
rpws documentation

Release 0

eirannejad

Dec 06, 2018

Contents

1 Data Models	9
Python Module Index	19

Python wrapper for Autodesk Revit Server.

Example

```
>>> name = '<server name>'
>>> version = '2017'      # server version in XXXX format
>>> rserver = RevitServer(name, version)
>>> # listing all files, folders, and models in a server
>>> for parent, folders, files, models in rserver.walk():
...     print(parent)
...     for fd in folders:
...         print('\t@d {}'.format(fd.path))
...     for f in files:
...         print('\t@f {}'.format(f.path))
...     for m in models:
...         print('\t@m {}'.format(m.path))
```

class rpws.server.RevitServer(name, version, username=None, machine=None)

Handles all communication with Revit Server as initialized.

Parameters

- **name** (*str*) – Server name.
- **version** (*str*) – Server version.
- **username** (*str*, *optional*) – Username to be passed to in http calls. This is set to current user if not provided.
- **machine** (*str*, *optional*) – Machine name to be passed to in http calls Set to current machine if not provided.

name

Server name.

Type str

version

Server version.

Type str

__base_uri

Base URI of the initialized server

Type str, private

__huser

Username of the initialized server.

Type str, private

__hmachine

Machine name of the initialized server

Type str, private

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> print(rserver)
<rpws.RevitServer name:server01 version:2017>
```

cancellock (*nodepath*)

Cancel any in-progress locks on model.

Parameters **nodepath** (*str*) – Path to a model on the server.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for model in rserver.listmodels('/example/path'):
...     rserver.cancellock(model.path)
```

copy (*nodepath*, *new_nodepath*, *overwrite=False*)

Copy a file, folder, or model to new location.

Parameters

- **nodepath** (*str*) – Path to a file, folder, or model on the server.
- **new_nodepath** (*str*) – Full path to new location and name.
- **overwrite** (*bool*) – True to overwrite any existing entries.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> rserver.copy('/example/model.rvt', '/example/newmodel.rvt')
```

delete (*nodepath*)

Delete a file, folder, or model.

Parameters **nodepath** (*str*) – Path to a file, folder, or model on the server.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> rserver.delete('/example/path/model.rvt')
```

deletedescendentlocks (*nodepath*)

Delete the decendent locks.

API command: /descendent/locks

Parameters **nodepath** (*str*) – Path to a directory on the server.

Returns list of str for list of entries with failed locks

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for folder in rserver.listfolders('/example/path'):
...     rserver.deletedescendentlocks(folder.path)
```

getdescendentlocks (*nodepath*)

Return the decendent locks info.

API command: /descendent/locks

Parameters **nodepath** (*str*) – Path to a directory on the server.

Returns rpws.models.ChildrenLockInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for folder in rserver.listfolders('/example/path'):
...     clockinfo = rserver.getdescendentlocks(folder.path)
...     for locked_model_path in clockinfo.items:
...         print(locked_model_path)
```

getdriveinfo ()

Return server drive information.

API command: /contents

API keys: DriveFreeSpace and DriveSpace

Returns rpws.models.ServerDriveInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> dinfo = rserver.getdriveinfo()
>>> print(dinfo.drive_space)
```

getfolderinfo (*nodepath*)

Return directory info for provided path.

API command: /directoryinfo

Parameters **nodepath** (*str*) – Path to an entry on the server.

Returns rpws.models.EntryDirInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for folder in rserver.listfolders('/example/path'):
...     finfo = rserver.getfolderinfo(folder.path)
...     print(finfo.date_created)
```

getinfo()

Return server properties.

API command: /serverproperties

Returns rpws.models.ServerInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> sinfo = rserver.getinfo()
```

getmodelhistory(nodepath)

Return model info from provided model path.

API command: /history

Parameters **nodepath** (*str*) – Path to a model on the server.

Returns rpws.models.MHistoryInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for model in rserver.listmodels('/example/path'):
...     mhist = rserver.getmodelhistory(model.path)
...     for hist in mhist.items:
...         print(hist.comment, hist.user)
```

getmodelinfo(nodepath)

Return model info from provided model path.

API command: /modelinfo

Parameters **nodepath** (*str*) – Path to a model on the server.

