)
Wrapper for reading token data from json object.
- bool [token_expired](#) ()
Resturns if token is expired.
- int [token_time_to_expire](#) ()
Delivers time in seconds when how long the token is valid.
- std::string [get_token_type](#) ()
Get token type.
- std::string [get_access_token](#) ()
Get access token.
- std::string [get_refresh_token](#) ()
Get refresh token.
- int [get_expire_time](#) ()
Get expire time of token.
- time_t [get_expire_date](#) ()
Get expire date.

6.7.1 Member Function Documentation

6.7.1.1 `get_access_token()`

```
std::string SessionToken::get_access_token ( )
```

Get access token.

Returns

access token

6.7.1.2 `get_expire_date()`

```
time_t SessionToken::get_expire_date ( )
```

Get expire date.

Returns

expire date of the access token as unix timestamp

6.7.1.3 `get_expire_time()`

```
int SessionToken::get_expire_time ( )
```

Get expire time of token.

Returns

expire time of the access token in seconds

6.7.1.4 `get_refresh_token()`

```
std::string SessionToken::get_refresh_token ( )
```

Get refresh token.

Returns

a refresh token to request a new access token

6.7.1.5 get_token_type()

```
std::string SessionToken::get_token_type ( )
```

Get token type.

Returns

Type of the token

6.7.1.6 read_token()

```
int SessionToken::read_token (
    const std::string & token )
```

Wrapper for reading token data from json object.

Parameters

<i>token</i>	the token data as json contained in a string
--------------	--

Returns

error code

6.7.1.7 set_date_parse_descr()

```
void SessionToken::set_date_parse_descr (
    std::string formatDesc )
```

Setter for data format.

Must have a format like it is described in strftime. <https://www.cplusplus.com/reference/ctime/strftime/>

Parameters

<i>formatDesc</i>	strftime like date description
-------------------	--------------------------------

6.7.1.8 set_keys_access_token()

```
void SessionToken::set_keys_access_token (
    const std::vector< std::string > & keysAccessTokenIn )
```

Set access token.

Parameters

<i>keysAccess</i> ↔ <i>TokenIn</i>	contains keys of json file where to find the access token ["token", "accessToken"]
---------------------------------------	--

6.7.1.9 set_keys_by_token_type()

```
void SessionToken::set_keys_by_token_type (
    const SessionTokenTypes token )
```

Sets keys to extract session token more easily.

Parameters

<i>token</i>	struct with json keys for token attributes
--------------	--

6.7.1.10 set_keys_expire_date()

```
void SessionToken::set_keys_expire_date (
    const std::vector< std::string > & keysExpireDateIn )
```

Set expire date.

Parameters

<i>keysExpire</i> ↔ <i>DateIn</i>	contains keys of json file where to find the expire date ["token", "date"]
--------------------------------------	--

6.7.1.11 set_keys_expire_time()

```
void SessionToken::set_keys_expire_time (
    const std::vector< std::string > & keysExpireTimeIn )
```

Set expire time.

Parameters

<i>keysExpire</i> ↔ <i>TimeIn</i>	contains keys of json file where to find the expire time ["token", "time"]
--------------------------------------	--

6.7.1.12 set_keys_refresh_token()

```
void SessionToken::set_keys_refresh_token (
    const std::vector< std::string > & keysRefreshTokenIn )
```

Set refresh token.

Parameters

<i>keysRefreshTokenIn</i>	contains keys of json file where to find the refresh token ["token", "refreshToken"]
---------------------------	--

6.7.1.13 set_keys_token_type()

```
void SessionToken::set_keys_token_type (
    const std::vector< std::string > & keysTokenTypeIn )
```

Set token type.

Parameters

<i>keysTokenTypeIn</i>	contains keys of json file where to find the token type ["token", "tokenType"]
------------------------	--

6.7.1.14 token_expired()

```
bool SessionToken::token_expired ( )
```

Returns if token is expired.

