Autoscope

Generated by Doxygen 1.8.16

1 Hierarchical Index 1

1.1 Class Hierarchy 1

2 Class Index 3

2.1 Class List 3

3 File Index 5

3.1 File List 5

4 Class Documentation 7

4.1 Autoscope Class Reference 7

4.1.1 Detailed Description 10

4.1.2 Constructor & Destructor Documentation 10

4.1.2.1 Autoscope() 10

4.1.2.2 ∼Autoscope() 11

4.1.3 Member Function Documentation 11

4.1.3.1 clearTrackedObject() 11

4.1.3.2 configureGui() 11

4.1.3.3 connectToAutoscope() 11

4.1.3.4 deconnectFromAutoscope() 12

4.1.3.5 draw() 12

4.1.3.6 getAltAzi() 12

4.1.3.7 getAutoscopePictureWindow() 12

4.1.3.8 getAutoscopeWindow() 13

4.1.3.9 getCallOrder() 13

4.1.3.10 getScreenHeight() 13

4.1.3.11 getScreenWidth() 14

4.1.3.12 init() 14

4.1.3.13 loadConfiguration() 14

4.1.3.14 moveObserverToObject() 14

4.1.3.15 restoreDefaultConfiguration() 15

4.1.3.16 searchAnObject() 15

4.1.3.17 setIpAddress() 15

4.1.3.18 setPort() 15

4.1.3.19 setTrackObject() 16

4.1.3.20 showGui 16

4.1.3.21 slotConnected 16

4.1.3.22 slotEnablePictureDispaly 16

4.1.3.23 slotTakePicture 16

4.1.3.24 slotTrackObject 17

4.1.3.25 slotUnTrackOject 17

4.1.3.26 trackSearchedObject() 17

4.1.3.27 trackSelectedObject() 17

4.1.3.28 update() 17
4.1.4 Member Data Documentation 18
4.1.4.1 conf 18
4.1.4.2 displayedAtStartup 18
4.1.4.3 EnablePictureDisplay 18
4.1.4.4 font 18
4.1.4.5 gui 18
4.1.4.6 guiIsVisible 18
4.1.4.7 m_autoscopeIp 18
4.1.4.8 m_autoscopePictureWindow 19
4.1.4.9 m_autoscopePort 19
4.1.4.10 m_autoscopeWindow 19
4.1.4.11 m_client 19
4.1.4.12 m_core 19
4.1.4.13 m_screenHeight 20
4.1.4.14 m_screenWidth 20
4.1.4.15 markColor 20
4.1.4.16 markFader 20
4.1.4.17 menuButton 20
4.1.4.18 mvMgr 20
4.1.4.19 newSelected 20
4.1.4.20 objectMgr 21
4.1.4.21 objectPosition 21
4.1.4.22 searchedObject 21
4.1.4.23 searchObjectFound 21
4.1.4.24 selectedObject 21
4.1.4.25 Sun 21
4.1.4.26 takePicture 21
4.1.4.27 trackButton 22
4.1.4.28 trackObject 22
4.1.4.29 unTrackButton 22
4.2 AutoscopePictureWindowForm Class Reference 22
4.2.1 Detailed Description 24
4.2.2 Constructor & Destructor Documentation 24
4.2.2.1 AutoscopePictureWindowForm() 24
4.2.2.2 ∼AutoscopePictureWindowForm() 24
4.2.3 Member Function Documentation 24
4.2.3.1 createDialogContent() 25
4.2.3.2 getGuiHeight() 25
4.2.3.3 getGuiHorizontalPosition() 25
4.2.3.4 getGuiVerticalPosition() 25
4.2.3.5 getGuiWidth() 26

4.2.3.6 retranslate 26
4.2.3.7 setAutoscopeWindow() 26
4.2.3.8 setGuiHorizontalPosition() 26
4.2.3.9 setGuiOpacity() 27
4.2.3.10 setGuiSize() 27
4.2.3.11 setGuiVerticalPosition() 27
4.2.3.12 update() 27
4.2.3.13 updateGuiPosition() 28
4.2.3.14 updateGuiSize() 28
4.2.3.15 updateImage() 28
4.2.4 Member Data Documentation 28
4.2.4.1 m_autoscope 28
4.2.4.2 m_autoscopeWindow 28
4.2.4.3 m_guiHorizontalPosition 29
4.2.4.4 m_guiVerticalPosition 29
4.2.4.5 m_height 29
4.2.4.6 m_screenHeight 29
4.2.4.7 m_screenWidth 29
4.2.4.8 m_width 29
4.2.4.9 ui 29
4.3 AutoscopeStelPluginInterface Class Reference 30
4.3.1 Detailed Description 30
4.3.2 Member Function Documentation 30
4.3.2.1 getExtensionList() 31
4.3.2.2 getPluginInfo() 31
4.3.2.3 getStelModule() 31
4.4 AutoscopeWindowForm Class Reference 31
4.4.1 Detailed Description 34
4.4.2 Constructor & Destructor Documentation 34
4.4.2.1 AutoscopeWindowForm() 34
4.4.2.2 ∼AutoscopeWindowForm() 34
4.4.3 Member Function Documentation 34
4.4.3.1 altitudeChanged 35
4.4.3.2 azimuthChanged 35
4.4.3.3 connectionButtonPressed 35
4.4.3.4 createDialogContent() 35
4.4.3.5 deconnectionButtonPressed 35
4.4.3.6 displayOpacityChanged 36
4.4.3.7 displaySizeChanged 36
4.4.3.8 downloadPictureButtonPressed 36
4.4.3.9 exposureTimeChanged 36
4.4.3.10 getGuiHorizontalPosition() 36