Returns rpws.models.ModelInfoEx

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for model in rserver.listmodels('/example/path'):
...     minfo = rserver.getmodelinfo(model.path)
...     print(minfo.size)
```

getprojectinfo(nodepath)

Return project info from provided model path.

API command: /projectinfo

Parameters **nodepath** (*str*) – Path to an entry on the server.

Returns rpws.models.ProjectInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for model in rserver.listmodels('/example/path'):
...     pinfo = rserver.getprojectinfo(model.path)
...     for pparam in pinfo.parameters:
...         print(pparam.name, pparam.value)
```

listfiles (nodepath=None)

Return files from root or provided path.

API command: /contents

API key: Files

Parameters **nodepath** (*str*, *optional*) – Path to an entry on the server. Root if not provided.

Returns list of rpws.models.FileInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for file in rserver.listfiles('/example/path'):
...     print(file.path)
```

listfolders (nodepath=None)

Return folders from root or provided path.

API command: /contents

API key: Folders

Parameters **nodepath** (*str*, *optional*) – Path to an entry on the server. Root if not provided.

Returns list of rpws.models.FolderInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for folder in rserver.listfolders('/example/path'):
...     print(folder.path)
```

listmodels (nodepath=None)

Return models from root or provided path.

API command: /contents

API key: Models

Parameters **nodepath** (*str*, *optional*) – Path to an entry on the server. Root if not provided.

Returns list of rpws.models.ModelInfo

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for model in rserver.listmodels('/example/path'):
...     print(model.path)
```

lock (*nodepath*)

Lock model.

Parameters **nodepath** (*str*) – Path to a model on the server.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for model in rserver.listmodels('/example/path'):
...     rserver.lock(model.path)
```

mkdir (*nodepath*)

Create a new directory.

Parameters **nodepath** (*str*) – Path to a directory on the server.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> rserver.mkdir('/example/path')
```

move (*nodepath*, *new_nodepath*, *overwrite=False*)

Move a file, folder, or model to new location.

Parameters

- **nodepath** (*str*) – Path to a file, folder, or model on the server.
- **new_nodepath** (*str*) – Full path to new location and name.
- **overwrite** (*bool*) – True to overwrite any existing entries.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> rserver.move('/example/model.rvt', '/example/newmodel.rvt')
```

path

Root path of server.

Returns Root path

Return type str

rename (*nodepath*, *new_nodename*)

Rename a file, folder, or model.

Parameters

- **nodepath** (*str*) – Path to a file, folder, or model on the server.
- **new_nodename** (*str*) – New name for file, folder, or model

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> rserver.rename('/example/path', '/example/newpath')
```

rmdir (*nodepath*)

Delete a file, folder, or model.

Parameters **nodepath** (*str*) – Path to a file, folder, or model on the server.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> rserver.rmdir('/example/path')
```

scandir (*nodepath=None*)

Return files, folders, and models from root or provided path.

API command: /contents

Parameters **nodepath** (*str, optional*) – Path to an entry on the server. Root if not provided.

Returns rpws.models.EntryContents

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for entry in rserver.scandir('/example/path'):
...     print(entry.path)
```

unlock (*nodepath*)

Unlock model.

Parameters **nodepath** (*str*) – Path to a model on the server.

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for model in rserver.listmodels('/example/path'):
...     rserver.unlock(model.path)
```

walk (*top=None, topdown=True, digmodels=False*)

Walk the provided path or root.

Yields a 4-tuple of parent directory, folders, files, and models

Parameters

- **top** (*str, optional*) – Parent directory. Root if not provided.
- **topdown** (*bool*) – True to start from top and walk down
- **digmodels** (*bool*) – True to list entries under a model folder Revit models on Revit Server are actually folders with files, and other subfolders.