Returns

simple bool true == expired

6.7.1.15 token_time_to_expire()

```
int SessionToken::token_time_to_expire ( )
```

Delivers time in seconds when how long the token is valid.

Returns

time in seconds

The documentation for this class was generated from the following files:

- /home/ntippman/projects/trex/src/batchExchange/httpSession.h
- /home/ntippman/projects/trex/src/batchExchange/httpSession.cpp

6.8 SessionTokenTypes Struct Reference

Public Attributes

- `std::vector< std::string > keysTokenType`
- `std::vector< std::string > keysAccessToken`
- `std::vector< std::string > keysRefreshToken`
- `std::vector< std::string > keysExpireTime`
- `std::vector< std::string > keysExpireDate`

The documentation for this struct was generated from the following file:

- `/home/ntippman/projects/trex/src/batchExchange/sessionTokenTypes.h`

Chapter 7

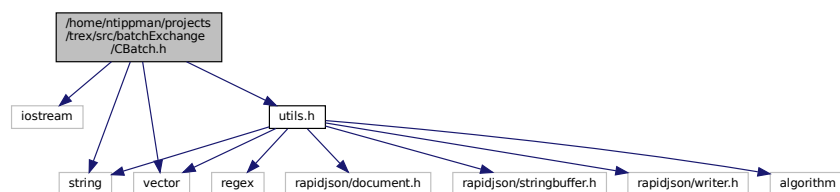
File Documentation

7.1 /home/ntippman/projects/trex/src/batchExchange/CBatch.h File Reference

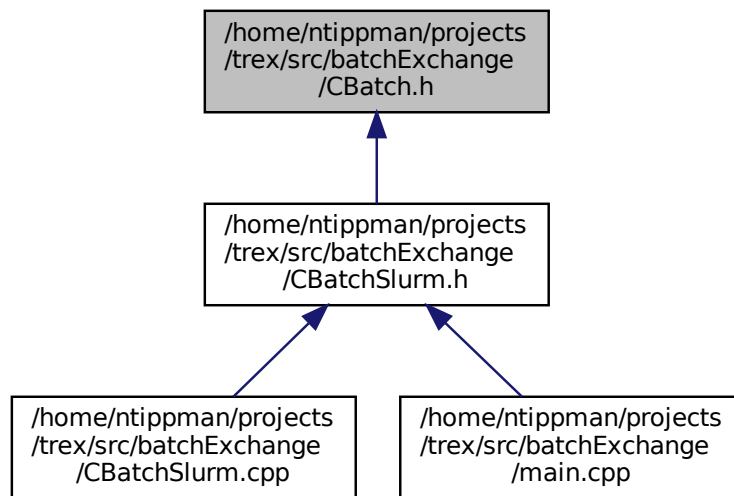
Definitions for [CBatch.h](#).

```
#include <iostream>
#include <string>
#include <vector>
#include "utils.h"
```

Include dependency graph for CBatch.h:



This graph shows which files directly or indirectly include this file:



Classes

- class `CBatch`
Interface.

7.1.1 Detailed Description

Definitions for `CBatch.h`.

Interface definition for inheriting batch classes.

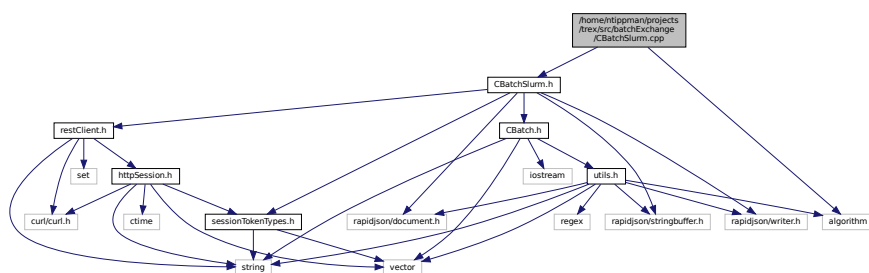
7.2 `/home/ntippman/projects/trex/src/batchExchange/CBatchSlurm.cpp` File Reference