4.4.3.11 getGuiVerticalPosition() . . . 37
4.4.3.12 getScreenSizePercent() . . . 37
4.4.3.13 horizontalDisplayPositionChanged . . . 37
4.4.3.14 ipAddressChanged . . . 37
4.4.3.15 ipPortChanged . . . 37
4.4.3.16 moveToButtonPressed . . . 38
4.4.3.17 numberOfPictureChanged . . . 38
4.4.3.18 outputPictureDirectoryButtonPressed . . . 38
4.4.3.19 outputPictureDirectoryChanged . . . 38
4.4.3.20 retranslate . . . 38
4.4.3.21 searchButtonPressed . . . 39
4.4.3.22 searchObjectChanged . . . 39
4.4.3.23 setAutoscopePictureWindow() . . . 39
4.4.3.24 startButtonPressed . . . 39
4.4.3.25 takePictureButtonPressed . . . 39
4.4.3.26 toggleDisplay() . . . 40
4.4.3.27 toggleDisplayButtonPressed . . . 40
4.4.3.28 trackButtonPressed . . . 40
4.4.3.29 untrackButtonPressed . . . 40
4.4.3.30 update() . . . 40
4.4.3.31 updateGuiPosition() . . . 40
4.4.3.32 updateGuiSize() . . . 41
4.4.3.33 updateIpMessengerText() . . . 41
4.4.3.34 updateMaxMinSlider() . . . 41
4.4.3.35 verticalDisplayPositionChanged . . . 41
4.4.3.36 zoomChanged . . . 41
4.4.4 Member Data Documentation . . . 42
4.4.4.1 m_autoscope . . . 42
4.4.4.2 m_autoscopePictureWindow . . . 42
4.4.4.3 m_guiHorizontalPosition . . . 42
4.4.4.4 m_guiVerticalPosition . . . 42
4.4.4.5 m_height . . . 42
4.4.4.6 m_screenHeight . . . 43
4.4.4.7 m_screenWidth . . . 43
4.4.4.8 m_width . . . 43
4.4.4.9 picturedirectoryPath . . . 43
4.4.4.10 pictureWindowIsVisible . . . 43
4.4.4.11 screenSizePercent . . . 43
4.4.4.12 ui . . . 43
4.5 TcpClient Class Reference . . . 44
4.5.1 Detailed Description . . . 45
4.5.2 Constructor & Destructor Documentation . . . 45

4.5.2.1 TcpClient() [1/2] 45
4.5.2.2 TcpClient() [2/2] 45
4.5.2.3 ∼TcpClient() 45
4.5.3 Member Function Documentation 46
4.5.3.1 connected_handler 46
4.5.3.2 host_found_handler 46
4.5.4 Member Data Documentation 46
4.5.4.1 m_host_address 46
4.5.4.2 m_port 46
4.6 TcpServer Class Reference 47
4.6.1 Detailed Description 48
4.6.2 Constructor & Destructor Documentation 48
4.6.2.1 TcpServer() [1/3] 48
4.6.2.2 TcpServer() [2/3] 48
4.6.2.3 TcpServer() [3/3] 48
4.6.2.4 ∼TcpServer() 48
4.6.3 Member Function Documentation 49
4.6.3.1 create_socket 49
4.6.3.2 get_client() 49
4.6.3.3 set_address() 49
4.6.3.4 set_port() 49
4.6.3.5 start() 50
4.6.4 Member Data Documentation 50
4.6.4.1 m_host_address 50
4.6.4.2 m_port 50
4.6.4.3 m_tcp_client 50

5 File Documentation **51**
5.1 src/Autoscope.cpp File Reference 51
5.2 src/Autoscope.hpp File Reference 51
5.2.1 Detailed Description 52
5.3 src/gui/AutoscopePictureWindowForm.cpp File Reference 53
5.4 src/gui/AutoscopePictureWindowForm.hpp File Reference 53
5.4.1 Detailed Description 54
5.5 src/gui/AutoscopeWindowForm.cpp File Reference 54
5.6 src/gui/AutoscopeWindowForm.hpp File Reference 54
5.6.1 Detailed Description 55
5.7 src/network/tcp_client.cpp File Reference 55
5.8 src/network/tcp_client.hpp File Reference 56
5.8.1 Detailed Description 56
5.9 src/network/tcp_server.cpp File Reference 57
5.10 src/network/tcp_server.hpp File Reference 57

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

QObject
- AutoscopeStelPluginInterface . . . 30

QTcpServer
- TcpServer . . . 47

QTcpSocket
- TcpClient . . . 44

StelDialog
- AutoscopePictureWindowForm . . . 22
- AutoscopeWindowForm . . . 31

StelModule
- Autoscope . . . 7

StelPluginInterface
- AutoscopeStelPluginInterface . . . 30

Chapter 2

Class Index

2.1 Class List

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

Autoscope
This class is use to retrieve and compute data for sending to the telescope, and call user interfaces windows 7
AutoscopePictureWindowForm
This class is use to build a picture display interface to show to the user the last taken picture . 22
AutoscopeStelPluginInterface
This class is used by Qt to manage a plug-in interface 30
AutoscopeWindowForm
This class is use to build a configuration interface between the user and the plugin 31
TcpClient
Allow to make usage of QTcpSocket more suitable for development 44
TcpServer
Allow to make usage of QTcpServer more suitable for development 47

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

src/Autoscope.cpp 51
src/Autoscope.hpp
Main plugin class 51
src/gui/AutoscopePictureWindowForm.cpp 53
src/gui/AutoscopePictureWindowForm.hpp
Header file including the picture display window definition 53
src/gui/AutoscopeWindowForm.cpp 54
src/gui/AutoscopeWindowForm.hpp
Header file including configuration window definition 54
src/network/tcp_client.cpp 55
src/network/tcp_client.hpp
Header file including TCP client definition 56
src/network/tcp_server.cpp 57
src/network/tcp_server.hpp 57

Chapter 4

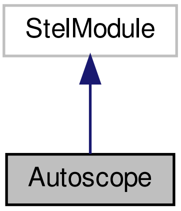
Class Documentation

4.1 Autoscope Class Reference

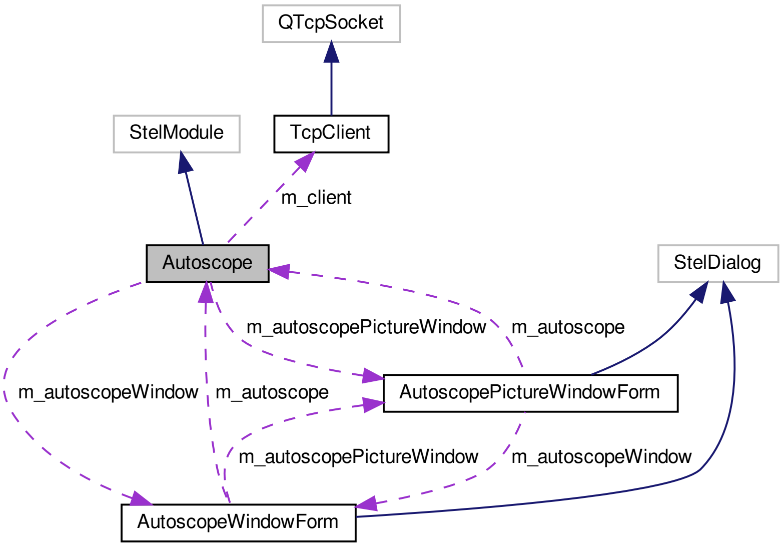
This class is use to retrieve and compute data for sending to the telescope, and call user interfaces windows.

```
#include <Autoscope.hpp>
```

Inheritance diagram for Autoscope:



Collaboration diagram for Autoscope:



Public Slots

- void showGui (void)

  Enable the configuration menu.
- void slotTrackObject (void)

  Start tracking the selected object.
- void slotUnTrackOject (void)

  Stop tracking the tracked object.
- void slotEnablePictureDispaly (void)

  Toggle the visibility of the picture display.
- void slotTakePicture (void)

  Send a command to the Autoscope to take a picture.
- void slotConnected (void)

  Triggered when the plugin succeed to connect to the Autoscope.