Returns (parent, folders, files, models)

Return type tuple

Example

```
>>> rserver = RevitServer('server01', '2017')
>>> for parent, folders, files, models in rserver.walk():
...     print(parent)
...     for fd in folders:
...         print('\t@d {}'.format(fd.path))
...     for f in files:
...         print('\t@f {}'.format(f.path))
...     for m in models:
...         print('\t@m {}'.format(m.path))
```

CHAPTER 1

Data Models

```
class rpws.models.ServerInfo(name, version, machine_name, roles, access_level_types,  
                             max_path_length, max_name_length, servers)  
    namedtuple for server properties  
  
    name  
        Server name  
        Type str  
  
    version  
        Revit Server version  
        Type str  
  
    machine_name  
        Revit Server machine name  
        Type str  
  
    roles  
        List of server roles  
        Type list<ServerRole>  
  
    access_level_types  
        List of access level types  
        Type list<str>  
  
    max_path_length  
        Maximum allowed length for server paths  
        Type int  
  
    max_name_length  
        Maximum allowed length for server entry names  
        Type int
```

servers

List of neighbour servers

Type list<str>**class** rpws.models.ServerRole

Enum representing various server role codes.

Host = 0**Accelerator** = 1**Admin** = 2**class** rpws.models.ServerDriveInfo (*drive_space, drive_freespace*)

namedtuple for server drive info

drive_space

Total drive space in bytes

Type int**drive_freespace**

Free drive space in bytes

Type int**class** rpws.models.EntryContents (*path, drive_space, drive_freespace, files, folders, lock_context, lock_state, locks_inprogress, models*)

namedtuple for server entry contents (Encapsulates result of /contents)

path

Full path of this entry on server

Type str**drive_space**

Total server drive space in bytes

Type int**drive_freespace**

Free server drive space in bytes

Type int**files**

List of all files under this entry

Type list<FileInfo>**folders**

List of all folders under this entry

Type list<FolderInfo>**lock_context**

Lock context on this entry

Type str**lock_state**

Lock state on this entry

Type *LockState*

locks_inprogress

List of all in-progress locks

Type list<IPLockInfo>

models

List of all models under this entry

Type list<ModelInfo>

class rpws.models.**EntryDirInfo** (*path, name, size, date_created, date_modified, exists, folder_count, is_folder, last_modified_by, lock_context, lock_state, model_count, model_size, locks_inprogress*)

namedtuple for server entry directory info (Encapsulates result of /directoryinfo)

path

Full path of this entry on server

Type str

name

Entry name including extension

Type str

size

Entry size in bytes

Type int

date_created

Date entry was created

Type *DateEntry*

date_modified

Date entry was last modified

Type *DateEntry*

exists

True if entry exists

Type bool

folder_count

Number of sub folders

Type int

is_folder

True of this entry if a directory

Type bool

last_modified_by

Username of user who last modified the model

Type str

lock_context

Lock context on this entry

Type str

lock_state

Lock state on this entry

Type *LockState*

model_count

Number of models under this entry

Type int

model_size

If this entry is a model, size of model in bytes

Type int

locks_inprogress

List of all in-progress locks

Type list<IPLockInfo>

class rpws.models.**LockState**

Enum representing Revit Server lock states.

Unlocked = 0

Locked = 1

LockedParent = 2

LockedChild = 3

BeingUnlocked = 4

BeingLocked = 5

class rpws.models.**LockOptions**

Enum representing Revit Server lock options.

NotSet = 0

Read = 1

Write = 2

NonExclusiveReadWrite = 128

ReadAndNonExclusiveReadWrite = 129

WriteAndNonExclusiveReadWrite = 130

ReadWriteAndNonExclusiveReadWrite = 130

class rpws.models.**LockType**

Enum representing Revit Server lock type.

Data = 0

Permission = 1

class rpws.models.**FileInfo** (*path, name, size, is_text*)

namedtuple for info on a server file

path

Full path of this file on server

Type str

name

File name including extension

Type str

size
File size in bytes
Type int

is_text
True if text file
Type bool

class rpws.models.**FolderInfo**(*path, name, size, has_contents, lock_context, lock_state, locks_inprogress*)
namedtuple for info on a server folder

path
Full path of this folder on server
Type str

name
File name including extension
Type str

size
File size in bytes
Type int

has_contents
True if folder has contents
Type bool

lock_context
Lock context on this entry
Type str

lock_state
Lock state on this entry
Type *LockState*

locks_inprogress
List of all in-progress locks
Type list<IPLockInfo>

class rpws.models.**ModelInfo**(*path, name, size, support_size, product_version, lock_context, lock_state, locks_inprogress*)
namedtuple for info on a server model