`CBatch` Slurm implementation.

```
#include "CBatchSlurm.h"
```

```
#include <algorithm>
```

Include dependency graph for `CBatchSlurm.cpp`:



7.3.1 Detailed Description

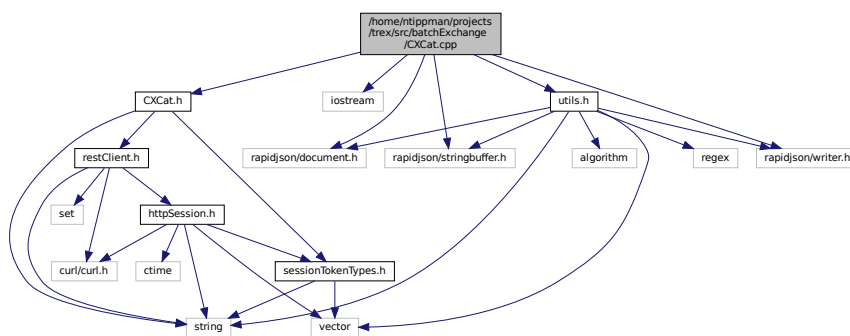
Header for [CBatchSlurm.cpp](#).

7.4 /home/ntippman/projects/trex/src/batchExchange/CXCat.cpp File Reference

[CXCat](#) implementation.

```
#include "CXCat.h"
#include <iostream>
#include "rapidjson/document.h"
#include "rapidjson/stringbuffer.h"
#include "rapidjson/writer.h"
#include "utils.h"
```

Include dependency graph for CXCat.cpp:



7.4.1 Detailed Description

[CXCat](#) implementation.

7.5 /home/ntippman/projects/trex/src/batchExchange/CXCat.h File Reference

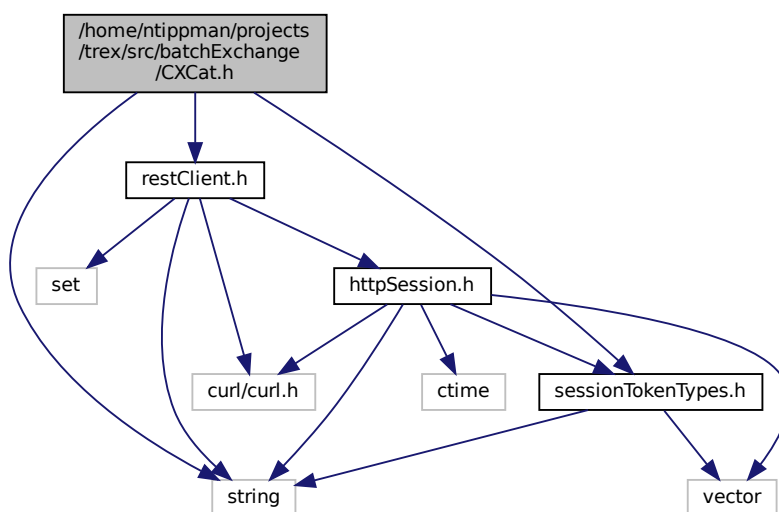
Header for [CXCat](#).

```
#include <string>
#include "restClient.h"
```

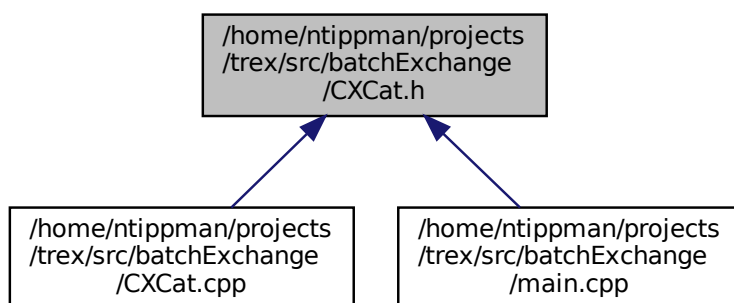


```
#include "sessionTokenTypes.h"
```