Public Member Functions

- Autoscope ()

  Builder of the Autoscope class.
- virtual ∼Autoscope ()

  Destroyer of the Autoscope class.
- virtual void init ()

  Init method is inherited form StelModule class and is use to initialize components sush as StelButton.
- virtual void update (double)

  Update method is inherited form StelModule class and is use to update components or methods. It's automatically called by StelModuleMgr.
- virtual void draw (StelCore ∗core)

Draw method is inherited form StelModule class and is use to draw components on the user interface.

- virtual double getCallOrder (StelModuleActionName actionName) const

  getCallOrder method
- virtual bool configureGui (bool show)

  ConfigureGui method is inherited from StelModule class and is use to show the user interface through the plugin manager.
- void loadConfiguration ()

  Method used to retrieve the configuration of the plugin.
- void restoreDefaultConfiguration ()

  Method used to restore the default configuration of the plugin.
- void getAltAzi (StelObjectP object)

  Method used to retrieve the altitude and the azimuth of an object.
- QString searchAnObject (QString objectName)

  Method used to search an object by it's name.
- void setTrackObject (StelObjectP object)

  Method used to modify the object that the Autoscope should track.
- void trackSelectedObject (void)

  Method used to tarck the object selected by the user.
- QString trackSearchedObject (void)

  Method used to tarck the object search by the user.
- void clearTrackedObject (void)

  Method used to clear the current tracked object.
- void moveObserverToObject (StelObjectP object)

  Method used to move the view in Stellarium to an object.
- AutoscopeWindowForm ∗ getAutoscopeWindow (void)

  Getter which allow any class which have an instance of Autoscope class to retrieve an instance of the configuration window.
- AutoscopePictureWindowForm ∗ getAutoscopePictureWindow (void)

  Getter which allow any class which have an instance of Autoscope class to retrieve an instance of the picture window.
- int getScreenWidth (void)

  Getter allow any class which have an instance of Autoscope class to retrieve the width of the screen.
- int getScreenHeight (void)

  Getter allow any class which have an instance of Autoscope class to retrieve the height of the screen.
- void setIpAddress (QString addr)

  Setter used to set the IP address of the Autoscope.
- void setPort (int port)

  Setter used to set the port of the Autoscope.
- void connectToAutoscope (void)

  Method used to connect the plugin to the Autoscope.
- void deconnectFromAutoscope (void)

  Method used to disconnect the plugin from the Autoscope.

Public Attributes

- StelMovementMgr ∗ mvMgr

  An instance of Stellarium movement manager and it's use to move the view in Stellarium.
- StelObjectMgr ∗ objectMgr

  An instance of Stellarium object manager and it's use to retrieve selected object or search object by name.

Private Attributes

- QSettings ∗ conf
- StelGui ∗ gui
- AutoscopeWindowForm ∗ m_autoscopeWindow

    An instance of AutoscopeWindowForm class.
- AutoscopePictureWindowForm ∗ m_autoscopePictureWindow

    An instance of AutoscopePictureWindowForm class.
- StelCore ∗ m_core

    An instance of StelCore.
- Vec3f markColor
- LinearFader markFader
- bool displayedAtStartup
- bool guiIsVisible = false
- StelButton ∗ menuButton
- StelButton ∗ trackButton
- StelButton ∗ unTrackButton
- StelButton ∗ EnablePictureDisplay
- StelButton ∗ takePicture
- QFont font
- StelObjectP Sun
- StelObjectP trackObject
- StelObjectP selectedObject
- StelObjectP searchedObject
- Vec3d objectPosition
- QList< StelObjectP > newSelected
- bool searchObjectFound = false
- int m_screenWidth
- int m_screenHeight
- TcpClient ∗ m_client

    An instance of TcpClient class.
- QHostAddress m_autoscopeIp
- int m_autoscopePort = 4444

4.1.1 Detailed Description

This class is use to retrieve and compute data for sending to the telescope, and call user interfaces windows.

4.1.2 Constructor & Destructor Documentation

4.1.2.1 Autoscope()

```
Autoscope::Autoscope ( )
```

Builder of the Autoscope class.

4.1.2.2 ∼Autoscope()

```
Autoscope::∼Autoscope ( )  [virtual]
```

Destroyer of the Autoscope class.

4.1.3 Member Function Documentation

4.1.3.1 clearTrackedObject()

```
void Autoscope::clearTrackedObject (
             void  )
```

Method used to clear the current tracked object.

4.1.3.2 configureGui()

```
bool Autoscope::configureGui (
             bool show )  [virtual]
```

ConfigureGui method is inherited from StelModule class and is use to show the user interface through the plugin manager.

Parameters

show	If show is true, set the gui visible, else set it invisible

Returns

true if it doesn't crash!

See also

StelModule

4.1.3.3 connectToAutoscope()

```
void Autoscope::connectToAutoscope (
             void  )
```

Method used to connect the plugin to the Autoscope.

4.1.3.4 deconnectFromAutoscope()

```
void Autoscope::deconnectFromAutoscope (
             void  )
```

Method used to disconnect the plugin from the Autoscope.

4.1.3.5 draw()

```
void Autoscope::draw (
             StelCore * core )  [virtual]
```

Draw method is inherited form StelModule class and is use to draw components on the user interface.

Parameters

core	The object given by StelModuleMgr to draw forms

See also

StelModule

4.1.3.6 getAltAzi()

```
void Autoscope::getAltAzi (
             StelObjectP object )
```

Method used to retrieve the altitude and the azimuth of an object.

Parameters

object	The object that we want to retrieve data from

4.1.3.7 getAutoscopePictureWindow()

```
AutoscopePictureWindowForm* Autoscope::getAutoscopePictureWindow (
             void  )  [inline]
```

Getter which allow any class which have an instance of Autoscope class to retrieve an instance of the picture window.

Returns

An instance of the picture window

4.1.3.8 getAutoscopeWindow()

```
AutoscopeWindowForm* Autoscope::getAutoscopeWindow (
             void  )  [inline]
```

Getter which allow any class which have an instance of Autoscope class to retrieve an instance of the configuration window.

