

# Parallels Server Python API Reference

API Documentation

June 11, 2008

## Contents

<b>Contents</b>	<b>1</b>
<b>1 Module prlsdk</b>	<b>2</b>
1.1 Modules . . . . .	2
1.2 Functions . . . . .	2
1.3 Variables . . . . .	3
1.4 Class Base . . . . .	3
1.4.1 Methods . . . . .	3
1.4.2 Properties . . . . .	4
1.5 Class BaseAsync . . . . .	4
1.5.1 Methods . . . . .	5
1.5.2 Properties . . . . .	5
1.6 Class BaseServer . . . . .	6
1.6.1 Methods . . . . .	6
1.6.2 Properties . . . . .	8
1.7 Class BaseVm . . . . .	8
1.7.1 Methods . . . . .	8
1.7.2 Properties . . . . .	9
1.8 Class Event . . . . .	9
1.8.1 Methods . . . . .	10
1.8.2 Properties . . . . .	11
1.9 Class Job . . . . .	11
1.9.1 Methods . . . . .	11
1.9.2 Properties . . . . .	12
1.10 Class OwnerRefAsync . . . . .	13
1.10.1 Methods . . . . .	13
1.10.2 Properties . . . . .	13
1.11 Class ParallelsError . . . . .	14
1.11.1 Methods . . . . .	14
1.11.2 Properties . . . . .	14
1.12 Class PrlJobTimeoutError . . . . .	15
1.12.1 Methods . . . . .	15
1.12.2 Properties . . . . .	15
1.13 Class PrlNotImplError . . . . .	16
1.13.1 Methods . . . . .	16
1.13.2 Properties . . . . .	16
1.14 Class PrlSDKAsyncError . . . . .	17

1.14.1	Methods . . . . .	17
1.14.2	Properties . . . . .	17
1.15	Class PrlSDKError . . . . .	18
1.15.1	Methods . . . . .	18
1.15.2	Properties . . . . .	19
1.16	Class PrlStateError . . . . .	19
1.16.1	Methods . . . . .	19
1.16.2	Properties . . . . .	19
1.17	Class Question . . . . .	20
1.17.1	Methods . . . . .	20
1.17.2	Properties . . . . .	21
1.18	Class Result . . . . .	21
1.18.1	Methods . . . . .	21
1.18.2	Properties . . . . .	22
1.19	Class ServerInfo . . . . .	23
1.19.1	Methods . . . . .	23
1.19.2	Properties . . . . .	24
<b>2</b>	<b>Module prlsdk.consts</b>	<b>25</b>
2.1	Variables . . . . .	25
<b>3</b>	<b>Module prlsdk.errors</b>	<b>38</b>
3.1	Variables . . . . .	38
<b>4</b>	<b>Module prlsdk'</b>	<b>73</b>
4.1	Functions . . . . .	73
4.2	Variables . . . . .	74
4.3	Class Base . . . . .	74
4.3.1	Methods . . . . .	74
4.3.2	Properties . . . . .	75
4.4	Class BaseAsync . . . . .	75
4.4.1	Methods . . . . .	76
4.4.2	Properties . . . . .	76
4.5	Class BaseServer . . . . .	77
4.5.1	Methods . . . . .	77
4.5.2	Properties . . . . .	79
4.6	Class BaseVm . . . . .	79
4.6.1	Methods . . . . .	79
4.6.2	Properties . . . . .	80
4.7	Class Event . . . . .	80
4.7.1	Methods . . . . .	81
4.7.2	Properties . . . . .	82
4.8	Class Job . . . . .	82
4.8.1	Methods . . . . .	82
4.8.2	Properties . . . . .	83
4.9	Class OwnerRefAsync . . . . .	84
4.9.1	Methods . . . . .	84
4.9.2	Properties . . . . .	84
4.10	Class ParallelsError . . . . .	85
4.10.1	Methods . . . . .	85
4.10.2	Properties . . . . .	85
4.11	Class PrlJobTimeoutError . . . . .	86

4.11.1	Methods . . . . .	86
4.11.2	Properties . . . . .	86
4.12	Class PrlNotImplError . . . . .	87
4.12.1	Methods . . . . .	87
4.12.2	Properties . . . . .	87
4.13	Class PrlSDKAsyncError . . . . .	88
4.13.1	Methods . . . . .	88
4.13.2	Properties . . . . .	88
4.14	Class PrlSDKError . . . . .	89
4.14.1	Methods . . . . .	89
4.14.2	Properties . . . . .	90
4.15	Class PrlStateError . . . . .	90
4.15.1	Methods . . . . .	90
4.15.2	Properties . . . . .	90
4.16	Class Question . . . . .	91
4.16.1	Methods . . . . .	91
4.16.2	Properties . . . . .	92
4.17	Class Result . . . . .	92
4.17.1	Methods . . . . .	92
4.17.2	Properties . . . . .	93
4.18	Class ServerInfo . . . . .	94
4.18.1	Methods . . . . .	94
4.18.2	Properties . . . . .	95
<b>5</b>	<b>Package prlsdkapi</b>	<b>96</b>
5.1	Functions . . . . .	96
5.2	Variables . . . . .	97
5.3	Class DispConfig . . . . .	97
5.3.1	Methods . . . . .	97
5.3.2	Properties . . . . .	98
5.3.3	Class Variables . . . . .	99
5.4	Class DispNet . . . . .	100
5.4.1	Methods . . . . .	100
5.4.2	Properties . . . . .	100
5.4.3	Class Variables . . . . .	101
5.5	Class FSEntry . . . . .	102
5.5.1	Methods . . . . .	102
5.5.2	Properties . . . . .	102
5.5.3	Class Variables . . . . .	103
5.6	Class FSInfo . . . . .	103
5.6.1	Methods . . . . .	103
5.6.2	Properties . . . . .	103
5.6.3	Class Variables . . . . .	104
5.7	Class FoundVm . . . . .	104
5.7.1	Methods . . . . .	104
5.7.2	Properties . . . . .	105
5.7.3	Class Variables . . . . .	105
5.8	Class LicenceInfo . . . . .	105
5.8.1	Methods . . . . .	105
5.8.2	Properties . . . . .	106
5.8.3	Class Variables . . . . .	106
5.9	Class LoginResponse . . . . .	107

5.9.1	Methods . . . . .	107
5.9.2	Properties . . . . .	107
5.9.3	Class Variables . . . . .	108
5.10	Class NetServiceStatus . . . . .	108
5.10.1	Methods . . . . .	108
5.10.2	Properties . . . . .	108
5.10.3	Class Variables . . . . .	109
5.11	Class PrlIterator . . . . .	109
5.11.1	Methods . . . . .	109
5.11.2	Properties . . . . .	110
5.12	Class PrlObject . . . . .	110
5.12.1	Methods . . . . .	110
5.12.2	Properties . . . . .	110
5.13	Class PrlReadOnlyAttrError . . . . .	111
5.13.1	Methods . . . . .	111
5.13.2	Properties . . . . .	111
5.14	Class PrlResultIter . . . . .	112
5.14.1	Methods . . . . .	112
5.14.2	Properties . . . . .	112
5.15	Class RunningTask . . . . .	112
5.15.1	Methods . . . . .	113
5.15.2	Properties . . . . .	113
5.15.3	Class Variables . . . . .	113
5.16	Class Server . . . . .	114
5.16.1	Methods . . . . .	114
5.16.2	Properties . . . . .	120
5.16.3	Class Variables . . . . .	120
5.17	Class SrvCfgDev . . . . .	121
5.17.1	Methods . . . . .	121
5.17.2	Properties . . . . .	121
5.17.3	Class Variables . . . . .	122
5.18	Class SrvCfgHdd . . . . .	122
5.18.1	Methods . . . . .	122
5.18.2	Properties . . . . .	123
5.18.3	Class Variables . . . . .	123
5.19	Class SrvCfgHddPart . . . . .	123
5.19.1	Methods . . . . .	123
5.19.2	Properties . . . . .	124
5.19.3	Class Variables . . . . .	124
5.20	Class SrvCfgNet . . . . .	125
5.20.1	Methods . . . . .	125
5.20.2	Properties . . . . .	125
5.20.3	Class Variables . . . . .	125
5.21	Class SrvConfig . . . . .	126
5.21.1	Methods . . . . .	126
5.21.2	Properties . . . . .	126
5.21.3	Class Variables . . . . .	127
5.22	Class SrvInfo . . . . .	127
5.22.1	Methods . . . . .	127
5.22.2	Properties . . . . .	127
5.22.3	Class Variables . . . . .	128

5.23	Class UserConfig . . . . .	128
5.23.1	Methods . . . . .	128
5.23.2	Properties . . . . .	129
5.23.3	Class Variables . . . . .	129
5.24	Class UsrInfo . . . . .	129
5.24.1	Methods . . . . .	130
5.24.2	Properties . . . . .	130
5.24.3	Class Variables . . . . .	130
5.25	Class Vm . . . . .	131
5.25.1	Methods . . . . .	131
5.25.2	Properties . . . . .	142
5.25.3	Class Variables . . . . .	146
5.26	Class VmAccessRights . . . . .	146
5.26.1	Methods . . . . .	146
5.26.2	Properties . . . . .	147
5.27	Class VmBaseDev . . . . .	147
5.27.1	Methods . . . . .	147
5.27.2	Properties . . . . .	149
5.27.3	Class Variables . . . . .	149
5.28	Class VmBaseDiskDev . . . . .	150
5.28.1	Methods . . . . .	150
5.28.2	Properties . . . . .	150
5.28.3	Class Variables . . . . .	151
5.29	Class VmBasePortDev . . . . .	151
5.29.1	Methods . . . . .	151
5.29.2	Properties . . . . .	152
5.29.3	Class Variables . . . . .	152
5.30	Class VmBootDev . . . . .	153
5.30.1	Methods . . . . .	153
5.30.2	Properties . . . . .	153
5.30.3	Class Variables . . . . .	154
5.31	Class VmCPUStat . . . . .	154
5.31.1	Methods . . . . .	154
5.31.2	Properties . . . . .	154
5.31.3	Class Variables . . . . .	155
5.32	Class VmDiskPartStat . . . . .	155
5.32.1	Methods . . . . .	155
5.32.2	Properties . . . . .	155
5.32.3	Class Variables . . . . .	156
5.33	Class VmDiskStat . . . . .	156
5.33.1	Methods . . . . .	156
5.33.2	Properties . . . . .	157
5.33.3	Class Variables . . . . .	157
5.34	Class VmDisplayDev . . . . .	157
5.34.1	Methods . . . . .	158
5.34.2	Properties . . . . .	160
5.34.3	Class Variables . . . . .	160
5.35	Class VmFloppyDev . . . . .	161
5.35.1	Methods . . . . .	161
5.35.2	Properties . . . . .	161
5.35.3	Class Variables . . . . .	162

5.36	Class VmHddDev . . . . .	162
5.36.1	Methods . . . . .	162
5.36.2	Properties . . . . .	163
5.36.3	Class Variables . . . . .	163
5.37	Class VmIFaceStat . . . . .	164
5.37.1	Methods . . . . .	164
5.37.2	Properties . . . . .	164
5.37.3	Class Variables . . . . .	165
5.38	Class VmInfo . . . . .	165
5.38.1	Methods . . . . .	165
5.38.2	Properties . . . . .	165
5.38.3	Class Variables . . . . .	166
5.39	Class VmNetDev . . . . .	166
5.39.1	Methods . . . . .	166
5.39.2	Properties . . . . .	167
5.39.3	Class Variables . . . . .	167
5.40	Class VmObjEmulation . . . . .	167
5.40.1	Methods . . . . .	168
5.40.2	Properties . . . . .	168
5.41	Class VmOpticalDev . . . . .	168
5.41.1	Methods . . . . .	169
5.41.2	Properties . . . . .	169
5.41.3	Class Variables . . . . .	169
5.42	Class VmParallelPortDev . . . . .	170
5.42.1	Methods . . . . .	170
5.42.2	Properties . . . . .	170
5.42.3	Class Variables . . . . .	171
5.43	Class VmProcStat . . . . .	171
5.43.1	Methods . . . . .	171
5.43.2	Properties . . . . .	171
5.43.3	Class Variables . . . . .	172
5.44	Class VmScrRes . . . . .	172
5.44.1	Methods . . . . .	173
5.44.2	Properties . . . . .	173
5.44.3	Class Variables . . . . .	173
5.45	Class VmSerialDev . . . . .	174
5.45.1	Methods . . . . .	174
5.45.2	Properties . . . . .	174
5.45.3	Class Variables . . . . .	175
5.46	Class VmShare . . . . .	175
5.46.1	Methods . . . . .	175
5.46.2	Properties . . . . .	176
5.46.3	Class Variables . . . . .	176
5.47	Class VmSoundDev . . . . .	177
5.47.1	Methods . . . . .	177
5.47.2	Properties . . . . .	177
5.47.3	Class Variables . . . . .	178
5.48	Class VmStat . . . . .	178
5.48.1	Methods . . . . .	178
5.48.2	Properties . . . . .	180
5.48.3	Class Variables . . . . .	180

---

5.49	Class VmToolsInfo . . . . .	181
5.49.1	Methods . . . . .	181
5.49.2	Properties . . . . .	181
5.49.3	Class Variables . . . . .	181
5.50	Class VmUsbDev . . . . .	182
5.50.1	Methods . . . . .	182
5.50.2	Properties . . . . .	182
5.50.3	Class Variables . . . . .	183
5.51	Class VmUserStat . . . . .	183
5.51.1	Methods . . . . .	183
5.51.2	Properties . . . . .	184
5.51.3	Class Variables . . . . .	184
5.52	Class VmVNCConfig . . . . .	184
5.52.1	Methods . . . . .	184
5.52.2	Properties . . . . .	185
5.52.3	Class Variables . . . . .	185
5.53	Class deinit_sdk . . . . .	185
5.53.1	Methods . . . . .	185
5.53.2	Properties . . . . .	185
5.53.3	Class Variables . . . . .	186

# 1 Module prlsdk

## 1.1 Modules

- **consts** (Section 2, p. 25)
- **errors** (Section 3, p. 38)

## 1.2 Functions

<b>CheckServerAlive</b> ( <i>hostname</i> , <i>timeout=DefaultTimeout</i> ) <hr/> <p>Lets to determine whether Parallels server on specified host alive If alive - ServerInfo will be returned else None.</p> <p><b>Return Value</b> ServerInfo</p>
<b>DefaultJobTimeout</b> (...)
<b>DefaultKeyActWait</b> (...)
<b>DefaultTimeout</b> (...)
<b>DeinitializeSDK</b> () <hr/> <p>Method force SDK deinitialization.</p>
<b>GetSDKLibraryPath</b> () <hr/> <p>Return path from witch library will be loaded.</p> <p><b>Return Value</b> string</p>
<b>GetServerList</b> ( <i>msec</i> ) <hr/> <p>Lookups Parallels servers at the LAN during specified timeout.</p> <p><b>Return Value</b> list of ServerInfo objects</p>
<b>InitializeSDK</b> () <hr/> <p>Method initialize SDK. Return False if it was already initialized, else True.</p> <p><b>Return Value</b> bool</p>

<b>IsSDKInitialized()</b>
Return True if SDK allready loaded.
<b>Return Value</b> bool

<b>SetSDKLibraryPath(<i>new_path</i>)</b>
Return path from witch library will be loaded. Raise exception if SDK API allready initialized.
<b>Return Value</b> string

<b>__debug_level__(...)</b>
-----------------------------

### 1.3 Variables

Name	Description
__SDK__	<b>Value:</b> <prlsdk.__SDK__ instance at 0x00A55130, version '3.0.2130...>

### 1.4 Class Base



**Known Subclasses:** prlsdk.BaseAsync, prlsdk.BaseServer, prlsdk.Event, prlsdk.Job, prlsdk.ServerInfo, prlsdkapi.FoundVm

Base objects

#### 1.4.1 Methods

<b>__cmp__(<i>x, y</i>)</b>
cmp(x,y)

<b>__init__(...)</b>
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

<b><code>--new--</code></b> ( <i>T</i> , <i>S</i> , ...)
<b>Return Value</b> a new object with type <i>S</i> , a subtype of <i>T</i>
Overrides: <code>object.--new--</code>

<b><code>--repr--</code></b> ( <i>x</i> )
<code>repr(x)</code>
Overrides: <code>object.--repr--</code>

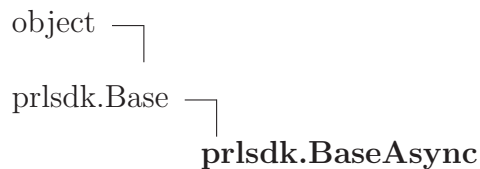
### *Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`,  
`--str--()`

#### 1.4.2 Properties

Name	Description
<code>--handle--</code>	
<code>handle_type</code>	handle type
<i>Inherited from object</i>	
<code>--class--</code>	

## 1.5 Class BaseAsync



**Known Subclasses:** `prlsdk.BaseVm`, `prlsdk.OwnerRefAsync`, `prlsdk.Result`, `prlsdkapi.FSEntry`,  
`prlsdkapi.FSInfo`, `prlsdkapi.LicenceInfo`, `prlsdkapi.LoginResponse`, `prlsdkapi.NetServiceStatus`,  
`prlsdkapi.RunningTask`, `prlsdkapi.SrvCfgDev`, `prlsdkapi.SrvCfgHddPart`, `prlsdkapi.SrvConfig`,  
`prlsdkapi.SrvInfo`, `prlsdkapi.UserConfig`, `prlsdkapi.UsrInfo`, `prlsdkapi.VmCPUStat`, `prlsdkapi.VmDiskPartS`,  
`prlsdkapi.VmDiskStat`, `prlsdkapi.VmIFaceStat`, `prlsdkapi.VmInfo`, `prlsdkapi.VmProcStat`, `prls-`  
`dkapi.VmStat`, `prlsdkapi.VmToolsInfo`, `prlsdkapi.VmUserStat`

Base async objects

## 1.5.1 Methods

---

**\_\_cmp\_\_**(*x, y*)

---

 cmp(x,y)

---

 Overrides: prlsdk.Base.\_\_cmp\_\_

---

**\_\_init\_\_**(...)

---

 x.\_\_init\_\_(...) initializes x; see x.\_\_class\_\_.\_\_doc\_\_ for signature

---

 Overrides: object.\_\_init\_\_

---

**\_\_new\_\_**(*T, S, ...*)
**Return Value**

a new object with type S, a subtype of T

---

 Overrides: object.\_\_new\_\_

---

**\_\_repr\_\_**(*x*)

---

 repr(x)

---

 Overrides: object.\_\_repr\_\_

---

**cancel**(*BaseAsync*)

---

 Start canceling this job, this method is asynchone so it return Result object.
**Return Value**

Result

---

**wait**(*BaseAsync, msec=DefaultJobWait*)

---

 Waits for job finished for passed milliseconds if successefully finished self object is returned If timeout -> PrlJobTimeoutError will be raised . If job finished with error ParallelsError will be raised.
**Return Value**

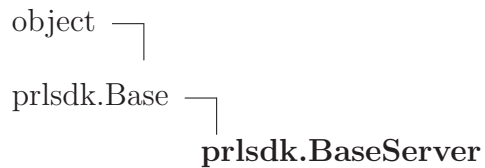
BaseAsync

**Inherited from object**
 \_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(),  
 \_\_str\_\_()

## 1.5.2 Properties

Name	Description
<code>--expected_h_type--</code>	
<code>--job--</code>	
<code>finished</code>	Shows job status
<code>handle_type</code>	Returns handle type
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

## 1.6 Class BaseServer



**Known Subclasses:** prlsdkapi.Server

Server base object class

### 1.6.1 Methods

<code>--init--(...)</code>
<code>x.__init__(...)</code> initializes x; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

<code>--new--(T, S, ...)</code>
<b>Return Value</b>
a new object with type S, a subtype of T
Overrides: <code>object.__new__</code>

**login**(*Server*, *host*, *user*, *password*, *port*=0, *secur\_level*=LOW, *session\_uid*=, *timeout*=0)

*secur\_level* – see `prlsdk.consts.PSL_xxx` Login operation should be performed before to allow further server operations. Through Result object you could check that operation is finished. All method that checks for connection will wait until login finished.

**Return Value**

Result object

**login\_local**(*Server*, *port*=0, *secur\_level*=LOW, *session\_uid*=)

*secur\_level* – see `prlsdk.consts.PSL_xxx` Login operation should be performed before to allow further server operations. Through Result object you could check that operation is finished. All method that checks for connection will wait until login finished.

**Return Value**

Result object

**logoff**(*Server*)

Logoff operation will finish interactive connection session.

**Return Value**

Job object

**reg\_event\_handler**(*Server*, *handler*, *user\_data*=None)

Register server event handler. handler should take 2 arguments: Event object, *user\_data*.

**shutdown**(*srv*, *Srv*, *force*=False)

Let to shutdown server connection established with

**Return Value**

Job

**unreg\_event\_handler**(*Server*, *handler*)

Unregister server event handler.

*Inherited from prlsdk.Base(Section 4.3)*

`--cmp--()`, `--repr--()`

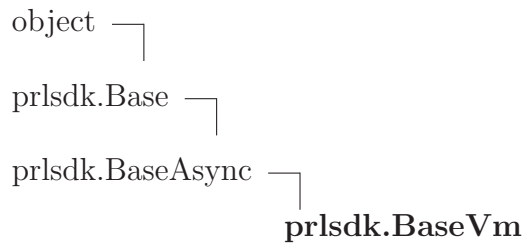
*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__str__()`

### 1.6.2 Properties

Name	Description
<code>__editing_common__</code>	
<code>__editing_user__</code>	
<code>__login_result__</code>	
<code>autostart_editing</code>	
<code>connected</code>	Return connected state
<code>last_session_uuid</code>	
<code>session_uuid</code>	Retruns last server session uuid
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code> , <code>handle_type</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

## 1.7 Class BaseVm



**Known Subclasses:** `prlsdkapi.Vm`

Vm base object class

### 1.7.1 Methods

<code>__init__(...)</code>
<code>x.__init__(...)</code> initializes x; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

```
__new__(T, S, ...)
```

**Return Value**

a new object with type S, a subtype of T

Overrides: object.\_\_new\_\_

**Inherited from prlsdk.BaseAsync (Section 4.4)**

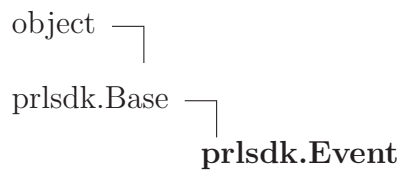
```
__cmp__(), __repr__(), cancel(), wait()
```

**Inherited from object**

```
__delattr__(), __getattr__(), __hash__(), __reduce__(), __reduce_ex__(), __setattr__(),
__str__()
```

**1.7.2 Properties**

Name	Description
<code>__editing__</code>	
<code>__unfin_op__</code>	
<code>autostart_editing</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>__expected_h_type__</code> , <code>__job__</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

**1.8 Class Event**

**Known Subclasses:** prlsdk.Question

Event object class

## 1.8.1 Methods

<code>__getitem__(x, y)</code>
<code>x[y]</code>

<code>__init__(...)</code>
<code>x.__init__(...)</code> initializes x; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

<code>__iter__(x)</code>
<code>iter(x)</code>

<code>__len__(x)</code>
<code>len(x)</code>

<code>__new__(T, S, ...)</code>
<b>Return Value</b> a new object with type S, a subtype of T
Overrides: <code>object.__new__</code>

<code>get_param(Event, index)</code>
Returns depending on parameter type some value with passed index.
<b>Return Value</b> value of any type

<code>get_params_count(Event, index)</code>
Returns parameters count.
<b>Return Value</b> int

*Inherited from `prlsdk.Base` (Section 4.3)*

`__cmp__()`, `__repr__()`

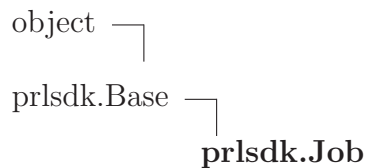
*Inherited from `object`*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`,  
`__str__()`

### 1.8.2 Properties

Name	Description
event_code	Returns event code, see prlsdk.consts.PMT_XXX.
event_str_code	Returns event code string representaion.
event_type	Returns event type, see prlsdk.consts.PMT_XXX.
explicit_text	Returns explicit event text.
params_count	Returns event param count
sender	Returns question sender, see prlsdk.consts.PIE_XXX.
text	Returns event text.
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__, handle_type	
<i>Inherited from object</i>	
__class__	

## 1.9 Class Job



Job objects

### 1.9.1 Methods

<b>__init__</b> (...) <hr/> x.__init__(...) initializes x; see x.__class__.__doc__ for signature Overrides: object.__init__
<b>__new__</b> (T, S, ...) <b>Return Value</b> a new object with type S, a subtype of T Overrides: object.__new__

**cancel**(*Job*)

Start canceling this job, this method is asynchone so it return another job.

**Return Value**

Result

**get\_held\_object**(*Job, name*)

Returns held onject by name, None if not held.

**Return Value**

object

**hold\_object**(*Job, name, obj*)

Add object to holding list.

**Return Value**

self

**wait**(*Job, msec=DefaultJobWait*)

Waits for job finished for passed milliseconds If timeout ->return False. If job finished with error ParallelsError will be raised.

**Return Value**

bool

***Inherited from prlsdk.Base (Section 4.3)***

`--cmp--()`, `--repr--()`

***Inherited from object***

`--delattr--()`, `--getattribute--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

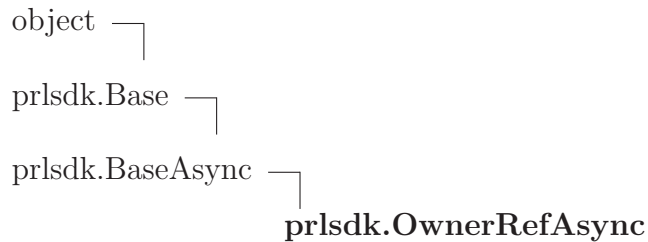
**1.9.2 Properties**

Name	Description
<code>--description--</code>	
<code>--ret_code--</code>	
<code>finished</code>	Shows job status
<code>op_code</code>	Return job operation code, see <code>prlsdk.consts.PJOC_xxx</code> .
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code> , <code>handle_type</code>	
<i>Inherited from object</i>	

*continued on next page*

Name	Description
__class__	

## 1.10 Class OwnerRefAsync



**Known Subclasses:** prlsdkapi.DispConfig, prlsdkapi.DispNet, prlsdkapi.VmBaseDev, prlsdkapi.VmBootDev, prlsdkapi.VmScrRes, prlsdkapi.VmShare

Base async object with owner reference

### 1.10.1 Methods

<b>__init__</b> (...)
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

<b>__new__</b> (T, S, ...)
<b>Return Value</b>
a new object with type S, a subtype of T
Overrides: object.__new__

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

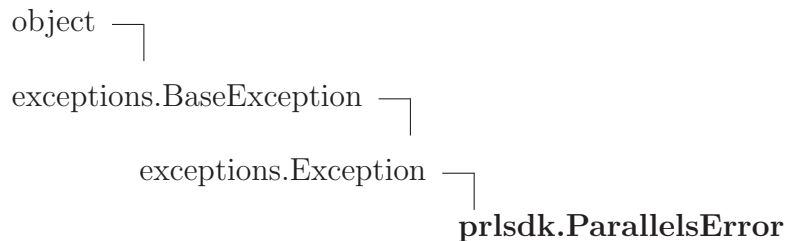
*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

### 1.10.2 Properties

Name	Description
<code>--owner--</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

## 1.11 Class ParallelsError



**Known Subclasses:** `prlsdk.PrlJobTimeoutError`, `prlsdk.PrlNotImplError`, `prlsdk.PrlSDKError`, `prlsdk.PrlStateError`

### 1.11.1 Methods

*Inherited from exceptions.Exception*

`--init--()`, `--new--()`

*Inherited from exceptions.BaseException*

`--delattr--()`, `--getattr--()`, `--getitem--()`, `--reduce--()`, `--repr--()`, `--setattr--()`, `--setstate--()`, `--str--()`

*Inherited from object*

`--hash--()`, `--reduce_ex--()`

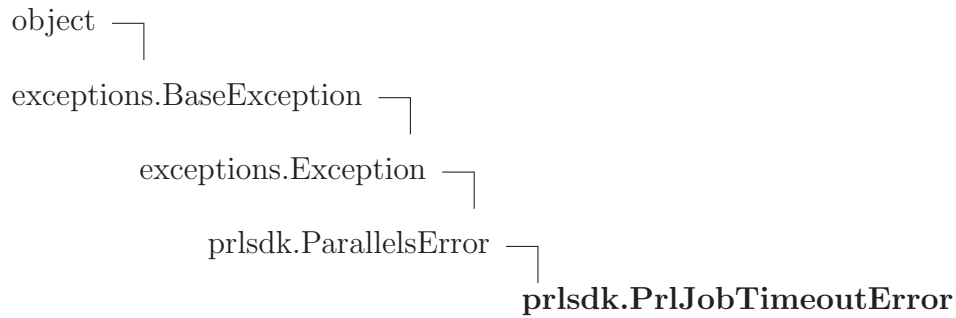
### 1.11.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	
<code>args</code> , <code>message</code>	
<i>Inherited from object</i>	

*continued on next page*

Name	Description
__class__	

## 1.12 Class PrlJobTimeoutError



### 1.12.1 Methods

#### *Inherited from exceptions.Exception*

`__init__()`, `__new__()`

#### *Inherited from exceptions.BaseException*

`__delattr__()`, `__getattr__()`, `__getitem__()`, `__reduce__()`, `__repr__()`, `__setattr__()`, `__setstate__()`, `__str__()`

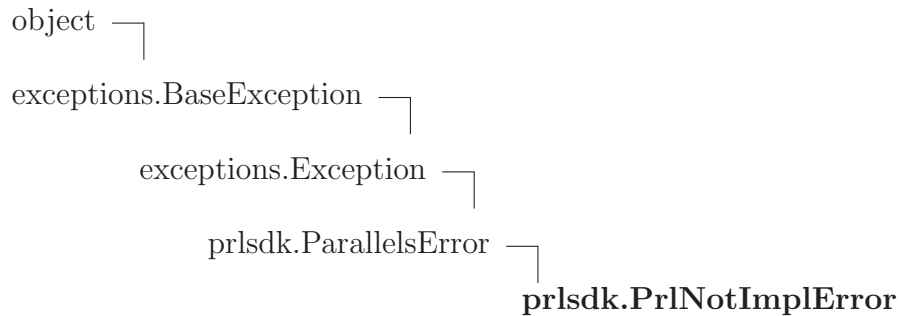
#### *Inherited from object*

`__hash__()`, `__reduce_ex__()`

### 1.12.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	
args, message	
<i>Inherited from object</i>	
__class__	

### 1.13 Class *PrlNotImplError*



#### 1.13.1 Methods

##### *Inherited from exceptions.Exception*

`__init__()`, `__new__()`

##### *Inherited from exceptions.BaseException*

`__delattr__()`, `__getattr__()`, `__getitem__()`, `__reduce__()`, `__repr__()`, `__setattr__()`,  
`__setstate__()`, `__str__()`

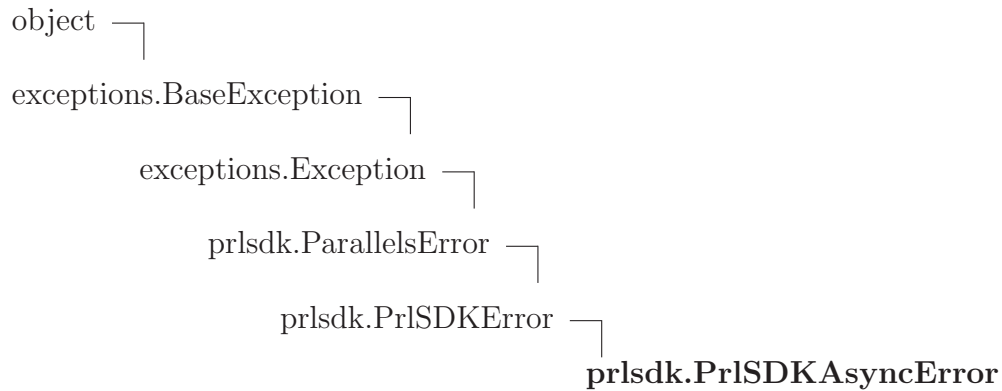
##### *Inherited from object*

`__hash__()`, `__reduce_ex__()`

#### 1.13.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	
<code>args</code> , <code>message</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

## 1.14 Class *PrlSDKAsyncError*



*SDKAsyncError* exception type

### 1.14.1 Methods

**`__init__`**(...)

`x.__init__`(...) initializes `x`; see `x.__class__.__doc__` for signature

Overrides: `object.__init__`

**`__new__`**(*T*, *S*, ...)

**Return Value**

a new object with type *S*, a subtype of *T*

Overrides: `object.__new__`

#### *Inherited from exceptions.BaseException*

`__delattr__`(), `__getattr__`(), `__getitem__`(), `__reduce__`(), `__repr__`(), `__setattr__`(),  
`__setstate__`(), `__str__`()

#### *Inherited from object*

`__hash__`(), `__reduce_ex__`()

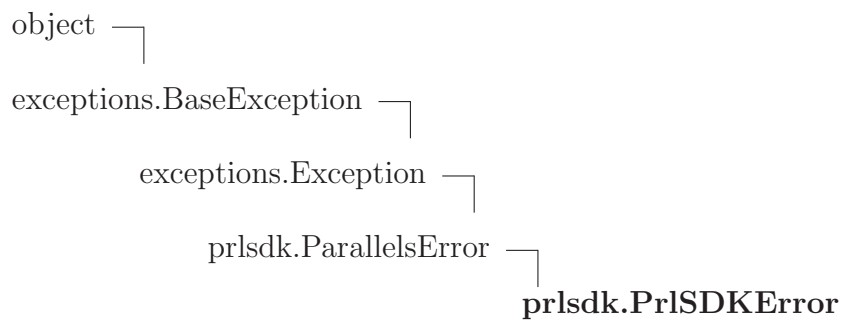
### 1.14.2 Properties

Name	Description
<code>details</code>	Returns error details
<code>error_params</code>	
<code>job_op_code</code>	

*continued on next page*

Name	Description
<i>Inherited from</i> <i>prlsdk.PrlSDKError</i> (Section 4.14)	<code>__call_descr__</code> , <code>error_code</code>
<i>Inherited from</i> <i>exceptions.BaseException</i>	<code>args</code> , <code>message</code>
<i>Inherited from</i> <i>object</i>	<code>__class__</code>

## 1.15 Class *PrlSDKError*



**Known Subclasses:** *prlsdk.PrlSDKAsyncError*

SDKError exception type

### 1.15.1 Methods

<b><code>__init__</code></b> (...)
<code>x.__init__(...)</code> initializes <code>x</code> ; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

<b><code>__new__</code></b> ( <i>T</i> , <i>S</i> , ...)
<b>Return Value</b> a new object with type <i>S</i> , a subtype of <i>T</i>
Overrides: <code>object.__new__</code>

*Inherited from* *exceptions.BaseException*

`__delattr__`(), `__getattr__`(), `__getitem__`(), `__reduce__`(), `__repr__`(), `__setattr__`(),  
`__setstate__`(), `__str__`()