Include dependency graph for CXCat.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [CXCat](#)

7.5.1 Detailed Description

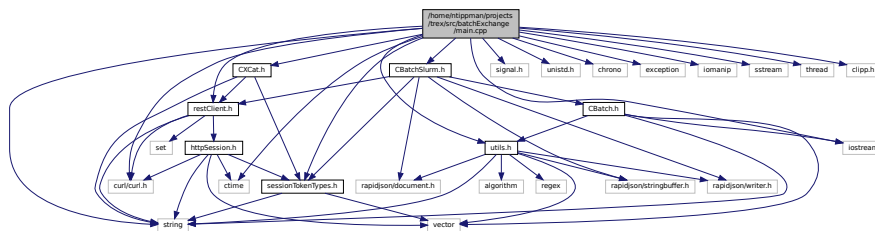
Header for [CXCat](#).

7.6 /home/ntippman/projects/trex/src/batchExchange/main.cpp File Reference

CLI.

```
#include <curl/curl.h>
#include <signal.h>
#include <unistd.h>
#include <chrono>
#include <ctime>
#include <exception>
#include <iomanip>
#include <iostream>
#include <sstream>
#include <string>
#include <thread>
#include "CBatchSlurm.h"
#include "CXCat.h"
#include "clipp.h"
#include "restClient.h"
#include "sessionTokenTypes.h"
#include "utils.h"
```

Include dependency graph for main.cpp:



Macros

- `#define DRAIN_SLEEP 3000`

Functions

- bool **canceled** (false)
- void **sigHandler** (int signal)
Handle caught signal.
- int **main** (int argc, char **argv)

7.6.1 Detailed Description

CLI.

7.6.2 Function Documentation

7.6.2.1 sigHandler()

```
void sigHandler (
    int signal )
```

Handle caught signal.

This function is called when SIGINT is caught.

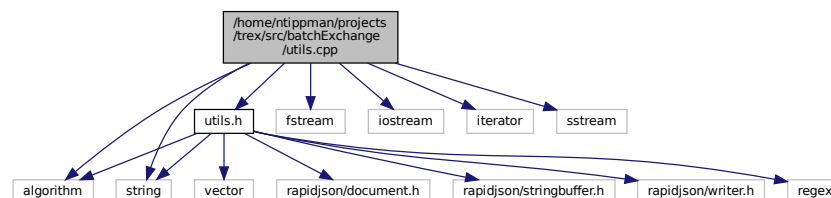
Parameters

<i>signal</i>	Number of signal
---------------	------------------

7.7 /home/ntippman/projects/trex/src/batchExchange/utils.cpp File Reference

Collection of helper functions.

```
#include "utils.h"
#include <algorithm>
#include <fstream>
#include <iostream>
#include <iterator>
#include <sstream>
#include <string>
Include dependency graph for utils.cpp:
```



7.7.1 Detailed Description

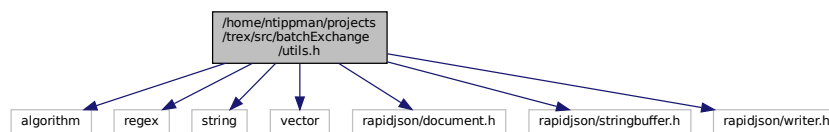
Collection of helper functions.