Returns

An instance of the configuration window

4.1.3.9 getCallOrder()

```
double Autoscope::getCallOrder (
             StelModuleActionName actionName ) const  [virtual]
```

getCallOrder method

Parameters

actionName	

See also

StelModule

Returns

4.1.3.10 getScreenHeight()

```
int Autoscope::getScreenHeight (
             void  )  [inline]
```

Getter allow any class which have an instance of Autoscope class to retrieve the height of the screen.

Returns

The height of the screen

4.1.3.11 getScreenWidth()

```
int Autoscope::getScreenWidth (
             void ) [inline]
```

Getter allow any class which have an instance of Autoscope class to retrieve the width of the screen.

Returns

The width of the screen

4.1.3.12 init()

```
void Autoscope::init ( ) [virtual]
```

Init method is inherited form StelModule class and is use to initialize components sush as StelButton.

See also

StelModule

4.1.3.13 loadConfiguration()

```
void Autoscope::loadConfiguration ( )
```

Method used to retrieve the configuration of the plugin.

4.1.3.14 moveObserverToObject()

```
void Autoscope::moveObserverToObject (
             StelObjectP object )
```

Method used to move the view in Stellarium to an object.

Parameters

object	

4.1.3.15 restoreDefaultConfiguration()

```
void Autoscope::restoreDefaultConfiguration ( )
```

Method used to restore the default configuration of the plugin.

4.1.3.16 searchAnObject()

```
QString Autoscope::searchAnObject (
             QString objectName )
```

Method used to search an object by it's name.

Parameters

objectName	The name of the object

Returns

A string to display in the searchMessenger

4.1.3.17 setIpAddress()

```
void Autoscope::setIpAddress (
             QString addr )  [inline]
```

Setter used to set the IP address of the Autoscope.

Parameters

addr	The IP address of the Autoscope

4.1.3.18 setPort()

```
void Autoscope::setPort (
             int port )  [inline]
```

Setter used to set the port of the Autoscope.

Parameters

port	The port of the Autoscope

4.1.3.19 setTrackObject()

```
void Autoscope::setTrackObject (
            StelObjectP object )
```

Method used to modify the object that the Autoscope should track.

Parameters

object	The object to track

4.1.3.20 showGui

```
void Autoscope::showGui (
            void ) [slot]
```

Enable the configuration menu.

4.1.3.21 slotConnected

```
void Autoscope::slotConnected (
            void ) [slot]
```

Triggered when the plugin succeed to connect to the Autoscope.

4.1.3.22 slotEnablePictureDispaly

```
void Autoscope::slotEnablePictureDispaly (
            void ) [slot]
```

Toggle the visibility of the picture display.

4.1.3.23 slotTakePicture

```
void Autoscope::slotTakePicture (
            void ) [slot]
```

Send a command to the Autoscope to take a picture.

4.1.3.24 slotTrackObject

```
void Autoscope::slotTrackObject (
             void  )  [slot]
```

Start tracking the selected object.

4.1.3.25 slotUnTrackOject

```
void Autoscope::slotUnTrackOject (
             void  )  [slot]
```

Stop tracking the tracked object.

4.1.3.26 trackSearchedObject()

```
QString Autoscope::trackSearchedObject (
             void  )
```

Method used to tarck the object search by the user.

Returns

A string to display in the searchMessenger

4.1.3.27 trackSelectedObject()

```
void Autoscope::trackSelectedObject (
             void  )
```

Method used to tarck the object selected by the user.

4.1.3.28 update()

```
void Autoscope::update (
             double t )  [virtual]
```

Update method is inherited form StelModule class and is use to update components or methods. It's automatically called by StelModuleMgr.

See also

StelModule

4.1.4 Member Data Documentation

4.1.4.1 conf

```
QSettings* Autoscope::conf  [private]
```

An instance of QSettings

4.1.4.2 displayedAtStartup

```
bool Autoscope::displayedAtStartup  [private]
```

4.1.4.3 EnablePictureDisplay

```
StelButton* Autoscope::EnablePictureDisplay  [private]
```

An instance of StelButton

4.1.4.4 font

```
QFont Autoscope::font  [private]
```

An instance of Qfont

4.1.4.5 gui

```
StelGui* Autoscope::gui  [private]
```

An instance of StelGui

4.1.4.6 guiIsVisible

```
bool Autoscope::guiIsVisible = false  [private]
```

A flag which represent the visibility of the configuration window

4.1.4.7 m_autoscopeIp

```
QHostAddress Autoscope::m_autoscopeIp  [private]
```

The Autoscope IP address

4.1.4.8 m_autoscopePictureWindow

```
AutoscopePictureWindowForm* Autoscope::m_autoscopePictureWindow  [private]
```

An instance of AutoscopePictureWindowForm class.

See also

AutoscopePictureWindowForm

4.1.4.9 m_autoscopePort

```
int Autoscope::m_autoscopePort = 4444  [private]
```

The Autoscope IP port

4.1.4.10 m_autoscopeWindow

```
AutoscopeWindowForm* Autoscope::m_autoscopeWindow  [private]
```

An instance of AutoscopeWindowForm class.

See also

AutoscopeWindowForm

4.1.4.11 m_client

```
TcpClient* Autoscope::m_client  [private]
```

An instance of TcpClient class.

See also

TcpClient

4.1.4.12 m_core

```
StelCore* Autoscope::m_core  [private]
```

An instance of StelCore.

4.1.4.13 m_screenHeight

```
int Autoscope::m_screenHeight  [private]
```

The screen height

4.1.4.14 m_screenWidth

```
int Autoscope::m_screenWidth  [private]
```

The screen width

4.1.4.15 markColor

```
Vec3f Autoscope::markColor  [private]
```

4.1.4.16 markFader

```
LinearFader Autoscope::markFader  [private]
```

4.1.4.17 menuButton

```
StelButton* Autoscope::menuButton  [private]
```

An instance of StelButton

4.1.4.18 mvMgr

```
StelMovementMgr* Autoscope::mvMgr
```

An instance of Stellarium movement manager and it's use to move the view in Stellarium.

See also

StelMovementMgr

4.1.4.19 newSelected

```
QList<StelObjectP> Autoscope::newSelected  [private]
```

A list of StelObjectP

4.1.4.20 objectMgr

```
StelObjectMgr* Autoscope::objectMgr
```

An instance of Stellarium object manager and it's use to retrieve selected object or search object by name.