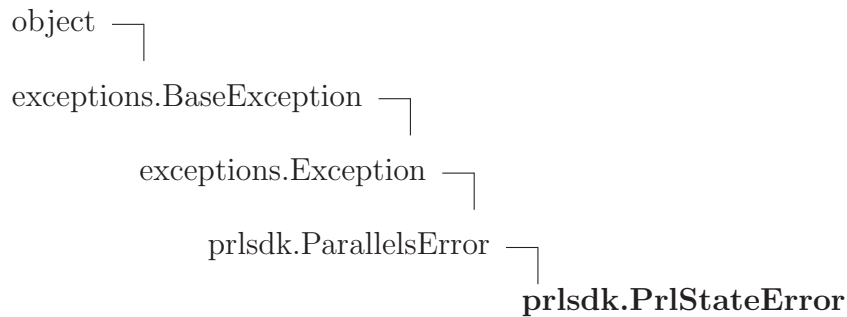
*Inherited from* *object*

`__hash__()`, `__reduce_ex__()`

### 1.15.2 Properties

Name	Description
<code>__call_descr__</code>	
<code>details</code>	Returns error details
<code>error_code</code>	
<i>Inherited from exceptions.BaseException</i>	
args, message	
<i>Inherited from object</i>	
<code>__class__</code>	

## 1.16 Class PrlStateError



### 1.16.1 Methods

*Inherited from exceptions.Exception*

`__init__()`, `__new__()`

*Inherited from exceptions.BaseException*

`__delattr__()`, `__getattr__()`, `__getitem__()`, `__reduce__()`, `__repr__()`, `__setattr__()`,  
`__setstate__()`, `__str__()`

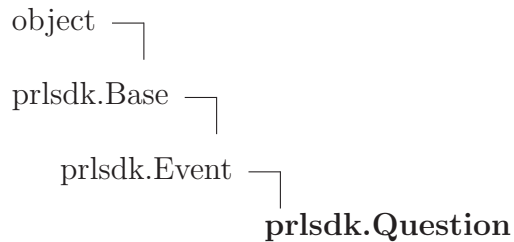
*Inherited from object*

`__hash__()`, `__reduce_ex__()`

### 1.16.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	
args, message	
<i>Inherited from object</i>	
__class__	

## 1.17 Class Question



Question object class

### 1.17.1 Methods

<b>__init__</b> (...)
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

<b>__new__</b> (T, S, ...)
<b>Return Value</b> a new object with type S, a subtype of T
Overrides: object.__new__

<b>send_answer</b> (Question, answer_code)
Send answer.
<b>Return Value</b> Job

*Inherited from prlsdk.Event(Section 4.7)*

\_\_getitem\_\_(), \_\_iter\_\_(), \_\_len\_\_(), get\_param(), get\_params\_count()

*Inherited from prlsdk.Base(Section 4.3)*

`__cmp__()`, `__repr__()`

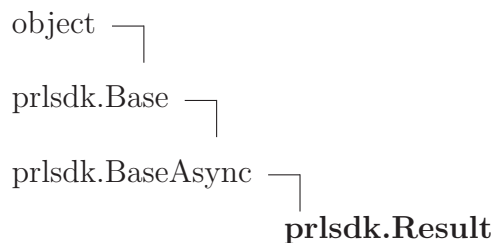
### *Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`,  
`__str__()`

#### 1.17.2 Properties

Name	Description
answers	Returns answer variants tuple ((description, value, name), ).
<i>Inherited from prlsdk.Event (Section 4.7)</i>	
event_code, event_str_code, event_type, explicit_text, params_count, sender, text	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__, handle_type	
<i>Inherited from object</i>	
__class__	

## 1.18 Class Result



Result object class

#### 1.18.1 Methods

<code>__getitem__(x, y)</code>
<code>x[y]</code>
<code>__init__(...)</code>
<code>x.__init__(...)</code> initializes x; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

<code>--iter--(<i>x</i>)</code>
<code>iter(<i>x</i>)</code>

<code>--len--(<i>x</i>)</code>
<code>len(<i>x</i>)</code>

<code>--new--(<i>T</i>, <i>S</i>, ...)</code>
<b>Return Value</b> a new object with type <i>S</i> , a subtype of <i>T</i>
Overrides: <code>object.--new--</code>

<code>get_param(<i>Result</i>, <i>index</i>)</code>
Returns depending on parameter type some SDK object or string with passed <i>index</i> .
<b>Return Value</b> some sdk object/string

<code>get_param_as_string(<i>Result</i>, <i>index</i>)</code>
Returns as string representation of parameter with passed <i>index</i> .
<b>Return Value</b> string

<code>get_params_count(<i>Result</i>, <i>index</i>)</code>
Returns parameters count.
<b>Return Value</b> int

***Inherited from prlsdk.BaseAsync(Section 4.4)***

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

***Inherited from object***

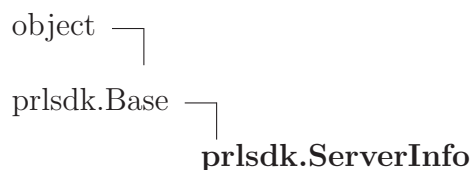
`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

**1.18.2 Properties**

*continued on next page*

Name	Description
<b>Name</b>	<b>Description</b>
params_count	Returns count of resulted params
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

## 1.19 Class *ServerInfo*



*ServerInfo* object class

### 1.19.1 Methods

<b>__init__</b> (...)
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

<b>__new__</b> ( <i>T</i> , <i>S</i> , ...)
<b>Return Value</b>
a new object with type <i>S</i> , a subtype of <i>T</i>
Overrides: object.__new__

<b>__str__</b> ( <i>x</i> )
str( <i>x</i> )
Overrides: object.__str__

*Inherited from prlsdk.Base (Section 4.3)*

\_\_cmp\_\_(), \_\_repr\_\_()

***Inherited from object***

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`

**1.19.2 Properties**

Name	Description
host	host name
os_version	os version
port	port
uuid	server uuid
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code> , <code>handle_type</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

## 2 Module prlsdk.consts

### 2.1 Variables

Name	Description
MB_LEFT_BUTTON	<b>Value:</b> 1342177285
MB_MIDDLE_BUTTON	<b>Value:</b> 1342177287
MB_NOBUTTON	<b>Value:</b> 0
MB_RIGHT_BUTTON	<b>Value:</b> 1342177286
ME_PRESS	<b>Value:</b> 1342177283
ME_RELEASE	<b>Value:</b> 1342177284
PAO_VM_NOT_SHARED	<b>Value:</b> 0
PAO_VM_SHARED_ON_- FULL_ACCESS	<b>Value:</b> 3
PAO_VM_SHARED_ON_- VIEW	<b>Value:</b> 1
PAO_VM_SHARED_ON_- VIEW_AND_RUN	<b>Value:</b> 2
PAO_VM_START_MANU- AL	<b>Value:</b> 0
PAO_VM_START_ON_L- OAD	<b>Value:</b> 1
PAO_VM_START_ON_R- ELOAD	<b>Value:</b> 2
PAO_VM_STOP	<b>Value:</b> 0
PAO_VM_SUSPEND	<b>Value:</b> 1
PBO_CD_HDD_FLOPPY	<b>Value:</b> 2
PBO_FLOPPY_HDD_CD	<b>Value:</b> 1
PBO_HDD_CD_FLOPPY	<b>Value:</b> 0
PCM_CPU_AMD_V	<b>Value:</b> 2
PCM_CPU_INTEL_VT_X	<b>Value:</b> 1
PCM_CPU_MODE_32	<b>Value:</b> 0
PCM_CPU_MODE_64	<b>Value:</b> 1
PCM_CPU_NONE_HV	<b>Value:</b> 0
PDA_DEVICE_IDLE	<b>Value:</b> 0
PDA_DEVICE_READ	<b>Value:</b> 1
PDA_DEVICE_WRITE	<b>Value:</b> 2
PDE_CLUSTERED_DEVI- CE	<b>Value:</b> 1
PDE_FLOPPY_DISK	<b>Value:</b> 3
PDE_GENERIC_DEVICE	<b>Value:</b> 0
PDE_GENERIC_NETWO- RK_ADAPTER	<b>Value:</b> 8

*continued on next page*

Name	Description
PDE_GENERIC_PCI_DEVICE	<b>Value:</b> 17
PDE_GENERIC_PORT	<b>Value:</b> 9
PDE_GENERIC_SCSI_DEVICE	<b>Value:</b> 18
PDE_HARD_DISK	<b>Value:</b> 6
PDE_MASSSTORAGE_DEVICE	<b>Value:</b> 4
PDE_MAX	<b>Value:</b> 19
PDE_MIXER_DEVICE	<b>Value:</b> 13
PDE_OPTICAL_DISK	<b>Value:</b> 5
PDE_PARALLEL_PORT	<b>Value:</b> 11
PDE_PRINTER	<b>Value:</b> 16
PDE_SERIAL_PORT	<b>Value:</b> 10
PDE_SOUND_DEVICE	<b>Value:</b> 12
PDE_STORAGE_DEVICE	<b>Value:</b> 2
PDE_USB_DEVICE	<b>Value:</b> 15
PDT_ANY_TYPE	<b>Value:</b> 65535
PDT_USE_IMAGE_FILE	<b>Value:</b> 1
PDT_USE_OTHER	<b>Value:</b> 3
PDT_USE_OUTPUT_FILE	<b>Value:</b> 2
PDT_USE_REAL_DEVICE	<b>Value:</b> 0
PFS_UNIX_LIKE_FS	<b>Value:</b> 1
PFS_WINDOWS_LIKE_FS	<b>Value:</b> 0
PHD_EXPANDING_HARD_DISK	<b>Value:</b> 1
PHD_PLAIN_HARD_DISK	<b>Value:</b> 0
PHI_REAL_NET_ADAPTER	<b>Value:</b> 0
PHI_VIRTUAL_NET_ADAPTER	<b>Value:</b> 1
PHO_LIN	<b>Value:</b> 1
PHO_MAC	<b>Value:</b> 0
PHO_WIN	<b>Value:</b> 2
PHT_ACCESS_RIGHTS	<b>Value:</b> 268435504
PHT_BOOT_DEVICE	<b>Value:</b> 268435497
PHT_DISP_CONFIG	<b>Value:</b> 268435478

*continued on next page*

Name	Description
PHT_DISP_NET_ADAPTE- R	<b>Value:</b> 268435481
PHT_EVENT	<b>Value:</b> 268435474
PHT_EVENT_PARAME- TER	<b>Value:</b> 268435486
PHT_FOUND_VM_INFO	<b>Value:</b> 268435506
PHT_HANDLES_LIST	<b>Value:</b> 268435502
PHT_HW_GENERIC_DE- VICE	<b>Value:</b> 268435482
PHT_HW_HARD_DISK	<b>Value:</b> 268435483
PHT_HW_HARD_DISK_P- ARTITION	<b>Value:</b> 268435484
PHT_HW_NET_ADAPTE- R	<b>Value:</b> 268435485
PHT_JOB	<b>Value:</b> 268435473
PHT_LICENSE	<b>Value:</b> 268435495
PHT_LOGIN_RESPONSE	<b>Value:</b> 268435499
PHT_NET_SERVICE_ST- ATUS	<b>Value:</b> 268435501
PHT_REMOTEDDEV_CM- D	<b>Value:</b> 268435487
PHT_REMOTE_FILESYS- TEM_ENTRY	<b>Value:</b> 268435460
PHT_REMOTE_FILESYS- TEM_INFO	<b>Value:</b> 268435459
PHT_RESULT	<b>Value:</b> 268435475
PHT_RUNNING_TASK	<b>Value:</b> 268435500
PHT_SCREEN_RESOLU- TION	<b>Value:</b> 268435480
PHT_SERVER	<b>Value:</b> 268435457
PHT_SERVER_CONFIG	<b>Value:</b> 268435458
PHT_SERVER_INFO	<b>Value:</b> 268435496
PHT_SHARE	<b>Value:</b> 268435479
PHT_STRINGS_LIST	<b>Value:</b> 268435498
PHT_SYSTEM_STATIST- ICS	<b>Value:</b> 268435488
PHT_SYSTEM_STATIST- ICS_CPU	<b>Value:</b> 268435489
PHT_SYSTEM_STATIST- ICS_DISK	<b>Value:</b> 268435492
PHT_SYSTEM_STATIST- ICS_DISK_PARTITION	<b>Value:</b> 268435493

*continued on next page*

Name	Description
PHT_SYSTEM_STATISTICS_IFACE	<b>Value:</b> 268435490
PHT_SYSTEM_STATISTICS_PROCESS	<b>Value:</b> 268435494
PHT_SYSTEM_STATISTICS_USER_SESSION	<b>Value:</b> 268435491
PHT_USER_INFO	<b>Value:</b> 268435508
PHT_USER_PROFILE	<b>Value:</b> 268435477
PHT_VIRTUAL_DEV_DISPLAY	<b>Value:</b> 268435462
PHT_VIRTUAL_DEV_FLOPPY	<b>Value:</b> 268435465
PHT_VIRTUAL_DEV_HARDDISK	<b>Value:</b> 268435466
PHT_VIRTUAL_DEV_KEYBOARD	<b>Value:</b> 268435463
PHT_VIRTUAL_DEV_MOUSE	<b>Value:</b> 268435464
PHT_VIRTUAL_DEV_NETWORK_ADAPTER	<b>Value:</b> 268435467
PHT_VIRTUAL_DEV_OPTICAL_DISK	<b>Value:</b> 268435470
PHT_VIRTUAL_DEV_PARALLEL_PORT	<b>Value:</b> 268435468
PHT_VIRTUAL_DEV_SERIAL_PORT	<b>Value:</b> 268435469
PHT_VIRTUAL_DEV_SOUND	<b>Value:</b> 268435472
PHT_VIRTUAL_DEV_USB_DEVICE	<b>Value:</b> 268435471
PHT_VIRTUAL_DISK	<b>Value:</b> 268435503
PHT_VIRTUAL_MACHINE	<b>Value:</b> 268435461
PHT_VM_INFO	<b>Value:</b> 268435476
PHT_VM_TOOLS_INFO	<b>Value:</b> 268435505
PIF_BMP	<b>Value:</b> 1342177289
PIF_JPG	<b>Value:</b> 1342177290
PIF_PNG	<b>Value:</b> 1342177291
PIF_RAW	<b>Value:</b> 1342177288
PJOC_JOB_CANCEL	<b>Value:</b> 1
PJOC_SRV_ADD_NETWORK_ADAPTER	<b>Value:</b> 35

*continued on next page*

Name	Description
PJOC_SRV_ATTACH_TO_LOST_TASK	<b>Value:</b> 39
PJOC_SRV_COMMON_PREFS_BEGIN_EDIT	<b>Value:</b> 8
PJOC_SRV_COMMON_PREFS_COMMIT	<b>Value:</b> 9
PJOC_SRV_DELETE_NETWORK_ADAPTER	<b>Value:</b> 36
PJOC_SRV_FS_CAN_CREATE_FILE	<b>Value:</b> 23
PJOC_SRV_FS_CREATE_DIR	<b>Value:</b> 21
PJOC_SRV_FS_GENERATE_ENTRY_NAME	<b>Value:</b> 25
PJOC_SRV_FS_GET_DIR_ENTRIES	<b>Value:</b> 20
PJOC_SRV_FS_GET_DISK_LIST	<b>Value:</b> 19
PJOC_SRV_FS_REMOVE_ENTRY	<b>Value:</b> 22
PJOC_SRV_FS_RENAME_ENTRY	<b>Value:</b> 24
PJOC_SRV_GET_COMMON_PREFS	<b>Value:</b> 7
PJOC_SRV_GET_LICENSE_INFO	<b>Value:</b> 27
PJOC_SRV_GET_NETWORK_SERVICE_STATUS	<b>Value:</b> 34
PJOC_SRV_GET_PROBLEM_REPORT	<b>Value:</b> 38
PJOC_SRV_GET_SRV_CONFIG	<b>Value:</b> 6
PJOC_SRV_GET_STATISTICS	<b>Value:</b> 11
PJOC_SRV_GET_USER_INFO	<b>Value:</b> 41
PJOC_SRV_GET_USER_INFO_LIST	<b>Value:</b> 40
PJOC_SRV_GET_USER_PROFILE	<b>Value:</b> 10
PJOC_SRV_GET_VM_LIST	<b>Value:</b> 15

*continued on next page*

Name	Description
PJOC_SRV_LOGIN	Value: 3
PJOC_SRV_LOGIN_LOCAL	Value: 4
PJOC_SRV_LOGOFF	Value: 5
PJOC_SRV_LOOKUP_PARALLELSERVERS	Value: 2
PJOC_SRV_NET_SERVICE_RESTART	Value: 32
PJOC_SRV_NET_SERVICE_RESTORE_DEFAULTS	Value: 33
PJOC_SRV_NET_SERVICE_START	Value: 30
PJOC_SRV_NET_SERVICE_STOP	Value: 31
PJOC_SRV_REGISTER_VM	Value: 14
PJOC_SRV_SEND_ANSWER	Value: 28
PJOC_SRV_SHUTDOWN	Value: 18
PJOC_SRV_START_SEARCH_VMS	Value: 29
PJOC_SRV_SUBSCRIBE_TO_HOST_STATISTICS	Value: 16
PJOC_SRV_UNSUBSCRIBE_FROM_HOST_STATISTICS	Value: 17
PJOC_SRV_UPDATE_LICENSE	Value: 26
PJOC_SRV_UPDATE_NETWORK_ADAPTER	Value: 37
PJOC_SRV_USER_PROFILE_BEGIN_EDIT	Value: 12
PJOC_SRV_USER_PROFILE_COMMIT	Value: 13
PJOC_UNKNOWN	Value: 0
PJOC_VM_BEGIN_EDIT	Value: 62
PJOC_VM_CLONE	Value: 50
PJOC_VM_COMMIT	Value: 63
PJOC_VM_CREATE_UNATTENDED_FLOPPY	Value: 64
PJOC_VM_DELETE	Value: 51

*continued on next page*

Name	Description
PJOC_VM_DEV_CONNECT	<b>Value:</b> 68
PJOC_VM_DEV_CREATE_IMAGE	<b>Value:</b> 70
PJOC_VM_DEV_DISCONNECT	<b>Value:</b> 69
PJOC_VM_DROP_SUSPENDED_STATE	<b>Value:</b> 49
PJOC_VM_GENERATE_VM_DEV_FILENAME	<b>Value:</b> 54
PJOC_VM_GET_PROBLEM_REPORT	<b>Value:</b> 52
PJOC_VM_GET_STATE	<b>Value:</b> 53
PJOC_VM_GET_STATISTICS	<b>Value:</b> 57
PJOC_VM_GET_TOOLS_STATE	<b>Value:</b> 55
PJOC_VM_INITIATE_DEV_STATE_NOTIFICATIONS	<b>Value:</b> 65
PJOC_VM_INSTALL_TOOLS	<b>Value:</b> 71
PJOC_VM_PAUSE	<b>Value:</b> 44
PJOC_VM_REFRESH_CONFIG	<b>Value:</b> 56
PJOC_VM_REG	<b>Value:</b> 60
PJOC_VM_RESET	<b>Value:</b> 45
PJOC_VM_RESUME	<b>Value:</b> 42
PJOC_VM_START	<b>Value:</b> 42
PJOC_VM_STOP	<b>Value:</b> 43
PJOC_VM_SUBSCRIBE_TO_GUEST_STATISTICS	<b>Value:</b> 58
PJOC_VM_SUSPEND	<b>Value:</b> 46
PJOC_VM_UNREG	<b>Value:</b> 61
PJOC_VM_UNSUBSCRIBE_FROM_GUEST_STATISTICS	<b>Value:</b> 59
PJOC_VM_UPDATE_SECURITY	<b>Value:</b> 66
PJOC_VM_VALIDATE_CONFIG	<b>Value:</b> 67

*continued on next page*

Name	Description
PKE_CLICK	Value: 255
PKE_PRESS	Value: 0
PKE_RELEASE	Value: 128
PLM_ROOT_ACOUNT	Value: 1
PLM_START_ACOUNT	Value: 0
PLM_USER_ACOUNT	Value: 2
PMS_IDE_DEVICE	Value: 0
PMS_SCSI_DEVICE	Value: 1
PMT_ANSWER	Value: 4
PMT_CRITICAL	Value: 2
PMT_INFORMATION	Value: 1
PMT_QUESTION	Value: 3
PMT_WARNING	Value: 0
PNA_BRIDGED_ETHER- NET	Value: 2
PNA_HOST_ONLY	Value: 0
PNA_SHARED	Value: 1
PRD_AUTO	Value: 1
PRD_DISABLED	Value: 0
PRD_MANUAL	Value: 2
PRL_ERR_FLOPPY_IMA- GE_ALREADY_EXIST	Value: -2147483050
PRL_ERR_HDD_IMAGE_- IS_ALREADY_EXIST	Value: -2147483000
PSE_DIRECTORY	Value: 1
PSE_DRIVE	Value: 0
PSE_FILE	Value: 2
PSK_VM_REBOOT	Value: 1
PSK_VM_SHUTDOWN	Value: 0
PSL_HIGH_SECURITY	Value: 2
PSL_LOW_SECURITY	Value: 0
PSL_NORMAL_SECURITY	Value: 1
PSP_SERIAL_SOCKET_- CLIENT	Value: 1
PSP_SERIAL_SOCKET_S- ERVER	Value: 0
PSS_NOT_INSTALLED	Value: 2
PSS_STARTED	Value: 0
PSS_STOPPED	Value: 1
PSS_UNKNOWN	Value: 3
PTIS_RECORD_ACTIVE	Value: 1

*continued on next page*

Name	Description
PTIS_RECORD_CANCELED	Value: 2
PTIS_RECORD_EMPTY	Value: 0
PTS_INSTALLED	Value: 2
PTS_NOT_INSTALLED	Value: 1
PTS_OUTDATED	Value: 3
PTS_UNKNOWN	Value: 0
PTU_CMD_INVALID	Value: 0
PTU_CMD_PTU_ALIVE	Value: 1
PVA_ACCELERATION_DISABLED	Value: 0
PVA_ACCELERATION_HIGH	Value: 2
PVA_ACCELERATION_NORMAL	Value: 1
PVC_ALL	Value: 1
PVC_BOOT_OPTION	Value: 3
PVC_CD_DVD_ROM	Value: 10
PVC_CPU	Value: 6
PVC_FLOPPY_DISK	Value: 9
PVC_GENERAL_PARAMETERS	Value: 2
PVC_HARD_DISK	Value: 11
PVC_IDE_DEVICES	Value: 16
PVC_MAIN_MEMORY	Value: 7
PVC_NETWORK_ADAPTER	Value: 12
PVC_PARALLEL_PORT	Value: 15
PVC_REMOTE_DISPLAY	Value: 4
PVC_SCSI_DEVICES	Value: 17
PVC_SERIAL_PORT	Value: 14
PVC_SHARED_FOLDERS	Value: 5
PVC_SOUND	Value: 13
PVC_VIDEO_MEMORY	Value: 8
PVR_PRIORITY_HIGH	Value: 2
PVR_PRIORITY_LOW	Value: 0
PVR_PRIORITY_NORMAL	Value: 1
PVS_GUEST_TYPE_FREEBSD	Value: 10

*continued on next page*

Name	Description
PVS_GUEST_TYPE_LINUX	Value: 9
PVS_GUEST_TYPE_MACOS	Value: 7
PVS_GUEST_TYPE_MS-DOS	Value: 12
PVS_GUEST_TYPE_NETWARE	Value: 13
PVS_GUEST_TYPE_OS2	Value: 11
PVS_GUEST_TYPE_OTHER	Value: 255
PVS_GUEST_TYPE_SOLARIS	Value: 14
PVS_GUEST_TYPE_WINDOWS	Value: 8
PVS_GUEST_VER_BSD_4X	Value: 2561
PVS_GUEST_VER_BSD_5X	Value: 2562
PVS_GUEST_VER_BSD_6X	Value: 2563
PVS_GUEST_VER_BSD_7X	Value: 2564
PVS_GUEST_VER_BSD_OTHER	Value: 2815
PVS_GUEST_VER_DOS_MS622	Value: 3073
PVS_GUEST_VER_DOS_OTHER	Value: 3327
PVS_GUEST_VER_LIN_DEBIAN	Value: 2310
PVS_GUEST_VER_LIN_FEDORA	Value: 2311
PVS_GUEST_VER_LIN_FEDORA_5	Value: 2312
PVS_GUEST_VER_LIN_KRNL_24	Value: 2308
PVS_GUEST_VER_LIN_KRNL_26	Value: 2309
PVS_GUEST_VER_LIN_MANDRAKE	Value: 2307
PVS_GUEST_VER_LIN_OTHER	Value: 2559

*continued on next page*

Name	Description
PVS_GUEST_VER_LIN_R-EDHAT	<b>Value:</b> 2305
PVS_GUEST_VER_LIN_S-USE	<b>Value:</b> 2306
PVS_GUEST_VER_LIN_U-BUNTU	<b>Value:</b> 2314
PVS_GUEST_VER_LIN_X-ANDROS	<b>Value:</b> 2313
PVS_GUEST_VER_MAC-OS_LEOPARD	<b>Value:</b> 1794
PVS_GUEST_VER_MAC-OS_TIGER	<b>Value:</b> 1793
PVS_GUEST_VER_NET_-4X	<b>Value:</b> 3329
PVS_GUEST_VER_NET_-5X	<b>Value:</b> 3330
PVS_GUEST_VER_NET_-6X	<b>Value:</b> 3331
PVS_GUEST_VER_NET_-OTHER	<b>Value:</b> 3583
PVS_GUEST_VER_OS2_E-CS11	<b>Value:</b> 2820
PVS_GUEST_VER_OS2_E-CS12	<b>Value:</b> 2821
PVS_GUEST_VER_OS2_-OTHER	<b>Value:</b> 3071
PVS_GUEST_VER_OS2_-WARP3	<b>Value:</b> 2817
PVS_GUEST_VER_OS2_-WARP4	<b>Value:</b> 2818
PVS_GUEST_VER_OS2_-WARP45	<b>Value:</b> 2819
PVS_GUEST_VER_OTH_-OPENSTEP	<b>Value:</b> 65282
PVS_GUEST_VER_OTH_-OTHER	<b>Value:</b> 65535
PVS_GUEST_VER_OTH_-QNX	<b>Value:</b> 65281
PVS_GUEST_VER_SOL_1-0	<b>Value:</b> 3586
PVS_GUEST_VER_SOL_9	<b>Value:</b> 3585

*continued on next page*

Name	Description
PVS_GUEST_VER_SOL_- OTHER	<b>Value:</b> 3839
PVS_GUEST_VER_WIN_- 2003	<b>Value:</b> 2056
PVS_GUEST_VER_WIN_- 2008	<b>Value:</b> 2058
PVS_GUEST_VER_WIN_- 2K	<b>Value:</b> 2054
PVS_GUEST_VER_WIN_- 311	<b>Value:</b> 2049
PVS_GUEST_VER_WIN_- 95	<b>Value:</b> 2050
PVS_GUEST_VER_WIN_- 98	<b>Value:</b> 2051
PVS_GUEST_VER_WIN_- ME	<b>Value:</b> 2052
PVS_GUEST_VER_WIN_- NT	<b>Value:</b> 2053
PVS_GUEST_VER_WIN_- OTHER	<b>Value:</b> 2303
PVS_GUEST_VER_WIN_- VISTA	<b>Value:</b> 2057
PVS_GUEST_VER_WIN_- XP	<b>Value:</b> 2055
PWM_COHERENCE_WI- NDOW_MODE	<b>Value:</b> 3
PWM_DEFAULT_WIND- OW_MODE	<b>Value:</b> 0
PWM_FULL_SCREEN_W- INDOW_MODE	<b>Value:</b> 2
PWM_WINDOWED_WIN- DOW_MODE	<b>Value:</b> 1
RTT_RUNNING_TASK_- CLONE_VM	<b>Value:</b> 3
RTT_RUNNING_TASK_- CREATE_IMAGE	<b>Value:</b> 6
RTT_RUNNING_TASK_- CREATE_VM	<b>Value:</b> 1
RTT_RUNNING_TASK_- DELETE_VM	<b>Value:</b> 4
RTT_RUNNING_TASK_- REGISTER_VM	<b>Value:</b> 2

*continued on next page*

Name	Description
RTT_RUNNING_TASK_-UNKNOWN	<b>Value:</b> 0
RTT_RUNNING_TASK_-UNREG_VM	<b>Value:</b> 5
ScanCodesList	<b>Value:</b> {'0': [11], '1': [2], '2': [3], '3': [4], '4': [5], '5': ...}
VMS_COMPACTING	<b>Value:</b> 805306376
VMS_PAUSED	<b>Value:</b> 805306373
VMS_RESTOREING	<b>Value:</b> 805306371
VMS_RUNNING	<b>Value:</b> 805306372
VMS_SNAPSHOTING	<b>Value:</b> 805306378
VMS_STARTING	<b>Value:</b> 805306370
VMS_STOPPED	<b>Value:</b> 805306369
VMS_STOPPING	<b>Value:</b> 805306375
VMS_SUSPENDED	<b>Value:</b> 805306377
VMS_SUSPENDING	<b>Value:</b> 805306374
VMS_UNKNOWN	<b>Value:</b> 0

### 3 Module prlsdk.errors

#### 3.1 Variables

Name	Description
GUI_ERR_CANT_SAVE_CONFIG_AS_VM_WAS_DELETED	<b>Value:</b> 2147569730
GUI_ERR_WRONG_NUMBER_OF_CPU	<b>Value:</b> 2147569696
PRL_ERR_ACCESS_DENIED	<b>Value:</b> 2147483653
PRL_ERR_ACCESS_DENIED_TO_CHANGE_PERMISSIONS	<b>Value:</b> 2147484545
PRL_ERR_ACCESS_DENIED_TO_DISK_IMAGE	<b>Value:</b> 2147483768
PRL_ERR_ACCESS_DENIED_TO_RUN_WIZARD	<b>Value:</b> 2147484512
PRL_ERR_ACCESS_TOKEN_INVALID	<b>Value:</b> 2147483744
PRL_ERR_ACCESS_TO_VM_DENIED	<b>Value:</b> 2147484217
PRL_ERR_ACCESS_DENIED_FILE_TO_PARENT_PARENT_DIR	<b>Value:</b> 2147483904
PRL_ERR_ADD_HW_FLOPPY_IMAGE_NOT_SPECIFIED	<b>Value:</b> 2147484439
PRL_ERR_ADD_VM_OPERATION_WAS_CANCELLED	<b>Value:</b> 2147484441
PRL_ERR_ALIGN_ON_4	<b>Value:</b> 2147484163
PRL_ERR_ALREADY_CONNECTED_TO_DISPATCHER	<b>Value:</b> 2147484242
PRL_ERR_ANOTHER_USER_SESSIONS_PRESENT	<b>Value:</b> 2147484513
PRL_ERR_API_WASNT_INITIALIZED	<b>Value:</b> 2147484295
PRL_ERR_ATTACH_TO_TASK_BAD_SESSION	<b>Value:</b> 2147484489

*continued on next page*

Name	Description
PRL_ERR_AUTHENTIC- ATION_FAILED	<b>Value:</b> 2147484464
PRL_ERR_BAD_DISP_C- ONFIG_FILE_SPECIFIE- D	<b>Value:</b> 2147483680
PRL_ERR_BAD_PARAM- ETERS	<b>Value:</b> 2147483779
PRL_ERR_BAD_VM_CO- NFIG_FILE_SPECIFIED	<b>Value:</b> 2147483698
PRL_ERR_BAD_VM_DIR- _CONFIG_FILE_SPECIFI- ED	<b>Value:</b> 2147483687
PRL_ERR_BUFFER_OVE- RRUN	<b>Value:</b> 2147483654
PRL_ERR_CANTS_CRE- ATE_DISK_IMAGE_ON_- FAT	<b>Value:</b> 2147484550
PRL_ERR_CANTS_CRE- ATE_DISK_IMAGE_ON_- FAT32	<b>Value:</b> 2147484551
PRL_ERR_CANT_ALOC- A_MEM_FILE_COPY	<b>Value:</b> 2147483913
PRL_ERR_CANT_CHAN- GE_FILE_PERMISSIONS	<b>Value:</b> 2147484473
PRL_ERR_CANT_CHAN- GE_OWNER_OF_DISK_I- MAGE_FILE	<b>Value:</b> 2147484564
PRL_ERR_CANT_CHAN- GE_OWNER_OF_FILE	<b>Value:</b> 2147484289
PRL_ERR_CANT_CHAN- GE_OWNER_OF_VM_FIL- E	<b>Value:</b> 2147484563
PRL_ERR_CANT_CONFI- GURE_PHYSICAL_HDD	<b>Value:</b> 2147484472
PRL_ERR_CANT_CONN- ECT_TO_DISPATCHER	<b>Value:</b> 2147484233
PRL_ERR_CANT_CONV- ERT_HDD_IMAGE_FILE	<b>Value:</b> 2147484432
PRL_ERR_CANT_CONV- ERT_OLD_VM_WITH_IN- STALLED_TOOLS	<b>Value:</b> 2147484549

*continued on next page*

Name	Description
PRL_ERR_CANT_CONVERT_VM_CONFIG_DUE_UNDO_DISKS_PRESENT	<b>Value:</b> 2147484424
PRL_ERR_CANT_CONVERT_VM_CONFIG_DUE_UNDO_SNAPSHOTS_PRESENT	<b>Value:</b> 2147484425
PRL_ERR_CANT_CREATE_FLOPPY_IMAGE	<b>Value:</b> 2147483801
PRL_ERR_CANT_CREATE_HDD_IMAGE	<b>Value:</b> 2147483800
PRL_ERR_CANT_CREATE_HDD_IMAGE_NO_SPACE	<b>Value:</b> 2147484553
PRL_ERR_CANT_CREATE_PARALLEL_PORT_IMAGE	<b>Value:</b> 2147483905
PRL_ERR_CANT_CREATE_SERIAL_PORT_IMAGE	<b>Value:</b> 2147484672
PRL_ERR_CANT_DELETE_FILE	<b>Value:</b> 2147484290
PRL_ERR_CANT_EDIT_SUSPENDED_VM	<b>Value:</b> 2147483908
PRL_ERR_CANT_GET_FILE_PERMISSIONS	<b>Value:</b> 2147484546
PRL_ERR_CANT_INIT_REAL_CPUS_INFO	<b>Value:</b> 2147484274
PRL_ERR_CANT_PARSE_DISP_EVENT	<b>Value:</b> 2147484228
PRL_ERR_CANT_PARSE_VM_CONFIG	<b>Value:</b> 2147484201
PRL_ERR_CANT_RECREATE_HDD	<b>Value:</b> 2147484457
PRL_ERR_CANT_REMOVE_DIR_ENTRY	<b>Value:</b> 2147484680
PRL_ERR_CANT_REMOVE_ENTRY	<b>Value:</b> 2147483778
PRL_ERR_CANT_REMOVE_INVALID_VM_AS_NON_ADMIN	<b>Value:</b> 2147484547
PRL_ERR_CANT_RENAME_DIR_ENTRY	<b>Value:</b> 2147484681