path
Full path of this model on server
Type str

name
File name including extension
Type str

size
File size in bytes
Type int

support_size
Size of all support files in bytes

Type int

product_version
Revit version of this model

Type int

lock_context
Lock context on this entry

Type str

lock_state
Lock state on this entry

Type *LockState*

locks_inprogress
List of all in-progress locks

Type list<IPLockInfo>

class rpws.models.**ModelInfoEx** (*path, name, size, guid, date_created, date_modified,*
last_modified_by, support_size)
namedtuple for extended info on a server model

path
Full path of this model on server

Type str

name
File name including extension

Type str

size
File size in bytes

Type int

guid
GUID of the model

Type str

date_created
Date model was created

Type *DateEntry*

date_modified
Date model was last modified

Type *DateEntry*

last_modified_by
Username of user who last modified the model

Type str

support_size
Size of all support files in bytes

Type int

class rpws.models.**ProjectInfo** (*parameters*)
namedtuple for project info of a hosted model

parameters

List of project parameters in this model

Type list<ProjParameter>

class rpws.models.**ProjParameter** (*name, value, id, category, type, datatype*)
namedtuple for info project parameters in a hosted model

name

Parameter display name

Type str

value

Parameter value

Type str

id

Parameter unique id

Type str

category

Parameter category name

Type str

type

Parameter Type

Type *ParamType*

datatype

Parameter storage type

Type *ParamDataType*

class rpws.models.**ParamType**
Enum representing parameter types.

System = 'system'

Custom = 'custom'

Shared = 'shared'

Unknown = 'unknown'

class rpws.models.**ParamDataType**
Enum representing parameter storage types.

Length = 'length'

Number = 'number'

Material = 'material'

Text = 'text'

MultilineText = "multiline text"

YesNo = 'yes/no'

```
Unknown = 'unknown'

class rpws.models.MHistoryInfo(path, items)
    namedtuple for model history info

    path
        Full path of the hosted model
        Type str

    items
        List of history items
        Type list<MHistoryItemInfo>

class rpws.models.MHistoryItemInfo(id, comment, date, model_size, overwrittenby, support_size, user)
    namedtuple for model history item info

    id
        History item id
        Type str

    comment
        Comment recorded when syncing
        Type str

    date
        Date this history item was recorded
        Type DateEntry

    size
        File size in bytes
        Type int

    overwrittenby
        Id of history item overwriting this item
        Type str

    support_size
        Size of all support files in bytes
        Type int

    user
        Username recording the history item (by syncing model)
        Type str

class rpws.models.IPLockInfo(age, lock_options, lock_type, model_path, timestamp, username)
    namedtuple for info on an in-progress entry lock

    age
        Age of the lock
        Type str

    lock_options
        Lock options
        Type int
```

lock_type

Lock type

Type int**model_path**

Full path of the model being locked

Type str**timestamp**

Timestamp for when the lock was initiated

Type *DateEntry***username**

Username who initiated this lock

Type str**class** rpws.models.**ChildrenLockInfo** (*path, items, lock_context*)

namedtuple for info locked children under an entry

path

Full path of parent entry

Type str**items**

List of paths of locked models under this entry

Type list<str>**lock_context**

Lock context of parent entry

Type str**class** rpws.models.**DateEntry**

Timestamp data type wrapping Revit Server string timestamps.

Wraps Revit Server string timestamps in a typical python datetime.datetime subclass.

Example

```
>>> ts = DateEntry.fromrsdatestring("/Date(1483465201000)/")
DateEntry(2017, 1, 3, 17, 40, 1)
```

class rpws.models.**TimeSpanEntry**

Timespan data type wrapping Revit Server timespan.

Wraps Revit Server string timespan in a typical python datetime.timedelta subclass

Example

```
>>> ts = TimeSpanEntry.fromrstimespanstring("PT11M42.5154811S")
TimeSpanEntry(0, 5856, 811000)
```


r

`rpws.server, ??`

Symbols

`_base_uri` (*rpws.server.RevitServer* attribute), 1
`_hmachine` (*rpws.server.RevitServer* attribute), 1
`_huser` (*rpws.server.RevitServer* attribute), 1

A

`access_level_types` (*rpws.models.ServerInfo* attribute), 9
`age` (*rpws.models.IPLockInfo* attribute), 16