See also

StelObjectMgr

4.1.4.21 objectPosition

```
Vec3d Autoscope::objectPosition  [private]
```

An instance of Vec3d

4.1.4.22 searchedObject

```
StelObjectP Autoscope::searchedObject  [private]
```

An instance of StelObjectP

4.1.4.23 searchObjectFound

```
bool Autoscope::searchObjectFound = false  [private]
```

A flag which indicates if the searched object has been found

4.1.4.24 selectedObject

```
StelObjectP Autoscope::selectedObject  [private]
```

An instance of StelObjectP

4.1.4.25 Sun

```
StelObjectP Autoscope::Sun  [private]
```

An instance of StelObjectP

4.1.4.26 takePicture

```
StelButton* Autoscope::takePicture  [private]
```

An instance of StelButton

4.1.4.27 trackButton

```
StelButton* Autoscope::trackButton  [private]
```

An instance of StelButton

4.1.4.28 trackObject

```
StelObjectP Autoscope::trackObject  [private]
```

An instance of StelObjectP

4.1.4.29 unTrackButton

```
StelButton* Autoscope::unTrackButton  [private]
```

An instance of StelButton

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

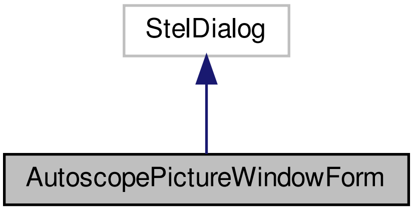
- src/Autoscope.hpp
- src/Autoscope.cpp

4.2 AutoscopePictureWindowForm Class Reference

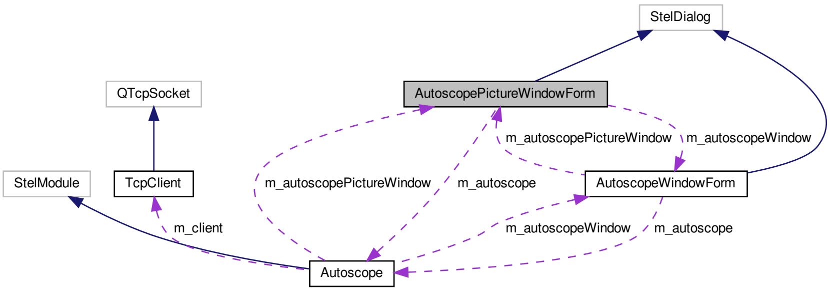
This class is use to build a picture display interface to show to the user the last taken picture.

```
#include <AutoscopePictureWindowForm.hpp>
```

Inheritance diagram for AutoscopePictureWindowForm:



Collaboration diagram for AutoscopePictureWindowForm:



Public Slots

- void retranslate ()

  Inherited from the StelDialog class.

Public Member Functions

- AutoscopePictureWindowForm ()

  Builder of the AutoscopePictureWindowForm class.
- ∼AutoscopePictureWindowForm ()

  Destroyer of the AutoscopePictureWindowForm class.
- void update ()

  Method used to update calculation and displayed components.
- int getGuiHorizontalPosition (void)

  Getter used to retrieve the horizontal gui position.
- int getGuiVerticalPosition (void)

  Getter used to retrieve the vertical gui position.
- void setGuiHorizontalPosition (int)

  Setter used to set the horizontal gui position.
- void setGuiVerticalPosition (int)

  Setter used to set the vertical gui position.
- int getGuiWidth (void)

  Getter used to retrieve the gui width.
- int getGuiHeight (void)

  Getter used to retrieve the gui height.
- void setGuiSize (int)

  Setter used to set the gui size in percent of the screen.
- void updateGuiSize (void)

  Method used to update the size of the gui.
- void setGuiOpacity (double)

  Setter used to set the gui opacity.
- void updateGuiPosition (void)

  Method used to update the gui position.
- void updateImage (QPixmap image)

  Method used to update the image of the gui.
- void setAutoscopeWindow (AutoscopeWindowForm ∗autoscopeWindow)

  Setter used to initialize an instance of AutoscopeWindowForm class.

Protected Member Functions

- void createDialogContent ()

  This method is inherited form the StelDialog class and is use to create the content of the dialog box.

Private Attributes

- Ui_AutoscopePictureWindowForm ∗ ui
- Autoscope ∗ m_autoscope
- AutoscopeWindowForm ∗ m_autoscopeWindow
- int m_width = 192
- int m_height = 108
- int m_screenWidth
- int m_screenHeight
- int m_guiHorizontalPosition = 0
- int m_guiVerticalPosition = 0

4.2.1 Detailed Description

This class is use to build a picture display interface to show to the user the last taken picture.

4.2.2 Constructor & Destructor Documentation

4.2.2.1 AutoscopePictureWindowForm()

```
AutoscopePictureWindowForm::AutoscopePictureWindowForm ( )
```

Builder of the AutoscopePictureWindowForm class.

4.2.2.2 ∼AutoscopePictureWindowForm()

```
AutoscopePictureWindowForm::∼AutoscopePictureWindowForm ( )
```

Destroyer of the AutoscopePictureWindowForm class.

4.2.3 Member Function Documentation

4.2.3.1 createDialogContent()

```
void AutoscopePictureWindowForm::createDialogContent ( )  [protected]
```

This method is inherited form the StelDialog class and is use to create the content of the dialog box.

See also

StelDialog

4.2.3.2 getGuiHeight()

```
int AutoscopePictureWindowForm::getGuiHeight (
            void  )
```

Getter used to retrieve the gui height.

Returns

The gui height

4.2.3.3 getGuiHorizontalPosition()

```
int AutoscopePictureWindowForm::getGuiHorizontalPosition (
            void  )
```

Getter used to retrieve the horizontal gui position.

Returns

The horizontal gui position

4.2.3.4 getGuiVerticalPosition()

```
int AutoscopePictureWindowForm::getGuiVerticalPosition (
            void  )
```

Getter used to retrieve the vertical gui position.

Returns

The vertical gui position

4.2.3.5 getGuiWidth()

```
int AutoscopePictureWindowForm::getGuiWidth (
             void  )
```

Getter used to retrieve the gui width.

Returns

The gui width

4.2.3.6 retranslate

```
void AutoscopePictureWindowForm::retranslate ( )  [slot]
```

Inherited from the StelDialog class.

See also

StelDialog

4.2.3.7 setAutoscopeWindow()

```
void AutoscopePictureWindowForm::setAutoscopeWindow (
             AutoscopeWindowForm * autoscopeWindow )  [inline]
```

Setter used to initialize an instance of AutoscopeWindowForm class.