*continued on next page*

Name	Description
PRL_ERR_CANT_RENAME_ENTRY	<b>Value:</b> 2147483777
PRL_ERR_CANT_REPLACE_FLOPPY_IMAGE	<b>Value:</b> 2147484467
PRL_ERR_CANT_TO_CHANGE_PERMISSIONS_ON_REMOTE_LOCATION	<b>Value:</b> 2147484695
PRL_ERR_CANT_TO_START_VM_TEMPLATE	<b>Value:</b> 2147484520
PRL_ERR_CANT_UPDATE_DEVICE_INFO	<b>Value:</b> 2147484466
PRL_ERR_CAN_NOT_GET_DISK_FREE_SPACE	<b>Value:</b> 2147484422
PRL_ERR_CDD_IMAGE_NOT_SPECIFIED	<b>Value:</b> 2147484168
PRL_ERR_CLONE_OPERATION_CANCELED	<b>Value:</b> 2147483911
PRL_ERR_COMMON_SERVER_PREFS_BLOCKED_TO_CHANGE	<b>Value:</b> 2147484273
PRL_ERR_COMMON_SERVER_PREFS_WERE_CHANGED	<b>Value:</b> 2147483938
PRL_ERR_COM_DEVICE_ALREADY_EXIST	<b>Value:</b> 2147484244
PRL_ERR_COM_OUTPUT_FILE_IS_NOT_SPECIFIED	<b>Value:</b> 2147484181
PRL_ERR_COM_OUTPUT_FILE_NOT_EXIST	<b>Value:</b> 2147484183
PRL_ERR_CONCURRENT_COMMAND_EXECUTING	<b>Value:</b> 2147484226
PRL_ERR_CONFIG_BEGIN_EDIT_NOT_FOUND_OBJECT_UUID	<b>Value:</b> 2147484486
PRL_ERR_CONFIG_BEGIN_EDIT_NOT_FOUND_USER_UUID	<b>Value:</b> 2147484487
PRL_ERR_CONFIG_EDIT_COLLISION	<b>Value:</b> 2147484451

*continued on next page*

Name	Description
PRL_ERR_CONFIG_FILE_NOT_SET	<b>Value:</b> 2147483704
PRL_ERR_CORE_STATE_CHANGE_VM_CONFIG	<b>Value:</b> 2147614724
PRL_ERR_CORE_STATE_CORRUPT_MEM_FILE	<b>Value:</b> 2147614723
PRL_ERR_CORE_STATE_CORRUPT_SAV_FILE	<b>Value:</b> 2147614722
PRL_ERR_CORE_STATE_ERROR_COMMON	<b>Value:</b> 2147614720
PRL_ERR_CORE_STATE_INV_SAV_VERSION	<b>Value:</b> 2147614721
PRL_ERR_COULDNT_CREATE_AUTHORIZATION_FILE	<b>Value:</b> 2147484309
PRL_ERR_COULDNT_SET_PERMISSIONS_TO_AUTHORIZATION_FILE	<b>Value:</b> 2147484310
PRL_ERR_CPU_RESTART	<b>Value:</b> 2147484544
PRL_ERR_CPU_SHUTDOWN	<b>Value:</b> 2147484537
PRL_ERR_CREATE_PRIVILEGED_PROCESS	<b>Value:</b> 2147577858
PRL_ERR_DELETING_VM_FROM_CATALOGUE	<b>Value:</b> 2147484243
PRL_ERR_DEV_ALREADY_CONNECTED	<b>Value:</b> 2147520512
PRL_ERR_DEV_ALREADY_DISCONNECTED	<b>Value:</b> 2147520513
PRL_ERR_DEV_DISABLE_AFTER_LOST	<b>Value:</b> 2147484483
PRL_ERR_DEV_FLOPPY_CONNECT_FAILED	<b>Value:</b> 2147487744
PRL_ERR_DEV_MAX_CD_HDD_EXCEEDED	<b>Value:</b> 2147549185
PRL_ERR_DEV_MAX_NUMBER_EXCEEDED	<b>Value:</b> 2147549184
PRL_ERR_DEV_PARALLEL_PORT_FILE_CONNECT_FAILED	<b>Value:</b> 2147512322

*continued on next page*

Name	Description
PRL_ERR_DEV_PARALLEL_PORT_PHYSICAL_CONNECT_FAILED	<b>Value:</b> 2147512320
PRL_ERR_DEV_PARALLEL_PORT_PRINTER_CONNECT_FAILED	<b>Value:</b> 2147512321
PRL_ERR_DEV_PARALLEL_PORT_REMOTE_CONNECT_FAILED	<b>Value:</b> 2147512323
PRL_ERR_DEV_SERIAL_PORT_FILE_CONNECT_FAILED	<b>Value:</b> 2147508226
PRL_ERR_DEV_SERIAL_PORT_PHYSICAL_CONNECT_FAILED	<b>Value:</b> 2147508224
PRL_ERR_DEV_SERIAL_PORT_PIPE_CONNECT_FAILED	<b>Value:</b> 2147508225
PRL_ERR_DEV_SERIAL_PORT_REMOTE_CONNECT_FAILED	<b>Value:</b> 2147508227
PRL_ERR_DEV_USB_INSTALL_DRIVER_FAILED	<b>Value:</b> 2147516417
PRL_ERR_DEV_USB_OPEN_MANAGER_FAILED	<b>Value:</b> 2147516416
PRL_ERR_DIRECTORY_DOES_NOT_EXIST	<b>Value:</b> 2147483765
PRL_ERR_DISK_BLOCK_CREATED	<b>Value:</b> 2147618885
PRL_ERR_DISK_BLOCK_PARTIALLY_PROCESSED	<b>Value:</b> 2147618841
PRL_ERR_DISK_BLOCK_SEARCH_FAILED	<b>Value:</b> 2147622915
PRL_ERR_DISK_BLOCK_SKIPPED	<b>Value:</b> 2147618840
PRL_ERR_DISK_CANT_INITIALIZE_IMAGE	<b>Value:</b> 2147618889
PRL_ERR_DISK_COMPRESSED_FILE_EMPTY	<b>Value:</b> 2147622913
PRL_ERR_DISK_CREATE_IMAGE_ERROR	<b>Value:</b> 2147618838

*continued on next page*

Name	Description
PRL_ERR_DISK_DATA_NOT_FOUND	<b>Value:</b> 2147627008
PRL_ERR_DISK_DIR_CREATE_ERROR	<b>Value:</b> 2147618837
PRL_ERR_DISK_DISK_NOT_OPENED	<b>Value:</b> 2147618849
PRL_ERR_DISK_FAT32_SIZE_EXCEEDED	<b>Value:</b> 2147618851
PRL_ERR_DISK_FILE_CREATE_ERROR	<b>Value:</b> 2147618835
PRL_ERR_DISK_FILE_EXISTS	<b>Value:</b> 2147618834
PRL_ERR_DISK_FILE_OPEN_ERROR	<b>Value:</b> 2147618836
PRL_ERR_DISK_FSYNC_FAILED	<b>Value:</b> 2147618865
PRL_ERR_DISK_FULLFSYNC_FAILED	<b>Value:</b> 2147618866
PRL_ERR_DISK_GENERIC_ERROR	<b>Value:</b> 2147618816
PRL_ERR_DISK_GET_MOUNTPATH_FAILED	<b>Value:</b> 2147631104
PRL_ERR_DISK_GET_PARAMS_FAILED	<b>Value:</b> 2147631107
PRL_ERR_DISK_GROUP_INTERSECT	<b>Value:</b> 2147618886
PRL_ERR_DISK_IDENTIFY_FAILED	<b>Value:</b> 2147631108
PRL_ERR_DISK_IMAGES_CORRUPTED	<b>Value:</b> 2147618824
PRL_ERR_DISK_IMAGE_BUSY	<b>Value:</b> 2147618884
PRL_ERR_DISK_INCORRECTLY_CLOSED	<b>Value:</b> 2147618887
PRL_ERR_DISK_INSUFFICIENT_SPACE	<b>Value:</b> 2147618850
PRL_ERR_DISK_INTERNAL_CLASS_ERROR	<b>Value:</b> 2147618853
PRL_ERR_DISK_INVALID_BLOCKSIZE	<b>Value:</b> 2147618825
PRL_ERR_DISK_INVALID_FORMAT	<b>Value:</b> 2147618867

*continued on next page*

Name	Description
PRL_ERR_DISK_INVALID_PARAMETERS	<b>Value:</b> 2147618833
PRL_ERR_DISK_MEMORY_ERROR	<b>Value:</b> 2147618848
PRL_ERR_DISK_MOUNT_FAILED	<b>Value:</b> 2147631105
PRL_ERR_DISK_NOT_IMPLEMENTED	<b>Value:</b> 2147618882
PRL_ERR_DISK_NOT_PERMITTED	<b>Value:</b> 2147618852
PRL_ERR_DISK_NULL_PARTITION_SIZE	<b>Value:</b> 2147618897
PRL_ERR_DISK_OFFSETS_FIXED	<b>Value:</b> 2147618888
PRL_ERR_DISK_OPERATION_ABORTED	<b>Value:</b> 2147618870
PRL_ERR_DISK_OPERATION_IN_PROGRESS	<b>Value:</b> 2147618839
PRL_ERR_DISK_OPERATION_NOT_ALLOWED	<b>Value:</b> 2147618869
PRL_ERR_DISK_PARTITIONS_TABLE_CYCLE	<b>Value:</b> 2147618883
PRL_ERR_DISK_POINTERS_MIXED_UP	<b>Value:</b> 2147622916
PRL_ERR_DISK_READ_FAILED	<b>Value:</b> 2147618857
PRL_ERR_DISK_READ_OUT_DISK	<b>Value:</b> 2147618856
PRL_ERR_DISK_RENAME_ERROR	<b>Value:</b> 2147618868
PRL_ERR_DISK_RESERVED_WORD	<b>Value:</b> 2147627009
PRL_ERR_DISK_SET_CACHING_FAILED	<b>Value:</b> 2147618864
PRL_ERR_DISK_SHARED_BLOCK	<b>Value:</b> 2147622912
PRL_ERR_DISK_SHARING_VIOLATION	<b>Value:</b> 2147618871
PRL_ERR_DISK_SMALL_IMAGE_SIZE	<b>Value:</b> 2147618896
PRL_ERR_DISK_SNAPSHOTS_CORRUPTED	<b>Value:</b> 2147618822

*continued on next page*

Name	Description
PRL_ERR_DISK_STATE-S_ERROR	<b>Value:</b> 2147618832
PRL_ERR_DISK_STORAGE_CORRUPTED	<b>Value:</b> 2147618823
PRL_ERR_DISK_TRUNCATE_FAILED	<b>Value:</b> 2147622914
PRL_ERR_DISK_UNMOUNT_FAILED	<b>Value:</b> 2147631106
PRL_ERR_DISK_USER_INTERRUPTED	<b>Value:</b> 2147618872
PRL_ERR_DISK_WRITE_FAILED	<b>Value:</b> 2147618855
PRL_ERR_DISK_WRITE_OUT_DISK	<b>Value:</b> 2147618854
PRL_ERR_DISK_WRITE_REAL_FAILED	<b>Value:</b> 2147618881
PRL_ERR_DISK_XML_INVALID	<b>Value:</b> 2147618818
PRL_ERR_DISK_XML_INVALID_VERSION	<b>Value:</b> 2147618819
PRL_ERR_DISK_XML_LARGE_FILE	<b>Value:</b> 2147618821
PRL_ERR_DISK_XML_OPEN_FAILED	<b>Value:</b> 2147618817
PRL_ERR_DISK_XML_SAVE_ERROR	<b>Value:</b> 2147618820
PRL_ERR_DISK_XML_SAVE_REMOVE_ERROR	<b>Value:</b> 2147618873
PRL_ERR_DISK_XML_SAVE_RENAME_ERROR	<b>Value:</b> 2147618880
PRL_ERR_DISPATCHER_SERVICE_MODE	<b>Value:</b> 2147484516
PRL_ERR_DISP_CONFIG_ALREADY_EXISTS	<b>Value:</b> 2147483685
PRL_ERR_DISP_CONFIG_FILE_NOT_SET	<b>Value:</b> 2147483712
PRL_ERR_DISP_CONFIG_WRITE_ERR	<b>Value:</b> 2147483750
PRL_ERR_DISP_LOGON_ACTIONS_REACHED_UP_LIMIT	<b>Value:</b> 2147484693
PRL_ERR_DISP_SHUTDOWN_IN_PROCESS	<b>Value:</b> 2147484676

*continued on next page*

Name	Description
PRL_ERR_DISP_VM_COMMAND_CANT_BE_EXECUTED	<b>Value:</b> 2147484694
PRL_ERR_DISP_VM_IS_NOT_STARTED	<b>Value:</b> 2147484535
PRL_ERR_DOUBLE_INIT	<b>Value:</b> 2147483665
PRL_ERR_DROP_SUSPEND_FOR_HDD_QUERY	<b>Value:</b> 2147484677
PRL_ERR_ENTRY_ALREADY_EXISTS	<b>Value:</b> 2147483776
PRL_ERR_ENTRY_DIR_ALREADY_EXISTS	<b>Value:</b> 2147484688
PRL_ERR_ENTRY_DIR_DOES_NOT_EXIST	<b>Value:</b> 2147484689
PRL_ERR_ENTRY_DOES_NOT_EXIST	<b>Value:</b> 2147483769
PRL_ERR_EXCEED_LIMIT_MAX_RUNNING_VMS	<b>Value:</b> 2147484552
PRL_ERR_EXCEED_MEMORY_LIMIT	<b>Value:</b> 2147484165
PRL_ERR_FAILURE	<b>Value:</b> 2147483657
PRL_ERR_FDD_IMAGE_CLONE_TO_SELF	<b>Value:</b> 2147484437
PRL_ERR_FDD_IMAGE_COPY	<b>Value:</b> 2147484436
PRL_ERR_FDD_IMAGE_NOT_SPECIFIED	<b>Value:</b> 2147484169
PRL_ERR_FILE_DISK_SPACE_ERROR	<b>Value:</b> 2147484531
PRL_ERR_FILE_NOT_EXIST	<b>Value:</b> 2147484450
PRL_ERR_FILE_NOT_FOUND	<b>Value:</b> 2147483664
PRL_ERR_FILE_OR_DIR_ALREADY_EXISTS	<b>Value:</b> 2147484501
PRL_ERR_FILE_READ_ERROR	<b>Value:</b> 2147483920
PRL_ERR_FILE_TRANSFER_CANT_CREATE_DST_FILE	<b>Value:</b> 2147639300

*continued on next page*

Name	Description
PRL_ERR_FILE_TRANSFER_CANT_LOCATE_SRC_FILE	<b>Value:</b> 2147639297
PRL_ERR_FILE_TRANSFER_CANT_READ_SRC_FILE	<b>Value:</b> 2147639298
PRL_ERR_FILE_TRANSFER_CANT_WRITE_DATA_TO_DST_FILE	<b>Value:</b> 2147639301
PRL_ERR_FILE_TRANSFER_CLIENT_NOT_CONNECTED	<b>Value:</b> 2147639296
PRL_ERR_FILE_TRANSFER_DST_FILE_ALREADY_EXIST	<b>Value:</b> 2147639302
PRL_ERR_FILE_TRANSFER_INVALID_ARGUMENTS	<b>Value:</b> 2147639305
PRL_ERR_FILE_TRANSFER_INVALID_CREDENTIALS	<b>Value:</b> 2147639299
PRL_ERR_FILE_TRANSFER_OPERATION_NOT_SUPPORTED	<b>Value:</b> 2147639303
PRL_ERR_FILE_TRANSFER_UPLOAD_CANCELLED_BY_USER	<b>Value:</b> 2147639304
PRL_ERR_FILE_WRITE_ERROR	<b>Value:</b> 2147484530
PRL_ERR_FIXME	<b>Value:</b> 2147484199
PRL_ERR_FLOPPY_DRIVE_INVALID	<b>Value:</b> 2147484198
PRL_ERR_FLOPPY_IMAGE_ALREADY_EXIST	<b>Value:</b> 2147484246
PRL_ERR_FLOPPY_IMAGE_NOT_EXIST	<b>Value:</b> 2147483797
PRL_ERR_FREE_DISK_SPACE_FOR_CLONE	<b>Value:</b> 2147484529
PRL_ERR_GET_LICENSE_INVALID_ARG	<b>Value:</b> 2147483959
PRL_ERR_GET_MON_STATE_INVALID_ARG	<b>Value:</b> 2147483944

*continued on next page*

Name	Description
PRL_ERR_GET_MON_STATE_VM_NOT_CONFIGURED	<b>Value:</b> 2147483945
PRL_ERR_GET_MON_STATE_VM_NOT_CREATED	<b>Value:</b> 2147483943
PRL_ERR_GET_NET_SERVICE_STATUS_FAILED	<b>Value:</b> 2147484499
PRL_ERR_GET_RESOLUTION_TOOL_INVALID_ARG	<b>Value:</b> 2147484022
PRL_ERR_GET_RESOLUTION_TOOL_VMNOTCREATED	<b>Value:</b> 2147484021
PRL_ERR_GET_USER_HOME_DIR	<b>Value:</b> 2147484227
PRL_ERR_GUEST_MAC_INVALID_VERSION	<b>Value:</b> 2147484505
PRL_ERR_GUEST_MAC_NOT_ENOUGH_MEMORY	<b>Value:</b> 2147484504
PRL_ERR_GUEST_MAC_NOT_MACSERVER_HOST	<b>Value:</b> 2147484567
PRL_ERR_HANDSHAKE_FAILED	<b>Value:</b> 2147484690
PRL_ERR_HARD_DISK_IMAGE_CORRUPTED	<b>Value:</b> 2147491842
PRL_ERR_HDD_IMAGE_CLONE_TO_SELF	<b>Value:</b> 2147483796
PRL_ERR_HDD_IMAGE_COPY	<b>Value:</b> 2147484215
PRL_ERR_HDD_IMAGE_IS_ALREADY_EXIST	<b>Value:</b> 2147484296
PRL_ERR_HDD_IMAGE_NOT_EXIST	<b>Value:</b> 2147483795
PRL_ERR_HDD_IMAGE_NOT_SPECIFIED	<b>Value:</b> 2147484166
PRL_ERR_HOST_AMD_GUEST_MAC	<b>Value:</b> 2147484488
PRL_ERR_HVT_DISABLED	<b>Value:</b> 2147484481

*continued on next page*

Name	Description
PRL_ERR_HVT_NOT_PRESENT	Value: 2147484470
PRL_ERR_HVT_TURNED_OFF_IN_CONFIG	Value: 2147484519
PRL_ERR_IMAGE_NEED_TO_CONVERT	Value: 2147491841
PRL_ERR_IMPERSONATE_FAILED	Value: 2147484496
PRL_ERR_INCONSISTENCY_VM_CONFIG	Value: 2147483910
PRL_ERR_INCORRECT_CDROM_PATH	Value: 2147487745
PRL_ERR_INCORRECT_FDD_PATH	Value: 2147484469
PRL_ERR_INCORRECT_PATH	Value: 2147484452
PRL_ERR_INSERTING_VM_TO_CATALOGUE	Value: 2147484216
PRL_ERR_INSTALLATION_PROBLEM	Value: 2147484292
PRL_ERR_INVALID_ACCESS_TOKEN_RECEIVED	Value: 2147484229
PRL_ERR_INVALID_ARG	Value: 2147483651
PRL_ERR_INVALID_CONFIGURATION	Value: 2147484517
PRL_ERR_INVALID_HANDLE	Value: 2147483652
PRL_ERR_INVALID_HDD_GEOMETRY	Value: 2147484167
PRL_ERR_INVALID_MEMORY_SIZE	Value: 2147484164
PRL_ERR_INVALID_PARALLELS_DISK	Value: 2147491840
PRL_ERR_INVALID_RAM	Value: 2147483672
PRL_ERR_IO_AUTHENTICATION_FAILED	Value: 2147484420
PRL_ERR_IO_CONNECTION_TIMEOUT	Value: 2147484419
PRL_ERR_IO_DISABLED	Value: 2147484480

*continued on next page*

Name	Description
PRL_ERR_IO_NO_CONNECTION	<b>Value:</b> 2147484673
PRL_ERR_IO_SEND_QUEUE_IS_FULL	<b>Value:</b> 2147484416
PRL_ERR_IO_STOPPED	<b>Value:</b> 2147484417
PRL_ERR_IO_UNKNOWN_VM_ID	<b>Value:</b> 2147484418
PRL_ERR_LICENSE_AUTH_FAILED	<b>Value:</b> 2147553287
PRL_ERR_LICENSE_EXPIRED	<b>Value:</b> 2147553281
PRL_ERR_LICENSE_NOT_VALID	<b>Value:</b> 2147553280
PRL_ERR_LICENSE_WRONG_ADVANCED_FIELD	<b>Value:</b> 2147553288
PRL_ERR_LICENSE_WRONG_DISTRIBUTOR	<b>Value:</b> 2147553286
PRL_ERR_LICENSE_WRONG_LANGUAGE	<b>Value:</b> 2147553285
PRL_ERR_LICENSE_WRONG_PLATFORM	<b>Value:</b> 2147553284
PRL_ERR_LICENSE_WRONG_PRODUCT	<b>Value:</b> 2147553283
PRL_ERR_LICENSE_WRONG_VERSION	<b>Value:</b> 2147553282
PRL_ERR_LOCAL_AUTHENTICATION_FAILED	<b>Value:</b> 2147484465
PRL_ERR_LPT_DEVICE_ALREADY_EXIST	<b>Value:</b> 2147484245
PRL_ERR_LPT_OUTPUT_FILE_IS_NOT_SPECIFIED	<b>Value:</b> 2147484182
PRL_ERR_LPT_OUTPUT_FILE_NOT_EXIST	<b>Value:</b> 2147484184
PRL_ERR_MAC_ADDRESS_INCORRECT	<b>Value:</b> 2147484456
PRL_ERR_MAC_ADDRESS_IN_CORRECT_LENGTH	<b>Value:</b> 2147484177
PRL_ERR_MAC_ADDRESS_IS_EMPTY	<b>Value:</b> 2147484176

*continued on next page*

Name	Description
PRL_ERR_MAC_ADDRESSES_WITHIN_CORRECT_SYMBOLS	<b>Value:</b> 2147484180
PRL_ERR_MAC_ADDRESSES_WITH_2_ZERO_START	<b>Value:</b> 2147484178
PRL_ERR_MAC_ADDRESSES_WITH_ALL_ZEROS	<b>Value:</b> 2147484179
PRL_ERR_MAKE_DIRECTORY	<b>Value:</b> 2147484210
PRL_ERR_NETWORK_ROLLBACK_FAILED	<b>Value:</b> 2147484272
PRL_ERR_NOT_ALL_FILES_WAS_DELETED	<b>Value:</b> 2147483906
PRL_ERR_NOT_CONNECTED_TO_DISPATCHER	<b>Value:</b> 2147484241
PRL_ERR_NOT_ENOUGH_DISK_FREE_SPACE	<b>Value:</b> 2147484423
PRL_ERR_NOT_ENOUGH_PERMS_TO_OPEN_AUTHORIZATION_FILE	<b>Value:</b> 2147484312
PRL_ERR_NO_BOOTING_DEVICE_SELECTED	<b>Value:</b> 2147484468
PRL_ERR_NO_DATA	<b>Value:</b> 2147483668
PRL_ERR_NO_DISP_CONFIG_FOUND	<b>Value:</b> 2147483673
PRL_ERR_NO_MORE_FREE_INTERFACE_SLOTS	<b>Value:</b> 2147484521
PRL_ERR_NO_PROBLEM_REPORT_FOUND	<b>Value:</b> 2147483781
PRL_ERR_NO_TARGET_DIR_PATH_SPECIFIED	<b>Value:</b> 2147484679
PRL_ERR_NO_TARGET_PATH_SPECIFIED	<b>Value:</b> 2147483764
PRL_ERR_NO_VM_DIR_CONFIG_FOUND	<b>Value:</b> 2147483686
PRL_ERR_OBJECT_WAS_REMOVED	<b>Value:</b> 2147484678
PRL_ERR_ONLY_ADMIN_CAN_SET_PARAMETER_STARTLOGINMODE_ROOT	<b>Value:</b> 2147484565

*continued on next page*

Name	Description
PRL_ERR_OPEN_DISP_CONFIG_READ	<b>Value:</b> 2147483681
PRL_ERR_OPEN_DISP_CONFIG_WRITE	<b>Value:</b> 2147483682
PRL_ERR_OPEN_PROBLEM_REPORT_READ	<b>Value:</b> 2147483782
PRL_ERR_OPEN_VM_CONFIG_READ	<b>Value:</b> 2147483699
PRL_ERR_OPEN_VM_CONFIG_WRITE	<b>Value:</b> 2147483700
PRL_ERR_OPEN_VM_DIR_CONFIG_READ	<b>Value:</b> 2147483688
PRL_ERR_OPEN_VM_DIR_CONFIG_WRITE	<b>Value:</b> 2147483689
PRL_ERR_OPERATION_FAILED	<b>Value:</b> 2147483670
PRL_ERR_OPERATION_PENDING	<b>Value:</b> 2147483667
PRL_ERR_OPERATION_WAS_CANCELED	<b>Value:</b> 2147484277
PRL_ERR_OUT_OF_DISK_SPACE	<b>Value:</b> 2147484311
PRL_ERR_OUT_OF_MEMORY	<b>Value:</b> 2147483650
PRL_ERR_PARALLEL_PORT_IMAGE_NOT_EXIST	<b>Value:</b> 2147483799
PRL_ERR_PARALLEL_PORT_IMG_COPY	<b>Value:</b> 2147484214
PRL_ERR_PARAM_NOT_FOUND	<b>Value:</b> 2147483671
PRL_ERR_PARSE_CLIENT_PREFS	<b>Value:</b> 2147483927
PRL_ERR_PARSE_COMMON_SERVER_PREFS	<b>Value:</b> 2147483937
PRL_ERR_PARSE_DISP_CONFIG	<b>Value:</b> 2147483683
PRL_ERR_PARSE_FILESYSTEM_INFO	<b>Value:</b> 2147483762
PRL_ERR_PARSE_HARD_DISK_HW_INFO	<b>Value:</b> 2147483761

*continued on next page*

Name	Description
PRL_ERR_PARSE_HOST_HW_INFO	<b>Value:</b> 2147483760
PRL_ERR_PARSE_PROBLEM_REPORT	<b>Value:</b> 2147483780
PRL_ERR_PARSE_STATISTICS	<b>Value:</b> 2147484278
PRL_ERR_PARSE_USER_PROFILE	<b>Value:</b> 2147483763
PRL_ERR_PARSE_VM_CONFIG	<b>Value:</b> 2147483702
PRL_ERR_PARSE_VM_DIR_CONFIG	<b>Value:</b> 2147483696
PRL_ERR_PARSING_EVENT	<b>Value:</b> 2147483684
PRL_ERR_PREV_SESSION_IS_ACTIVE	<b>Value:</b> 2147483748
PRL_ERR_PROBLEM_REPORT_ALREADY_EXISTS	<b>Value:</b> 2147483784
PRL_ERR_PROBLEM_REPORT_FILE_NOTSET	<b>Value:</b> 2147483783
PRL_ERR_PROBLEM_REPORT_WRITE	<b>Value:</b> 2147483785
PRL_ERR_READONLY_FILESYSTEM	<b>Value:</b> 2147484502
PRL_ERR_READ_XML_CONTENT	<b>Value:</b> 2147483701
PRL_ERR_REBOOT_HOST	<b>Value:</b> 2147484498
PRL_ERR_REMOTE_DEVICE_EXIST	<b>Value:</b> 2147484281
PRL_ERR_REMOTE_DEVICE_NOT_EXIST	<b>Value:</b> 2147484288
PRL_ERR_REMOTE_DEVICE_NOT_REGISTERED	<b>Value:</b> 2147484280
PRL_ERR_REMOTE_DISPLAY_HOST_NOT_SPECIFIED	<b>Value:</b> 2147484484
PRL_ERR_REMOTE_DISPLAY_WRONG_PORT_NUMBER	<b>Value:</b> 2147484485

*continued on next page*

Name	Description
PRL_ERR_RETRIEVE_VM_CONFIG	Value: 2147484224
PRL_ERR_RETURN_CODE_RANG_EEND	Value: 2147483923
PRL_ERR_REVERT_IMPERSONATE_FAILED	Value: 2147484497
PRL_ERR_SAVE_VM_CATALOG	Value: 2147483792
PRL_ERR_SAVE_VM_CONFIG	Value: 2147484212
PRL_ERR_SEARCH_CONFIG_OPERATION_CANCELLED	Value: 2147483912
PRL_ERR_SEND_COMMAND_TOWS_FAILED	Value: 2147484240
PRL_ERR_SERIAL_IMG_COPY	Value: 2147484213
PRL_ERR_SERIAL_IMG_NOT_FOUND	Value: 2147484211
PRL_ERR_SERIAL_PORT_IMAGE_NOT_EXIST	Value: 2147483798
PRL_ERR_SERVER_GOES_DOWN	Value: 2147484232
PRL_ERR_SERVER_PREFS_EDIT_COLLISION	Value: 2147484454
PRL_ERR_SET_LICENSE_INVALID_ARG	Value: 2147483960
PRL_ERR_SET_LICENSE_INVALID_KEY	Value: 2147483961
PRL_ERR_SHUT_DOWN_NOTIFICATION_SERVICE	Value: 2147484231
PRL_ERR_SMC_ERROR	Value: 2147483921
PRL_ERR_SOME_TASKS_PRESENT	Value: 2147484515
PRL_ERR_SOME_VMS_RUNNING	Value: 2147484514
PRL_ERR_SOUND_DEVICE_WRITE_FAILED	Value: 2147504128
PRL_ERR_SOUND_IN_DEVICE_OPEN_FAILED	Value: 2147504130
PRL_ERR_SOUND_OUT_DEVICE_OPEN_FAILED	Value: 2147504129

*continued on next page*

Name	Description
PRL_ERR_SSL_HANDSHAKE_FAILED	<b>Value:</b> 2147484692
PRL_ERR_STATE_ALREADY_EXISTS	<b>Value:</b> 2147586069
PRL_ERR_STATE_CANT_LOAD_SPECIFIED_CFG	<b>Value:</b> 2147586084
PRL_ERR_STATE_CANT_OPEN_FOR_WRITE	<b>Value:</b> 2147586064
PRL_ERR_STATE_CANT_OPEN_IMAGE	<b>Value:</b> 2147586057
PRL_ERR_STATE_CANT_OPEN_LOCKED	<b>Value:</b> 2147586065
PRL_ERR_STATE_CANT_SAVE_LOCKED	<b>Value:</b> 2147586066
PRL_ERR_STATE_DELETE_NONCLOSED	<b>Value:</b> 2147586083
PRL_ERR_STATE_ERROR_CREATE_IMAGE	<b>Value:</b> 2147586055
PRL_ERR_STATE_ERROR_RENAMING_IMAGE	<b>Value:</b> 2147586056
PRL_ERR_STATE_FULL_DELETE_FAILED	<b>Value:</b> 2147586072
PRL_ERR_STATE_GETFREESPACE_FAILED	<b>Value:</b> 2147586085
PRL_ERR_STATE_INT_CORRUPTED	<b>Value:</b> 2147586067
PRL_ERR_STATE_INVALID_IMAGE_TYPE	<b>Value:</b> 2147586082
PRL_ERR_STATE_INVALID_PARAMETERS	<b>Value:</b> 2147586054
PRL_ERR_STATE_MEMORY_ERROR	<b>Value:</b> 2147586081
PRL_ERR_STATE_MERGE_FAILED	<b>Value:</b> 2147586073
PRL_ERR_STATE_MERGE_NO_SPACE	<b>Value:</b> 2147586080
PRL_ERR_STATE_NOT_OPENED	<b>Value:</b> 2147586070
PRL_ERR_STATE_NOT_PERMITTED	<b>Value:</b> 2147586071
PRL_ERR_STATE_NOT_RELEASED	<b>Value:</b> 2147586087

*continued on next page*

<b>Name</b>	<b>Description</b>
PRL_ERR_STATE_NO_DISKS	<b>Value:</b> 2147586049
PRL_ERR_STATE_NO_STATE	<b>Value:</b> 2147586068
PRL_ERR_STATE_PROCESS_RUNNING	<b>Value:</b> 2147586050
PRL_ERR_STATE_ROLLBACK_ERROR	<b>Value:</b> 2147586053
PRL_ERR_STATE_ROLLBACK_IN_PROGRESS	<b>Value:</b> 2147586052
PRL_ERR_STATE_STATFS_FAILED	<b>Value:</b> 2147586086
PRL_ERR_STATE_STOPPING_STATE	<b>Value:</b> 2147586051
PRL_ERR_STATE_UNEXPECTED_ERROR	<b>Value:</b> 2147586048
PRL_ERR_SYMBOL_NOT_FOUND	<b>Value:</b> 2147483666
PRL_ERR_TARGET_NAME_ALREADY_OCCUPIED	<b>Value:</b> 2147483767
PRL_ERR_TARGET_PATH_IS_NOT_DIRECTORY	<b>Value:</b> 2147483766
PRL_ERR_TASK_NOT_FOUND	<b>Value:</b> 2147483794
PRL_ERR_TIMEOUT	<b>Value:</b> 2147483669
PRL_ERR_TIS_INVALID_UID	<b>Value:</b> 2147645440
PRL_ERR_TOOLS_UNSUPPORTED_GUEST	<b>Value:</b> 2147484528
PRL_ERR_TOO_LOW_HDD_SIZE	<b>Value:</b> 2147484256
PRL_ERR_UNABLE_DROP_SUSPENDED_STATE	<b>Value:</b> 2147483907
PRL_ERR_UNABLE_SEND_REQUEST	<b>Value:</b> 2147484018
PRL_ERR_UNDER_OLD_HYPERVISOR	<b>Value:</b> 2147484248
PRL_ERR_UNEXPECTED	<b>Value:</b> 2147483649

*continued on next page*

Name	Description
PRL_ERR_UNIMPLEM- ENTED	<b>Value:</b> 2147483656
PRL_ERR_UNINITIALIZ- ED	<b>Value:</b> 2147483655
PRL_ERR_UNRECOGNI- ZED_REQUEST	<b>Value:</b> 2147483793
PRL_ERR_UNSUPPORTED_ DEVICE_TYPE	<b>Value:</b> 2147487746
PRL_ERR_UNSUPPORTED_ NETWORK_FILE_SY- STEM	<b>Value:</b> 2147484569
PRL_ERR_UPDATE_ME- M_VM_NOT_CREATED	<b>Value:</b> 2147484017
PRL_ERR_UPD_TOOLS_ VER_VM_NOT_CONFIG- URED	<b>Value:</b> 2147484020
PRL_ERR_UPD_TOOLS_ VER_VM_NOT_CREATE- D	<b>Value:</b> 2147484019
PRL_ERR_UPD_UPDAT- ER_CONFIG	<b>Value:</b> 2147577856
PRL_ERR_UPD_UPDAT- ES	<b>Value:</b> 2147577857
PRL_ERR_USER_CANT_ CHANGE_ACCESS_PRO- FILE	<b>Value:</b> 2147483929
PRL_ERR_USER_CANT_ CHANGE_PROFILE_AC- CESS_PART	<b>Value:</b> 2147484249
PRL_ERR_USER_CANT_ CHANGE_READ_ONLY_ VALUE	<b>Value:</b> 2147483936
PRL_ERR_USER_DIREC- TORY_NOT_SET	<b>Value:</b> 2147483751
PRL_ERR_USER_IS_ALR- EADY_LOGGED	<b>Value:</b> 2147483745
PRL_ERR_USER_NOT_F- OUND	<b>Value:</b> 2147484560
PRL_ERR_USER_NO_AU- TH_TO_CREATE_VM_IN- _DIR	<b>Value:</b> 2147484438

