



Telescope Interface Standard 1.4 Programmer's Reference

Written by [Robert B. Denny](#)

Please conduct discussions regarding this interface and its usage on the [ASCOM-Talk List](#).

[Click here to join ASCOM-Talk](#)

Introduction

This document describes the interface used by low-level telescope "driver" components as part of the Astronomy Common Object Model (ASCOM). Components that implement this interface can provide way for programs to control various telescopes via a standard set of properties and methods. The characteristics of this interface comply with the [ASCOM Quality Guidelines](#), assuring consistent behavior and compatibility with the widest possible variety of Windows Automation clients.

To control a particular telescope type, a program would create an instance of the driver for that telescope type, then use the standard properties and methods described in this document to effect control of that scope. Thus any program or script that uses the standard driver interface automatically gains access to any telescope type for which a driver exists. For more information on ASCOM, see the [ASCOM Home Page](#).

This specification covers a *simple, low-level* telescope control interface for reading and writing coordinates, slewing, and synchronizing. It does not provide for features (such as drive speed controls) that vary widely between mount types, nor for accessories such as focusers, flip mirrors, etc. Those sorts of things will have their own interfaces.

Equatorial coordinates are local topocentric, equinox of the current date, hours for right ascension, and degrees for declination. Thus, "inner loop" issues such as mechanical compensation must be contained within the driver or it's underlying controller(s). Time is in Coordinated Universal Time (UTC).

For best results

This will display best if you have Internet Explorer 5 or later, text size set to "smaller", and the following fonts installed:

- Arial
- Verdana
- Lucida Console

Telescope Release Notes

V1.4 (Release)

This is the 1.4 version of the ASCOM Telescope Interface, the first release as an ASCOM standard. For more information see the [ASCOM Initiative web site](#).

Copyright © 2001, The ASCOM Initiative

Telescope Object

Represents the ASCOM low-level telescope driver interface.

Remarks

Use this interface to perform basic operations on a robotic telescope. The interface is designed to be called from client applications that wish to provide telescope control capabilities. By using this interface, your application will be freed from dealing with the details of serial port control and low-level protocols in use by various telescopes.

Concurrency

This interface does not provide for concurrency control. In the interest of keeping things simple at this level, the architecture assumes that client applications will refrain from performing conflicting operations.

Telescope.AlignmentMode Property

 **Telescope.AlignmentMode** (read-only, [AlignmentModes](#))

The alignment mode of the mount.

Syntax

`Telescope.AlignmentMode`

The property syntax has these parts:

Part	Description
Value (AlignmentModes)	The alignment mode of the mount.

Remarks

German equatorial mounts are distinct because they require a flip at the meridian.

Symbolic Constants

The (symbolic) values for **AlignmentModes** are:

Constant	Value	Description
teleAltAz	0	Altitude-Azimuth mount
teleGermanPolar	2	German equatorial mount

Telescope.Altitude Property

 **Telescope.Altitude** (read-only, Double)

The Altitude (degrees, positive up) of the telescope's current alt/az coordinates

Syntax

Telescope.**Altitude**

The property syntax has these parts:

Part	Description
Value (Double)	The Altitude (degrees, positive up) of the telescope's current alt/az coordinates

Remarks

Telescope.ApertureDiameter Property

 **Telescope.ApertureDiameter** (read-only, Double)

The telescope's effective aperture diameter (metres)

Syntax

Telescope.**ApertureDiameter**

The property syntax has these parts:

Part	Description
Value (Double)	The telescope's effective aperture diameter (metres)

Remarks

Telescope.Azimuth Property

 **Telescope.Azimuth** (read-only, Double)

The azimuth (degrees, North-referenced, positive East/clockwise) of the telescope's current alt/az coordinates.

Syntax

Telescope.**Azimuth**

The property syntax has these parts:

Part	Description
Value (Double)	The azimuth (degrees, North-referenced, positive East/clockwise) of the telescope's current alt/az coordinates.