C

`cancellock()` (*rpws.server.RevitServer* method), 2
`category` (*rpws.models.ProjParameter* attribute), 15
`ChildrenLockInfo` (class in *rpws.models*), 17
`comment` (*rpws.models.MHistoryItemInfo* attribute), 16
`copy()` (*rpws.server.RevitServer* method), 2

D

`datatype` (*rpws.models.ProjParameter* attribute), 15
`date` (*rpws.models.MHistoryItemInfo* attribute), 16
`date_created` (*rpws.models.EntryDirInfo* attribute), 11
`date_created` (*rpws.models.ModelInfoEx* attribute), 14
`date_modified` (*rpws.models.EntryDirInfo* attribute), 11
`date_modified` (*rpws.models.ModelInfoEx* attribute), 14
`DateEntry` (class in *rpws.models*), 17
`delete()` (*rpws.server.RevitServer* method), 2
`deletedescendentlocks()` (*rpws.server.RevitServer* method), 2
`drive_freespace` (*rpws.models.EntryContents* attribute), 10
`drive_freespace` (*rpws.models.ServerDriveInfo* attribute), 10
`drive_space` (*rpws.models.EntryContents* attribute), 10

`drive_space` (*rpws.models.ServerDriveInfo* attribute), 10

E

`EntryContents` (class in *rpws.models*), 10
`EntryDirInfo` (class in *rpws.models*), 11
`exists` (*rpws.models.EntryDirInfo* attribute), 11

F

`FileInfo` (class in *rpws.models*), 12
`files` (*rpws.models.EntryContents* attribute), 10
`folder_count` (*rpws.models.EntryDirInfo* attribute), 11
`FolderInfo` (class in *rpws.models*), 13
`folders` (*rpws.models.EntryContents* attribute), 10

G

`getdescendentlocks()` (*rpws.server.RevitServer* method), 3
`getdriveinfo()` (*rpws.server.RevitServer* method), 3
`getfolderinfo()` (*rpws.server.RevitServer* method), 3
`getinfo()` (*rpws.server.RevitServer* method), 3
`getmodelhistory()` (*rpws.server.RevitServer* method), 4
`getmodelinfo()` (*rpws.server.RevitServer* method), 4
`getprojectinfo()` (*rpws.server.RevitServer* method), 4
`guid` (*rpws.models.ModelInfoEx* attribute), 14

H

`has_contents` (*rpws.models.FolderInfo* attribute), 13

I

`id` (*rpws.models.MHistoryItemInfo* attribute), 16
`id` (*rpws.models.ProjParameter* attribute), 15
`IPLockInfo` (class in *rpws.models*), 16
`is_folder` (*rpws.models.EntryDirInfo* attribute), 11
`is_text` (*rpws.models.FileInfo* attribute), 13

items (*rpws.models.ChildrenLockInfo* attribute), 17
 items (*rpws.models.MHHistoryInfo* attribute), 16

L

last_modified_by (*rpws.models.EntryDirInfo* attribute), 11
 last_modified_by (*rpws.models.ModelInfoEx* attribute), 14
 listfiles() (*rpws.server.RevitServer* method), 5
 listfolders() (*rpws.server.RevitServer* method), 5
 listmodels() (*rpws.server.RevitServer* method), 5
 lock() (*rpws.server.RevitServer* method), 6
 lock_context (*rpws.models.ChildrenLockInfo* attribute), 17
 lock_context (*rpws.models.EntryContents* attribute), 10
 lock_context (*rpws.models.EntryDirInfo* attribute), 11
 lock_context (*rpws.models.FolderInfo* attribute), 13
 lock_context (*rpws.models.ModelInfo* attribute), 14
 lock_options (*rpws.models.IPLockInfo* attribute), 16
 lock_state (*rpws.models.EntryContents* attribute), 10
 lock_state (*rpws.models.EntryDirInfo* attribute), 11
 lock_state (*rpws.models.FolderInfo* attribute), 13
 lock_state (*rpws.models.ModelInfo* attribute), 14
 lock_type (*rpws.models.IPLockInfo* attribute), 16
 LockOptions (class in *rpws.models*), 12
 locks_inprogress (*rpws.models.EntryContents* attribute), 10
 locks_inprogress (*rpws.models.EntryDirInfo* attribute), 12
 locks_inprogress (*rpws.models.FolderInfo* attribute), 13
 locks_inprogress (*rpws.models.ModelInfo* attribute), 14
 LockState (class in *rpws.models*), 12
 LockType (class in *rpws.models*), 12