Parameters

autoscopeWindow	an instance of AutoscopeWindowForm class

4.2.3.8 setGuiHorizontalPosition()

```
void AutoscopePictureWindowForm::setGuiHorizontalPosition (
             int i )
```

Setter used to set the horizontal gui position.

Parameters

The	horizontal gui position in pixel

4.2.3.9 setGuiOpacity()

```
void AutoscopePictureWindowForm::setGuiOpacity (
             double opacity )
```

Setter used to set the gui opacity.

Parameters

the	gui opacity

4.2.3.10 setGuiSize()

```
void AutoscopePictureWindowForm::setGuiSize (
             int percent )
```

Setter used to set the gui size in percent of the screen.

Parameters

The	gui size in percent of the screen

4.2.3.11 setGuiVerticalPosition()

```
void AutoscopePictureWindowForm::setGuiVerticalPosition (
             int i )
```

Setter used to set the vertical gui position.

Parameters

The	vertical gui position in pixel

4.2.3.12 update()

```
void AutoscopePictureWindowForm::update ( )
```

Method used to update calculation and displayed components.

4.2.3.13 updateGuiPosition()

```
void AutoscopePictureWindowForm::updateGuiPosition (
            void )
```

Method used to update the gui position.

4.2.3.14 updateGuiSize()

```
void AutoscopePictureWindowForm::updateGuiSize (
            void )
```

Method used to update the size of the gui.

4.2.3.15 updateImage()

```
void AutoscopePictureWindowForm::updateImage (
            QPixmap image )
```

Method used to update the image of the gui.

Parameters

image	

4.2.4 Member Data Documentation

4.2.4.1 m_autoscope

```
Autoscope* AutoscopePictureWindowForm::m_autoscope [private]
```

An instance of the Autoscope class

4.2.4.2 m_autoscopeWindow

```
AutoscopeWindowForm* AutoscopePictureWindowForm::m_autoscopeWindow [private]
```

An instance of the AutoscopeWindowForm class

4.2.4.3 m_guiHorizontalPosition

```
int AutoscopePictureWindowForm::m_guiHorizontalPosition = 0 [private]
```

The horizontal position of the dialog box

4.2.4.4 m_guiVerticalPosition

```
int AutoscopePictureWindowForm::m_guiVerticalPosition = 0 [private]
```

The vertical position of the dialog box

4.2.4.5 m_height

```
int AutoscopePictureWindowForm::m_height = 108 [private]
```

The height of the dialog box

4.2.4.6 m_screenHeight

```
int AutoscopePictureWindowForm::m_screenHeight [private]
```

The height of the screen

4.2.4.7 m_screenWidth

```
int AutoscopePictureWindowForm::m_screenWidth [private]
```

The width of the screen

4.2.4.8 m_width

```
int AutoscopePictureWindowForm::m_width = 192 [private]
```

The width of the dialog box

4.2.4.9 ui

```
Ui_AutoscopePictureWindowForm* AutoscopePictureWindowForm::ui [private]
```

An instance of the Ui_AutoscopePictureWindowForm class

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

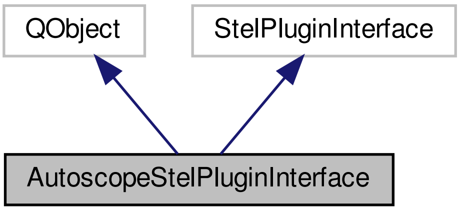
- src/gui/AutoscopePictureWindowForm.hpp
- src/gui/AutoscopePictureWindowForm.cpp

4.3 AutoscopeStelPluginInterface Class Reference

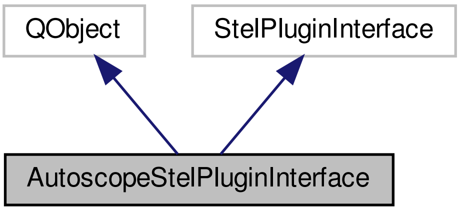
This class is used by Qt to manage a plug-in interface.

```
#include <Autoscope.hpp>
```

Inheritance diagram for AutoscopeStelPluginInterface:



Collaboration diagram for AutoscopeStelPluginInterface:



Public Member Functions

- virtual StelModule ∗ getStelModule () const
- virtual StelPluginInfo getPluginInfo () const
- virtual QObjectList getExtensionList () const

4.3.1 Detailed Description

This class is used by Qt to manage a plug-in interface.

4.3.2 Member Function Documentation

4.3.2.1 getExtensionList()

```
virtual QObjectList AutoscopeStelPluginInterface::getExtensionList ( ) const  [inline], [virtual]
```

4.3.2.2 getPluginInfo()

```
StelPluginInfo AutoscopeStelPluginInterface::getPluginInfo ( ) const  [virtual]
```

4.3.2.3 getStelModule()

```
StelModule * AutoscopeStelPluginInterface::getStelModule ( ) const  [virtual]
```

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

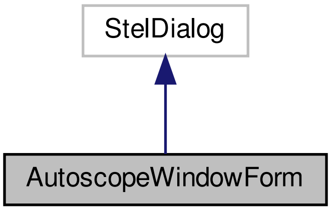
- src/Autoscope.hpp
- src/Autoscope.cpp

4.4 AutoscopeWindowForm Class Reference

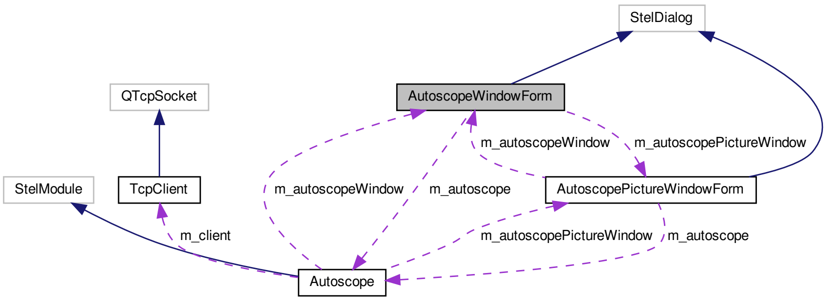
This class is use to build a configuration interface between the user and the plugin.

```
#include <AutoscopeWindowForm.hpp>
```

Inheritance diagram for AutoscopeWindowForm:



Collaboration diagram for AutoscopeWindowForm:



Public Slots

- void retranslate ()

  Inherited from the StelDialog class.
- void startButtonPressed (void)

  Start initialization of the telescope.
- void trackButtonPressed (void)

  Start tracking the selected object.
- void untrackButtonPressed (void)

  Stop tracking the tracked object.
- void takePictureButtonPressed (void)

  Send a command to the telescope to take a picture.
- void toggleDisplayButtonPressed (void)

  Toggle the visibility of the picture display.