Remarks

Telescope.CanFindHome Property

 **Telescope.CanFindHome** (read-only, Boolean)

True if this telescope is capable of programmed finding its home position (FindHome() method)

Syntax

Telescope.**CanFindHome**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed finding its home position (FindHome() method)

Remarks

Telescope.CanPark Property

 **Telescope.CanPark** (read-only, Boolean)

True if this telescope is capable of programmed parking (Park() method)

Syntax

Telescope.**CanPark**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed parking (Park() method)

Remarks

Telescope.CanPulseGuide Property

 **Telescope.CanPulseGuide** (read-only, Boolean)

True if this telescope is capable of programmed guiding (PulseGuide() method)

Syntax

Telescope.**CanPulseGuide**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed guiding (PulseGuide() method)

Remarks

Telescope.CanSetPark Property

 **Telescope.CanSetPark** (read-only, Boolean)

True if this telescope is capable of programmed setting of its park position (SetPark() method)

Syntax

Telescope.**CanSetPark**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed setting of its park position (SetPark() method)

Remarks

Telescope.CanSetTracking Property

 **Telescope.CanSetTracking** (read-only, Boolean)

True if the Telescope.Tracking property can be changed, turning telescope tracking on and off.

Syntax

Telescope.**CanSetTracking**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if the Telescope.Tracking property can be changed, turning telescope tracking on and off.

Remarks

Telescope.CanSlew Property

 **Telescope.CanSlew** (read-only, Boolean)

True if this telescope is capable of programmed slewing (synchronous or asynchronous)

Syntax

Telescope.**CanSlew**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed slewing (synchronous or asynchronous)

Remarks

If this is true, then only the synchronous slewing methods are guaranteed to be supported. See the CanSlewAsync property for the asynchronous slewing capability flag.

Telescope.CanSlewAsync Property

 **Telescope.CanSlewAsync** (read-only, Boolean)

True if this telescope is capable of programmed asynchronous slewing

Syntax

Telescope.**CanSlewAsync**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed asynchronous slewing

Remarks

This indicates the the asynchronous slewing methods are supported. If this is True, then CanSlew will also be true.

Telescope.CanSync Property

 **Telescope.CanSync** (read-only, Boolean)

True if this telescope is capable of programmed synching

Syntax

Telescope.**CanSync**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed synching

Remarks

Telescope.CanUnpark Property

 **Telescope.CanUnpark** (read-only, Boolean)

True if this telescope is capable of programmed unparking (Unpark() method)

Syntax

Telescope.**CanUnpark**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if this telescope is capable of programmed unparking (Unpark() method)

Remarks

If this is true, then CanPark will also be true.

Telescope.Connected Property

 **Telescope.Connected** (read-write, Boolean)

True if telescope connected, False otherwise

Syntax

Telescope.**Connected** [= *Boolean*]

The property syntax has these parts:

Part	Description
Value (Boolean)	True if telescope connected, False otherwise

Remarks

Set this property to True to connect to the telescope. Raises an error if there is a problem connecting.

Telescope.Declination Property

 **Telescope.Declination** (read-only, Double)

The declination (degrees) of the telescope's current local topocentric equatorial coordinates

Syntax

Telescope.**Declination**

The property syntax has these parts:

Part	Description
Value (Double)	The declination (degrees) of the telescope's current local topocentric equatorial coordinates

Remarks

Reading the property will raise an error if the value is unavailable.

Telescope.DeclinationRate Property

 **Telescope.DeclinationRate** (read-write, Double)

The declination tracking rate (arcseconds per second, default = 0.0)

Syntax

Telescope.**DeclinationRate** [= *Double*]

The property syntax has these parts:

Part	Description
Value (Double)	The declination tracking rate (arcseconds per second, default = 0.0)

Remarks

The range of values supported is telescope specific.

Telescope.Description Property

 **Telescope.Description** (read-only, String)

The long description of the telescope.

Syntax

Telescope.**Description**

The property syntax has these parts:

Part	Description
Value (String)	The long description of the telescope.