*continued on next page*

Name	Description
PRL_ERR_USER_NO_AUTH_TO_EDIT_SERVER_SETTINGS	Value: 2147484247
PRL_ERR_USER_NO_AUTH_TO_EDIT_VM	Value: 2147484449
PRL_ERR_USER_NO_AUTH_TO_SAVE_FILES	Value: 2147484435
PRL_ERR_USER_OPERATION_NOT_AUTHORIZED	Value: 2147483737
PRL_ERR_USER_PROFILE_WAS_CHANGED	Value: 2147483928
PRL_ERR_VA_CONFIG	Value: 2147581952
PRL_ERR_VMCONF_AUTOSTART_FROM_CURRENT_USER_FORBIDDEN	Value: 2147643475
PRL_ERR_VMCONF_BOOT_OPTION_DEVICE_NOT_EXISTS	Value: 2147643474
PRL_ERR_VMCONF_BOOT_OPTION_DUPLICATE_DEVICE	Value: 2147643473
PRL_ERR_VMCONF_BOOT_OPTION_INVALID_DEVICE_TYPE	Value: 2147643472
PRL_ERR_VMCONF_CD_DVD_ROM_DUPLICATE_SYS_NAME	Value: 2147644417
PRL_ERR_VMCONF_CD_DVD_ROM_IMAGE_IS_NOT_EXIST	Value: 2147644418
PRL_ERR_VMCONF_CD_DVD_ROM_SYS_NAME_IS_EMPTY	Value: 2147644416
PRL_ERR_VMCONF_CPU_COUNT_MORE_HOST_CPU_COUNT	Value: 2147643906
PRL_ERR_VMCONF_CPU_COUNT_MORE_MAX_CPU_COUNT	Value: 2147643905
PRL_ERR_VMCONF_CPU_ZERO_COUNT	Value: 2147643904

*continued on next page*

Name	Description
PRL_ERR_VMCONF_FLOPPY_DISK_IMAGE_IS_NOT_EXIST	<b>Value:</b> 2147644241
PRL_ERR_VMCONF_FLOPPY_DISK_IS_NOT_ACCESSIBLE	<b>Value:</b> 2147644242
PRL_ERR_VMCONF_FLOPPY_DISK_SYS_NAME_IS_EMPTY	<b>Value:</b> 2147644240
PRL_ERR_VMCONF_HARD_DISK_DUPLICATE_SYS_NAME	<b>Value:</b> 2147644499
PRL_ERR_VMCONF_HARD_DISK_IMAGE_IS_NOT_EXIST	<b>Value:</b> 2147644497
PRL_ERR_VMCONF_HARD_DISK_IMAGE_IS_NOT_VALID	<b>Value:</b> 2147644498
PRL_ERR_VMCONF_HARD_DISK_SYS_NAME_IS_EMPTY	<b>Value:</b> 2147644496
PRL_ERR_VMCONF_IDE_DEVICES_COUNT_OUT_OF_RANGE	<b>Value:</b> 2147645184
PRL_ERR_VMCONF_IDE_DEVICES_DUPLICATE_STACK_INDEX	<b>Value:</b> 2147645185
PRL_ERR_VMCONF_MAIN_MEMORY_NOT_4_RATIO_SIZE	<b>Value:</b> 2147643986
PRL_ERR_VMCONF_MAIN_MEMORY_OUT_OF_RANGE	<b>Value:</b> 2147643985
PRL_ERR_VMCONF_MAIN_MEMORY_ZERO_SIZE	<b>Value:</b> 2147643984
PRL_ERR_VMCONF_NETWORK_ADAPTER_DUPLICATE_MAC_ADDRESS	<b>Value:</b> 2147644674

*continued on next page*

Name	Description
PRL_ERR_VMCONF_NETWORK_ADAPTER_INVALID_BOUND_INDEX	<b>Value:</b> 2147644672
PRL_ERR_VMCONF_NETWORK_ADAPTER_INVALID_MAC_ADDRESS	<b>Value:</b> 2147644673
PRL_ERR_VMCONF_PARALLEL_PORT_IMAGE_IS_NOT_EXIST	<b>Value:</b> 2147645009
PRL_ERR_VMCONF_PARALLEL_PORT_SYS_NAME_IS_EMPTY	<b>Value:</b> 2147645008
PRL_ERR_VMCONF_REMOTE_DISPLAY_HOST_IP_ADDRESS_IS_ZERO	<b>Value:</b> 2147643649
PRL_ERR_VMCONF_REMOTE_DISPLAY_INVALID_HOST_IP_ADDRESS	<b>Value:</b> 2147643650
PRL_ERR_VMCONF_REMOTE_DISPLAY_PORT_NUMBER_IS_ZERO	<b>Value:</b> 2147643648
PRL_ERR_VMCONF_SCSI_DEVICES_COUNT_OUT_OF_RANGE	<b>Value:</b> 2147645264
PRL_ERR_VMCONF_SCSI_DEVICES_DUPLICATE_STACK_INDEX	<b>Value:</b> 2147645265
PRL_ERR_VMCONF_SERIAL_PORT_IMAGE_IS_NOT_EXIST	<b>Value:</b> 2147644929
PRL_ERR_VMCONF_SERIAL_PORT_SYS_NAME_IS_EMPTY	<b>Value:</b> 2147644928
PRL_ERR_VMCONF_SHARED_FOLDERS_DUPLICATE_FOLDER_NAME	<b>Value:</b> 2147643729
PRL_ERR_VMCONF_SHARED_FOLDERS_DUPLICATE_FOLDER_PATH	<b>Value:</b> 2147643731
PRL_ERR_VMCONF_SHARED_FOLDERS_EMPTY_FOLDER_NAME	<b>Value:</b> 2147643728

*continued on next page*

Name	Description
PRL_ERR_VMCONF_SHARED_FOLDERS_INVALID_FOLDER_PATH	Value: 2147643730
PRL_ERR_VMCONF_SOUND_MIXER_IS_EMPTY	Value: 2147644752
PRL_ERR_VMCONF_SOUND_OUTPUT_IS_EMPTY	Value: 2147644753
PRL_ERR_VMCONF_UNKNOWN_OS_TYPE	Value: 2147643394
PRL_ERR_VMCONF_UNKNOWN_OS_VERSION	Value: 2147643395
PRL_ERR_VMCONF_VALIDATION_FAILED	Value: 2147643392
PRL_ERR_VMCONF_VIDEO_MEMORY_OUT_OF_RANGE	Value: 2147644160
PRL_ERR_VMCONF_VM_NAME_HAS_INVALID_SYMBOL	Value: 2147643396
PRL_ERR_VMCONF_VM_NAME_IS_EMPTY	Value: 2147643393
PRL_ERR_VMDIR_INVALID_PATH	Value: 2147484293
PRL_ERR_VMDIR_PATH_IS_NOT_ABSOLUTE	Value: 2147484294
PRL_ERR_VM_ABORT	Value: 2147484453
PRL_ERR_VM_ALLOC_MEM_DRV_NOT_STARTED	Value: 2147484036
PRL_ERR_VM_ALLOC_VM_MEMORY	Value: 2147484037
PRL_ERR_VM_ALREADY_CONNECTED	Value: 2147484561
PRL_ERR_VM_ALREADY_CREATED	Value: 2147483940
PRL_ERR_VM_ALREADY_REGISTERED_UNIQUE_PARAMS	Value: 2147483924
PRL_ERR_VM_ALREADY_REGISTERED_VM_NAME	Value: 2147483925

*continued on next page*

Name	Description
PRL_ERR_VM_ALREADY_REGISTERED_VM_PATH	<b>Value:</b> 2147483747
PRL_ERR_VM_ALREADY_REGISTERED_VM_UUID	<b>Value:</b> 2147483746
PRL_ERR_VM_ALREADY_RESUMED	<b>Value:</b> 2147483730
PRL_ERR_VM_ALREADY_RUNNING	<b>Value:</b> 2147484674
PRL_ERR_VM_ALREADY_SUSPENDED	<b>Value:</b> 2147483728
PRL_ERR_VM_APPLY_CHANGES_PENDING	<b>Value:</b> 2147484518
PRL_ERR_VM_APPLY_CONFIG_FAILED	<b>Value:</b> 2147484304
PRL_ERR_VM_APPLY_CONFIG_NEEDS_REBOOT	<b>Value:</b> 2147484305
PRL_ERR_VM_ASYNC_CD_CONSTRUCTOR	<b>Value:</b> 2147484032
PRL_ERR_VM_BAD_OS_TYPE	<b>Value:</b> 2147484162
PRL_ERR_VM_CANT_CLONE_HDD_MISSING	<b>Value:</b> 2147484448
PRL_ERR_VM_CANT_CLONE_RUNNING	<b>Value:</b> 2147484306
PRL_ERR_VM_CANT_DELETE_RUNNING	<b>Value:</b> 2147484307
PRL_ERR_VM_CANT_UNREG_RUNNING	<b>Value:</b> 2147484308
PRL_ERR_VM_COMPACT_PROCESSING	<b>Value:</b> 2147647488
PRL_ERR_VM_CONFIG_ALREADY_EXISTS	<b>Value:</b> 2147483703
PRL_ERR_VM_CONFIG_DOESNT_EXIST	<b>Value:</b> 2147484291
PRL_ERR_VM_CONFIG_INVALID_SERVER_UUID	<b>Value:</b> 2147484533
PRL_ERR_VM_CONFIG_INVALID_VM_UUID	<b>Value:</b> 2147484532

*continued on next page*

Name	Description
PRL_ERR_VM_CONFIG_WAS_CHANGED	<b>Value:</b> 2147483922
PRL_ERR_VM_CONF_CHANGED_INVALID_ARG	<b>Value:</b> 2147483956
PRL_ERR_VM_CONF_CHANGED_NOT_CREATED	<b>Value:</b> 2147483955
PRL_ERR_VM_CONF_CHANGED_NOT_STARTED	<b>Value:</b> 2147483957
PRL_ERR_VM_CONF_CHANGED_UNSUPPORTED_DEV	<b>Value:</b> 2147483958
PRL_ERR_VM_COULDNT_BE_STARTED_UNDER_SPECIFIED_USER	<b>Value:</b> 2147484568
PRL_ERR_VM_CREATE_HDD_IMG_INVALID_ARG	<b>Value:</b> 2147484007
PRL_ERR_VM_CREATE_HDD_IMG_INVALID_CREATE	<b>Value:</b> 2147484008
PRL_ERR_VM_CREATE_INVALID_ARG	<b>Value:</b> 2147483939
PRL_ERR_VM_DEVICES_INITIALIZATION_FAILED	<b>Value:</b> 2147484275
PRL_ERR_VM_DEVICES_TERMINATION_FAILED	<b>Value:</b> 2147484276
PRL_ERR_VM_DEV_CHANGE_MEDIA_FAILED	<b>Value:</b> 2147483735
PRL_ERR_VM_DEV_CONNECT_FAILED	<b>Value:</b> 2147483733
PRL_ERR_VM_DEV_DISCONNECT_FAILED	<b>Value:</b> 2147483734
PRL_ERR_VM_DIRECTORY_NOT_EXIST	<b>Value:</b> 2147484455
PRL_ERR_VM_DIRECTORY_NOT_INITIALIZED	<b>Value:</b> 2147483749
PRL_ERR_VM_DIR_CONFIG_ALREADY_EXISTS	<b>Value:</b> 2147483697

*continued on next page*

Name	Description
PRL_ERR_VM_DIR_FILE_NOT_SET	<b>Value:</b> 2147483705
PRL_ERR_VM_DOES_NOT_STOPPED	<b>Value:</b> 2147484225
PRL_ERR_VM_EMPTY_NAME_OF_CLONE	<b>Value:</b> 2147484209
PRL_ERR_VM_FILES_ALREADY_REMOVED	<b>Value:</b> 2147484421
PRL_ERR_VM_FREE_VM_MEMORY	<b>Value:</b> 2147484038
PRL_ERR_VM_GET_CONFIG_FAILED	<b>Value:</b> 2147483721
PRL_ERR_VM_GET_CONFIG_INVALID_ARG	<b>Value:</b> 2147483952
PRL_ERR_VM_GET_CONFIG_NOT_CONFIGURED	<b>Value:</b> 2147483953
PRL_ERR_VM_GET_DEVICE_STATE_FAILED	<b>Value:</b> 2147483736
PRL_ERR_VM_GET_HDD_IMG_INVALID_ARG	<b>Value:</b> 2147484009
PRL_ERR_VM_GET_HDD_IMG_NOT_OPEN	<b>Value:</b> 2147484016
PRL_ERR_VM_GET_INDICATORS_INVALID_ARG	<b>Value:</b> 2147484005
PRL_ERR_VM_GET_INDICATORS_NOT_CREATED	<b>Value:</b> 2147484004
PRL_ERR_VM_GET_INDICATORS_NOT_RUNNING	<b>Value:</b> 2147484006
PRL_ERR_VM_GET_PROBLEM_REPORT_FAILED	<b>Value:</b> 2147483720
PRL_ERR_VM_GET_SCR_SIZE_NOT_CREATED	<b>Value:</b> 2147483976
PRL_ERR_VM_GET_SCR_UPDATED_NOT_CREATED	<b>Value:</b> 2147483977

*continued on next page*

Name	Description
PRL_ERR_VM_GET_SCR_UPDATED_NOT_RUNNING	<b>Value:</b> 2147483984
PRL_ERR_VM_GET_STATUS_FAILED	<b>Value:</b> 2147483719
PRL_ERR_VM_GET_STATUS_INVALID_ARG	<b>Value:</b> 2147483942
PRL_ERR_VM_GUESTMEM_FAIL	<b>Value:</b> 2147484562
PRL_ERR_VM_HDD_SIZE	<b>Value:</b> 2147484023
PRL_ERR_VM_HOME_PATH_IS_NOT_EMPTY	<b>Value:</b> 2147484200
PRL_ERR_VM_HYPERVISOR_COMM	<b>Value:</b> 2147484056
PRL_ERR_VM_INIT_MONITOR	<b>Value:</b> 2147484048
PRL_ERR_VM_INIT_VCPU	<b>Value:</b> 2147484049
PRL_ERR_VM_INTERACT_PRLS_DRIVER	<b>Value:</b> 2147484039
PRL_ERR_VM_INTERNAL_OS_ERROR	<b>Value:</b> 2147484265
PRL_ERR_VM_INVALID_SWAP_REGION	<b>Value:</b> 2147484536
PRL_ERR_VM_IN_HDD_CONSTRUCTOR	<b>Value:</b> 2147484024
PRL_ERR_VM_LOAD_BINARY_FAILED	<b>Value:</b> 2147484675
PRL_ERR_VM_LOAD_MONITOR	<b>Value:</b> 2147484053
PRL_ERR_VM_MAP_VM_MEM0	<b>Value:</b> 2147484041
PRL_ERR_VM_MONITOR	<b>Value:</b> 2147484160
PRL_ERR_VM_MUST_BE_STOPPED_BEFORE_RENAMING	<b>Value:</b> 2147484534
PRL_ERR_VM_MUST_BE_STOPPED_FOR_CHANGE_DEVICES	<b>Value:</b> 2147484548
PRL_ERR_VM_NAME_IS_EMPTY	<b>Value:</b> 2147483926

*continued on next page*

Name	Description
PRL_ERR_VM_NOT_CREATED	<b>Value:</b> 2147483941
PRL_ERR_VM_OPERATION_FAILED	<b>Value:</b> 2147483753
PRL_ERR_VM_PAUSE_ALREADY_PAUSED	<b>Value:</b> 2147483975
PRL_ERR_VM_PAUSE_FAILED	<b>Value:</b> 2147483729
PRL_ERR_VM_PAUSE_NOT_CREATED	<b>Value:</b> 2147483973
PRL_ERR_VM_PAUSE_NOT_STARTED	<b>Value:</b> 2147483974
PRL_ERR_VM_POWER_OFF_FAILED	<b>Value:</b> 2147483717
PRL_ERR_VM_POWER_ON_FAILED	<b>Value:</b> 2147483716
PRL_ERR_VM_PRLS_DESCRIPTOR_ALLOC_PHYS_MEM	<b>Value:</b> 2147484057
PRL_ERR_VM_PROCESS_IS_NOT_STARTED	<b>Value:</b> 2147483752
PRL_ERR_VM_REQUEST_NOT_SUPPORTED	<b>Value:</b> 2147484161
PRL_ERR_VM_RESET_FAILED	<b>Value:</b> 2147483718
PRL_ERR_VM_RESTART_NOT_CREATED	<b>Value:</b> 2147483971
PRL_ERR_VM_RESTART_NOT_STARTED	<b>Value:</b> 2147483972
PRL_ERR_VM_RESTORE_STATE_FAILED	<b>Value:</b> 2147483732
PRL_ERR_VM_RESUME_FAILED	<b>Value:</b> 2147614726
PRL_ERR_VM_SAVE_STATE_FAILED	<b>Value:</b> 2147483731
PRL_ERR_VM_SEND_ABS_MOUSE_NOT_CREATED	<b>Value:</b> 2147483992
PRL_ERR_VM_SEND_ABS_MOUSE_NOT_RUNNING	<b>Value:</b> 2147483993

*continued on next page*

Name	Description
PRL_ERR_VM_SEND_ABS_MOUSE_PAUSED	<b>Value:</b> 2147484000
PRL_ERR_VM_SEND_KEYBOARD_NOT_CREATED	<b>Value:</b> 2147484001
PRL_ERR_VM_SEND_KEYBOARD_NOT_RUNNING	<b>Value:</b> 2147484002
PRL_ERR_VM_SEND_KEYBOARD_PAUSED	<b>Value:</b> 2147484003
PRL_ERR_VM_SEND_REL_MOUSE_NOT_CREATED	<b>Value:</b> 2147483989
PRL_ERR_VM_SEND_REL_MOUSE_NOT_RUNNING	<b>Value:</b> 2147483990
PRL_ERR_VM_SEND_REL_MOUSE_PAUSED	<b>Value:</b> 2147483991
PRL_ERR_VM_SET_CONFIG_FAILED	<b>Value:</b> 2147483715
PRL_ERR_VM_SET_CONFIG_INVALID_ARG	<b>Value:</b> 2147483954
PRL_ERR_VM_SET_LICENSE_FAILED	<b>Value:</b> 2147483714
PRL_ERR_VM_SET_VISIBLE_NOT_CREATED	<b>Value:</b> 2147483985
PRL_ERR_VM_SPECIFY_GUEST_INSTALL_FILES	<b>Value:</b> 2147484313
PRL_ERR_VM_START_FAILED	<b>Value:</b> 2147483713
PRL_ERR_VM_START_NOT_CONFIGURED	<b>Value:</b> 2147483969
PRL_ERR_VM_START_NOT_CREATED	<b>Value:</b> 2147483968
PRL_ERR_VM_STOP_NOT_CREATED	<b>Value:</b> 2147483970
PRL_ERR_VM_SUSPENDED_CHANGED_VM_CONFIG	<b>Value:</b> 2147614728
PRL_ERR_VM_SUSPENDED_FAILED	<b>Value:</b> 2147614725

*continued on next page*

Name	Description
PRL_ERR_VM_TO_REGISTER_IS_NOT_SPECIFIED	<b>Value:</b> 2147484471
PRL_ERR_VM_UNABLE_ALLOC_MEM	<b>Value:</b> 2147484040
PRL_ERR_VM_UNABLE_ALLOC_MEM_MONITOR	<b>Value:</b> 2147484054
PRL_ERR_VM_UNABLE_CREATE_TIMER	<b>Value:</b> 2147484055
PRL_ERR_VM_UNABLE_GET_GUEST_CPU	<b>Value:</b> 2147484050
PRL_ERR_VM_UNABLE_OPEN_DISK_IMAGE	<b>Value:</b> 2147484033
PRL_ERR_VM_UNABLE_SEND_REQUEST	<b>Value:</b> 2147484035
PRL_ERR_VM_UNABLE_TO_OPEN_VIRTUAL_BRIDGE	<b>Value:</b> 2147484025
PRL_ERR_VM_UNDEFINED_API_CALL	<b>Value:</b> 2147484034
PRL_ERR_VM_USER_AUTHENTICATION_FAILED	<b>Value:</b> 2147484566
PRL_ERR_VM_UUID_EMPTY	<b>Value:</b> 2147484208
PRL_ERR_VM_UUID_NOT_FOUND	<b>Value:</b> 2147483909
PRL_ERR_VM_VALLOC_MON_BODY	<b>Value:</b> 2147484051
PRL_ERR_VM_VALLOC_PE_IMG	<b>Value:</b> 2147484052
PRL_ERR_VM_VIEW_SCR_INVALID_ARG	<b>Value:</b> 2147483987
PRL_ERR_VM_VIEW_SCR_NOT_CREATED	<b>Value:</b> 2147483986
PRL_ERR_VM_VIEW_SCR_NOT_RUNNING	<b>Value:</b> 2147483988
PRL_ERR_VTD_DEVICE_NOT_MAPPED	<b>Value:</b> 2147557376
PRL_ERR_VTX_ENABLED_ONLY_IN_SMX	<b>Value:</b> 2147484482

*continued on next page*

Name	Description
PRL_ERR_WINDOWS_EX- PRESS_INSTALL_USE- R_NAME_EMPTY	<b>Value:</b> 2147484440
PRL_ERR_WRONG_CON- NECTION_SECURITY_L- LEVEL	<b>Value:</b> 2147484279
PRL_ERR_WRONG_PRO- TOCOL_VERSION	<b>Value:</b> 2147484691
PRL_ERR_WS_DISP_CO- NNECTION_CLOSED	<b>Value:</b> 2147484230
PRL_ERR_X64GUEST_O- N_X32HOST	<b>Value:</b> 2147484500
PRL_ERR_X64GUEST_O- N_X64VM_HVT_DISABL- ED	<b>Value:</b> 2147484503
PRL_NET_ADAPTER_N- OT_EXIST	<b>Value:</b> 2147500036
PRL_NET_BIND_FAILE- D	<b>Value:</b> 2147500041
PRL_NET_CABLE_DISC- ONNECTED	<b>Value:</b> 2147500048
PRL_NET_CONNECTIO- N_SHARING_CONFLICT	<b>Value:</b> 2147500053
PRL_NET_ERR_ADAPT- ER_CONFIG	<b>Value:</b> 2147500037
PRL_NET_ERR_ETH_NO- BINDABLE_ADAPTER	<b>Value:</b> 2147500038
PRL_NET_ERR_PRL_NO- BINDABLE_ADAPTER	<b>Value:</b> 2147500039
PRL_NET_ETHLIST_CR- EATE_ERROR	<b>Value:</b> 2147500032
PRL_NET_INSTALL_FAI- LED	<b>Value:</b> 2147500051
PRL_NET_INSTALL_TIM- EOUT	<b>Value:</b> 2147500050
PRL_NET_PRLNET_OPE- N_FAILED	<b>Value:</b> 2147500040
PRL_NET_RENAME_FAI- LED	<b>Value:</b> 2147500049
PRL_NET_RESTORE_DE- FAULTS_PARTITIAL_S- UCCESS	<b>Value:</b> 2147500054

*continued on next page*

Name	Description
PRL_NET_SRV_NOTIFY_ERROR	<b>Value:</b> 2147500035
PRL_NET_SYSTEM_ERROR	<b>Value:</b> 2147500033
PRL_NET_UNINSTALL_FAILED	<b>Value:</b> 2147500052
PRL_NET_VLAN_UNSUPPORTED_IN_THIS_VERSION	<b>Value:</b> 2147500055
PRL_NET_WINSCM_OPEN_ERROR	<b>Value:</b> 2147500034
PRL_REQUEST_CDD_IMAGE_NOT_EXIST	<b>Value:</b> 2147484192
PRL_REQUEST_CD_DRIVE_NOT_EXIST	<b>Value:</b> 2147484193
PRL_REQUEST_EXCEED_RECOMMENDED_MEMORY	<b>Value:</b> 2147484185
PRL_REQUEST_FDD_DRIVE_NOT_EXIST	<b>Value:</b> 2147484197
PRL_REQUEST_FDD_IMAGE_NOT_EXIST	<b>Value:</b> 2147484196
PRL_REQUEST_HDD_IMAGE_NOT_EXIST	<b>Value:</b> 2147484195
PRL_REQUEST_HDD_SAME_IMAGE	<b>Value:</b> 2147484194
PRL_REQUEST_POWER_OFF	<b>Value:</b> 2147484433
PRL_REQUEST_RESET	<b>Value:</b> 2147484434
PRL_WARN_FAT32_FILE_OVERRUN	<b>Value:</b> 2147484263
PRL_WARN_HDD_ERROR	<b>Value:</b> 2147484264
PRL_WARN_IMAGE_IS_STOLEN	<b>Value:</b> 2147484260
PRL_WARN_LAN_NOT_CONNECTED	<b>Value:</b> 2147484257
PRL_WARN_TARGET_HOST_DISK_IS_FULL	<b>Value:</b> 2147484261
PRL_WARN_TOO_LOW_HDD_SIZE	<b>Value:</b> 2147484297
PRL_WARN_UNABLE_OPEN_DEVICE	<b>Value:</b> 2147484259

*continued on next page*

---

Name	Description
PRL_WARN_UNABLE_OPEN_DEVICE_NAME	<b>Value:</b> 2147484258
PRL_WARN_UNKNOWN_HDD_REQUEST_TYPE	<b>Value:</b> 2147484262

## 4 Module prlsdk'

### 4.1 Functions

**CheckServerAlive**(*hostname*, *timeout=DefaultTimeout*)

Lets to determine whether Parallels server on specified host alive If alive - ServerInfo will be returned else None.

**Return Value**  
ServerInfo

**DefaultJobTimeout**(...)

**DefaultKeyActWait**(...)

**DefaultTimeout**(...)

**DeinitializeSDK**()

Method force SDK deinitialization.

**GetSDKLibraryPath**()

Return path from witch library will be loaded.

**Return Value**  
string

**GetServerList**(*msec*)

Lookups Parallels servers at the LAN during specified timeout.

**Return Value**  
list of ServerInfo objects

**InitializeSDK**()

Method initialize SDK. Return False if it was already initialized, else True.

**Return Value**  
bool

<b>IsSDKInitialized()</b>
Return True if SDK allready loaded.
<b>Return Value</b>
bool

<b>SetSDKLibraryPath(<i>new_path</i>)</b>
Return path from witch library will be loaded. Raise exception if SDK API allready initialized.
<b>Return Value</b>
string

<b>__debug_level__(...)</b>
-----------------------------

## 4.2 Variables

Name	Description
__SDK__	<b>Value:</b> <prlsdk.__SDK__ instance at 0x00A55130, version '3.0.2130...

## 4.3 Class Base



**Known Subclasses:** prlsdk.BaseAsync, prlsdk.BaseServer, prlsdk.Event, prlsdk.Job, prlsdk.ServerInfo, prlsdkapi.FoundVm

Base objects

### 4.3.1 Methods

<b>__cmp__(<i>x, y</i>)</b>
cmp(x,y)

<b>__init__(...)</b>
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

```
--new--(T, S, ...)
```

**Return Value**

a new object with type *S*, a subtype of *T*

Overrides: object...new--

```
--repr--(x)
```

```
repr(x)
```

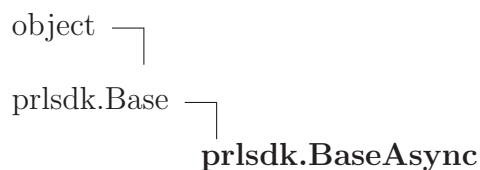
Overrides: object...repr--

**Inherited from object**

```
--delattr--(), --getattr--(), --hash--(), --reduce--(), --reduce_ex--(), --setattr--(),
--str--()
```

**4.3.2 Properties**

Name	Description
<code>--handle--</code>	
<code>handle_type</code>	handle type
<i>Inherited from object</i>	
<code>--class--</code>	

**4.4 Class BaseAsync**

**Known Subclasses:** prlsdk.BaseVm, prlsdk.OwnerRefAsync, prlsdk.Result, prlsdkapi.FSEntry, prlsdkapi.FSInfo, prlsdkapi.LicenceInfo, prlsdkapi.LoginResponse, prlsdkapi.NetServiceStatus, prlsdkapi.RunningTask, prlsdkapi.SrvCfgDev, prlsdkapi.SrvCfgHddPart, prlsdkapi.SrvConfig, prlsdkapi.SrvInfo, prlsdkapi.UserConfig, prlsdkapi.UsrInfo, prlsdkapi.VmCPUStat, prlsdkapi.VmDiskPartS, prlsdkapi.VmDiskStat, prlsdkapi.VmIFaceStat, prlsdkapi.VmInfo, prlsdkapi.VmProcStat, prlsdkapi.VmStat, prlsdkapi.VmToolsInfo, prlsdkapi.VmUserStat

Base async objects

## 4.4.1 Methods

<code>--cmp--(x, y)</code>
----------------------------

<code>cmp(x,y)</code>
-----------------------

Overrides: <code>prlsdk.Base.--cmp--</code>
---

<code>--init--(...)</code>
----------------------------

<code>x.--init--(...)</code> initializes x; see <code>x.--class--.--doc--</code> for signature
--

Overrides: <code>object.--init--</code>
---

<code>--new--(T, S, ...)</code>
---------------------------------

**Return Value**

a new object with type S, a subtype of T
--

Overrides: <code>object.--new--</code>
--

<code>--repr--(x)</code>
--------------------------

<code>repr(x)</code>
----------------------

Overrides: <code>object.--repr--</code>
---

<code>cancel(BaseAsync)</code>
--------------------------------

Start canceling this job, this method is asynchone so it return Result object.
--

**Return Value**

Result
--------

<code>wait(BaseAsync, msec=DefaultJobWait)</code>
---

Waits for job finished for passed milliseconds if successefully finished self object is returned If timeout -> <code>PrlJobTimeoutError</code> will be raised . If job finished with error <code>ParallelsError</code> will be raised.
--

**Return Value**

<code>BaseAsync</code>
------------------------

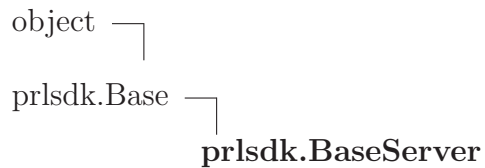
*Inherited from object*

<code>--delattr--()</code> , <code>--getattr--()</code> , <code>--hash--()</code> , <code>--reduce--()</code> , <code>--reduce_ex--()</code> , <code>--setattr--()</code> , <code>--str--()</code>
--

## 4.4.2 Properties

Name	Description
<code>--expected_h_type--</code>	
<code>--job--</code>	
<code>finished</code>	Shows job status
<code>handle_type</code>	Returns handle type
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

## 4.5 Class BaseServer



**Known Subclasses:** prlsdkapi.Server

Server base object class

### 4.5.1 Methods

<code>--init--(...)</code>
<code>x.__init__(...)</code> initializes x; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

<code>--new--(T, S, ...)</code>
<b>Return Value</b>
a new object with type S, a subtype of T
Overrides: <code>object.__new__</code>

**login**(*Server*, *host*, *user*, *password*, *port*=0, *secur\_level*=LOW, *session\_uid*=, *timeout*=0)

*secur\_level* – see *prlsdk.consts.PSL\_xxx* Login operation should be performed before to allow further server operations. Through Result object you could check that operation is finished. All method that checks for connection will wait until login finished.

**Return Value**

Result object

**login\_local**(*Server*, *port*=0, *secur\_level*=LOW, *session\_uid*=)

*secur\_level* – see *prlsdk.consts.PSL\_xxx* Login operation should be performed before to allow further server operations. Through Result object you could check that operation is finished. All method that checks for connection will wait until login finished.

**Return Value**

Result object

**logoff**(*Server*)

Logoff operation will finish interactive connection session.

**Return Value**

Job object

**reg\_event\_handler**(*Server*, *handler*, *user\_data*=None)

Register server event handler. handler should take 2 arguments: Event object, *user\_data*.

**shutdown**(*srv*, *Srv*, *force*=False)

Let to shutdown server connection established with

**Return Value**

Job

**unreg\_event\_handler**(*Server*, *handler*)

Unregister server event handler.