Public Member Functions

- AutoscopeWindowForm ()

  Builder of the AutoscopeWindowForm class.
- ∼AutoscopeWindowForm ()

  Destroyer of the AutoscopeWindowForm class.
- void update ()

  Method used to update calculation and displayed components.
- void setAutoscopePictureWindow (AutoscopePictureWindowForm ∗autoscopePictureWindow)

  Setter used to initialize an instance of AutoscopePictureWindowForm class.
- int getGuiHorizontalPosition (void)

  Getter used to retrieve the horizontal gui position.
- int getGuiVerticalPosition (void)

  Getter used to retrieve the vertical gui position.
- int getScreenSizePercent (void)

  Getter used to retrieve the value of the display size editor.
- void updateGuiSize (void)

  Method used to update the gui size.
- void updateGuiPosition (void)

  Method used to update the gui position.
- void toggleDisplay (void)

  Method used to toggle the visibility of the picture display.
- void updateIpMessengerText (QString)

  Method used to update the IP messenger text.

Protected Member Functions

- void createDialogContent ()

  This method is inherited form the StelDialog class and it's use to create the content of the dialog box.

Private Slots

- void moveToButtonPressed (void)

  Move to an azimuth and an altitude.
- void azimuthChanged (double)

  Update azimuth value.
- void altitudeChanged (double)

  Update altitude value.
- void searchObjectChanged (QString)

  Update searched object.
- void searchButtonPressed (void)

  Move to the searched object and track it.
- void zoomChanged (int)

  Update zoom value.
- void exposureTimeChanged (double)

  Update exposure time.
- void numberOfPictureChanged (int)

  Update the number of picture taken at each time.
- void displaySizeChanged (int)

  Update the size value of the picture display.
- void horizontalDisplayPositionChanged (int)

  Update the horizontal position value of the picture display.
- void verticalDisplayPositionChanged (int)

  Update the vertical position value of the picture display.
- void displayOpacityChanged (int)

  Update the opacity value of the picture display.
- void outputPictureDirectoryChanged (void)

  Update the path to put picture into.
- void outputPictureDirectoryButtonPressed (void)

  Show directory browser to select the directory to put picture into.
- void downloadPictureButtonPressed (void)

  Download the last picture taken.
- void ipAddressChanged (QString)

  Update the IP address of the telescope.
- void ipPortChanged (int)

  Update the IP port of the telescope.
- void connectionButtonPressed (void)

  Connect to the IP address and port.
- void deconnectionButtonPressed (void)

  Disconnect form the telescope.

Private Member Functions

- void updateMaxMinSlider (void)

  Method used to update max and min value of the position slider.

Private Attributes

- Ui_AutoscopeWindowForm ∗ ui
- int m_width = 600
- int m_height = 300
- int m_screenWidth
- int m_screenHeight
- int m_guiHorizontalPosition
- int m_guiVerticalPosition
- int screenSizePercent
- bool pictureWindowIsVisible = false
- Autoscope ∗ m_autoscope

  An instance of Autoscope class.
- AutoscopePictureWindowForm ∗ m_autoscopePictureWindow

  An instance of AutoscopePictureWindowForm class.
- QString picturedirectoryPath = ""

4.4.1 Detailed Description

This class is use to build a configuration interface between the user and the plugin.

4.4.2 Constructor & Destructor Documentation

4.4.2.1 AutoscopeWindowForm()

```
AutoscopeWindowForm::AutoscopeWindowForm ( )
```

Builder of the AutoscopeWindowForm class.

4.4.2.2 ∼AutoscopeWindowForm()

```
AutoscopeWindowForm::∼AutoscopeWindowForm ( )
```

Destroyer of the AutoscopeWindowForm class.

4.4.3 Member Function Documentation

4.4.3.1 altitudeChanged

```
void AutoscopeWindowForm::altitudeChanged (
             double d )  [private], [slot]
```

Update altitude value.

4.4.3.2 azimuthChanged

```
void AutoscopeWindowForm::azimuthChanged (
             double d )  [private], [slot]
```

Update azimuth value.

4.4.3.3 connectionButtonPressed

```
void AutoscopeWindowForm::connectionButtonPressed (
             void  )  [private], [slot]
```

Connect to the IP address and port.

4.4.3.4 createDialogContent()

```
void AutoscopeWindowForm::createDialogContent ( )  [protected]
```

This method is inherited form the StelDialog class and it's use to create the content of the dialog box.

See also

StelDialog

4.4.3.5 deconnectionButtonPressed

```
void AutoscopeWindowForm::deconnectionButtonPressed (
             void  )  [private], [slot]
```

Disconnect form the telescope.

4.4.3.6 displayOpacityChanged

```
void AutoscopeWindowForm::displayOpacityChanged (
             int opacity )  [private], [slot]
```

Update the opacity value of the picture display.

4.4.3.7 displaySizeChanged

```
void AutoscopeWindowForm::displaySizeChanged (
             int percent )  [private], [slot]
```

Update the size value of the picture display.

4.4.3.8 downloadPictureButtonPressed

```
void AutoscopeWindowForm::downloadPictureButtonPressed (
             void  )  [private], [slot]
```

Download the last picture taken.

4.4.3.9 exposureTimeChanged

```
void AutoscopeWindowForm::exposureTimeChanged (
             double d )  [private], [slot]
```

Update exposure time.

4.4.3.10 getGuiHorizontalPosition()

```
int AutoscopeWindowForm::getGuiHorizontalPosition (
             void  )
```

Getter used to retrieve the horizontal gui position.

Returns

The horizontal gui position

4.4.3.11 getGuiVerticalPosition()

```
int AutoscopeWindowForm::getGuiVerticalPosition (
            void )
```

Getter used to retrieve the vertical gui position.

Returns

The vertical gui position

4.4.3.12 getScreenSizePercent()

```
int AutoscopeWindowForm::getScreenSizePercent (
            void )
```

Getter used to retrieve the value of the display size editor.

Returns

The value of the display size editor

4.4.3.13 horizontalDisplayPositionChanged

```
void AutoscopeWindowForm::horizontalDisplayPositionChanged (
            int x )  [private], [slot]
```

Update the horizontal position value of the picture display.

4.4.3.14 ipAddressChanged

```
void AutoscopeWindowForm::ipAddressChanged (
            QString ipAddress )  [private], [slot]
```

Update the IP address of the telescope.

4.4.3.15 ipPortChanged

```
void AutoscopeWindowForm::ipPortChanged (
            int port )  [private], [slot]
```

Update the IP port of the telescope.