Remarks

This string may contain line endings and may be hundreds to thousands of characters long. It is intended to display detailed information on the telescope itself. See the [DriverInfo property](#) for descriptive info on the driver itself.

Telescope.DriverInfo Property

 **Telescope.DriverInfo** (read-only, String)

Descriptive and version information about this ASCOM Telescope driver

Syntax

Telescope.**DriverInfo**

The property syntax has these parts:

Part	Description
Value (String)	Descriptive and version information about this ASCOM Telescope driver

Remarks

This string may contain line endings and may be hundreds to thousands of characters long. It is intended to display detailed information on the ASCOM driver, including version and copyright data.. See the [Description property](#) for descriptive info on the telescope itself.

Telescope.FocalLength Property

 **Telescope.FocalLength** (read-only, Double)

The telescope's focal length, metres

Syntax

Telescope.**FocalLength**

The property syntax has these parts:

Part	Description
Value (Double)	The telescope's focal length, metres

Remarks

Telescope.Name Property

 **Telescope.Name** (read-only, String)

The short name of the telescope, for display purposes

Syntax

Telescope.**Name**

The property syntax has these parts:

Part	Description
Value (String)	The short name of the telescope, for display purposes

Remarks

Telescope.RightAscension Property

 **Telescope.RightAscension** (read-only, Double)

The right ascension (hours) of the telescope's current local topocentric equatorial coordinates

Syntax

Telescope.**RightAscension**

The property syntax has these parts:

Part	Description
Value (Double)	The right ascension (hours) of the telescope's current local topocentric equatorial coordinates

Remarks

Reading the property will raise an error if the value is unavailable.

Telescope.RightAscensionRate Property

 **Telescope.RightAscensionRate** (read-write, Double)

The right ascension tracking rate *offset* from sidereal (seconds per *sidereal* second, default = 0.0)

Syntax

Telescope.**RightAscensionRate** [= *Double*]

The property syntax has these parts:

Part	Description
Value (Double)	The right ascension tracking rate <i>offset</i> from sidereal (seconds per <i>sidereal</i> second, default = 0.0)

Remarks

The range of values supported is telescope specific.

Use the [Tracking property](#) to enable and disable sidereal tracking (if supported).

Telescope.SiderealTime Property

 **Telescope.SiderealTime** (read-only, Double)

The local apparent sidereal time from the telescope's internal clock

Syntax

Telescope.**SiderealTime**

The property syntax has these parts:

Part	Description
Value (Double)	The local apparent sidereal time from the telescope's internal clock

Remarks

Reading the property will raise an error if the value is unavailable.

Telescope.SiteElevation Property

 **Telescope.SiteElevation** (read-write, Double)

The elevation above mean sea level (metres) of the site at which the telescope is located

Syntax

Telescope.**SiteElevation** [= *Double*]

The property syntax has these parts:

Part	Description
Value (Double)	The elevation above mean sea level (metres) of the site at which the telescope is located

Remarks

Setting this property will raise an error if the given value is outside the range -300 through +10000 metres. Reading the property will raise an error if the value has never been set or is otherwise unavailable.

Telescope.SiteLatitude Property

 **Telescope.SiteLatitude** (read-write, Double)

The *geodetic*(map) latitude (degrees, positive North) of the site at which the telescope is located

Syntax

Telescope.**SiteLatitude** [= *Double*]

The property syntax has these parts:

Part	Description
Value (Double)	The <i>geodetic</i> (map) latitude (degrees, positive North) of the site at which the telescope is located

Remarks

Setting this property will raise an error if the given value is outside the range -90 to +90 degrees. Reading the property will raise an error if the value has never been set or is otherwise unavailable.

Telescope.SiteLongitude Property

 **Telescope.SiteLongitude** (read-write, Double)

The longitude (degrees, positive East) of the site at which the telescope is located

Syntax

Telescope.**SiteLongitude** [= *Double*]

The property syntax has these parts:

Part	Description
Value (Double)	The longitude (degrees, positive East) of the site at which the telescope is located

Remarks

Setting this property will raise an error if the given value is outside the range -180 to +180 degrees. Reading the property will raise an error if the value has never been set or is otherwise unavailable.