*Inherited from prlsdk.Base(Section 4.3)*

*--cmp--*() , *--repr--*()

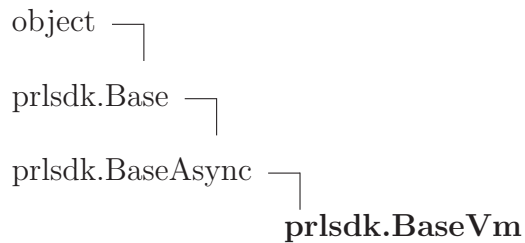
*Inherited from object*

```
__delattr__(), __getattr__(), __hash__(), __reduce__(), __reduce_ex__(), __setattr__(),
__str__()
```

#### 4.5.2 Properties

Name	Description
__editing_common__	
__editing_user__	
__login_result__	
autostart_editing	
connected	Return connected state
last_session_uuid	
session_uuid	Retruns last server session uuid
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__, handle_type	
<i>Inherited from object</i>	
__class__	

#### 4.6 Class BaseVm



**Known Subclasses:** prlsdkapi.Vm

Vm base object class

##### 4.6.1 Methods

<b>__init__</b> (...)
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

```
__new__(T, S, ...)
```

**Return Value**

a new object with type S, a subtype of T

Overrides: object.\_\_new\_\_

***Inherited from prlsdk.BaseAsync (Section 4.4)***

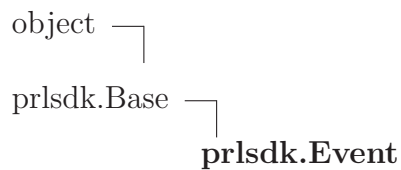
```
__cmp__(), __repr__(), cancel(), wait()
```

***Inherited from object***

```
__delattr__(), __getattr__(), __hash__(), __reduce__(), __reduce_ex__(), __setattr__(),
__str__()
```

**4.6.2 Properties**

Name	Description
__editing__	
__unfin_op__	
autostart_editing	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

**4.7 Class Event**

**Known Subclasses:** prlsdk.Question

Event object class

## 4.7.1 Methods

<code>__getitem__(x, y)</code>
<code>x[y]</code>

<code>__init__(...)</code>
<code>x.__init__(...)</code> initializes x; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

<code>__iter__(x)</code>
<code>iter(x)</code>

<code>__len__(x)</code>
<code>len(x)</code>

<code>__new__(T, S, ...)</code>
<b>Return Value</b> a new object with type S, a subtype of T
Overrides: <code>object.__new__</code>

<code>get_param(Event, index)</code>
Returns depending on parameter type some value with passed index.
<b>Return Value</b> value of any type

<code>get_params_count(Event, index)</code>
Returns parameters count.
<b>Return Value</b> int

*Inherited from `prlsdk.Base` (Section 4.3)*

`__cmp__()`, `__repr__()`

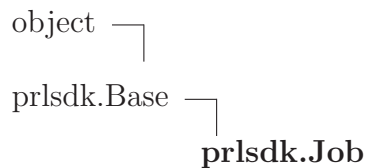
*Inherited from `object`*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`,  
`__str__()`

### 4.7.2 Properties

Name	Description
event_code	Returns event code, see prlsdk.consts.PMT_XXX.
event_str_code	Returns event code string representaion.
event_type	Returns event type, see prlsdk.consts.PMT_XXX.
explicit_text	Returns explicit event text.
params_count	Returns event param count
sender	Returns question sender, see prlsdk.consts.PIE_XXX.
text	Returns event text.
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__, handle_type	
<i>Inherited from object</i>	
__class__	

## 4.8 Class Job



Job objects

### 4.8.1 Methods

<b>__init__</b> (...) <hr/> x.__init__(...) initializes x; see x.__class__.__doc__ for signature Overrides: object.__init__
<b>__new__</b> (T, S, ...) <b>Return Value</b> a new object with type S, a subtype of T Overrides: object.__new__

<b>cancel</b> ( <i>Job</i> )
Start canceling this job, this method is asynchone so it return another job.
<b>Return Value</b> Result

<b>get_held_object</b> ( <i>Job, name</i> )
Returns held onject by name, None if not held.
<b>Return Value</b> object

<b>hold_object</b> ( <i>Job, name, obj</i> )
Add object to holding list.
<b>Return Value</b> self

<b>wait</b> ( <i>Job, msec=DefaultJobWait</i> )
Waits for job finished for passed milliseconds If timeout ->return False. If job finished with error ParallelsError will be raised.
<b>Return Value</b> bool

**Inherited from prlsdk.Base (Section 4.3)**

`--cmp--()`, `--repr--()`

**Inherited from object**

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

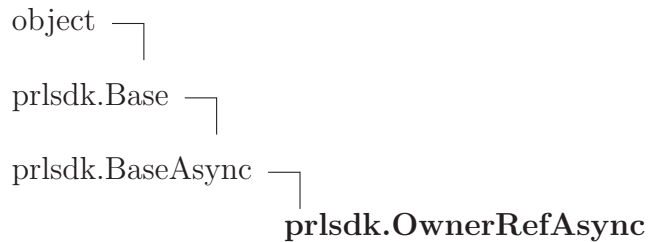
**4.8.2 Properties**

Name	Description
<code>--description--</code>	
<code>--ret_code--</code>	
<code>finished</code>	Shows job status
<code>op_code</code>	Return job operation code, see prlsdk.consts.PJOC_xxx.
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code> , <code>handle_type</code>	
<i>Inherited from object</i>	

*continued on next page*

Name	Description
__class__	

## 4.9 Class OwnerRefAsync



**Known Subclasses:** prlsdkapi.DispConfig, prlsdkapi.DispNet, prlsdkapi.VmBaseDev, prlsdkapi.VmBootDev, prlsdkapi.VmScrRes, prlsdkapi.VmShare

Base async object with owner reference

### 4.9.1 Methods

<b>__init__</b> (...)
x. <b>__init__</b> (...) initializes x; see x. <b>__class__</b> . <b>__doc__</b> for signature
Overrides: object. <b>__init__</b>

<b>__new__</b> (T, S, ...)
<b>Return Value</b> a new object with type S, a subtype of T
Overrides: object. <b>__new__</b>

*Inherited from prlsdk.BaseAsync(Section 4.4)*

**\_\_cmp\_\_**(), **\_\_repr\_\_**(), **cancel**(), **wait**()

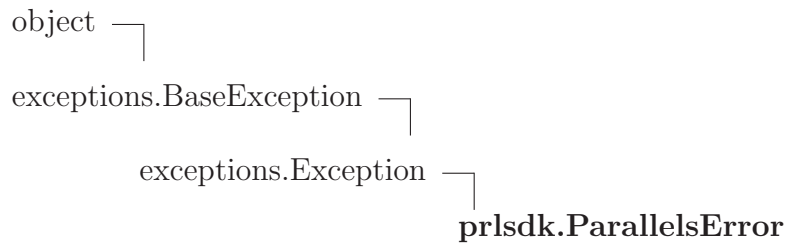
*Inherited from object*

**\_\_delattr\_\_**(), **\_\_getattr\_\_**(), **\_\_hash\_\_**(), **\_\_reduce\_\_**(), **\_\_reduce\_ex\_\_**(), **\_\_setattr\_\_**(), **\_\_str\_\_**()

### 4.9.2 Properties

Name	Description
<code>--owner--</code>	
	<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
	<i>Inherited from prlsdk.Base (Section 4.3)</i>
<code>--handle--</code>	
	<i>Inherited from object</i>
<code>--class--</code>	

#### 4.10 Class ParallelsError



**Known Subclasses:** prlsdk.PrlJobTimeoutError, prlsdk.PrlNotImplError, prlsdk.PrlSDKError, prlsdk.PrlStateError

##### 4.10.1 Methods

*Inherited from exceptions.Exception*

`--init--()`, `--new--()`

*Inherited from exceptions.BaseException*

`--delattr--()`, `--getattr--()`, `--getitem--()`, `--reduce--()`, `--repr--()`, `--setattr--()`, `--setstate--()`, `--str--()`

*Inherited from object*

`--hash--()`, `--reduce_ex--()`

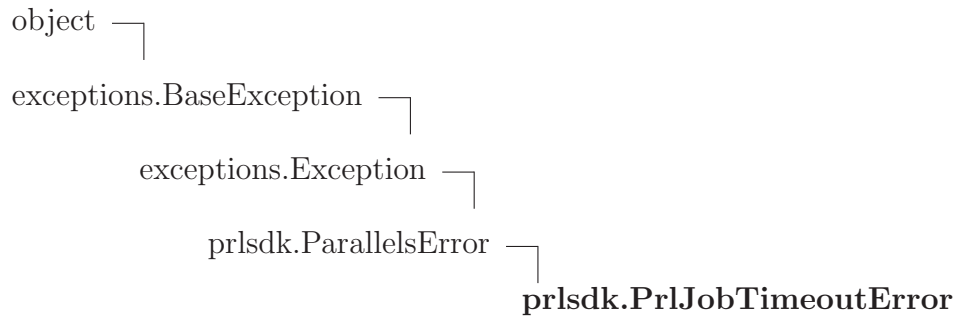
##### 4.10.2 Properties

Name	Description
	<i>Inherited from exceptions.BaseException</i>
<code>args</code> , <code>message</code>	
	<i>Inherited from object</i>

*continued on next page*

Name	Description
<code>--class--</code>	

#### 4.11 Class `PrlJobTimeoutError`



##### 4.11.1 Methods

###### *Inherited from exceptions.Exception*

`--init--()`, `--new--()`

###### *Inherited from exceptions.BaseException*

`--delattr--()`, `--getattr--()`, `--getitem--()`, `--reduce--()`, `--repr--()`, `--setattr--()`,  
`--setstate--()`, `--str--()`

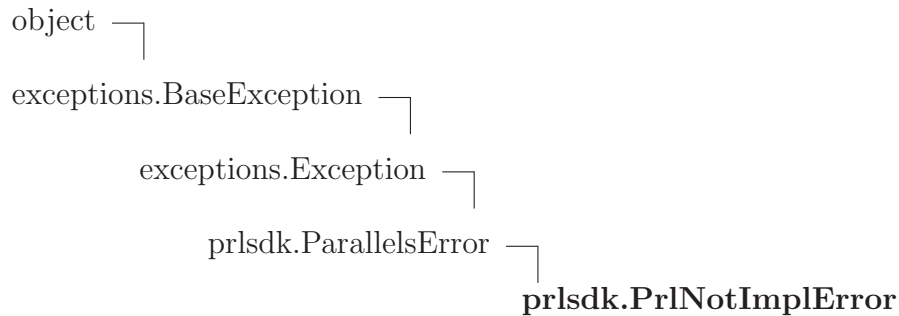
###### *Inherited from object*

`--hash--()`, `--reduce_ex--()`

##### 4.11.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	
args, message	
<i>Inherited from object</i>	
<code>--class--</code>	

## 4.12 Class *PrlNotImplError*



### 4.12.1 Methods

#### *Inherited from exceptions.Exception*

`__init__()`, `__new__()`

#### *Inherited from exceptions.BaseException*

`__delattr__()`, `__getattr__()`, `__getitem__()`, `__reduce__()`, `__repr__()`, `__setattr__()`,  
`__setstate__()`, `__str__()`

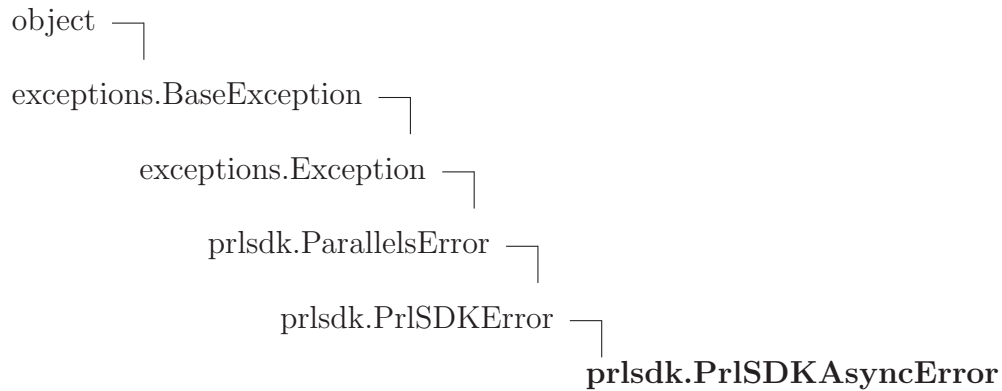
#### *Inherited from object*

`__hash__()`, `__reduce_ex__()`

### 4.12.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	
args, message	
<i>Inherited from object</i>	
<code>__class__</code>	

### 4.13 Class *PrlSDKAsyncError*



*SDKAsyncError* exception type

#### 4.13.1 Methods

**`__init__`**(...)

`x.__init__`(...) initializes `x`; see `x.__class__.__doc__` for signature

Overrides: `object.__init__`

**`__new__`**(*T*, *S*, ...)

**Return Value**

a new object with type *S*, a subtype of *T*

Overrides: `object.__new__`

*Inherited from exceptions.BaseException*

`__delattr__`(), `__getattr__`(), `__getitem__`(), `__reduce__`(), `__repr__`(), `__setattr__`(),  
`__setstate__`(), `__str__`()

*Inherited from object*

`__hash__`(), `__reduce_ex__`()

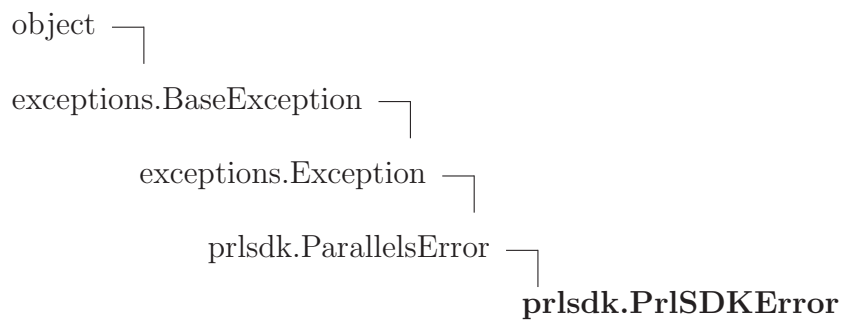
#### 4.13.2 Properties

Name	Description
<code>details</code>	Returns error details
<code>error_params</code>	
<code>job_op_code</code>	

*continued on next page*

Name	Description
<i>Inherited from prlsdk.PrlSDKError (Section 4.14)</i>	<code>--call_descr--</code> , <code>error_code</code>
<i>Inherited from exceptions.BaseException</i>	<code>args</code> , <code>message</code>
<i>Inherited from object</i>	<code>--class--</code>

#### 4.14 Class **PrlSDKError**



**Known Subclasses:** `prlsdk.PrlSDKAsyncError`

SDKError exception type

##### 4.14.1 Methods

<b><code>--init--(...)</code></b>
<code>x.--init--(...)</code> initializes x; see <code>x.--class--.--doc--</code> for signature
Overrides: <code>object.--init--</code>

<b><code>--new--(T, S, ...)</code></b>
<b>Return Value</b>
a new object with type S, a subtype of T
Overrides: <code>object.--new--</code>

*Inherited from exceptions.BaseException*

`--delattr--()`, `--getattr--()`, `--getitem--()`, `--reduce--()`, `--repr--()`, `--setattr--()`,  
`--setstate--()`, `--str--()`

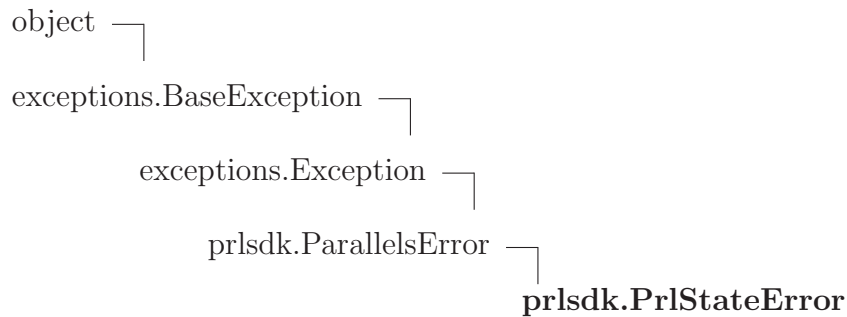
*Inherited from object*

`__hash__()`, `__reduce_ex__()`

#### 4.14.2 Properties

Name	Description
<code>__call_descr__</code>	
<code>details</code>	Returns error details
<code>error_code</code>	
<i>Inherited from exceptions.BaseException</i>	
args, message	
<i>Inherited from object</i>	
<code>__class__</code>	

#### 4.15 Class *PrlStateError*



##### 4.15.1 Methods

*Inherited from exceptions.Exception*

`__init__()`, `__new__()`

*Inherited from exceptions.BaseException*

`__delattr__()`, `__getattr__()`, `__getitem__()`, `__reduce__()`, `__repr__()`, `__setattr__()`,  
`__setstate__()`, `__str__()`

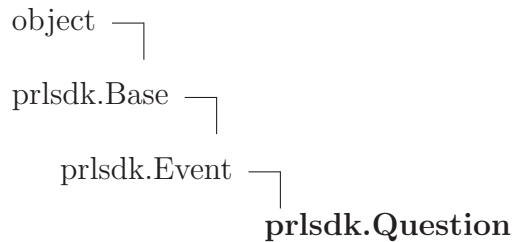
*Inherited from object*

`__hash__()`, `__reduce_ex__()`

##### 4.15.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	
args, message	
<i>Inherited from object</i>	
__class__	

## 4.16 Class Question



Question object class

### 4.16.1 Methods

<b>__init__</b> (...)
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

<b>__new__</b> (T, S, ...)
<b>Return Value</b> a new object with type S, a subtype of T
Overrides: object.__new__

<b>send_answer</b> (Question, answer_code)
Send answer.
<b>Return Value</b> Job

*Inherited from prlsdk.Event(Section 4.7)*

\_\_getitem\_\_(), \_\_iter\_\_(), \_\_len\_\_(), get\_param(), get\_params\_count()

*Inherited from prlsdk.Base(Section 4.3)*

`--cmp--()`, `--repr--()`

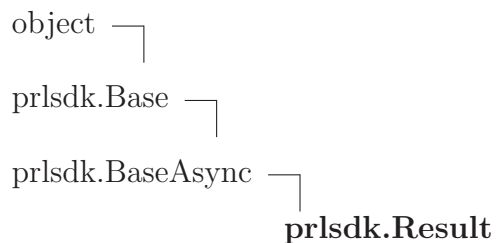
### *Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`,  
`--str--()`

#### 4.16.2 Properties

Name	Description
answers	Returns answer variants tuple ((description, value, name), ).
<i>Inherited from prlsdk.Event (Section 4.7)</i>	
event_code, event_str_code, event_type, explicit_text, params_count, sender, text	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code> , <code>handle_type</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

## 4.17 Class Result



Result object class

### 4.17.1 Methods

<code>--getitem--(x, y)</code>
<code>x[y]</code>
<code>--init--(...)</code>
<code>x.__init__(...)</code> initializes x; see <code>x.__class__.__doc__</code> for signature
Overrides: <code>object.__init__</code>

<code>--iter--(<i>x</i>)</code>
<code>iter(<i>x</i>)</code>

<code>--len--(<i>x</i>)</code>
<code>len(<i>x</i>)</code>

<code>--new--(<i>T</i>, <i>S</i>, ...)</code>
<b>Return Value</b> a new object with type <i>S</i> , a subtype of <i>T</i>
Overrides: <code>object.--new--</code>

<code>get_param(<i>Result</i>, <i>index</i>)</code>
Returns depending on parameter type some SDK object or string with passed <i>index</i> .
<b>Return Value</b> some sdk object/string

<code>get_param_as_string(<i>Result</i>, <i>index</i>)</code>
Returns as string representation of parameter with passed <i>index</i> .
<b>Return Value</b> string

<code>get_params_count(<i>Result</i>, <i>index</i>)</code>
Returns parameters count.
<b>Return Value</b> int

***Inherited from prlsdk.BaseAsync(Section 4.4)***

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

***Inherited from object***

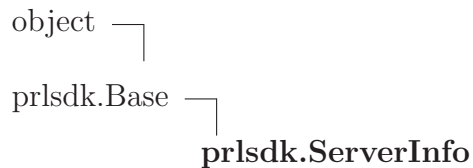
`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

**4.17.2 Properties**

*continued on next page*

Name	Description
params_count	Returns count of resulted params
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

## 4.18 Class *ServerInfo*



*ServerInfo* object class

### 4.18.1 Methods

<b>__init__</b> (...)
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__

<b>__new__</b> (T, S, ...)
<b>Return Value</b>
a new object with type S, a subtype of T
Overrides: object.__new__

<b>__str__</b> (x)
str(x)
Overrides: object.__str__

*Inherited from prlsdk.Base (Section 4.3)*

\_\_cmp\_\_(), \_\_repr\_\_()

***Inherited from object***

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`

**4.18.2 Properties**

Name	Description
host	host name
os_version	os version
port	port
uuid	server uuid
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code> , <code>handle_type</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

## 5 Package prlsdkapi

### 5.1 Functions

`GetClientConfig()`

`--prl_sdk_register_objects--(obj_type_list)`

`check_platform_and_version()`

`get_class_methods_doc(class_obj)`

`get_handle_type(obj)`

`get_os_type_from_version(os_version)`

Determine and return OS type based on the specified OS version.

**Parameters**

`os_version`: The version of the OS to get the type for. See `consts.PVS_GUEST_VER.xxx` for the list of OS versions.

**Return Value**

A Result object containing the OS type as a string. See `consts.PVS_GUEST_TYPE.xxx` for the list of OS types.

`prl_datetime_property(prop_name, prefix=None, doc=None)`

`prl_indirect_property(prop_name, prefix=None, doc=None)`

`prl_iter_property(count_methname, index_methodname, doc=None, arg1=None)`

`prl_property(prop_name, prefix=None, doc=None)`

`prl_sdk_call(meth_name, object, *args)`

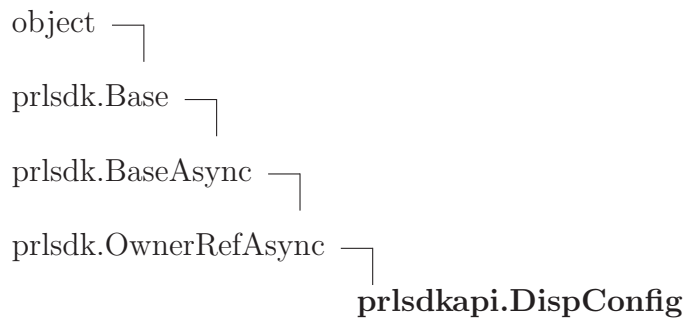
`prl_sdk_call_prefix(prefix, meth_name, object, *args)`

`prl_to_date_time(dt_string)`

## 5.2 Variables

Name	Description
DATE_TIME_SELECTOR	<b>Value:</b> re.compile(r'([0-9]{4}-[0-9]{2}-[0-9]{2})?([T])?([0-9]{2}...
PRL_DATETIME_FMTS	<b>Value:</b> ('%Y-%m-%d', '%H:%M:%S', '%Y-%m-%d %H:%M:%S', '%Y-%m-%dT%...
__deinit_sdk__	<b>Value:</b> <prlsdkapi.deinit_sdk object at 0x00D35530>

## 5.3 Class DispConfig



The DispConfig class provides methods for managing Parallels Server preferences.

### 5.3.1 Methods

#### **commit**(*self*)

Commit the changes to the server.

Before making any changes to Parallels server preferences, call `Server.begin_edit_common_prefs` to timestamp the beginning of the operation. When you are done making the changes, call this method to commit the changes to the server. This is an asynchronous method.

#### **Return Value**

A Job object.

**create\_net\_adapter**(*self*)

Create and return a new virtual network adapter object and add it to the network adapter list.

**Return Value**

A new `DispNet` instance.

**get\_net\_adapter**(*self*, *index*)

Return information about the specified virtual network adapter.

**Parameters**

**index**: The index specifying the net adapter in the adapter list. To obtain the total number of the available adapters, examine the `DispConfig.net_adapters_count` property.

**Return Value**

A `DispNet` object containing the specified virtual network adapter information.

**remove\_net\_adapter**(*self*, *net\_adapter*)

Remove the specified virtual network adapter object from the adapter list.

**Parameters**

**net\_adapter**: Identifies the network adapter. This could be a `DispNet` object or the adapter's index in the list. This is an asynchronous method.

**Return Value**

A `Job` object.

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

`__init__()`, `__new__()`

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`__cmp__()`, `__repr__()`, `cancel()`, `wait()`

*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__str__()`

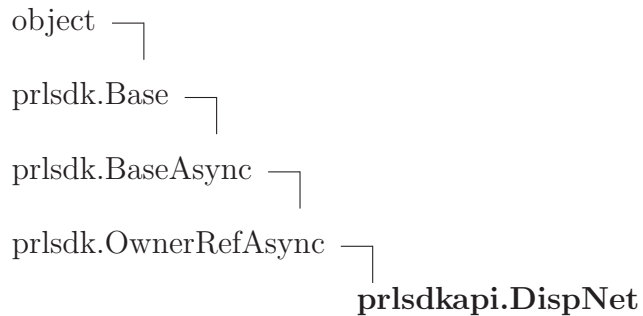
### 5.3.2 Properties

Name	Description
auto_mem_adjust	Specifies whether the size of the memory allocated to the Parallels server will be set automatically or manually.
can_change_default_settings	Indicates whether new users are by default allowed to modify the Parallels server preferences. True - can modify; False - cannot modify.
default_vm_dir	Name and path of the default virtual machine folder.
default_vnc_host_name	The default VNC host name of this Parallels server.
min_security_level	Indicates the lowest allowable security level that can be used to connect to this Parallels server. See <code>consts.PSL_xxx</code> for available security levels.
net_adapters	An iterator/sequence that allows to traverse through network adapter collection. Each network adapter is an instance of <code>DispNet</code> .
net_adapters_count	The total number of the available network adapters.
reserved_mem_limit	Specifies the memory size reserved for the Parallels server operations
vnc_base_port	The base VNC port number.
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
__owner__	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.3.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435478
__prefix__	<b>Value:</b> 'dispcfg'

## 5.4 Class DispNet



Contains information about a virtual network adapter.

### 5.4.1 Methods

**remove(*self*)**

Remove this object from memory.

This method does not delete the adapter from the Parallels server configuration. Use this method if you would like to cancel the changes that you've made to the object properties so far. The method must be called before calling `DispConfig.commit`. To delete an existing adapter from the configuration, use the `DispConfig.remove_net_adapter` method.

*Inherited from prlsdk.OwnerRefAsync (Section 4.9)*

`--init--()`, `--new--()`

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

### 5.4.2 Properties

Name	Description
<code>dhcp_enabled</code>	Indicates whether DHCP scope is enabled for the specified virtual network adapter. True - enabled; False - disabled.

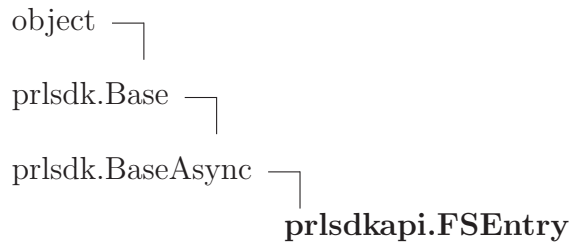
*continued on next page*

Name	Description
dhcp_scope_end	The end IP address of DHCP scope for the adapter.
dhcp_scope_mask	The subnet mask of DHCP scope for the adapter.
dhcp_scope_start	The start IP address of DHCP scope for the adapter.
enabled	Indicates whether the adapter is enabled. True - enabled; False - disabled.
index	The index identifying the adapter on the server.
name	The adapter name.
network_type	Indicates the virtual network adapter network type. See <code>consts.PNA_xxx</code> for the available network types.
sys_name	The name of the physical network adapter to which the specified virtual network adapter is bound.
uuid	The adapter UUID.
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
__owner__	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.4.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435481
__prefix__	<b>Value:</b> 'dispnet'

## 5.5 Class FSEntry



Contains information about a file system entry. The instance of this class is obtained from the FSInfo object and can contain a parent file system entry information or the information about any of its child entries.

### 5.5.1 Methods

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

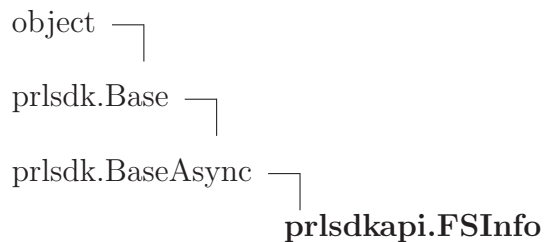
### 5.5.2 Properties

Name	Description
<code>abs_path</code>	The file system entry absolute path.
<code>last_modified_date</code>	The date on which the file system entry was last modified.
<code>permissions</code>	The file system entry permissions for the current user: 'r' - read; 'w' - write; 'x' - execute.
<code>relative_name</code>	The file system entry relative name.
<code>size</code>	The file system entry size.
<code>type</code>	The file system entry type (file, directory, drive). See <code>consts.PSE_xxx</code> for the available types.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.5.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435460
<code>--prefix--</code>	<b>Value:</b> 'fs_entry'

## 5.6 Class FSInfo



Contains information about a file system entry and its immediate child elements (files and directories).

### 5.6.1 Methods

<b>get_entry</b> ( <i>self</i> , <i>index</i> )
Return an <code>FSEntry</code> object containing information about a child file system element specified by <code>index</code> .
<b>Parameters</b>
<i>index</i> : The index of the child entry in the container (beginning with 0). The total number of child entries can be determined by examining the <code>FSInfo.entries_count</code> property.
<b>Return Value</b>
An <code>FSEntry</code> object containing the child entry information.

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattribute--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

### 5.6.2 Properties

Name	Description
entries	An iterator/sequence that allows to traverse through child files system entries. Each child entry is an instance of <code>FSEntry</code> .
entries_count	The number of child entries.
parent_entry	An instance of <code>FSEntry</code> containing information about this file system entry.
type	<code>SDK.fs_info__get_type(FSInfo) -&gt; int</code> Returns remote file system basic type: Windows or Unix like FS.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i> <code>__expected_h_type__</code> , <code>__job__</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i> <code>__handle__</code>	
<i>Inherited from object</i> <code>__class__</code>	

### 5.6.3 Class Variables

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435459
<code>__prefix__</code>	<b>Value:</b> 'fs_info'

## 5.7 Class FoundVm



Contains summary information about a virtual machine.

### 5.7.1 Methods

*Inherited from prlsdk.Base(Section 4.3)*

`__cmp__()`, `__init__()`, `__new__()`, `__repr__()`

*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`,

`__str__()`

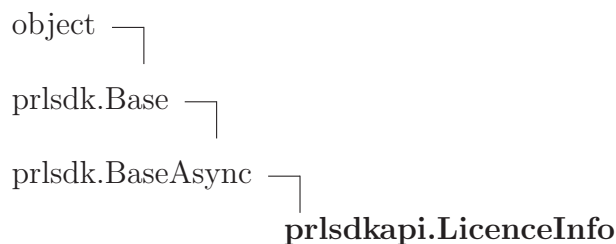
### 5.7.2 Properties

Name	Description
<code>config_path</code>	Name and path of the virtual machine directory.
<code>name</code>	Virtual machine name.
<code>old_config</code>	Indicates whether the virtual machine was created with an older version of Parallels Server: True - older version; False - current version.
<code>os_version</code>	Guest OS version.
<code>template</code>	Indicates whether the virtual machine is a template: True - it is a template; False - it is a regular virtual machine.
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code> , <code>handle_type</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

### 5.7.3 Class Variables

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435506
<code>__prefix__</code>	<b>Value:</b> 'foundvm'

## 5.8 Class LicenceInfo



Contains the Parallels Server license information.

### 5.8.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`__cmp__()`, `__init__()`, `__new__()`, `__repr__()`, `cancel()`, `wait()`

### *Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__str__()`

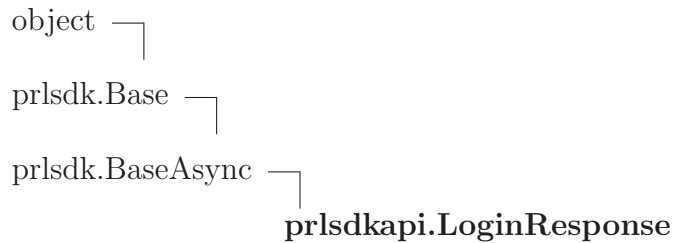
### 5.8.2 Properties

Name	Description
<code>company_name</code>	A string containing the company name on the license.
<code>distributor</code>	License distributor ID.
<code>expire_date</code>	License expiration date.
<code>flags</code>	License flags.
<code>id</code>	License ID.
<code>language</code>	License language ID.
<code>license_key</code>	A string containing the license key.
<code>platform</code>	License platform ID.
<code>product</code>	License product ID.
<code>trial</code>	Indicates whether this is a trial license: True - trial license; False - commercial license.
<code>user_name</code>	A string containing the user name on the license.
<code>valid</code>	Indicates whether the license is valid: True - valid; False - invalid.
<code>version</code>	License version ID.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>__expected_h_type__</code> , <code>__job__</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

### 5.8.3 Class Variables

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435495
<code>__prefix__</code>	<b>Value:</b> 'licinfo'

## 5.9 Class LoginResponse



Contains information pertinent to a successful Parallels server login operation.

### 5.9.1 Methods

<b>get_run_task</b> ( <i>self</i> , <i>index</i> )
Obtain and return a <code>RunningTask</code> instance.
<b>Parameters</b>
<i>index</i> : A numeric index of the task to obtain the handle for. Use <code>LoginResponse</code> to obtain the task count.
<b>Return Value</b>
An instance of <code>RunningTask</code> .

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`__cmp__()`, `__init__()`, `__new__()`, `__repr__()`, `cancel()`, `wait()`

*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__str__()`

### 5.9.2 Properties

Name	Description
<code>host_os_version</code>	OS version that the host server is running.
<code>product_version</code>	The Parallels Server version that the host server is running.
<code>run_tasks</code>	An iterator/sequence that allows to traverse through <code>RunningTask</code> instances contained in this object.
<code>run_tasks_count</code>	The number of <code>RunningTask</code> instances contained in this object.
<code>server_uuid</code>	The host server UUID.

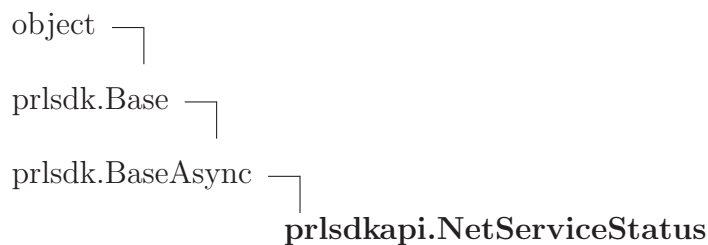
*continued on next page*

Name	Description
session.uuid	The session UUID string (used to restore a session).
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.9.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435499
__prefix__	<b>Value:</b> 'loginresponse'

## 5.10 Class NetServiceStatus



Contains the Parallels server network status information.

### 5.10.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_init\_\_(), \_\_new\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

### 5.10.2 Properties

Name	Description
status	Network service status. See <code>consts.PSS_xxx</code> for status list.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.10.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435501
__prefix__	<b>Value:</b> 'netsvc'

## 5.11 Class PrlIterator



**Known Subclasses:** prlsdkapi.PrlResultIter

### 5.11.1 Methods

<code>__getitem__(self, index)</code>
<code>__init__(self, count_getter, item_getter, *args)</code> x.__init__(...) initializes x; see x.__class__.__doc__ for signature Overrides: object.__init__ extit(inherited documentation)
<code>__iter__(self)</code>
<code>__len__(self)</code>
<code>get_count(self)</code>

<code>next(self)</code>
-------------------------

**Inherited from object**

`__delattr__()`, `__getattr__()`, `__hash__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`,  
`__repr__()`, `__setattr__()`, `__str__()`

**5.11.2 Properties**

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

**5.12 Class PrObject**

```

object └─
         prlsdkapi.PrObject

```

**Known Subclasses:** `prlsdkapi.VmAccessRights`, `prlsdkapi.VmObjEmulation`

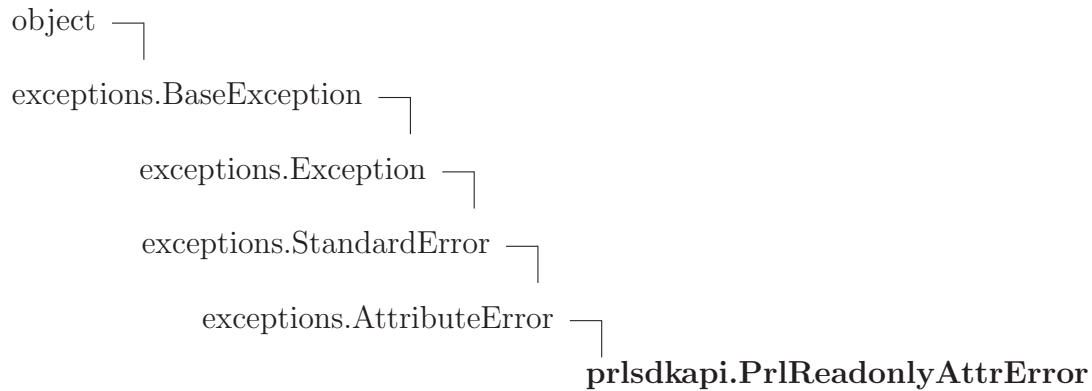
**5.12.1 Methods****Inherited from object**

`__delattr__()`, `__getattr__()`, `__hash__()`, `__init__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`,  
`__repr__()`, `__setattr__()`, `__str__()`

**5.12.2 Properties**

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

### 5.13 Class PriReadOnlyAttrError



#### 5.13.1 Methods

```

__init__(self, obj, prop_name)
x.__init__(...) initializes x; see x.__class__.__doc__ for signature
Overrides: object.__init__ extit(inherited documentation)
  
```

#### *Inherited from exceptions.AttributeError*

```
__new__()
```

#### *Inherited from exceptions.BaseException*

```
__delattr__(), __getattr__(), __getitem__(), __reduce__(), __repr__(), __setattr__(),
__setstate__(), __str__()
```

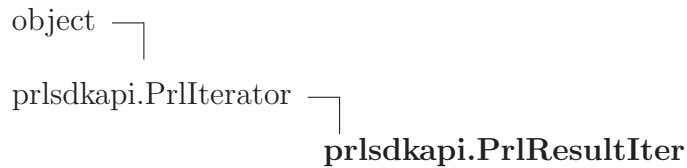
#### *Inherited from object*

```
__hash__(), __reduce_ex__()
```

#### 5.13.2 Properties

Name	Description
<i>Inherited from exceptions.BaseException</i>	args, message
<i>Inherited from object</i>	__class__

## 5.14 Class PrlResultIter



### 5.14.1 Methods

`__init__(self, result_obj)`  
`x.__init__(...)` initializes `x`; see `x.__class__.__doc__` for signature  
 Overrides: `object.__init__` `exitit` (inherited documentation)

*Inherited from prlsdkapi.PrlIterator (Section 5.11)*

`__getitem__()`, `__iter__()`, `__len__()`, `get_count()`, `next()`

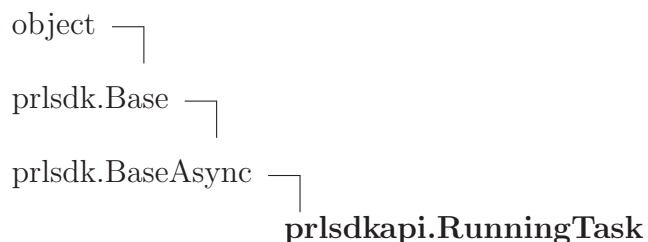
*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`,  
`__repr__()`, `__setattr__()`, `__str__()`

### 5.14.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

## 5.15 Class RunningTask



The `RunningTask` class is used in recovering from the lost server connection and attaching to an existing task currently running on the server. The class is a container that contains

information about a task that is still running on the server. For the list of tasks supported by this functionality, see `consts.RTT_RUNNING_TASK_XXX`.