M

machine_name (*rpws.models.ServerInfo* attribute), 9
 max_name_length (*rpws.models.ServerInfo* attribute), 9
 max_path_length (*rpws.models.ServerInfo* attribute), 9
 MHHistoryInfo (class in *rpws.models*), 16
 MHHistoryItemInfo (class in *rpws.models*), 16
 mkdir() (*rpws.server.RevitServer* method), 6
 model_count (*rpws.models.EntryDirInfo* attribute), 12
 model_path (*rpws.models.IPLockInfo* attribute), 17
 model_size (*rpws.models.EntryDirInfo* attribute), 12
 ModelInfo (class in *rpws.models*), 13
 ModelInfoEx (class in *rpws.models*), 14
 models (*rpws.models.EntryContents* attribute), 11

move() (*rpws.server.RevitServer* method), 6

N

name (*rpws.models.EntryDirInfo* attribute), 11
 name (*rpws.models.FileInfo* attribute), 12
 name (*rpws.models.FolderInfo* attribute), 13
 name (*rpws.models.ModelInfo* attribute), 13
 name (*rpws.models.ModelInfoEx* attribute), 14
 name (*rpws.models.ProjParameter* attribute), 15
 name (*rpws.models.ServerInfo* attribute), 9
 name (*rpws.server.RevitServer* attribute), 1

O

overwrittenby (*rpws.models.MHHistoryItemInfo* attribute), 16

P

ParamDataType (class in *rpws.models*), 15
 parameters (*rpws.models.ProjectInfo* attribute), 15
 ParamType (class in *rpws.models*), 15
 path (*rpws.models.ChildrenLockInfo* attribute), 17
 path (*rpws.models.EntryContents* attribute), 10
 path (*rpws.models.EntryDirInfo* attribute), 11
 path (*rpws.models.FileInfo* attribute), 12
 path (*rpws.models.FolderInfo* attribute), 13
 path (*rpws.models.MHHistoryInfo* attribute), 16
 path (*rpws.models.ModelInfo* attribute), 13
 path (*rpws.models.ModelInfoEx* attribute), 14
 path (*rpws.server.RevitServer* attribute), 6
 product_version (*rpws.models.ModelInfo* attribute), 14
 ProjectInfo (class in *rpws.models*), 15
 ProjParameter (class in *rpws.models*), 15

R

rename() (*rpws.server.RevitServer* method), 6
 RevitServer (class in *rpws.server*), 1
 rmdir() (*rpws.server.RevitServer* method), 7
 roles (*rpws.models.ServerInfo* attribute), 9
 rpws.server (module), 1

S

scandir() (*rpws.server.RevitServer* method), 7
 ServerDriveInfo (class in *rpws.models*), 10
 ServerInfo (class in *rpws.models*), 9
 ServerRole (class in *rpws.models*), 10
 servers (*rpws.models.ServerInfo* attribute), 9
 size (*rpws.models.EntryDirInfo* attribute), 11
 size (*rpws.models.FileInfo* attribute), 12
 size (*rpws.models.FolderInfo* attribute), 13
 size (*rpws.models.MHHistoryItemInfo* attribute), 16
 size (*rpws.models.ModelInfo* attribute), 13
 size (*rpws.models.ModelInfoEx* attribute), 14

support_size (rpws.models.MHistoryItemInfo attribute), 16
support_size (rpws.models.ModelInfo attribute), 13
support_size (rpws.models.ModelInfoEx attribute), 14

T

TimeSpanEntry (class in rpws.models), 17
timestamp (rpws.models.IPLockInfo attribute), 17
type (rpws.models.ProjParameter attribute), 15

U

unlock() (rpws.server.RevitServer method), 7
user (rpws.models.MHistoryItemInfo attribute), 16
username (rpws.models.IPLockInfo attribute), 17

V

value (rpws.models.ProjParameter attribute), 15
version (rpws.models.ServerInfo attribute), 9
version (rpws.server.RevitServer attribute), 1

W

walk() (rpws.server.RevitServer method), 7