Telescope.Slewing Property

 **Telescope.Slewing** (read-only, Boolean)

True if telescope is currently moving to new target coordinates, False at all other times.

Syntax

Telescope.**Slewing**

The property syntax has these parts:

Part	Description
Value (Boolean)	True if telescope is currently moving to new target coordinates, False at all other times.

Remarks

Reading the property will raise an error if the value is unavailable.

Telescope.SlewSettleTime Property

 **Telescope.SlewSettleTime** (read-write, Integer)

Specifies a post-slew settling time (sec.).

Syntax

Telescope.**SlewSettleTime** [= *Integer*]

The property syntax has these parts:

Part	Description
Value (Integer)	Specifies a post-slew settling time (sec.).

Remarks

Adds additional time to slew operations. Synchronous slewing methods will not return, and the Telescope.Slewing property will not become false, until the slew completes and the SlewSettleTime has elapsed. This feature (if supported) may be used with mounts that require extra settling time after a slew.

Telescope.TargetDeclination Property

 **Telescope.TargetDeclination** (read-write, Double)

The local topocentric declination (degrees, positive North) for the target of a slew or sync operation

Syntax

`Telescope.TargetDeclination` [= *Double*]

The property syntax has these parts:

Part	Description
Value (Double)	The local topocentric declination (degrees, positive North) for the target of a slew or sync operation

Remarks

Setting this property will raise an error if the given value is outside the range -90 to +90 degrees. Reading the property will raise an error if the value has never been set or is otherwise unavailable.

Telescope.TargetRightAscension Property

 **Telescope.TargetRightAscension** (read-write, Double)

The local topocentric right ascension (hours) for the target of a slew or sync operation

Syntax

`Telescope.TargetRightAscension` [= *Double*]

The property syntax has these parts:

Part	Description
Value (Double)	The local topocentric right ascension (hours) for the target of a slew or sync operation

Remarks

Setting this property will raise an error if the given value is outside the range 0 to 24 hours. Reading the property will raise an error if the value has never been set or is otherwise unavailable.

Telescope.Tracking Property

 **Telescope.Tracking** (read-write, Boolean)

The state of the telescope's sidereal tracking drive.

Syntax

`Telescope.Tracking [= Boolean]`

The property syntax has these parts:

Part	Description
Value (Boolean)	The state of the telescope's sidereal tracking drive.

Remarks

Telescopes may not support changing the value of this property and thus may not support turning tracking on and off. See the CanSetTracking property.

Telescope.UTCDate Property

 **Telescope.UTCDate** (read-write, Date)

The UTC date/time of the telescope's internal clock

Syntax

`Telescope.UTCDate [= Date]`

The property syntax has these parts:

Part	Description
Value (Date)	The UTC date/time of the telescope's internal clock

Remarks

Reading the property will raise an error if the value has never been set or is otherwise unavailable.

Telescope.AbortSlew() Method

Stops a slew in progress. Valid only after a call to Telescope.SlewToTargetAsync or Telescope.SlewToCoordinatesAsync. Does nothing if no slew is in progress.

Syntax

Telescope.AbortSlew()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Telescope.CommandBlind() Method

Send a string comand directly to the telescope without expecting response data.

Syntax

Telescope.CommandBlind(*Command*)

The method syntax has these parts:

Part	Description
<i>Command</i> (String)	The raw string to be sent to the telescope
Return (Nothing)	Does not return a value.

Remarks

Important

If you use this feature of the Telescope driver interface, your application will be dependent on the low-level protocol used by the particular scope you are connected to. Thus your application will not work with any arbitrary type of telescope.

Raises an error if there is a problem communicating with the telescope.

Telescope.CommandBool() Method

Send a string comand to the telescope, returning a true/false response

Syntax

Telescope.CommandBool(*Command*)

The method syntax has these parts:

Part	Description
<i>Command</i> (String)	The raw string to be sent to the telescope
Return (Boolean)	True if the response indicated true or success, else False.