### 5.15.1 Methods

#### *Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

#### *Inherited from object*

`--delattr--()`, `--getattribute--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

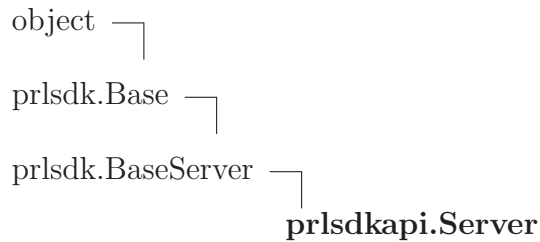
### 5.15.2 Properties

Name	Description
parameters	<code>runningtask.get_parameters(RunningTask) -&gt; string</code> Returns task parameters as string Value returns in UTF-8 encoding
type	The task type. See <code>consts.RTT_RUNNING_TASK_XXX</code> .
uuid	The the task UUID.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i> <code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i> <code>--handle--</code>	
<i>Inherited from object</i> <code>--class--</code>	

### 5.15.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435500
<code>--prefix--</code>	<b>Value:</b> 'runningtask'

## 5.16 Class Server



The Server class provides access to a Parallels server. The class is required to use most of the functionality within the Parallels Server API.

### 5.16.1 Methods

#### **add\_network\_adapter**(*self*, *net\_adapter*)

Add a new virtual network adapter to the specified Parallels server. This is an asynchronous method.

##### **Parameters**

**net\_adapter**: A DispNet object containing the new adapter information.

##### **Return Value**

A Job object.

#### **attach\_to\_lost\_task**(*self*, *task\_uuid*)

The method allows to obtain a Job object attached to a running task after loosing a connection with the server and reconnecting to it again. This is an asynchronous method.

##### **Parameters**

**task\_uuid**: The global task ID. The list of task IDs can be obtained from the LoginResponse object returned by the login operation.

##### **Return Value**

A Job object attached to the specified task.

#### **begin\_edit\_common\_prefs**(*self*)

This method must be called before making any changes to Parallels server common preferences (see DispConfig). When you are done making the changes, call DispConfig.commit to commit the changes to the server.

**begin\_edit\_user\_profile**(*self*)

This method must be called before making any changes to a Parallels server user profile. When you are done making the changes, call `UserConfig.commit` to commit the changes to the server.

**create\_vm**(*self*)

Create and return a new `Vm` instance.

**Return Value**

A new instance of `Vm`.

**delete\_network\_adapter**(*self*, *net\_adapter*)

Delete the specified virtual network adapter from the Parallels server configuration. This is an asynchronous method.

**Parameters**

`net_adapter`: A `DispNet` object identifying the adapter to delete.

**Return Value**

A `Job` object.

**fs\_can\_create\_file**(*self*, *file\_path*)

Determine if the current user has sufficient rights to create a file or a directory in the specified location on the host server. This is an asynchronous method.

**Parameters**

`file_path`: Name and path of the file to remove.

**Return Value**

An instance of the `Job` class containing the result. Success indicates that the user can create the specified file.

**fs\_create\_dir**(*self*, *path*)

Create a new directory on the host server. This is an asynchronous method.

**Parameters**

`path`: The new directory name and path.

**Return Value**

A `Job` object.

**fs\_remove\_dir**(*self*, *path*)

Remove the specified directory from the host server. This is an asynchronous method.

**Parameters**

**path**: Name and path of the directory to remove.

**Return Value**

A Job object.

**fs\_rename\_dir**(*self*, *path*, *new\_path*)

Rename the specified directory. This is an asynchronous method.

**Parameters**

**path**: Name and path of the directory to rename.

**new\_path**: The new directory name and path.

**Return Value**

A Job object.

**get\_dir\_entries**(*self*, *path*)

Obtain and return the information about the specified file system entry on the host server. This is an asynchronous method.

**Parameters**

**path**: The full path to the virtual machine directory.

**Return Value**

An instance of FSInfo containing the directory information.

```
get_unique_fs_entry_name(self, path, fn_prefix='', fn_suffix='',  
index_delim='')
```

Automatically generates a unique name for a new directory. This is an asynchronous method.

**Parameters**

- path:** A full path to the target location (directory).
- fn\_prefix:** Specifies a prefix to use in the directory name. Pass an empty string to use the default prefix 'tmpfile'. The name will be generated by adding a number to the prefix (e.g. tmpfile1, tmpfile2, etc.)
- fn\_suffix:** Specifies a suffix to use in the directory name.
- index\_delim:** A character(s) that will be inserted between the prefix and the index (e.g. tmpfile\_1, tmpfile\_2, etc.).

**Return Value**

A `Result` object containing the name.

```
get_user_info(self, user_id)
```

Return a `UsrInfo` class containing the information about the specified user.

**Parameters**

- user\_id:** The global ID of the user to get the information for.

**Return Value**

A `UsrInfo` object containing the user information.

```
get_vm_list(self)
```

Obtain and return a list of virtual machines from the server.

**Return Value**

A `Result` object containing a list of `Vm` instances. Individual `Vm` objects can be extracted using the `Result.get_param` method.

---

**net\_service**(*self*, *operation*='u')

Allows to start, stop, or restart the Parallels server network service. This is an asynchronous method.

**Parameters**

**operation:** Specifies the operation to perform. Possible values are:  
    'u' - start network service.  
    'd' - stop network service.  
    'r' - restart network service.

**Return Value**

A Job object.

---

**register\_vm**(*self*, *vm\_path*, *non\_interactive\_mode*=False)

Register an existing virtual machine with the Parallels server. This is an asynchronous method.

**Parameters**

**vm\_path:** Name and path of the virtual machine directory.  
**non\_interactive\_mode:** Set to **True** to use interactive mode. Set to **False** to use non-interactive mode.

**Return Value**

An instance of the **Vm** class containing information about the virtual machine that was registered.

---

**search\_vms**(*self*, *vm\_list*=None)

Search for unregistered virtual machines at the specified location(s) on the host server and return their information. This is an asynchronous method.

**Parameters**

**vm\_list:** A list of pathnames to search. Default value searches all available drives and directories.

**Return Value**

A **Result** object containing a list of **FoundVm** objects, each containing information about an individual virtual machine.

**send\_answer**(*self*, *answer\_event*)

Send an answer to the server in response to a question. This is an asynchronous method.

**Parameters**

*answer\_event*: An **Event** object containing the answer.

**Return Value**

A **Job** object.

**subscribe\_to\_host\_statistics**(*self*)

Subscribe to receive host server statistics on a periodic basis. This is an asynchronous method.

**Return Value**

A **Job** object.

**unsubscribe\_to\_host\_statistics**(*self*)

Cancel the server statistics subscription that was started using the **subscribe\_to\_host\_statistics** method. This is an asynchronous method.

**Return Value**

A **Job** object.

**update\_licence**(*self*, *key*, *user*, *company*)

Update the Parallels server license. This is an asynchronous method.

**Parameters**

*key*: A string containing the license key.

*user*: License user name.

*company*: License company name.

**Return Value**

A **Job** object.

**update\_network\_adapter**(*self*, *net\_adapter*)

Update the specified virtual network adapter on the Parallels server. This is an asynchronous method.

**Parameters**

*net\_adapter*: A **DispNet** object containing the updated adapter information.

**Return Value**

A **Job** object.

***Inherited from prlsdk.BaseServer(Section 4.5)***

`__init__()`, `__new__()`, `login()`, `login_local()`, `logoff()`, `reg_event_handler()`, `shutdown()`, `unreg_event_handler()`

***Inherited from prlsdk.Base(Section 4.3)***

`__cmp__()`, `__repr__()`

***Inherited from object***

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__str__()`

**5.16.2 Properties**

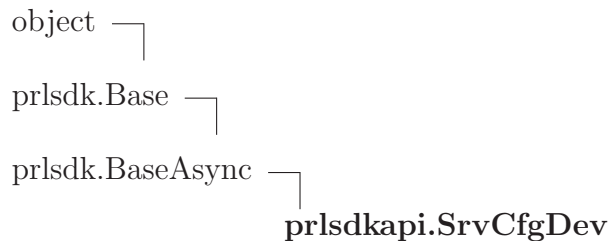
Name	Description
<code>common_prefs</code>	An instance of <code>DispConfig</code> .
<code>fs_disk_list</code>	An instance of <code>FSInfo</code> .
<code>is_local</code>	Indicates whether the current connection is local or remote: True - local connection; False - remote connection.
<code>licence_info</code>	An instance of <code>LicenceInfo</code> .
<code>net_service_status</code>	An instance of <code>NetServiceStatus</code> .
<code>problem_report</code>	A string containing a problem report in the event of a virtual machine operation failure.
<code>questions</code>	A tuple of instances of <code>Question</code> .
<code>server_config</code>	An instance of <code>SrvConfig</code> .
<code>server_info</code>	An instance of <code>SrvInfo</code> .
<code>statistics</code>	An instance of <code>VmStat</code> .
<code>user_profile</code>	An instance of <code>UserConfig</code> .
<code>users_info</code>	An instance of <code>UsrInfo</code> .
<code>vm_list</code>	Instance of <code>Result</code> containing <code>Vm</code> instances
<i>Inherited from prlsdk.BaseServer (Section 4.5)</i>	
<code>__editing_common__</code> , <code>__editing_user__</code> , <code>__login_result__</code> , <code>autostart_editing</code> , <code>connected</code> , <code>last_session_uuid</code> , <code>session_uuid</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code> , <code>handle_type</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

**5.16.3 Class Variables**

*continued on next page*

Name	Description
__ht__	Value: 268435457
__prefix__	Value: 'srv'

## 5.17 Class SrvCfgDev



**Known Subclasses:** prlsdkapi.SrvCfgHdd, prlsdkapi.SrvCfgNet

SrvCfgDev is a base class that provides methods for getting information about physical devices on the host server.

### 5.17.1 Methods

*Inherited from prlsdk.BaseAsync (Section 4.4)*

\_\_cmp\_\_(), \_\_init\_\_(), \_\_new\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

### 5.17.2 Properties

Name	Description
id	The device ID.
name	The name of the device.
type	The device type. See <code>consts.PDE_xxx</code> .
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	

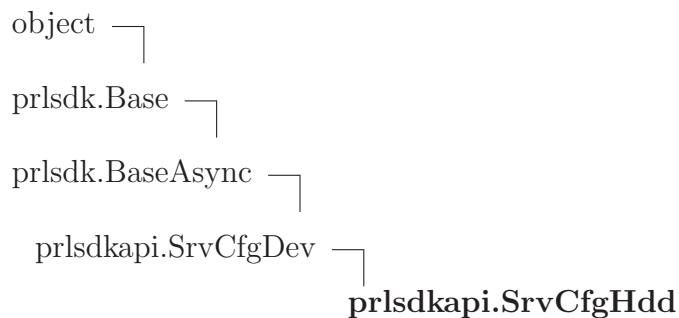
*continued on next page*

Name	Description
<i>Inherited from object</i>	
__class__	

### 5.17.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435482
__prefix__	<b>Value:</b> 'srvcfgdev'

## 5.18 Class SrvCfgHdd



Provides methods for getting information about a physical hard disk on the host server.

### 5.18.1 Methods

<b>get_part</b> ( <i>self</i> , <i>index</i> )
Obtain and return the specified partition information.
<b>Parameters</b>
<i>index</i> : A numeric index identifying the partition. To obtain the partition count, use <code>SrvCfgHdd.parts_count</code> .
<b>Return Value</b>
An instance of <code>SrvCfgHddPart</code> .

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`__cmp__()`, `__init__()`, `__new__()`, `__repr__()`, `cancel()`, `wait()`

*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`,

`__str__()`

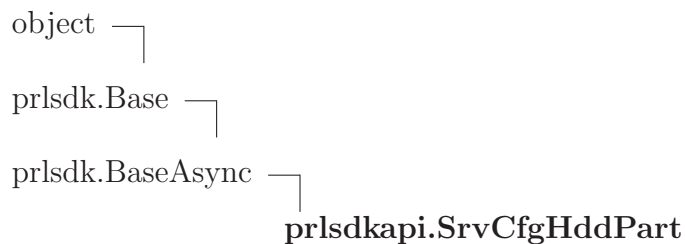
### 5.18.2 Properties

Name	Description
<code>index</code>	A numeric index identifying the disk.
<code>parts</code>	An iterator/sequence that allows to traverse through <code>SrvCfgHddPart</code> instances contained in this object.
<code>parts_count</code>	The number of partitions on the disk.
<code>size</code>	The hard disk size.
<i>Inherited from prlsdkapi.SrvCfgDev (Section 5.17)</i>	
<code>id</code> , <code>name</code> , <code>type</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>__expected_h_type__</code> , <code>__job__</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

### 5.18.3 Class Variables

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435483
<code>__prefix__</code>	<b>Value:</b> 'srvcfghdd'

## 5.19 Class SrvCfgHddPart



Contains information about a disk partition.

### 5.19.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

### *Inherited from object*

`--delattr--()`, `--getattr__()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`,  
`--str--()`

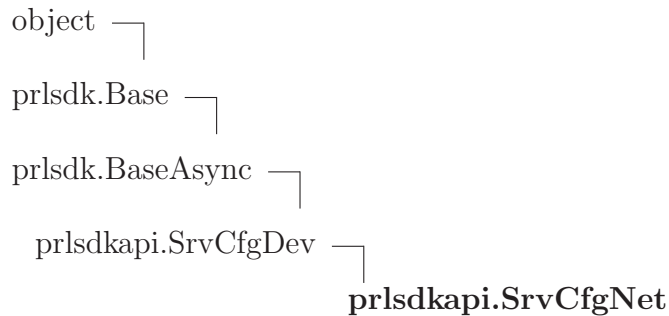
#### 5.19.2 Properties

Name	Description
active	Indicates whether the partition is active. True - active; False otherwise.
in_use	Indicates whether the partition is in use, that is, whether it contains a valid file system, being used for swap or by a database, etc. True - the partition is in use; False otherwise.
index	A numeric index identifying the partition.
logical	Indicates whether this is a logical partition. True - logical; False otherwise.
name	Partition name.
size	Partition size.
type	A numerical code identifying the type of the partition.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

#### 5.19.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435484
<code>--prefix--</code>	<b>Value:</b> 'srvcfghddpart'

## 5.20 Class SrvCfgNet



Provides methods for getting information about a network adapter on the host server.

### 5.20.1 Methods

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr__()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

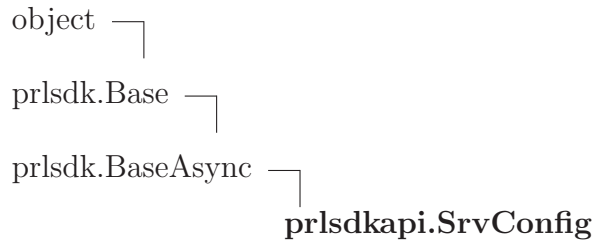
### 5.20.2 Properties

Name	Description
<code>enabled</code>	Indicates whether the adapter is enabled or disabled. True - enabled; False - disabled.
<code>net_adapter_type</code>	The adapter type. See <code>consts.PHI_xxx</code> .
<code>sys_index</code>	The network adapter system index.
<i>Inherited from prlsdkapi.SrvCfgDev (Section 5.17)</i>	
<code>id</code> , <code>name</code> , <code>type</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.20.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435485
<code>--prefix--</code>	<b>Value:</b> 'srvcfgnet'

## 5.21 Class SrvConfig



Contains the host server configuration information.

### 5.21.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

### 5.21.2 Properties

Name	Description
<code>cpu_count</code>	The number of CPUs on the host server.
<code>cpu_mode</code>	The CPU mode (32 bit or 64 bit).
<code>cpu_model</code>	The model of CPU on the host server.
<code>cpu_speed</code>	The host server CPU speed, in megahertz.
<code>os_version</code>	The host server OS version information. This is a dictionary with the following elements: '<code>type</code>': string - OS type. '<code>major</code>': int - Major version number. '<code>minor</code>': int - Minor version number. '<code>sub_minor</code>': int - Sub-minor version number. '<code>full</code>': string - The full OS information as a single string.

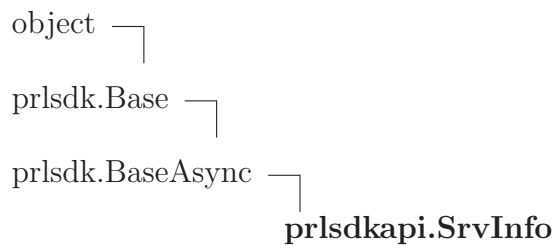
*continued on next page*

Name	Description
ram_size	The amount of memory (RAM) available on the host server, in megabytes.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.21.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435458
__prefix__	<b>Value:</b> 'srvcfg'

## 5.22 Class SrvInfo



Contains information about a host server. The information includes port number, host name, OS version, product version, and server UUID.

### 5.22.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_init\_\_(), \_\_new\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

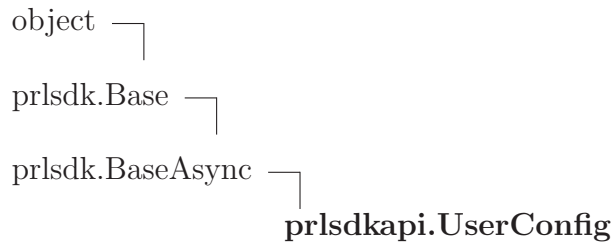
### 5.22.2 Properties

Name	Description
cmd_port	CMD port number.
host_name	Host name.
os_version	OS version that the host is running.
product_version	The Parallels Server version that the host server is running.
server_uuid	Server UUID (universally unique ID).
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.22.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435496
__prefix__	<b>Value:</b> 'srvinfo'

## 5.23 Class UserConfig



Contains information about a Parallels server user.

### 5.23.1 Methods

<b>commit</b> ( <i>self</i> )
<hr/> Commit user profile changes to the server. Every user profile modification must begin with the <code>Server.begin_edit_user_profile</code> call and end with this call.

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

### *Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`,  
`--str--()`

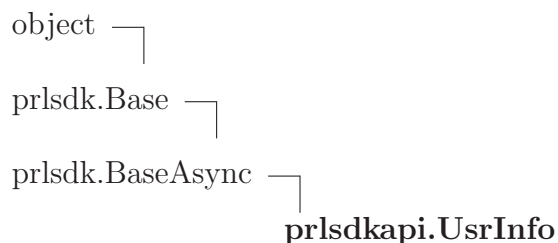
### 5.23.2 Properties

Name	Description
<code>can_change_srvcfg</code>	Indicates whether the user is allowed to modify server preferences.
<code>can_use_mng_console</code>	<code>SDK.usrcfg.get_can_use_mng_console(UserConfig)</code> -> bool Returns sign whether user can using server management console utility.
<code>default_folder</code>	Name and path of the default virtual machine directory for the user.
<code>dir_uuid</code>	The Universally Unique Identifier (UUID) of the user.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i> <code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i> <code>--handle--</code>	
<i>Inherited from object</i> <code>--class--</code>	

### 5.23.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435477
<code>--prefix--</code>	<b>Value:</b> 'usrcfg'

## 5.24 Class UsrInfo



Contains information about a Parallels server user.

### 5.24.1 Methods

#### *Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

#### *Inherited from object*

`--delattr--()`, `--getattribute--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

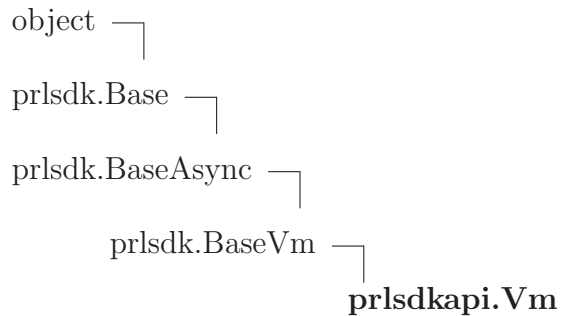
### 5.24.2 Properties

Name	Description
<code>can_change_srv_sets</code>	Indicates whether the user is allowed to modify Parallels server preferences. True - allowed; False - not allowed.
<code>default_vm_folder</code>	Name and path of the default virtual machine directory for the user.
<code>name</code>	User name.
<code>session_count</code>	Active session count.
<code>uuid</code>	User UUID.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.24.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435508
<code>--prefix--</code>	<b>Value:</b> 'usrinfo'

## 5.25 Class Vm



The Vm class provides methods for managing virtual machines. Whenever you want to get the information about, modify, or create a virtual machine, you have to obtain an instance of this class. The instance can be obtained using methods of various classes. The two main methods that return an instance of Vm are `Server.create_vm` and the `Server.get_vm_list`.

### 5.25.1 Methods

**`__init__(self, *args)`**

`x.__init__(...)` initializes x; see `x.__class__.__doc__` for signature

Overrides: `object.__init__` `exitit` (inherited documentation)

**`add_default_device(self, srv_config, type)`**

Adds a virtual device to the virtual machine. The method provides an easy way of setting devices in a virtual machine by automatically setting all the necessary device properties.

#### Parameters

**`srv_config`:** An instance of `SrvConfig` containing the host server configuration information.

**`type`:** The type of the device to add. See `{consts.PDE_xxx}` for available options.

**begin\_edit**(*self*)

Timestamp the beginning of the virtual machine editing operation. This method must be called before making any modifications to the virtual machine configuration. When the desired configuration properties are set, call `Vm.commit` to commit the changes to the server. The two functions are used to detect collisions with other clients trying to modify the same virtual machine at the same time.

This is an asynchronous method.

**Return Value**

A Job object.

**change\_access\_rights**(*self*, *others\_right*=None)

Modify the virtual machine security settings. This is an asynchronous method.

**Parameters**

**others\_right**: Specifies access rights that users of this Parallels server will have to this virtual machine. The owner of the virtual machine has full access rights and is not affected by this functionality. See `consts.PAO_VM_xxx` for available options.

**Return Value**

A Job object.

**clone**(*self*, *new\_name*, *root\_path*, *template*=False)

Clone the virtual machine. If this is a template, create a regular virtual machine from it. This is an asynchronous method.

**Parameters**

**new\_name**: The name to use for the new virtual machine.

**root\_path**: Name and path of the directory where the new virtual machine should be created. To create the machine in a default directory, pass an empty string.

**template**: Specifies whether the clone should be created as a virtual machine or as a template. True - create a template; False - create a regular virtual machine. When creating a virtual machine from a template, set the parameter value to False.

**Return Value**

A Job object.

**commit**(*self*)

Commit the virtual machine configuration changes to the server. This is an asynchronous method.

**Return Value**

A Job object.

**create\_boot\_dev**(*self*)

Create a new instance of VmBootDev.

**Return Value**

A new instance of VmBootDev.

**create\_device**(*self*, *device\_type*)

Create and return a new virtual device object based on the specified device type. Each virtual machine device type has a corresponding class in the Parallels Server Python API. You can search this documentation for available virtual device classes using the following rule: a class name begins with 'Vm' followed by the device type description followed by 'Dev'. For example: VmHddDev, VmNetDev, VmSoundDev, etc.

**Parameters**

**device\_type**: Device type. Based on the type specified here, an instance of the corresponding class will be created. See `consts.PDE_xxx` for available options.

**Return Value**

An instance of the requested class.

**create\_event**(*self*)**create\_scr\_res**(*self*)**create\_share**(*self*)

Create a new instance of VmShare.

**Return Value**

A new instance of VmShare.

**create\_unattended\_floppy**(*self*, *guest\_distro\_type*, *username*, *company\_name*, *serial\_key*)

Create a floppy disk image in the virtual machine home directory for unattended Windows installation. This is an asynchronous method.

**Parameters**

*guest\_distro\_type*: Windows version.  
*username*: User name.  
*company\_name*: Company name.  
*serial\_key*: Serial number.

**Return Value**

A Job object.

**delete**(*self*, *dev\_list*=None)

Delete the virtual machine from the server. This is an asynchronous method.

**Parameters**

*dev\_list*: Files and directories to be deleted from the virtual machine directory. If none specified, delete the entire directory.

**Return Value**

A Job object.

**drop\_suspended\_state**(*self*)

Drop suspended state; delete all temporary files from the hard drive. This is an asynchronous method.

**Return Value**

A Job object.

**execution**(*self*, *operation*, *use\_acpi=False*)

Perform the specified operation on the virtual machine. This is an asynchronous method

**Parameters**

- operation:** 'u'(up); 'd'(down); 'p'(pause); 'r'(reset); 's'(suspend); 'e'(resume).
- use\_acpi:** Specifies if ACPI should be used. Used only with down and pause operations. True - use ACPI; False - don't use ACPI.

**Return Value**

A Job object.

**generate\_device\_filename**(*self*, *fn\_prefix=''*, *fn\_suffix=''*, *index\_delim=''*)

Generate a unique name for a virtual device file that will be unique within the home directory of the virtual machine. This is an asynchronous function.

**Parameters**

- fn\_prefix:** A prefix to add to the name. Pass an empty string to use the default value: "tmpfile". The name will be generated by adding a numeric index to the prefix (e.g. tmpfile1, tmpfile2, tmpfile3, etc.)
- fn\_suffix:** A suffix to add to the name. Pass an empty string for no suffix.
- index\_delim:** A substring that will be inserted between the prefix and the numeric index. For example, if you specify an underscore character '\_' and use the default values for other parameters, the names will look like this: tmpfile\_1, tmpfile\_2, tmpfile\_3, etc. Specify an empty string if you don't want to insert any extra characters.

**Return Value**

A Job object.

**get\_boot\_dev**(*self*, *index*)

Return information about a boot device specified by index.

**Parameters**

- index:** The index of the boot device. Use `Vm.boot_devs` to obtain the boot device count.

**Return Value**

An instance of `VmBootDev`.

```
get_default_settings(os_version, ram_size=0)
```

```
get_device(self, device_type, index)
```

Return an object containing information about a device specified by device type.

**Parameters**

**device\_type:** The type of the device to return the information for. See `consts.PDE_xxx` for the available types.

**index:** The index identifying the device instance (beginning with 0). To determine the number of the devices of the specified type available in the virtual machine, use `Vm.get_devices_count`.

**Return Value**

An instance of a class corresponding to the specified device type. You can search this documentation for available virtual device classes using the following rule: a class name begins with 'Vm' followed by the device type description followed by 'Dev'. For example: `VmHddDev`, `VmNetDev`, `VmSoundDev`, etc.

```
get_devices_count(self, device_type)
```

Determine the total number of devices of the specified type installed in the virtual machine.

**Parameters**

**device\_type:** Device type. See `consts.PDE_xxx` for the available types.

```
get_display_dev(self)
```

```
get_floppy(self, index)
```

Return information about a floppy device specified by index.

**Parameters**

**index:** The index of the FDD device instance. Use `Vm.floppy_devs` to obtain the FDD device count.

**Return Value**

An instance of `VmFloppyDev`.

**get\_hdd**(*self*, *index*)

Return information about a hard disk specified by index.

**Parameters**

**index**: The index of the HDD device instance. Use `Vm.hdd_devs` to obtain the HDD device count.

**Return Value**

An instance of `VmHddDev`.

**get\_net\_adapter**(*self*, *index*)

Return information about a network adapter specified by index.

**Parameters**

**index**: The index of the network adapter instance. Use `Vm.net_adapters` to obtain the net adapter device count.

**Return Value**

An instance of `VmNetDev`.

**get\_optical**(*self*, *index*)

Return information about an optical disk specified by index.

**Parameters**

**index**: The index of the optical disk device instance. Use `Vm.optical_devs` to obtain the optical disk device count.

**Return Value**

An instance of `VmOpticalDev`.

**get\_parallel\_port**(*self*, *index*)

Return information about a parallel port specified by index.

**Parameters**

**index**: The index of the parallel port device instance. Use `Vm.parallel_ports` to obtain the device count.

**Return Value**

An instance of `VmParallelPortDev`.

**get\_scr\_res**(*self*, *index*)

Return screen resolution by index.

**Parameters**

**index**: A numeric index identifying the resolution.

**Return Value**

An instance of `VmScrRes`.

`get_serial_port(self, index)`

Return information about a serial port specified by index.

**Parameters**

`index`: The index of the serial port device instance. Use `Vm.serial_ports` to obtain the device count.

**Return Value**

An instance of `VmSerialDev`.

`get_share(self, index)`

Return information about a shared folder specified by index.

**Parameters**

`index`: The index identifying the shared folder instance. Use `Vm.shares` to obtain the shared folder instance count.

**Return Value**

An instance of `VmShare`.

`get_sound_dev(self, index)`

Return information about a sound device specified by index.

**Parameters**

`index`: The index of the sound device instance. Use `Vm.sound_devs` to obtain the sound device count.

**Return Value**

An instance of `VmSoundDev`.

`get_usb_dev(self, index)`

Return information about a USB device specified by index.

**Parameters**

`index`: The index of the USB device instance. Use `Vm.usb_devs` to obtain the USB device count.

**Return Value**

An instance of `VmUsbDev`.

`get_weak_objects(self)`

`initiate_dev_state_notifications(self)`

**install\_tools**(*self*)

Launch the Parallels Tools installation program. This is an asynchronous function.

**Return Value**

A Job object.

**pause**(*self*, *acpi=False*)

Pause the virtual machine. This is an asynchronous method.

**Return Value**

A Job object.

**refresh\_config**(*self*)**reg**(*self*, *config\_path*, *non\_interactive\_mode=False*)

Create a new virtual machine and register it with the Parallels server. This is an asynchronous method.

**Parameters**

**config\_path:** Name and path of the parent directory where you would like to create the new virtual machine directory. Pass an empty string to create the directory in the default location for this server.

**non\_interactive\_mode:** True - use interactive mode; False - use non-interactive mode.

**Return Value**

A {job} object.

**reg\_event\_handler**(*self*, *data*)**reset**(*self*)

Reset the virtual machine (stop then start; ACPI will not be used). This is an asynchronous method.

**Return Value**

A Job object.

**resume**(*self*)

Resume the virtual machine. This is an asynchronous method

**Return Value**

A Job object.

**send\_utility\_command**(*self*, *cmd*, *flags*, *buffer*)**set\_clipboard**(*self*, *buffer*)**set\_default\_config**(*self*, *srv\_config*, *os\_version*, *create\_devices*)

Automatically select and set a default configuration for the new virtual machine based on the operating system that the virtual machine will be running.

**Parameters**

- srv\_config:** An instance of SrvConfig containing the host server configuration information.
- os\_version:** The version of the operating system that the virtual machine will be running. See PVS.GUEST.VER.xxx for available options.
- create\_devices:** A flag indicating whether default virtual machine devices should be created or not. True - create devices; False - don't create.

**start**(*self*)

Start the virtual machine. This is an asynchronous method.

**Return Value**

A Job object.

**stop**(*self*, *acpi=False*)

Stop the virtual machine. This is an asynchronous method

**Return Value**

A Job object.

**subscribe\_to\_guest\_statistics**(*self*)

Subscribe to receive the virtual machine performance statistics on a periodic basis. This is an asynchronous method.

**Return Value**

A Job object.

**suspend**(*self*)

Suspend the virtual machine. This is an asynchronous method

**Return Value**

A Job object.

**tools\_send\_shutdown**(*self*, *kind=0*)

Gracefully shutdown the virtual machine.

**Parameters**

**kind**: Shutdown option. See `consts.PST_VM_xxx` for available options.

**unreg**(*self*)

Unregister the virtual machine from the Parallels server. The method removes the virtual machine from the Parallels server registry, but does not delete the virtual machine files from the hard drive.

This is an asynchronous method.

**Return Value**

A Job object.

**unreg\_event\_handler**(*self*, *data*)**unsubscribe\_from\_guest\_statistics**(*self*)

Cancel subscriptions started by `subscribe_to_guest_statistics` method.

This is an asynchronous method.

**Return Value**

A Job object.

**update\_config**(*self*)

This method is obsolete and is left for backward compatibility only.

<b>validate_config</b> ( <i>self</i> , <i>section</i> )
Validate the specified section of the virtual machine configuration. This is an asynchronous function.
<b>Parameters</b>
<i>section</i> : The name of the section of the virtual machine configuration. See <code>consts.PVC_xxx</code> for available options.
<b>Return Value</b>
A Job object.

*Inherited from prlsdk.BaseVm(Section 4.6)*

`--new--()`

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

### 5.25.2 Properties

Name	Description
<code>access_rights</code>	Returns an instance of <code>VmAccessRights</code> .
<code>app_in_dock_mode</code>	Determines the dock mode. See <code>consts.PDM_APP_IN_xxx</code> for available options.
<code>auto_start</code>	Determines whether the virtual machine starts automatically on host server start. True - start automatically; False - don't start.
<code>auto_start_delay</code>	Determines the time delay used during the virtual machine automatic startup, in seconds.
<code>auto_stop</code>	Determines the automatic shutdown mode for the virtual machine. See <code>consts.PAO_VM_xxx</code> for available options.
<code>background_priority</code>	Determines background process priority. See <code>PVR_PRIORITY_xxx</code> for available options.
<code>boot_dev_count</code>	The number of devices in the virtual machine boot device priority list.
<code>boot_devs</code>	An iterator/sequence that allows to traverse through boot device instances. See <code>VmBootDev</code> .

*continued on next page*

Name	Description
close_app_on_shutdown	vm.set_close_app_on_shutdown(Vm, close_app_on_shutdown) Sets Virtual Machine close app on shutdown sign.
cpu_accel_level	CPU acceleration level. See <code>consts.PVA_xxx</code> .
cpu_count	The number of CPUs in the virtual machine.
cpu_mode	CPU mode (32 bit or 64 bit). See <code>consts.PCM_CPU_xxx</code> .
cpu_vtx_enabled	vm.get_cpu_vtx_enabled(Vm) -> bool Returns Virtual Machine CPU VTx support enabling sign.
description	The virtual machine description.
disk_cache_write_back	Determines if disk cache write-back is enabled for this virtual machine. True - enabled; False - disabled.
display_dev	
dock_icon_type	Determines the virtual machine dock icon type. See <code>consts.PIT_DOCK_ICON_xxx</code> for available options.
exclude_dock	Determines the guest OS window behavior in coherence mode. True - don't redraw the portion of the window that moved below the dock; False - redraw the entire window.
floppy_count	The number of the available FDD devices.
floppy_devs	An iterator/sequence that allows to traverse through FDD device instances. See <code>VmFloppyDev</code> .
foreground_priority	Determines foreground process priority. See <code>consts.PVR_PRIORITY_xxx</code> for available options.
guest_sharing_auto_mount	Determines whether the host shared folders are mounted automatically in the virtual machine. True - mount automatically; False - don't mount.
guest_sharing_enabled	Determines whether guest sharing is enabled (the guest OS disk drives are visible in the host OS). True - enabled; False - disabled.
hdd_count	The number of the available HDD devices.
hdd_devs	An iterator/sequence that allows to traverse through HDD device instances. See <code>VmHddDev</code> .
home_path	The virtual machine home directory name and path.