7.8 /home/ntippman/projects/trex/src/batchExchange/utils.h File Reference

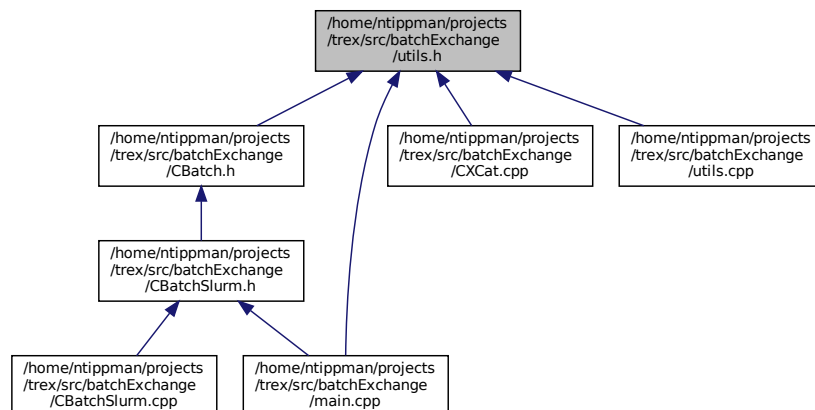
Header for [utils.cpp](#).

```
#include <algorithm>
#include <regex>
#include <string>
#include <vector>
#include "rapidjson/document.h"
#include "rapidjson/stringbuffer.h"
#include "rapidjson/writer.h"
```

Include dependency graph for utils.h:



This graph shows which files directly or indirectly include this file:



Classes

- struct [utils::loginData](#)

Macros

- `#define INVALID_JSON_ERROR_MSG "Error - Response is not valid JSON"`

Functions

- void `utils::str_split` (const std::string &input, const std::string delimiter, std::vector< std::string > &ret)
Split string at delimiter.
- int `utils::read_login_data` (const std::string &path, `utils::loginData` &megware, `utils::loginData` &xcat, `utils::loginData` &slurm, bool ignoreHeader=true)
- int `utils::read_file_to_string` (std::string, std::string &)
- int `utils::write_to_file` (std::string, const std::string &)
- void `utils::erase_lines_from_start` (std::string &, int)
- void `utils::to_lower` (std::string &)
- void `utils::to_upper` (std::string &)
- std::string `utils::join_vector_to_string` (const std::vector< std::string > &, const std::string)
- int `utils::check_errors` (const std::string &)
- void `utils::str_extract_regex_occurrences` (std::string, const std::regex &, std::vector< std::string > &)
Extract all regex matches from string.
- bool `utils::is_number` (const std::string &)
Check if string contains only numbers (integer only)
- void `utils::decode_brace` (const std::string &, std::vector< std::string > &)
Decode brace notation.
- bool `utils::str_match_wildcard` (const std::string &, const std::string &)
Check if string matches wildcard-string.
- bool `utils::ends_with` (const std::string &, const std::string &)
Check if string ends with <suffix>
- bool `utils::starts_with` (const std::string &, const std::string &)
Check if string starts with <prefix>
- bool `utils::str_match_any_wildcard` (const std::vector< std::string > &, const std::string &)
Check if string matches any wildcard-string of vector.
- void `utils::rapidjson_doc_to_str` (rapidjson::Document &, std::string &)
- template<typename T >
bool `utils::vector_contains` (const std::vector< T > &v, const T &e)
Template function to check whether a vector contains the specified element.

Variables

- const std::vector< std::string > `utils::slurmNodeStates`

7.8.1 Detailed Description

Header for `utils.cpp`.

7.8.2 Function Documentation

7.8.2.1 `decode_brace()`

```
void utils::decode_brace (
    const std::string & input,
    std::vector< std::string > & ret )
```

Decode brace notation.

Parameters

<i>input</i>	String to be decoded
<i>ret</i>	Vector of decoded strings

7.8.2.2 ends_with()

```
bool utils::ends_with (
    const std::string & s,
    const std::string & suffix )
```

Check if string ends with <suffix>

Parameters

<i>s</i>	Input string
<i>suffix</i>	Suffix

Returns

0 Suffix not present

1 Suffix present

7.8.2.3 is_number()

```
bool utils::is_number (
    const std::string & s )
```

Check if string contains only numbers (integer only)

Parameters

<i>pattern</i>	Wildcard pattern vector
<i>target</i>	Target to be checked

Returns

0 NaN

1 Is number

7.8.2.4 starts_with()

```
bool utils::starts_with (
    const std::string & s,
    const std::string & prefix )
```

Check if string starts with <prefix>

Parameters

<i>s</i>	Input string
<i>prefix</i>	Prefix

Returns

0 Prefix not present

1 Prefix present

7.8.2.5 str_extract_regex_occurrences()

```
void utils::str_extract_regex_occurrences (
    std::string input,
    const std::regex & regex,
    std::vector< std::string > & ret )
```

Extract all regex matches from string.