Remarks

Important

If you use this feature of the Telescope driver interface, your application will be dependent on the low-level protocol used by the particular scope you are connected to. Thus your application will not work with any arbitrary type of telescope.

Raises an error if there is a problem communicating with the telescope.

The returned value is the Automation-compatible Boolean type, True or False. It is the responsibility of the driver implementing this interface to translate raw response data to True/False values for return. If you want to see the raw response string, see

[Telescope.CommandString\(\)](#).

Copyright © 2001, The ASCOM Initiative

Telescope.CommandString() Method

Send a string comand to the telescope, returning the response string

Syntax

Telescope.CommandString(*Command*)

The method syntax has these parts:

Part	Description
<i>Command</i> (String)	
Return (String)	The response data from the telescope resulting from the sent command.

Remarks

Important

If you use this feature of the Telescope driver interface, your application will be dependent on the low-level protocol used by the particular scope you are connected to. Thus your application will not work with any arbitrary type of telescope.

Raises an error if there is a problem communicating with the telescope.

Telescope.FindHome() Method

Locates the telescope's "home" position

Syntax

Telescope.FindHome()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Raises an error if there is a problem.

Telescope.Park() Method

Move the telescope to its park position and stop tracking motion

Syntax

Telescope.Park()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Raises an error if there is a problem communicating with the telescope or if parking fails.

Telescope.PulseGuide() Method

Moves the scope in the given direction for the given interval or time, using the guiding rate of speed (which is dependent on the particular scope).

Syntax

Telescope.PulseGuide(*Direction*, *Duration*)

The method syntax has these parts:

Part	Description
<i>Direction</i> (GuideDirections)	The direction in which the guide-rate motion is to be made
<i>Duration</i> (Long)	The duration of the guide-rate motion (milliseconds)
Return (Nothing)	Does not return a value.

Remarks

This method returns only after the move has completed.

Symbolic Constants

The (symbolic) values for **GuideDirections** are:

Constant	Value	Description
guideNorth	0	North (+ declination/elevation)

guideSouth	1	South (- declination/elevation)
guideEast	2	East (+ right ascension/azimuth)
guideWest	3	West (+right ascension/azimuth)

Telescope.SetPark() Method

Sets the telescope's park position to be its current position

Syntax

Telescope.SetPark()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Raises an error if there is a problem.

Telescope.SetupDialog() Method

Displays a setup dialog, allowing the user to set telescope-specific values such as baud rate, geodetic position, etc.

Syntax

Telescope.SetupDialog()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Telescope.SlewToCoordinates() Method

Move the telescope to the given coordinates, return when slew is complete

Syntax

Telescope.SlewToCoordinates(*RightAscension*, *Declination*)

The method syntax has these parts:

Part	Description
<i>RightAscension</i> (Double)	The destination right ascension (local topocentric, hours). Copied to Telescope.TargetRightAscension.
<i>Declination</i> (Double)	The destination declination (local topocentric, degrees, positive North). Copied to Telescope.TargetDeclination.
Return (Nothing)	Does not return a value.

Remarks

Raises an error if the slew fails.

The slew may fail if the target coordinates are beyond limits imposed within the driver component. Such limits include airmass maximum, mechanical constraints imposed by the mount or attached instruments, building or dome enclosure restrictions, etc. The target coordinates are copied to Telescope.TargetRightAscension and Telescope.TargetDeclination whether or not the slew succeeds.

Telescope.SlewToCoordinatesAsync() Method

Move the telescope to the given coordinates, return immediately after starting the slew.

Syntax

Telescope.SlewToCoordinatesAsync(*RightAscension*, *Declination*)

The method syntax has these parts:

Part	Description
<i>RightAscension</i> (Double)	The destination right ascension (local topocentric, hours). Copied to Telescope.TargetRightAscension.
<i>Declination</i> (Double)	The destination declination (local topocentric, degrees, positive North). Copied to Telescope.TargetDeclination.
Return (Nothing)	Does not return a value.

Remarks

Raises an error if *starting* the slew failed.