*continued on next page*

Name	Description
host_sharing_enabled	Determines whether host sharing is enabled (host shared folders are visible in the guest OS). True - enabled; False - disabled.
icon	vm.set_icon(Vm, new_vm_icon) Sets Virtual Machine icon.
last_modified_date	Determines the date and time when the specified virtual machine was last modified.
last_modifier_name	The name of the user who modified the virtual machine the last time.
multi_display	vm.set_multi_display(Vm, vm_multi_display) Sets Virtual Machine multi display sign coherence option.
name	The virtual machine name.
net_adapter_count	The number of the available net adapter devices.
net_adapters	An iterator/sequence that allows to traverse through net adapter device instances. See <b>VmNetDev</b> .
optical_count	The number of the available optical disk devices.
optical_devs	An iterator/sequence that allows to traverse through optical disk device instances. See <b>VmOpticalDev</b> .
os_res_in_full_scr_mode	Determines if the guest OS screen resolution in full-screen mode is enabled or disabled. True - enabled; False - disabled.
os_type	The type of the OS the virtual machine is running.
os_version	The version of the OS that the virtual machine is running.
parallel_port_count	The number of the available parallel port devices.
parallel_ports	An iterator/sequence that allows to traverse through parallel port device instances. See <b>VmParallelPortDev</b> .
problem_report	Contains a problem report in the event of an abnormal virtual machine termination.
questions	vm.get_questions(Vm) -> QuestionsList This method is an alternative for working with questions synchronically
ram_size	The amount of memory in the virtual machine.

*continued on next page*

Name	Description
relocate_task_bar	Determines whether the Windows 'task bar relocation' feature is enabled in a virtual machine when it runs in the coherence mode. True - enabled; False - disabled.
scr_res_count	vm.get_scr_res_count(Vm) -> int Returns VM screen resolutions count.
scr_res_enabled	vm.set_scr_res_enabled(Vm, vm_scr_res_enabled) Sets Virtual Machine additional screen resolutions support enabling sign.
serial_port_count	The number of the available serial port devices.
serial_ports	An iterator/sequence that allows to traverse through serial port device instances. See <b>VmSerialDev</b> .
server_host	Host server name (hostname).
server_uuid	Host server universally unique ID.
shares	An iterator/sequence that allows to traverse through shared folder instances. See <b>VmShare</b> .
shares_count	The number of shared folders in a virtual machine.
show_task_bar	Determines whether the Windows task bar is displayed when the virtual machine runs in coherence mode. True - task bar is enabled; False - task bar is disabled.
sound_dev_count	The number of the available sound devices.
sound_devs	An iterator/sequence that allows to traverse through sound device instances. See <b>VmSoundDev</b> .
src_res	An iterator that allows to traverse through <b>VmScrRes</b> instances.
start_login_mode	Determines the automatic startup login mode. See <b>PLM_xxx</b> for available options.
start_user_login	Determines the name of the user which will be used during the virtual machine automatic startup.
state	An instance of <b>VmInfo</b> .
statistics	An instance of <b>VmStat</b> .
system_flags	vm.set_system_flags(Vm, new_vm_system_flags) Sets Virtual Machine system flags.
template	Determines whether the virtual machine is a template or a regular virtual machine. True - template; False - regular virtual machine.
tools_info	vm.get_tools_info(Vm) -> <b>VmToolsInfo</b> Extracts VM Tools descriptor

*continued on next page*

Name	Description
usb_dev_count	The number of the available USB devices.
usb_devs	An iterator/sequence that allows to traverse through USB device instances. See VmUsbDev.
uuid	The virtual machine universally unique ID.
video_ram_size	The amount of video memory in the machine.
vnc_config	An instance of VmVNCConfig.
window_mode	vm.set_window_mode(Vm, vm_window_mode) Sets Virtual Machine window mode.
<i>Inherited from prlsdk.BaseVm (Section 4.6)</i> __editing__, __unfin_op__, autostart_editing	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i> __expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i> __handle__	
<i>Inherited from object</i> __class__	

### 5.25.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435461
__prefix__	<b>Value:</b> 'vm'

## 5.26 Class VmAccessRights



### 5.26.1 Methods

<b>__init__</b> (self, dict=None) x.__init__(...) initializes x; see x.__class__.__doc__ for signature Overrides: object.__init__ extit(inherited documentation)
--

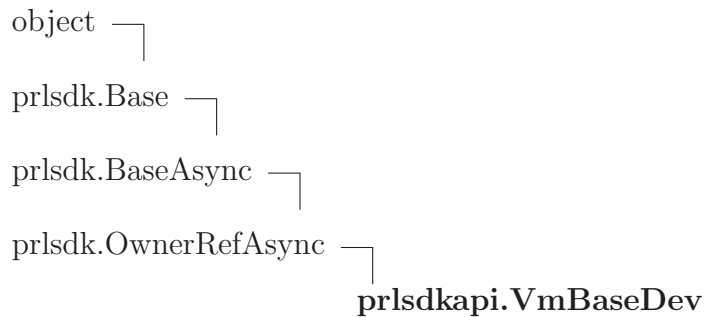
*Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--new--()`, `--reduce--()`, `--reduce_ex--()`,  
`--repr--()`, `--setattr--()`, `--str--()`

### 5.26.2 Properties

Name	Description
<code>current_is_owner</code>	
<code>others_rights</code>	
<code>owner</code>	
<code>owner_rights</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

## 5.27 Class VmBaseDev



**Known Subclasses:** `prlsdkapi.VmBaseDiskDev`, `prlsdkapi.VmBasePortDev`, `prlsdkapi.VmFloppyDev`,  
`prlsdkapi.VmNetDev`, `prlsdkapi.VmSoundDev`, `prlsdkapi.VmUsbDev`

`VmBaseDev` is a base class that provides methods for managing virtual devices.

### 5.27.1 Methods

<p><b>connect</b>(<i>self</i>)</p> <hr/> <p>Connect the device. This operation can be performed on a running virtual machine. This is an asynchronous method.</p> <p><b>Return Value</b>  A Job object.</p>
---

**create**(*self*)

Create a new instance of the same class as this object.

**Return Value**

A new class instance.

**create\_image**(*self*, *allow\_recreate=False*, *non\_interactive\_mode=False*)

Create a virtual device image on the server. This is an asynchronous method.

**Parameters**

**allow\_recreate:** Specifies whether the existing image should be replaced. True - replace the image; False - abort the operation if image exists.

**non\_interactive\_mode:** Specifies whether interactive mode should be used. True - use interactive mode; False - non-interactive mode.

**Return Value**

A Job object.

**disconnect**(*self*)

Disconnect the device. This operation can be performed on a running virtual machine. This is an asynchronous method.

**Return Value**

A Job object.

**remove**(*self*)

Remove this object from the virtual machine device list. The method only removes the object from memory. It does not delete the device from the existing virtual machine configuration on the server.

**set\_default\_stack\_index**(*self*)

Generate and automatically assign a stack index to the storage device. A corresponding device interface (IDE or SCSI) has to be set in advance. This method applies only to VmHddDev and VmOpticalDev.

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

`__init__()`, `__new__()`

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

### *Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`,  
`--str--()`

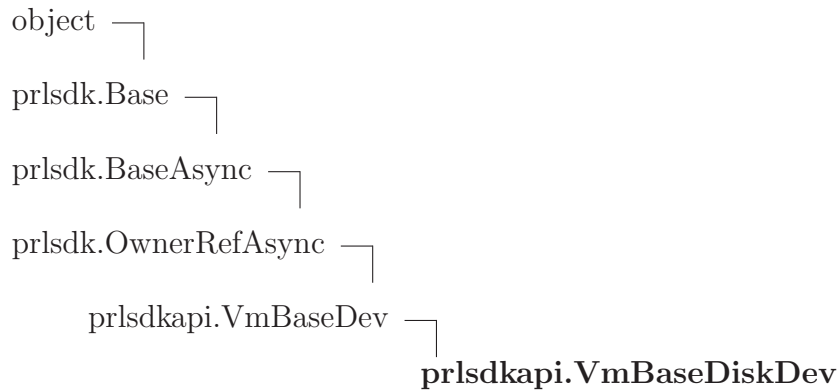
### 5.27.2 Properties

Name	Description
<code>connected</code>	Indicates whether the device is connected. True - connected; False - disconnected.
<code>emulated_type</code>	The device emulation type.
<code>enabled</code>	Indicates whether the device is enabled. True - enabled; False - disabled.
<code>friendly_name</code>	The user-friendly name of the device.
<code>index</code>	A numeric index identifying a device.
<code>remote</code>	Indicates whether this is a remote device. True - remote; False - local.
<code>sys_name</code>	The device system name.
<code>type</code>	The type of the device. See <code>consts.PDE_xxx</code> for available types.
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
<code>--owner--</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.27.3 Class Variables

Name	Description
<code>--int_prefix--</code>	<b>Value:</b> 'vmdev'
<code>--prefix--</code>	<b>Value:</b> ''

## 5.28 Class VmBaseDiskDev



**Known Subclasses:** prlsdkapi.VmHddDev, prlsdkapi.VmOpticalDev

VmBaseDiskDev is a base class that provides methods for managing virtual hard disks.

### 5.28.1 Methods

*Inherited from prlsdkapi.VmBaseDev(Section 5.27)*

connect(), create(), create\_image(), disconnect(), remove(), set\_default\_stack\_index()

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

\_\_init\_\_(), \_\_new\_\_()

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

### 5.28.2 Properties

Name	Description
iface_type	The disk interface type. See <code>consts.PMS_xxx</code> for available types.
image_path	The disk image path.
passthrough	Indicates whether the passthrough mode is enabled for the disk. True - enabled; False - disabled.

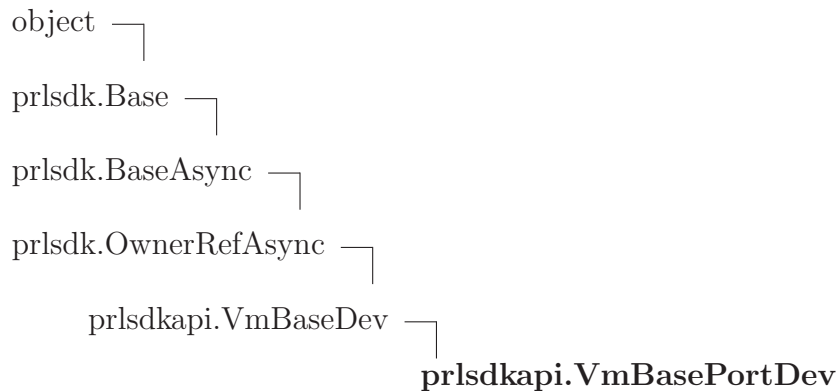
*continued on next page*

Name	Description
stack_index	The virtual disk stack index.
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type	
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
__owner__	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.28.3 Class Variables

Name	Description
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
__int_prefix__, __prefix__	

## 5.29 Class VmBasePortDev



**Known Subclasses:** prlsdkapi.VmParallelPortDev, prlsdkapi.VmSerialDev

Base class providing methods for managing virtual ports (serial, parallel).

### 5.29.1 Methods

*Inherited from prlsdkapi.VmBaseDev(Section 5.27)*

connect(), create(), create\_image(), disconnect(), remove(), set\_default\_stack\_index()

***Inherited from prlsdk.OwnerRefAsync (Section 4.9)***

\_\_init\_\_(), \_\_new\_\_()

***Inherited from prlsdk.BaseAsync (Section 4.4)***

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

***Inherited from object***

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(),  
\_\_str\_\_()

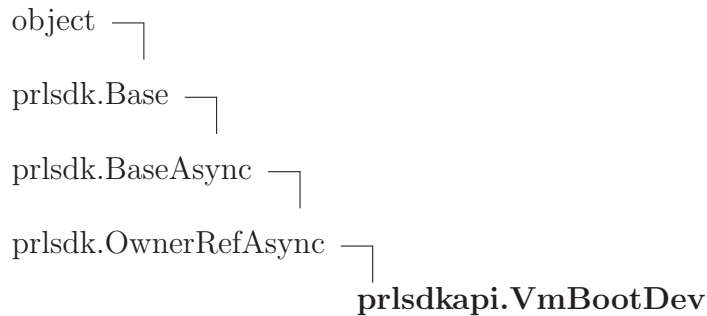
### 5.29.2 Properties

Name	Description
output_file	The contents of the virtual device output file.
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type	
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
__owner__	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.29.3 Class Variables

Name	Description
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
__int_prefix__, __prefix__	

### 5.30 Class VmBootDev



Provides methods for managing boot devices in a virtual machine.

#### 5.30.1 Methods

<b>remove(<i>self</i>)</b>
----------------------------

Remove the device from the boot device priority list.
---

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

`--init--()`, `--new--()`

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

#### 5.30.2 Properties

Name	Description
<code>in_use</code>	Determines whether the boot device is enabled or disabled in the boot priority list. True - enabled; False - disabled and will be ignored during boot.
<code>index</code>	A numeric index of the device. Device index is a property that, together with device type, is used to uniquely identify a device on a virtual machine
<code>sequence_index</code>	The boot device sequence index in the boot priority list.

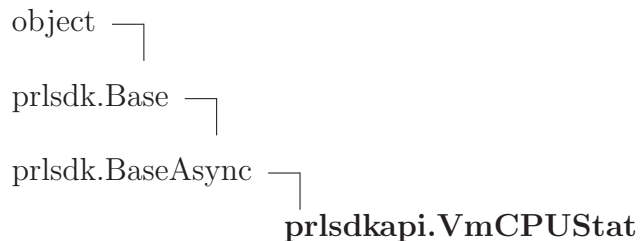
*continued on next page*

Name	Description
type	Identifies the boot device type. See <code>consts.PDE_xxx</code> for available types.
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
<code>__owner__</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>__expected_h_type__</code> , <code>__job__</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

### 5.30.3 Class Variables

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435497
<code>__prefix__</code>	<b>Value:</b> 'bootdev'

## 5.31 Class VmCPUStat



Contains CPU statistics.

### 5.31.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`__cmp__()`, `__init__()`, `__new__()`, `__repr__()`, `cancel()`, `wait()`

*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__str__()`

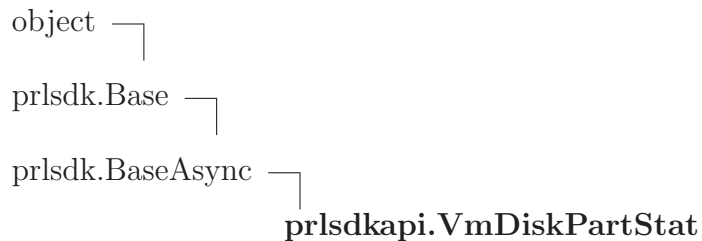
### 5.31.2 Properties

Name	Description
cpu_usage	The CPU usage, in percent.
system_time	CPU system time, in seconds.
total_time	CPU total time, in seconds.
user_time	CPU user time, in seconds.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.31.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435489
__prefix__	<b>Value:</b> 'statcpu'

## 5.32 Class VmDiskPartStat



Contains disk partition statistics.

### 5.32.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_init\_\_(), \_\_new\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

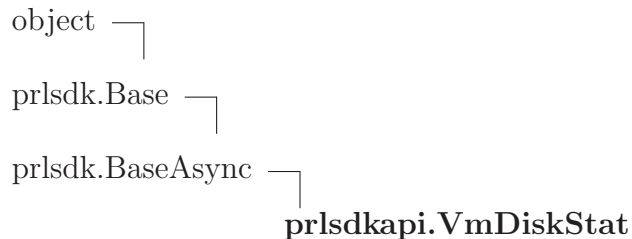
### 5.32.2 Properties

Name	Description
free_disk_space	Free disk space.
system_name	Partition system name.
usage_disk_space	Used disk space.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.32.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435493
__prefix__	<b>Value:</b> 'statdiskpart'

## 5.33 Class VmDiskStat



Contains disk drive statistics.

### 5.33.1 Methods

<b>get_part_stat</b> ( <i>self</i> , <i>index</i> )
Obtain a return disk partition statistics.
<b>Parameters</b>
<i>index</i> : A numeric index identifying a partition. To obtain the partition count, use <code>VmDiskStat.parts_stats_count</code> .
<b>Return Value</b>
An instance of <code>VmDiskPartStat</code> .

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

### Inherited from object

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

#### 5.33.2 Properties

Name	Description
<code>free_disk_space</code>	Free disk space, in bytes.
<code>part_stats</code>	An iterator/sequence that allows to traverse through <code>VmDiskPartStat</code> instances contained in this object.
<code>parts_stats_count</code>	The number of partitions on the disk.
<code>system_name</code>	The disk drive system name.
<code>usage_disk_space</code>	The size of the used space on the disk, in bytes.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

#### 5.33.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435492
<code>--prefix--</code>	<b>Value:</b> 'statdisk'

### 5.34 Class VmDisplayDev

```

object └─
prlsdkapi.PrlObject └─
prlsdkapi.VmObjEmulation └─
                                prlsdkapi.VmDisplayDev

```

The `VmDisplayDev` class is a part of the functionality provided by the Parallels Server API that allows you to write your own remote desktop applications.

## 5.34.1 Methods

`__del__(self)`

`__ensure_attr__(self)`

Overrides: prlsdkapi.VmObjEmulation.\_\_ensure\_attr\_\_

`__init__(self, vm)`

x.\_\_init\_\_(...) initializes x; see x.\_\_class\_\_.\_\_doc\_\_ for signature

Overrides: object.\_\_init\_\_ exitit(inherited documentation)

`capture_screen_to_file(self, filename, x, y, width, height, format=1342177291, quality=-1)`

Captures a screen area of a remote virtual machine directly to a file on the client computer. This is an asynchronous method.

**Parameters**

- `filename`: Name and path of the file to save the image to.
- `x`: The X coordinate of the area to capture.
- `y`: The Y coordinate of the area to capture.
- `width`: The width of the area to capture (pixels).
- `height`: The height of the area to capture (pixels).
- `format`: Image format to use. See `consts.PIF_XXX`.
- `quality`: Image quality factor. Must be in the range from 0 to 100, or -1. Specify 0 to use the highest data compression, 100 for no compression, or -1 for the default value.

**Return Value**

A Job object.

`check_connected(self)`

`connect_to_vm(self)`

Connect to a virtual machine to begin remote desktop input/output operations. This is an asynchronous method.

**Return Value**

A Job object.

---

**disconnect\_from\_vm**(*self*)

---

 Disconnect from a virtual machine.

---

**engage\_job**(*self*, *job*)

---

**get\_screen\_buffer**(*self*)

---

**move\_mouse**(*self*, *coord\_dxdydz*, *me*, *mb*)

---

 Move the mouse pointer to a relative position and send a mouse event (press or release the specified mouse button). The position to which you want to move the pointer is calculated relatively to the center of the screen.

**Parameters**

*coord\_dxdydz*: A tuple containing mouse pointer coordinate offsets (dx, dy, dz). The 'dz' offset (scroll) is optional.

*me*: Mouse event. See `consts.ME_xxx`.

*mb*: Mouse button event. See `consts.MB_xxx`.

---

**send\_key\_event**(*self*, *action*, *keys*, *action\_time\_wait=-1*)

---

 Send a set of keyboard key events to the virtual machine.

**Parameters**

*action*: Action. See `consts.PKE_xxx` for available actions.

*keys*: A tuple containing a list of key codes. See `consts.ScanCodesList` for available codes.

*action\_time\_wait*: A time delay imposed on key input after the event execution (in milliseconds).

---

**send\_keys\_event**(*self*, *action\_key\_pairs*, *action\_time\_wait=-1*)

---

 Send a set of keyboard key events to the virtual machine.

**Parameters**

*action\_key\_pairs*: A tuple containing action/key pairs (action, key\_codes). The key\_codes item is also a tuple containing a list of key codes. See `consts.ScanCodesList` for available codes. See `consts.PKE_xxx` for available actions.

*action\_time\_wait*: A time delay between actions specified in the *action\_key\_pairs* parameter (in milliseconds).

<b>send_screen_size_event</b> ( <i>self, width, height, depth</i> )
---

Send "screen size change" event to the virtual machine.

**Parameters**

**width:** The new resolution width.  
**height:** The new resolution height.  
**depth:** The new resolution depth.

<b>set_mouse_pos</b> ( <i>self, coord_xyz, me, mb</i> )
---

Set the mouse pointer position.

**Parameters**

**coord\_xyz:** A tuple containing mouse pointer coordinates (x, y, z).  
The 'z' coordinate (scroll) is optional.  
**me:** Mouse event. See `consts.ME_xxx`.  
**mb:** Mouse button event. See `consts.MB_xxx`.

*Inherited from prlsdkapi.VmObjEmulation(Section 5.40)*

`call_vm()`, `inderect_self()`, `safe_call()`

*Inherited from object*

`--delattr--()`, `--getattrattribute--()`, `--hash--()`, `--new--()`, `--reduce--()`, `--reduce_ex--()`,  
`--repr--()`, `--setattr--()`, `--str--()`

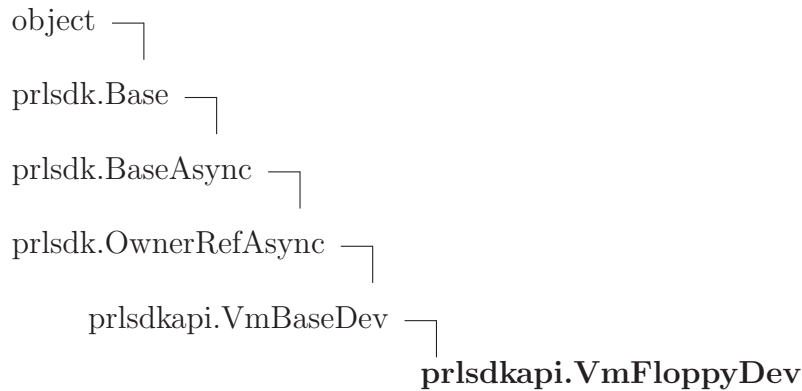
### 5.34.2 Properties

Name	Description
<code>dyn_res_tool_status</code>	Determines if Dynamic Resolution Tool is enabled. True - enabled; False - disabled.
<code>screen_size</code>	The virtual machine screen size (width, height, depth).
<code>sliding_mouse_enabled</code>	Determines if sliding mouse is enabled. True - enabled; False - disabled.
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.34.3 Class Variables

Name	Description
<code>--prefix--</code>	<b>Value:</b> 'devdisplay'

### 5.35 Class VmFloppyDev



Provides methods for managing virtual floppy disk drives.

#### 5.35.1 Methods

*Inherited from prlsdkapi.VmBaseDev(Section 5.27)*

connect(), create(), create\_image(), disconnect(), remove(), set\_default\_stack\_index()

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

\_\_init\_\_(), \_\_new\_\_()

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(),  
\_\_str\_\_()

#### 5.35.2 Properties

Name	Description
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	__owner__
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	__expected_h_type__, __job__, finished, handle_type

*continued on next page*

Name	Description
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

### 5.35.3 Class Variables

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435465
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
<code>__int_prefix__</code> , <code>__prefix__</code>	

## 5.36 Class VmHddDev



Provides methods for managing virtual hard disk drives.

### 5.36.1 Methods

*Inherited from prlsdkapi.VmBaseDev(Section 5.27)*

`connect()`, `create()`, `create_image()`, `disconnect()`, `remove()`, `set_default_stack_index()`

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

`__init__()`, `__new__()`

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

### Inherited from object

`--delattr--()`, `--getattr__()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

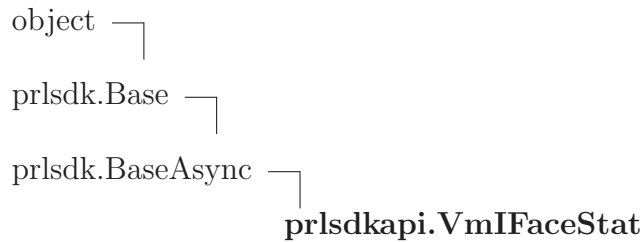
### 5.36.2 Properties

Name	Description
<code>disk_size</code>	Disk size in bytes.
<code>disk_type</code>	Disk type. See <code>consts.PHD_XXX</code> for available types.
<code>size_on_disk</code>	The size of the occupied space on the disk, in bytes.
<code>splitted</code>	Determines if the virtual hard disk is split into multiple files. True - multiple files; False - single file.
<i>Inherited from prlsdkapi.VmBaseDiskDev (Section 5.28)</i>	
<code>iface_type</code> , <code>image_path</code> , <code>passthrough</code> , <code>stack_index</code>	
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
<code>connected</code> , <code>emulated_type</code> , <code>enabled</code> , <code>friendly_name</code> , <code>index</code> , <code>remote</code> , <code>sys_name</code> , <code>type</code>	
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
<code>--owner--</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.36.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435466
<code>--prefix--</code>	<b>Value:</b> 'vmdevhdd'
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
<code>--int_prefix--</code>	

### 5.37 Class VmIFaceStat



Contains network interface statistics.

#### 5.37.1 Methods

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattribute--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

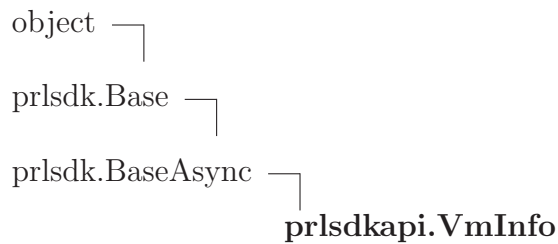
#### 5.37.2 Properties

Name	Description
<code>in_data_size</code>	The total number of bytes the network interface has received since the server was last started.
<code>in_pkgs_count</code>	The total number of packets the network interface has received since the server was last started.
<code>out_data_size</code>	The total number of bytes the network interface has sent since the server was last started.
<code>out_pkgs_count</code>	The total number of packets the network interface has sent since the server was last started.
<code>system_name</code>	The network interface system name.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.37.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435490
<code>--prefix--</code>	<b>Value:</b> 'statiface'

## 5.38 Class VmInfo



Contains a virtual machine state and access rights information.

### 5.38.1 Methods

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattribute--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

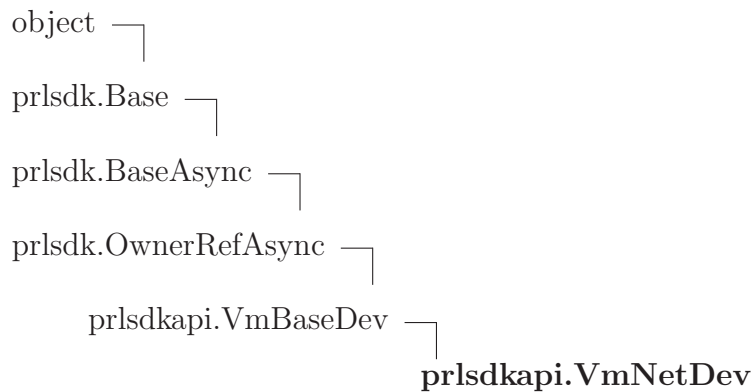
### 5.38.2 Properties

Name	Description
<code>access_rights</code>	An instance of <code>VmAccessRights</code> .
<code>state</code>	The virtual machine state. See <code>consts.VMS_xxx</code> for available states.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.38.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435476
<code>--prefix--</code>	<b>Value:</b> 'vminfo'

## 5.39 Class VmNetDev



Provides methods for managing virtual network adapters.

### 5.39.1 Methods

<b>generate_mac_addr</b> ( <i>self</i> )
Generate a unique MAC address for the adapter. The method generates a new MAC address and automatically assigns it to the adapter. The method does not return the address to the caller. To obtain the address, use <code>VmNetDev.mac_address</code> .

*Inherited from prlsdkapi.VmBaseDev(Section 5.27)*

`connect()`, `create()`, `create_image()`, `disconnect()`, `remove()`, `set_default_stack_index()`

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

`--init--()`, `--new--()`

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr__()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

### 5.39.2 Properties

Name	Description
<code>bound_adapter_index</code>	The index of the adapter to which this virtual adapter is bound.
<code>bound_adapter_name</code>	The name of the adapter to which this virtual adapter is bound
<code>mac_address</code>	The MAC address of the adapter.
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
<code>connected</code> , <code>emulated_type</code> , <code>enabled</code> , <code>friendly_name</code> , <code>index</code> , <code>remote</code> , <code>sys_name</code> , <code>type</code>	
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
<code>--owner--</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.39.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435467
<code>--prefix--</code>	<b>Value:</b> 'vmdevnet'
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
<code>--int_prefix--</code>	

## 5.40 Class VmObjEmulation



**Known Subclasses:** `prlsdkapi.VmDisplayDev`, `prlsdkapi.VmVNCCConfig`

## 5.40.1 Methods

`--ensure_attr__(self)``--init__(self, vm)`

x.\_\_init\_\_(...) initializes x; see x.\_\_class\_\_.\_\_doc\_\_ for signature

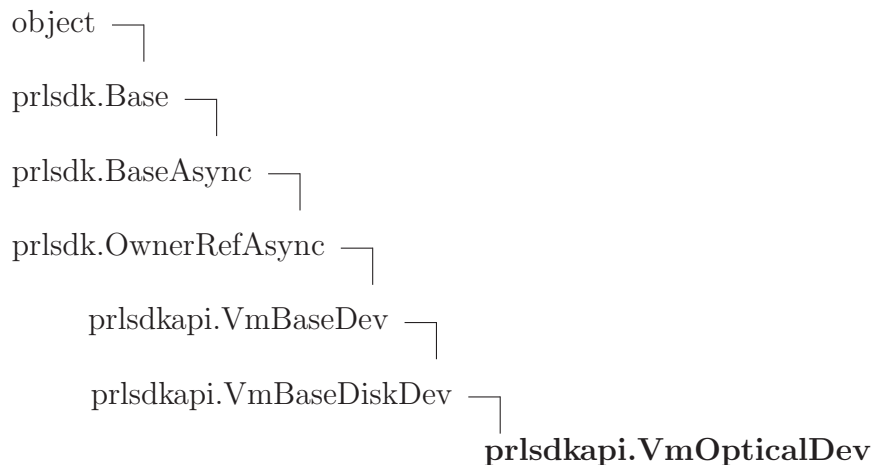
Overrides: object.\_\_init\_\_ exitit(inherited documentation)

`call_vm(self, method, *args)``indirect_self(self)``safe_call(self, method, *args)`*Inherited from object*`__delattr__()`, `__getattr__()`, `__hash__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`,  
`__repr__()`, `__setattr__()`, `__str__()`

## 5.40.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>__class__</code>	

## 5.41 Class VmOpticalDev



Provides methods for managing virtual optical disk drives.

### 5.41.1 Methods

#### *Inherited from prlsdkapi.VmBaseDev (Section 5.27)*

connect(), create(), create\_image(), disconnect(), remove(), set\_default\_stack\_index()

#### *Inherited from prlsdk.OwnerRefAsync (Section 4.9)*

\_\_init\_\_(), \_\_new\_\_()

#### *Inherited from prlsdk.BaseAsync (Section 4.4)*

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

#### *Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

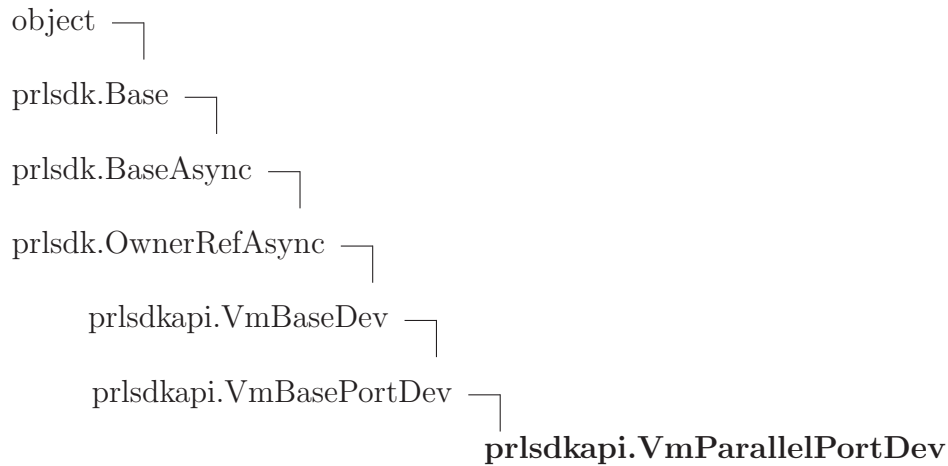
### 5.41.2 Properties

Name	Description
<i>Inherited from prlsdkapi.VmBaseDiskDev (Section 5.28)</i>	iface_type, image_path, passthrough, stack_index
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	__owner__
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	__expected_h_type__, __job__, finished, handle_type
<i>Inherited from prlsdk.Base (Section 4.3)</i>	__handle__
<i>Inherited from object</i>	__class__

### 5.41.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435470
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	__int_prefix__, __prefix__

## 5.42 Class VmParallelPortDev



Provides methods for managing virtual parallel ports.

### 5.42.1 Methods

#### *Inherited from prlsdkapi.VmBaseDev (Section 5.27)*

connect(), create(), create\_image(), disconnect(), remove(), set\_default\_stack\_index()

#### *Inherited from prlsdk.OwnerRefAsync (Section 4.9)*

\_\_init\_\_(), \_\_new\_\_()

#### *Inherited from prlsdk.BaseAsync (Section 4.4)*

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

#### *Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

### 5.42.2 Properties

Name	Description
<i>Inherited from prlsdkapi.VmBasePortDev (Section 5.29)</i> output_file	
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i> connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type	
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	

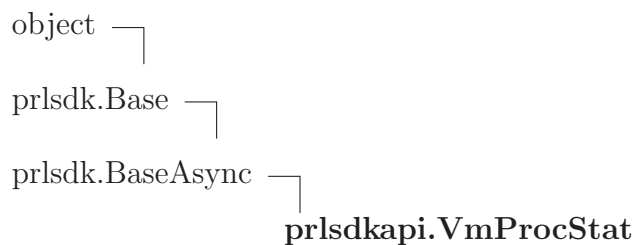
*continued on next page*

Name	Description
<code>--owner--</code>	
	<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
	<i>Inherited from prlsdk.Base (Section 4.3)</i>
<code>--handle--</code>	
	<i>Inherited from object</i>
<code>--class--</code>	

### 5.42.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435468
	<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>
<code>--int_prefix--</code> , <code>--prefix--</code>	

## 5.43 Class VmProcStat



Contains system process statistics.