Parameters

<i>input</i>	Input string to be checked
<i>regex</i>	Regex
<i>ret</i>	Vector of matches

7.8.2.6 str_match_any_wildcard()

```
bool utils::str_match_any_wildcard (
    const std::vector< std::string > & wildcards,
    const std::string & target )
```

Check if string matches any wildcard-string of vector.

Parameters

<i>pattern</i>	Wildcard pattern vector
<i>target</i>	Target to be checked

Returns

0 No match
1 Match

7.8.2.7 str_match_wildcard()

```
bool utils::str_match_wildcard (
    const std::string & pattern,
    const std::string & target )
```

Check if string matches wildcard-string.

Parameters

<i>pattern</i>	Wildcard pattern
<i>target</i>	Target to be checked

Returns

0 No match
1 Match

7.8.2.8 str_split()

```
void utils::str_split (
    const std::string & input,
    const std::string delimiter,
    std::vector< std::string > & ret )
```

Split string at delimiter.

Parameters

<i>input</i>	Input string
<i>delimiter</i>	Delimiter
<i>ret</i>	Output vector

7.8.2.9 vector_contains()

```
template<typename T >
bool utils::vector_contains (
```



```
const std::vector< T > & v,  
const T & e )
```

Template function to check whether a vector contains the specified element.

Parameters

<i>v</i>	Vector
<i>e</i>	Element

Returns

- 0 Element not in vector
- 1 Element in vector

Index

/home/ntippman/projects/trex/src/batchExchange/CBatch.h, [ends_with](#)
[39](#) [utils.h](#), [48](#)

/home/ntippman/projects/trex/src/batchExchange/CBatchSlurm.cpp,
[40](#) [get](#)

/home/ntippman/projects/trex/src/batchExchange/CBatchSlurm.h, [RestClient](#), [27](#)
[41](#) [get_access_token](#)

/home/ntippman/projects/trex/src/batchExchange/CXCat.cpp, [HttpSession](#), [23](#)
[42](#) [SessionToken](#), [32](#)

/home/ntippman/projects/trex/src/batchExchange/CXCat.h, [get_bootstate](#)
[42](#) [CXCat](#), [16](#)

/home/ntippman/projects/trex/src/batchExchange/main.cpp, [get_expire_date](#)
[44](#) [SessionToken](#), [32](#)

/home/ntippman/projects/trex/src/batchExchange/utils.cpp, [get_expire_time](#)
[45](#) [SessionToken](#), [32](#)

/home/ntippman/projects/trex/src/batchExchange/utils.h, [get_group](#)
[46](#) [CXCat](#), [17](#)

~HttpSession [get_group_members](#)
[HttpSession](#), [23](#) [CXCat](#), [17](#)

~RestClient [get_group_names](#)
[RestClient](#), [27](#) [CXCat](#), [18](#)

[get_jobs](#)
[CBatchSlurm](#), [13](#)

[get_last_http_code](#)
[RestClient](#), [27](#)

[get_last_request_time](#)
[RestClient](#), [28](#)

[get_last_url](#)
[RestClient](#), [28](#)

[get_node_states](#)
[CBatchSlurm](#), [13](#)

[get_nodes](#)
[CBatchSlurm](#), [14](#)
[CXCat](#), [18](#)

[get_os_image_names](#)
[CXCat](#), [18](#)

[get_os_images](#)
[CXCat](#), [19](#)