Returns immediately after starting the slew. The client may monitor the progress of the slew by reading the Telescope.RightAscension, Telescope.Declination, and Telescope.Slewing properties during the slew. When the slew completes, Telescope.Slewing becomes False.

The slew may fail to start if the target coordinates are beyond limits imposed within the driver component. Such limits include airmass maximum, mechanical constraints imposed by the mount or attached instruments, building or dome enclosure restrictions, etc. The target

coordinates are copied to Telescope.TargetRightAscension and Telescope.TargetDeclination whether or not the slew succeeds.

Copyright © 2001, The ASCOM Initiative

Telescope.SlewToTarget() Method

Move the telescope to the TargetRightAscension and TargetDeclination coordinates, return when slew complete.

Syntax

Telescope.SlewToTarget()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Raises an error if the slew failed.

The slew may fail if the target coordinates are beyond limits imposed within the driver component. Such limits include airmass maximum, mechanical constraints imposed by the mount or attached instruments, building or dome enclosure restrictions, etc.

Telescope.SlewToTargetAsync() Method

Move the telescope to the TargetRightAscension and TargetDeclination coordinates, returns immediately after starting the slew.

Syntax

Telescope.SlewToTargetAsync()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Raises an error if *starting* the slew failed.

Returns immediately after starting the slew. The client may monitor the progress of the slew by reading the Telescope.RightAscension, Telescope.Declination, and Telescope.Slewing properties during the slew. When the slew completes, Telescope.Slewing becomes False.

The slew may fail to start if the target coordinates are beyond limits imposed within the driver component. Such limits include airmass maximum, mechanical constraints imposed by the mount or attached instruments, building or dome enclosure restrictions, etc.

Telescope.SyncToCoordinates() Method

Matches the scope's coordinates to the given coordinates.

Syntax

Telescope.SyncToCoordinates(*RightAscension*, *Declination*)

The method syntax has these parts:

Part	Description
<i>RightAscension</i> (Double)	The corrected right ascension (local topocentric, hours). Copied to Telescope.TargetRightAscension.
<i>Declination</i> (Double)	The corrected declination (local topocentric, degrees, positive North). Copied to Telescope.TargetDeclination.
Return (Nothing)	Does not return a value.

Remarks

Sets Telescope.TargetRightAscension to the given right ascension, and Telescope.TargetDeclination to the given declination. Raises an error if matching fails.

Telescope.SyncToTarget() Method

Matches the telescope's current coordinates to Telescope.TargetRightAscension and Telescope.TargetDeclination

Syntax

Telescope.SyncToTarget()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Raises an error if matching fails.

Telescope.Unpark() Method

Start tracking from the parked position, valid only after Park().

Syntax

Telescope.Unpark()

The method syntax has these parts:

Part	Description
Return (Nothing)	Does not return a value.

Remarks

Raises an error if unparking fails.

Telescope Run-Time Errors

The Telescope object raises trappable errors when it cannot continue its current operation. The Contents tab of this document lists the error messages sorted alphabetically (under Error Messages) and the Index tab lists them by hexadecimal code (under Error Codes). Clicking on either of these entries leads to a page with a brief description of the error condition.

Error codes are Automation/ActiveX compatible 32-bit values, based on FACILITY_ITF (a.k.a. vbObjectError for VBA) with an offset of 400 hex, and are thus compatible with all ActiveX scripting languages, Visual Basic, Visual Basic for Applications (VBA), etc. it is imperative that drivers raise Automation errors with values equal to 80040400 hex and higher.

Drivers are free to choose their error codes and messages, except for one.

All drivers must implement the error "xxx is not implemented in this driver" with error code 80040400 hex. Drivers must report the specific property or method not implemented in the description part of the error object.

(0x80040400) This is the first error message and code

This is a placeholder for the first error message. Each error message added to the list will have a code one greater than the previous. The base is hex 80040400, which is FACILITY_ITF plus 1024, preventing clashes with Visual Basic internal Automation codes.

Copyright © 2001, The ASCOM Initiative