### 5.43.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

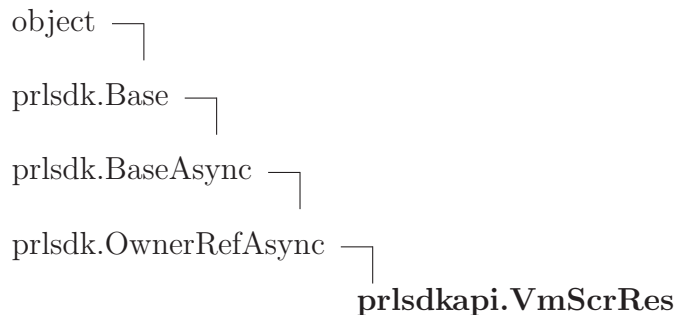
### 5.43.2 Properties

Name	Description
command_name	The process command name.
cpu_usage	The process CPU usage, in percent.
id	The process system ID.
owner_user_name	The name of the process owner.
real_mem_usage	Real memory usage size, in bytes
start_time	Start time of the process, in seconds (number of seconds since January 1, 1601 (UTC)).
state	The process state. See <code>consts.PPS_PROC_xxx</code> .
system_time	System running time of the process, in seconds.
total_mem_usage	Total memory usage size, in bytes.
total_time	Total running time of the process, in seconds
user_time	User running time of the process, in seconds.
virt_mem_usage	Virtual memory usage size, in bytes.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.43.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435494
__prefix__	<b>Value:</b> 'statproc'

## 5.44 Class VmScrRes



Contains a virtual machine screen resolution information.

### 5.44.1 Methods

<b>remove</b> ( <i>self</i> )
-------------------------------

Removes this screen resolution from the virtual machine configuration.
--

*Inherited from prlsdk.OwnerRefAsync (Section 4.9)*

`--init--()`, `--new--()`

*Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattr__()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

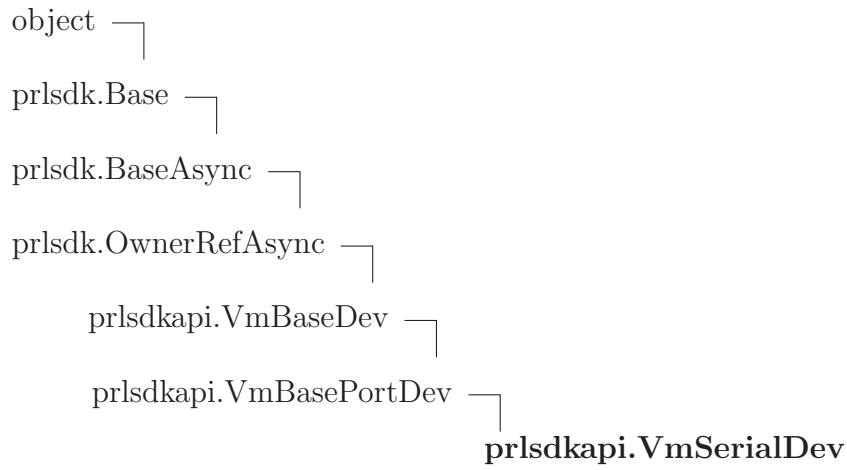
### 5.44.2 Properties

Name	Description
enabled	Determines whether this screen resolution is enabled in the virtual machine. True - enabled; False - disabled.
height	The screen resolution height.
width	The screen resolution width.
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
<code>--owner--</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.44.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435480
<code>--prefix--</code>	<b>Value:</b> 'scrres'

## 5.45 Class VmSerialDev



Provides methods for managing virtual serial ports.

### 5.45.1 Methods

*Inherited from prlsdkapi.VmBaseDev(Section 5.27)*

connect(), create(), create\_image(), disconnect(), remove(), set\_default\_stack\_index()

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

\_\_init\_\_(), \_\_new\_\_()

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(),  
\_\_str\_\_()

### 5.45.2 Properties

Name	Description
socket_mode	The socket mode of this serial port. See <code>consts.PSP_SERIAL_SOCKET_xxx</code> for available modes.
<i>Inherited from prlsdkapi.VmBasePortDev (Section 5.29)</i>	
output_file	
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	

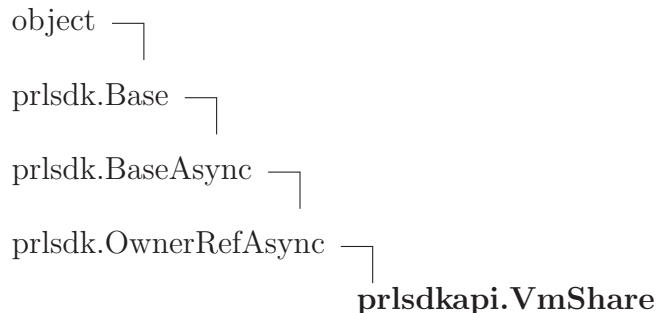
*continued on next page*

Name	Description
	connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type
	<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>
__owner__	
	<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>
__expected_h_type__, __job__, finished, handle_type	
	<i>Inherited from prlsdk.Base (Section 4.3)</i>
__handle__	
	<i>Inherited from object</i>
__class__	

### 5.45.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435469
__prefix__	<b>Value:</b> 'vmdevserial'
	<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>
__int_prefix__	

## 5.46 Class VmShare



Provides methods for managing host shared folder accessibility in a virtual machine. Using these methods, you can make a host OS folder visible and accessible in a virtual machine.

### 5.46.1 Methods

<b>remove(<i>self</i>)</b>
Remove this share from the virtual machine configuration.

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

`__init__()`, `__new__()`

***Inherited from prlsdk.BaseAsync (Section 4.4)***

`__cmp__()`, `__repr__()`, `cancel()`, `wait()`

***Inherited from object***

`__delattr__()`, `__getattr__()`, `__hash__()`, `__reduce__()`, `__reduce_ex__()`, `__setattr__()`, `__str__()`

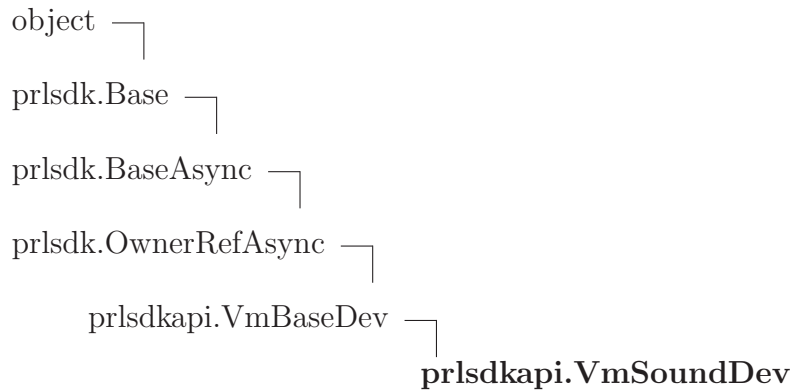
**5.46.2 Properties**

Name	Description
description	<code>share.set_description(Share, new_share_description)</code> Sets VM shared folder description.
enabled	Determines whether the shared folder is enabled. True - enabled; False - disabled.
name	Shared folder name.
path	Shared folder path.
read_only	Determines whether the shared folder is read-only. True - read only; False - writable.
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
<code>__owner__</code>	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>__expected_h_type__</code> , <code>__job__</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

**5.46.3 Class Variables**

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435479
<code>__prefix__</code>	<b>Value:</b> 'share'

## 5.47 Class VmSoundDev



Provides methods for managing virtual sound devices.

### 5.47.1 Methods

#### *Inherited from prlsdkapi.VmBaseDev (Section 5.27)*

connect(), create(), create\_image(), disconnect(), remove(), set\_default\_stack\_index()

#### *Inherited from prlsdk.OwnerRefAsync (Section 4.9)*

\_\_init\_\_(), \_\_new\_\_()

#### *Inherited from prlsdk.BaseAsync (Section 4.4)*

\_\_cmp\_\_(), \_\_repr\_\_(), cancel(), wait()

#### *Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

### 5.47.2 Properties

Name	Description
mixer_dev	Mixer device string.
output_dev	Output device string.
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type	
<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i>	
__owner__	
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	

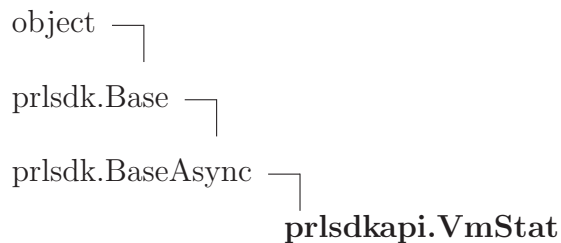
*continued on next page*

Name	Description
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.47.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435472
<code>--prefix--</code>	<b>Value:</b> 'vmdevsound'
<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i>	
<code>--int_prefix--</code>	

## 5.48 Class VmStat



Provides methods for obtaining performance statistics for host server and virtual machines.

### 5.48.1 Methods

<code>get_cpu_stat(self, index)</code>
Obtain and return CPU statistics.
<b>Parameters</b>
<b>index:</b> A numeric index identifying a CPU instance. To obtain the instance count, use <code>VmStat.cpus_stats_count</code> .
<b>Return Value</b>
An instance of <code>VmCPUStat</code> .

**get\_disk\_stat**(*self*, *index*)

Obtain and return disk drive statistics.

**Parameters**

**index**: A numeric index identifying a disk. To obtain the disk count, use `VmStat.disks_stats_count`.

**Return Value**

An instance of `VmDiskStat`.

**get\_iface\_stat**(*self*, *index*)

Obtain and return network interface statistics.

**Parameters**

**index**: A numeric instance identifying a network interface. To obtain the net interface count, use `VmStat.ifaces_stats_count`.

**Return Value**

An instance of `VmIFaceStat`.

**get\_proc\_stat**(*self*, *index*)

Obtain and return system process statistics.

**Parameters**

**index**: A numeric index identifying a process. To obtain the process count, use `VmStat.procs_stats_count`.

**Return Value**

An instance of `VmProcStat`.

**get\_user\_stat**(*self*, *index*)

Obtain and return the user session statistics.

**Parameters**

**index**: A numeric index identifying a user session. To obtain the user session count, use `VmStat.users_stats_count`.

**Return Value**

An instance of `VmUserStat`.

**Inherited from prlsdk.BaseAsync (Section 4.4)**

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

**Inherited from object**

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

## 5.48.2 Properties

Name	Description
cpu_stats	An iterator/sequence that allows to traverse through <code>VmCPUStat</code> instances.
cpus_stats_count	Indicates the number of instances of <code>VmCPUStat</code> contained in this object.
disk_stats	An iterator/sequence that allows to traverse through <code>VmDiskStat</code> instances.
disks_stats_count	Indicates the number of instances of <code>VmDiskStat</code> contained in this object.
disp_uptime	The Parallels Server dispatcher uptime in seconds.
free_ram_size	The amount of free RAM in bytes.
free_swap_size	The amount of free swap in bytes.
iface_stats	An iterator/sequence that allows to traverse through <code>VmIFaceStat</code> instances.
ifaces_stats_count	Indicates the number of instance of <code>VmIFaceStat</code> contained in this object.
os_uptime	The operating system uptime in seconds.
proc_stats	An iterator/sequence that allows to traverse through <code>VmProcStat</code> instances.
procs_stats_count	Indicates the number of instances of <code>VmProcStat</code> contained in this object.
total_ram_size	Total RAM size in bytes.
total_swap_size	Total swap size in bytes.
usage_ram_size	The amount of used RAM in bytes.
usage_swap_size	The amount of used swap in bytes.
user_stats	An iterator/sequence that allows to traverse through <code>VmUserStat</code> instances.
users_stats_count	Indicates the number of instances of <code>VmUserStat</code> contained in this object.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>__expected_h_type__</code> , <code>__job__</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>__handle__</code>	
<i>Inherited from object</i>	
<code>__class__</code>	

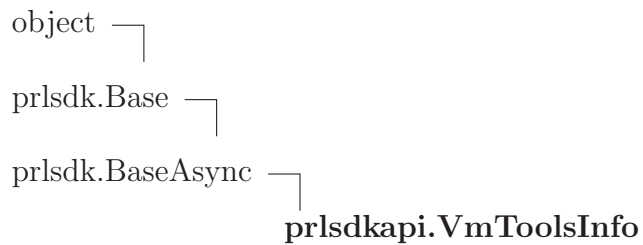
## 5.48.3 Class Variables

Name	Description
<code>__ht__</code>	<b>Value:</b> 268435488

*continued on next page*

Name	Description
<code>--prefix--</code>	<b>Value:</b> 'stat'

## 5.49 Class VmToolsInfo



Provides methods for determining whether the Parallels Tools package is installed in the specified virtual machine and for obtaining its status and version information.

### 5.49.1 Methods

#### *Inherited from prlsdk.BaseAsync (Section 4.4)*

`--cmp--()`, `--init--()`, `--new--()`, `--repr--()`, `cancel()`, `wait()`

#### *Inherited from object*

`--delattr--()`, `--getattr--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

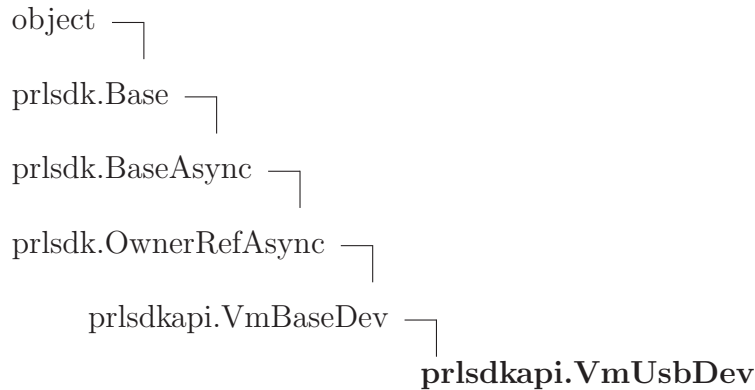
### 5.49.2 Properties

Name	Description
<code>state</code>	The Parallels Tools package state information. See <code>consts.PTS_xxx</code> for the list of states.
<code>version</code>	The Parallels Tools package version.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
<code>--expected_h_type--</code> , <code>--job--</code> , <code>finished</code> , <code>handle_type</code>	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
<code>--handle--</code>	
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.49.3 Class Variables

Name	Description
<code>--ht--</code>	<b>Value:</b> 268435505
<code>--prefix--</code>	<b>Value:</b> 'vmtoolsinfo'

## 5.50 Class VmUsbDev



Provides methods for managing virtual USB devices.

### 5.50.1 Methods

*Inherited from prlsdkapi.VmBaseDev(Section 5.27)*

`connect()`, `create()`, `create_image()`, `disconnect()`, `remove()`, `set_default_stack_index()`

*Inherited from prlsdk.OwnerRefAsync(Section 4.9)*

`--init--()`, `--new--()`

*Inherited from prlsdk.BaseAsync(Section 4.4)*

`--cmp--()`, `--repr--()`, `cancel()`, `wait()`

*Inherited from object*

`--delattr--()`, `--getattribute--()`, `--hash--()`, `--reduce--()`, `--reduce_ex--()`, `--setattr--()`, `--str--()`

### 5.50.2 Properties

Name	Description
<code>autoconnect_option</code>	USB controller autoconnect option. See <code>consts.PUD_xxx</code> for available options.

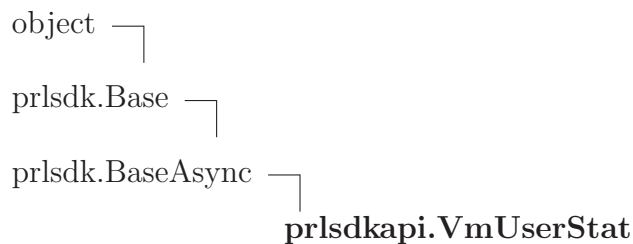
*continued on next page*

Name	Description
	<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i> connected, emulated_type, enabled, friendly_name, index, remote, sys_name, type
	<i>Inherited from prlsdk.OwnerRefAsync (Section 4.9)</i> __owner__
	<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i> __expected_h_type__, __job__, finished, handle_type
	<i>Inherited from prlsdk.Base (Section 4.3)</i> __handle__
	<i>Inherited from object</i> __class__

### 5.50.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435471
__prefix__	<b>Value:</b> 'vmdevusb'
	<i>Inherited from prlsdkapi.VmBaseDev (Section 5.27)</i> __int_prefix__

## 5.51 Class VmUserStat



Contains user session statistics.

### 5.51.1 Methods

*Inherited from prlsdk.BaseAsync(Section 4.4)*

\_\_cmp\_\_(), \_\_init\_\_(), \_\_new\_\_(), \_\_repr\_\_(), cancel(), wait()

*Inherited from object*

\_\_delattr\_\_(), \_\_getattr\_\_(), \_\_hash\_\_(), \_\_reduce\_\_(), \_\_reduce\_ex\_\_(), \_\_setattr\_\_(), \_\_str\_\_()

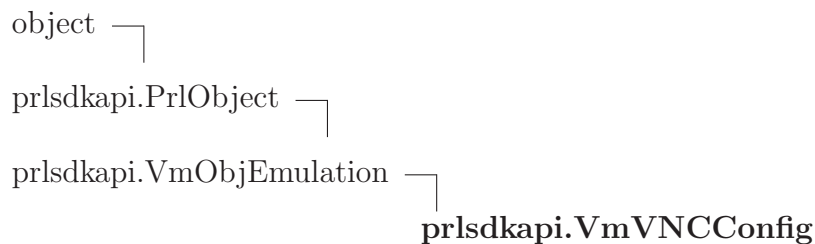
### 5.51.2 Properties

Name	Description
host_name	The host name of the client machine from which the session was initiated.
service_name	The name of the host system service that created the session
session_time	The session duration, in seconds.
user_name	The name of the session owner.
<i>Inherited from prlsdk.BaseAsync (Section 4.4)</i>	
__expected_h_type__, __job__, finished, handle_type	
<i>Inherited from prlsdk.Base (Section 4.3)</i>	
__handle__	
<i>Inherited from object</i>	
__class__	

### 5.51.3 Class Variables

Name	Description
__ht__	<b>Value:</b> 268435491
__prefix__	<b>Value:</b> 'statuser'

## 5.52 Class VmVNCConfig



Contains VNC properties of a virtual machine.

### 5.52.1 Methods

*Inherited from prlsdkapi.VmObjEmulation(Section 5.40)*

\_\_ensure\_attr\_\_(), \_\_init\_\_(), call\_vm(), inderect\_self(), safe\_call()

*Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`,  
`__repr__()`, `__setattr__()`, `__str__()`

### 5.52.2 Properties

Name	Description
host_name	The VNC hostname.
mode	The VNC mode. See <code>consts.PRD_xxx</code> for available modes.
password	The VNC password.
port	The VNC port number.
<i>Inherited from object</i>	
<code>__class__</code>	

### 5.52.3 Class Variables

Name	Description
<code>__prefix__</code>	<b>Value:</b> 'vm_vnc'

## 5.53 Class deinit\_sdk

object └─  
           prlsdkapi.deinit\_sdk

### 5.53.1 Methods

`__del__(self)`

`__init__(self)`  
 x.`__init__`(...) initializes x; see x.`__class__`.`__doc__` for signature  
 Overrides: object.`__init__` extit(inherited documentation)

### *Inherited from object*

`__delattr__()`, `__getattr__()`, `__hash__()`, `__new__()`, `__reduce__()`, `__reduce_ex__()`,  
`__repr__()`, `__setattr__()`, `__str__()`

### 5.53.2 Properties

Name	Description
<i>Inherited from object</i>	
<code>--class--</code>	

### 5.53.3 Class Variables

Name	Description
slots	Value: 'prlsdk'

## Index

- prlsdk (*module*), 2–24
  - prlsdk.\_\_debug\_level\_\_ (*function*), 3, 74
  - prlsdk.Base (*class*), 3–4, 74–75
    - prlsdk.Base.\_\_cmp\_\_ (*method*), 3, 74
  - prlsdk.BaseAsync (*class*), 4–6, 75–77
    - prlsdk.BaseAsync.cancel (*method*), 5, 76
    - prlsdk.BaseAsync.wait (*method*), 5, 76
  - prlsdk.BaseServer (*class*), 6–8, 77–79
    - prlsdk.BaseServer.login (*method*), 6, 77
    - prlsdk.BaseServer.login\_local (*method*), 7, 78
    - prlsdk.BaseServer.logoff (*method*), 7, 78
    - prlsdk.BaseServer.reg\_event\_handler (*method*), 7, 78
    - prlsdk.BaseServer.shutdown (*method*), 7, 78
    - prlsdk.BaseServer.unreg\_event\_handler (*method*), 7, 78
  - prlsdk.BaseVm (*class*), 8–9, 79–80
  - prlsdk.CheckServerAlive (*function*), 2, 73
  - prlsdk.consts (*module*), 25–37
  - prlsdk.DefaultJobTimeout (*function*), 2, 73
  - prlsdk.DefaultKeyActWait (*function*), 2, 73
  - prlsdk.DefaultTimeout (*function*), 2, 73
  - prlsdk.DeinitializeSDK (*function*), 2, 73
  - prlsdk.errors (*module*), 38–72
  - prlsdk.Event (*class*), 9–11, 80–82
    - prlsdk.Event.\_\_getitem\_\_ (*method*), 10, 81
    - prlsdk.Event.\_\_iter\_\_ (*method*), 10, 81
    - prlsdk.Event.\_\_len\_\_ (*method*), 10, 81
    - prlsdk.Event.get\_param (*method*), 10, 81
    - prlsdk.Event.get\_params\_count (*method*), 10, 81
  - prlsdk.GetSDKLibraryPath (*function*), 2, 73
  - prlsdk.GetServerList (*function*), 2, 73
  - prlsdk.InitializeSDK (*function*), 2, 73
  - prlsdk.IsSDKInitialized (*function*), 2, 73
  - prlsdk.Job (*class*), 11–13, 82–84
    - prlsdk.Job.cancel (*method*), 11, 82
    - prlsdk.Job.get\_held\_object (*method*), 12, 83
    - prlsdk.Job.hold\_object (*method*), 12, 83
    - prlsdk.Job.wait (*method*), 12, 83
  - prlsdk.OwnerRefAsync (*class*), 13–14, 84–85
  - prlsdk.ParallelsError (*class*), 14–15, 85–86
  - prlsdk.PrlJobTimeoutError (*class*), 15, 86
  - prlsdk.PrlNotImplError (*class*), 15–16, 86–87
  - prlsdk.PrlSDKAsyncError (*class*), 16–18, 87–89
  - prlsdk.PrlSDKError (*class*), 18–19, 89–90
  - prlsdk.PrlStateError (*class*), 19–20, 90–91
  - prlsdk.Question (*class*), 20–21, 91–92
    - prlsdk.Question.send\_answer (*method*), 20, 91
  - prlsdk.Result (*class*), 21–23, 92–94
    - prlsdk.Result.\_\_getitem\_\_ (*method*), 21, 92
    - prlsdk.Result.\_\_iter\_\_ (*method*), 21, 92
    - prlsdk.Result.\_\_len\_\_ (*method*), 22, 93
    - prlsdk.Result.get\_param (*method*), 22, 93
    - prlsdk.Result.get\_param\_as\_string (*method*), 22, 93
    - prlsdk.Result.get\_params\_count (*method*), 22, 93
  - prlsdk.ServerInfo (*class*), 23–24, 94–95
  - prlsdk.SetSDKLibraryPath (*function*), 3, 74
- prlsdk' (*module*), 73–95
- prlsdkapi (*package*), 96–186
  - prlsdkapi.\_prl\_sdk\_register\_objects\_\_ (*function*), 96
  - prlsdkapi.check\_platform\_and\_version (*function*), 96
  - prlsdkapi.deinit\_sdk (*class*), 185–186

- prlsdkapi.deinit\_sdk.\_\_del\_\_ (method), 185
- prlsdkapi.DispConfig (class), 97–99
  - prlsdkapi.DispConfig.commit (method), 97
  - prlsdkapi.DispConfig.create\_net\_adapter (method), 97
  - prlsdkapi.DispConfig.get\_net\_adapter (method), 98
  - prlsdkapi.DispConfig.remove\_net\_adapter (method), 98
- prlsdkapi.DispNet (class), 99–101
  - prlsdkapi.DispNet.remove (method), 100
- prlsdkapi.FoundVm (class), 104–105
- prlsdkapi.FSEntry (class), 101–103
- prlsdkapi.FSInfo (class), 103–104
  - prlsdkapi.FSInfo.get\_entry (method), 103
- prlsdkapi.get\_class\_methods\_doc (function), 96
- prlsdkapi.get\_handle\_type (function), 96
- prlsdkapi.get\_os\_type\_from\_version (function), 96
- prlsdkapi.GetClientConfig (function), 96
- prlsdkapi.LicenceInfo (class), 105–106
- prlsdkapi.LoginResponse (class), 106–108
  - prlsdkapi.LoginResponse.get\_run\_task (method), 107
- prlsdkapi.NetServiceStatus (class), 108–109
- prlsdkapi.prl\_datetime\_property (function), 96
- prlsdkapi.prl\_indirect\_property (function), 96
- prlsdkapi.prl\_iter\_property (function), 96
- prlsdkapi.prl\_property (function), 96
- prlsdkapi.prl\_sdk\_call (function), 96
- prlsdkapi.prl\_sdk\_call\_prefix (function), 96
- prlsdkapi.prl\_to\_date\_time (function), 96
- prlsdkapi.PrlIterator (class), 109–110
  - prlsdkapi.PrlIterator.\_\_getitem\_\_ (method), 109
  - prlsdkapi.PrlIterator.\_\_iter\_\_ (method), 109
  - prlsdkapi.PrlIterator.\_\_len\_\_ (method), 109
  - prlsdkapi.PrlIterator.get\_count (method), 109
  - prlsdkapi.PrlIterator.next (method), 109
- prlsdkapi.PrlObject (class), 110
- prlsdkapi.PrlReadOnlyAttrError (class), 110–111
- prlsdkapi.PrlResultIter (class), 111–112
- prlsdkapi.RunningTask (class), 112–113
- prlsdkapi.Server (class), 113–121
  - prlsdkapi.Server.add\_network\_adapter (method), 114
  - prlsdkapi.Server.attach\_to\_lost\_task (method), 114
  - prlsdkapi.Server.begin\_edit\_common\_prefs (method), 114
  - prlsdkapi.Server.begin\_edit\_user\_profile (method), 114
  - prlsdkapi.Server.create\_vm (method), 115
  - prlsdkapi.Server.delete\_network\_adapter (method), 115
  - prlsdkapi.Server.fs\_can\_create\_file (method), 115
  - prlsdkapi.Server.fs\_create\_dir (method), 115
  - prlsdkapi.Server.fs\_remove\_dir (method), 115
  - prlsdkapi.Server.fs\_rename\_dir (method), 116
  - prlsdkapi.Server.get\_dir\_entries (method), 116
  - prlsdkapi.Server.get\_unique\_fs\_entry\_name (method), 116
  - prlsdkapi.Server.get\_user\_info (method), 117
  - prlsdkapi.Server.get\_vm\_list (method), 117
  - prlsdkapi.Server.net\_service (method), 117
  - prlsdkapi.Server.register\_vm (method), 118
  - prlsdkapi.Server.search\_vms (method), 118
  - prlsdkapi.Server.send\_answer (method), 118
  - prlsdkapi.Server.subscribe\_to\_host\_statistics (method), 119
  - prlsdkapi.Server.unsubscribe\_to\_host\_statistics (method), 119
  - prlsdkapi.Server.update\_licence (method), 119

- prlsdkapi.Server.update\_network\_adapter (method), 119
- prlsdkapi.SrvCfgDev (class), 121–122
- prlsdkapi.SrvCfgHdd (class), 122–123
  - prlsdkapi.SrvCfgHdd.get\_part (method), 122
- prlsdkapi.SrvCfgHddPart (class), 123–124
- prlsdkapi.SrvCfgNet (class), 124–126
- prlsdkapi.SrvConfig (class), 126–127
- prlsdkapi.SrvInfo (class), 127–128
- prlsdkapi.UserConfig (class), 128–129
  - prlsdkapi.UserConfig.commit (method), 128
- prlsdkapi.UsrInfo (class), 129–130
- prlsdkapi.Vm (class), 130–146
  - prlsdkapi.Vm.add\_default\_device (method), 131
  - prlsdkapi.Vm.begin\_edit (method), 131
  - prlsdkapi.Vm.change\_access\_rights (method), 132
  - prlsdkapi.Vm.clone (method), 132
  - prlsdkapi.Vm.commit (method), 132
  - prlsdkapi.Vm.create\_boot\_dev (method), 133
  - prlsdkapi.Vm.create\_device (method), 133
  - prlsdkapi.Vm.create\_event (method), 133
  - prlsdkapi.Vm.create\_scr\_res (method), 133
  - prlsdkapi.Vm.create\_share (method), 133
  - prlsdkapi.Vm.create\_unattended\_floppy (method), 133
  - prlsdkapi.Vm.delete (method), 134
  - prlsdkapi.Vm.drop\_suspended\_state (method), 134
  - prlsdkapi.Vm.execution (method), 134
  - prlsdkapi.Vm.generate\_device\_filename (method), 135
  - prlsdkapi.Vm.get\_boot\_dev (method), 135
  - prlsdkapi.Vm.get\_default\_settings (static method), 135
  - prlsdkapi.Vm.get\_device (method), 136
  - prlsdkapi.Vm.get\_devices\_count (method), 136
  - prlsdkapi.Vm.get\_display\_dev (method), 136
  - prlsdkapi.Vm.get\_floppy (method), 136
  - prlsdkapi.Vm.get\_hdd (method), 136
  - prlsdkapi.Vm.get\_net\_adapter (method), 137
  - prlsdkapi.Vm.get\_optical (method), 137
  - prlsdkapi.Vm.get\_parallel\_port (method), 137
  - prlsdkapi.Vm.get\_scr\_res (method), 137
  - prlsdkapi.Vm.get\_serial\_port (method), 137
  - prlsdkapi.Vm.get\_share (method), 138
  - prlsdkapi.Vm.get\_sound\_dev (method), 138
  - prlsdkapi.Vm.get\_usb\_dev (method), 138
  - prlsdkapi.Vm.get\_weak\_objects (method), 138
  - prlsdkapi.Vm.initiate\_dev\_state\_notifications (method), 138
  - prlsdkapi.Vm.install\_tools (method), 138
  - prlsdkapi.Vm.pause (method), 139
  - prlsdkapi.Vm.refresh\_config (method), 139
  - prlsdkapi.Vm.reg (method), 139
  - prlsdkapi.Vm.reg\_event\_handler (method), 139
  - prlsdkapi.Vm.reset (method), 139
  - prlsdkapi.Vm.resume (method), 139
  - prlsdkapi.Vm.send\_utility\_command (method), 140
  - prlsdkapi.Vm.set\_clipboard (method), 140
  - prlsdkapi.Vm.set\_default\_config (method), 140
  - prlsdkapi.Vm.start (method), 140
  - prlsdkapi.Vm.stop (method), 140
  - prlsdkapi.Vm.subscribe\_to\_guest\_statistics (method), 140
  - prlsdkapi.Vm.suspend (method), 140
  - prlsdkapi.Vm.tools\_send\_shutdown (method), 141
  - prlsdkapi.Vm.unreg (method), 141
  - prlsdkapi.Vm.unreg\_event\_handler (method), 141
  - prlsdkapi.Vm.unsubscribe\_from\_guest\_statistics (method), 141
  - prlsdkapi.Vm.update\_config (method), 141
  - prlsdkapi.Vm.validate\_config (method),

- 141
- prlsdkapi.VmAccessRights (*class*), 146–147
- prlsdkapi.VmBaseDev (*class*), 147–149
- prlsdkapi.VmBaseDev.connect (*method*), 147
- prlsdkapi.VmBaseDev.create (*method*), 147
- prlsdkapi.VmBaseDev.create\_image (*method*), 148
- prlsdkapi.VmBaseDev.disconnect (*method*), 148
- prlsdkapi.VmBaseDev.remove (*method*), 148
- prlsdkapi.VmBaseDev.set\_default\_stack\_index (*method*), 148
- prlsdkapi.VmBaseDiskDev (*class*), 149–151
- prlsdkapi.VmBasePortDev (*class*), 151–152
- prlsdkapi.VmBootDev (*class*), 152–154
- prlsdkapi.VmBootDev.remove (*method*), 153
- prlsdkapi.VmCPUStat (*class*), 154–155
- prlsdkapi.VmDiskPartStat (*class*), 155–156
- prlsdkapi.VmDiskStat (*class*), 156–157
- prlsdkapi.VmDiskStat.get\_part\_stat (*method*), 156
- prlsdkapi.VmDisplayDev (*class*), 157–160
- prlsdkapi.VmDisplayDev.\_\_del\_\_ (*method*), 158
- prlsdkapi.VmDisplayDev.capture\_screen\_to\_file (*method*), 158
- prlsdkapi.VmDisplayDev.check\_connected (*method*), 158
- prlsdkapi.VmDisplayDev.connect\_to\_vm (*method*), 158
- prlsdkapi.VmDisplayDev.disconnect\_from\_vm (*method*), 158
- prlsdkapi.VmDisplayDev.engage\_job (*method*), 159
- prlsdkapi.VmDisplayDev.get\_screen\_buffer (*method*), 159
- prlsdkapi.VmDisplayDev.move\_mouse (*method*), 159
- prlsdkapi.VmDisplayDev.send\_key\_event (*method*), 159
- prlsdkapi.VmDisplayDev.send\_keys\_event (*method*), 159
- prlsdkapi.VmDisplayDev.send\_screen\_size\_event (*method*), 159
- prlsdkapi.VmDisplayDev.set\_mouse\_pos (*method*), 160
- prlsdkapi.VmFloppyDev (*class*), 160–162
- prlsdkapi.VmHddDev (*class*), 162–163
- prlsdkapi.VmIFaceStat (*class*), 163–165
- prlsdkapi.VmInfo (*class*), 165–166
- prlsdkapi.VmNetDev (*class*), 166–167
- prlsdkapi.VmNetDev.generate\_mac\_addr (*method*), 166
- prlsdkapi.VmObjEmulation (*class*), 167–168
- prlsdkapi.VmObjEmulation.\_\_ensure\_attr\_\_ (*method*), 168
- prlsdkapi.VmObjEmulation.call\_vm (*method*), 168
- prlsdkapi.VmObjEmulation.indirect\_self (*method*), 168
- prlsdkapi.VmObjEmulation.safe\_call (*method*), 168
- prlsdkapi.VmOpticalDev (*class*), 168–169
- prlsdkapi.VmParallelPortDev (*class*), 169–171
- prlsdkapi.VmProcStat (*class*), 171–172
- prlsdkapi.VmScrRes (*class*), 172–173
- prlsdkapi.VmScrRes.remove (*method*), 173
- prlsdkapi.VmSerialDev (*class*), 173–175
- prlsdkapi.VmShare (*class*), 175–176
- prlsdkapi.VmShare.remove (*method*), 175
- prlsdkapi.VmSoundDev (*class*), 176–178
- prlsdkapi.VmStat (*class*), 178–181
- prlsdkapi.VmStat.get\_cpu\_stat (*method*), 178
- prlsdkapi.VmStat.get\_disk\_stat (*method*), 178
- prlsdkapi.VmStat.get\_iface\_stat (*method*), 179
- prlsdkapi.VmStat.get\_proc\_stat (*method*), 179
- prlsdkapi.VmStat.get\_user\_stat (*method*), 179

179

- prlsdkapi.VmToolsInfo (*class*), 181–182
- prlsdkapi.VmUsbDev (*class*), 182–183
- prlsdkapi.VmUserStat (*class*), 183–184
- prlsdkapi.VmVNCConfig (*class*), 184–185