[get_queues](#)
[CBatchSlurm](#), [14](#)

[get_refresh_token](#)
[SessionToken](#), [32](#)

[get_token_type](#)
[SessionToken](#), [32](#)

[HttpSession](#), [22](#)
~HttpSession, [23](#)
[get_access_token](#), [23](#)
[HttpSession](#), [22](#)
[login](#), [23](#)
[logout](#), [23](#)

[decode_brace](#)
[utils.h](#), [47](#)

[del](#)
[RestClient](#), [27](#)

- set_date_parse_descr, 24
 - set_login_path, 24
 - set_logout_path, 24
 - set_token_type, 25
 - ssl_verify, 25
- is_number
 - utils.h, 48
- login
 - CBatchSlurm, 15
 - CXCat, 19
 - HttpSession, 23
 - RestClient, 28
- logout
 - CBatchSlurm, 15
 - CXCat, 19
 - HttpSession, 23
- main.cpp
 - sigHandler, 45
- patch
 - RestClient, 28
- post
 - RestClient, 29
- put
 - RestClient, 29
- read_token
 - SessionToken, 33
- reboot_nodes
 - CXCat, 20
- RestClient, 26
 - ~RestClient, 27
 - del, 27
 - get, 27
 - get_last_http_code, 27
 - get_last_request_time, 28
 - get_last_url, 28
 - login, 28
 - patch, 28
 - post, 29
 - put, 29
 - RestClient, 26
 - set_host_config, 29
 - set_user_credentials, 30
 - ssl_verify, 30
 - useragent, 30
- SessionToken, 31
 - get_access_token, 32
 - get_expire_date, 32
 - get_expire_time, 32
 - get_refresh_token, 32
 - get_token_type, 32
 - read_token, 33
 - set_date_parse_descr, 33
 - set_keys_access_token, 33
 - set_keys_by_token_type, 35
 - set_keys_expire_date, 35
 - set_keys_expire_time, 35
 - set_keys_refresh_token, 35
 - set_keys_token_type, 36
 - token_expired, 36
 - token_time_to_expire, 36
- SessionTokenTypes, 37
 - set_date_parse_descr
 - HttpSession, 24
 - SessionToken, 33
 - set_group_attributes
 - CXCat, 20
 - set_host_config
 - RestClient, 29
 - set_keys_access_token
 - SessionToken, 33
 - set_keys_by_token_type
 - SessionToken, 35
 - set_keys_expire_date
 - SessionToken, 35
 - set_keys_expire_time
 - SessionToken, 35
 - set_keys_refresh_token
 - SessionToken, 35
 - set_keys_token_type
 - SessionToken, 36
 - set_login_path
 - HttpSession, 24
 - set_logout_path
 - HttpSession, 24
 - set_node_attributes
 - CXCat, 21
 - set_node_state
 - CBatchSlurm, 15
 - set_os_image
 - CXCat, 21
 - set_token_type
 - HttpSession, 25
 - set_user_credentials
 - RestClient, 30
 - sigHandler
 - main.cpp, 45
 - ssl_verify
 - HttpSession, 25
 - RestClient, 30
 - starts_with
 - utils.h, 48
 - str_extract_regex_occurrences
 - utils.h, 49
 - str_match_any_wildcard
 - utils.h, 49
 - str_match_wildcard
 - utils.h, 50
 - str_split
 - utils.h, 50
 - token_expired
 - SessionToken, 36
 - token_time_to_expire

- SessionToken, [36](#)
- useragent
 - RestClient, [30](#)
- utils.h
 - decode_brace, [47](#)
 - ends_with, [48](#)
 - is_number, [48](#)
 - starts_with, [48](#)
 - str_extract_regex_occurrences, [49](#)
 - str_match_any_wildcard, [49](#)
 - str_match_wildcard, [50](#)
 - str_split, [50](#)
 - vector_contains, [50](#)
- utils::loginData, [25](#)
- vector_contains
 - utils.h, [50](#)