4.4.3.16 moveToButtonPressed

```
void AutoscopeWindowForm::moveToButtonPressed (
            void  )  [private], [slot]
```

Move to an azimuth and an altitude.

4.4.3.17 numberOfPictureChanged

```
void AutoscopeWindowForm::numberOfPictureChanged (
            int i )  [private], [slot]
```

Update the number of picture taken at each time.

4.4.3.18 outputPictureDirectoryButtonPressed

```
void AutoscopeWindowForm::outputPictureDirectoryButtonPressed (
            void  )  [private], [slot]
```

Show directory browser to select the directory to put picture into.

4.4.3.19 outputPictureDirectoryChanged

```
void AutoscopeWindowForm::outputPictureDirectoryChanged (
            void  )  [private], [slot]
```

Update the path to put picture into.

4.4.3.20 retranslate

```
void AutoscopeWindowForm::retranslate ( )  [slot]
```

Inherited from the StelDialog class.

See also

StelDialog

4.4.3.21 searchButtonPressed

```
void AutoscopeWindowForm::searchButtonPressed (
            void ) [private], [slot]
```

Move to the searched object and track it.

4.4.3.22 searchObjectChanged

```
void AutoscopeWindowForm::searchObjectChanged (
            QString s ) [private], [slot]
```

Update searched object.

4.4.3.23 setAutoscopePictureWindow()

```
void AutoscopeWindowForm::setAutoscopePictureWindow (
            AutoscopePictureWindowForm * autoscopePictureWindow ) [inline]
```

Setter used to initialize an instance of AutoscopePictureWindowForm class.

Parameters

autoscopePictureWindow	An instance of AutoscopePictureWindowForm class

4.4.3.24 startButtonPressed

```
void AutoscopeWindowForm::startButtonPressed (
            void ) [slot]
```

Start initialization of the telescope.

4.4.3.25 takePictureButtonPressed

```
void AutoscopeWindowForm::takePictureButtonPressed (
            void ) [slot]
```

Send a command to the telescope to take a picture.

4.4.3.26 toggleDisplay()

```
void AutoscopeWindowForm::toggleDisplay (
            void )
```

Method used to toggle the visibility of the picture display.

4.4.3.27 toggleDisplayButtonPressed

```
void AutoscopeWindowForm::toggleDisplayButtonPressed (
            void ) [slot]
```

Toggle the visibility of the picture display.

4.4.3.28 trackButtonPressed

```
void AutoscopeWindowForm::trackButtonPressed (
            void ) [slot]
```

Start tracking the selected object.

4.4.3.29 untrackButtonPressed

```
void AutoscopeWindowForm::untrackButtonPressed (
            void ) [slot]
```

Stop tracking the tracked object.

4.4.3.30 update()

```
void AutoscopeWindowForm::update ( )
```

Method used to update calculation and displayed components.

4.4.3.31 updateGuiPosition()

```
void AutoscopeWindowForm::updateGuiPosition (
            void )
```

Method used to update the gui position.

4.4.3.32 updateGuiSize()

```
void AutoscopeWindowForm::updateGuiSize (
             void  )
```

Method used to update the gui size.

4.4.3.33 updateIpMessengerText()

```
void AutoscopeWindowForm::updateIpMessengerText (
             QString message )
```

Method used to update the IP messenger text.

Parameters

the	text to put in the messenger

4.4.3.34 updateMaxMinSlider()

```
void AutoscopeWindowForm::updateMaxMinSlider (
             void  )  [private]
```

Method used to update max and min value of the position slider.

4.4.3.35 verticalDisplayPositionChanged

```
void AutoscopeWindowForm::verticalDisplayPositionChanged (
             int y )  [private], [slot]
```

Update the vertical position value of the picture display.

4.4.3.36 zoomChanged

```
void AutoscopeWindowForm::zoomChanged (
             int i )  [private], [slot]
```

Update zoom value.

4.4.4 Member Data Documentation

4.4.4.1 m_autoscope

```
Autoscope* AutoscopeWindowForm::m_autoscope [private]
```

An instance of Autoscope class.

See also

Autoscope

4.4.4.2 m_autoscopePictureWindow

```
AutoscopePictureWindowForm* AutoscopeWindowForm::m_autoscopePictureWindow [private]
```

An instance of AutoscopePictureWindowForm class.

See also

AutoscopePictureWindowForm

4.4.4.3 m_guiHorizontalPosition

```
int AutoscopeWindowForm::m_guiHorizontalPosition [private]
```

The gui horizontal position

4.4.4.4 m_guiVerticalPosition

```
int AutoscopeWindowForm::m_guiVerticalPosition [private]
```

The gui vertical position

4.4.4.5 m_height

```
int AutoscopeWindowForm::m_height = 300 [private]
```

The height of the dialog box

4.4.4.6 m_screenHeight

```
int AutoscopeWindowForm::m_screenHeight  [private]
```

The height of the screen

4.4.4.7 m_screenWidth

```
int AutoscopeWindowForm::m_screenWidth  [private]
```

The width of the screen

4.4.4.8 m_width

```
int AutoscopeWindowForm::m_width = 600  [private]
```

The width of the dialog box

4.4.4.9 picturedirectoryPath

```
QString AutoscopeWindowForm::picturedirectoryPath = ""  [private]
```

The path where to put downloaded picture

4.4.4.10 pictureWindowIsVisible

```
bool AutoscopeWindowForm::pictureWindowIsVisible = false  [private]
```

Flag which represent the visibility of the picture display window

4.4.4.11 screenSizePercent

```
int AutoscopeWindowForm::screenSizePercent  [private]
```

The size of the dialog box in percent of the screen

4.4.4.12 ui

```
Ui_AutoscopeWindowForm* AutoscopeWindowForm::ui  [private]
```

An instance of the Ui_AutoscopeWindowForm class

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

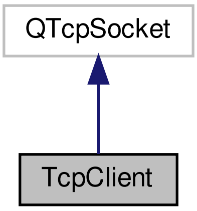
- src/gui/AutoscopeWindowForm.hpp
- src/gui/AutoscopeWindowForm.cpp

4.5 TcpClient Class Reference

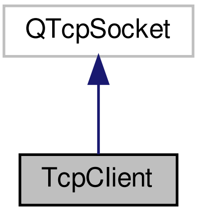
The TcpClient class allow to make usage of QTcpSocket more suitable for development.

```
#include <tcp_client.hpp>
```

Inheritance diagram for TcpClient:



Collaboration diagram for TcpClient:



Public Slots

- void host_found_handler (void)

  Handle host found.
- void connected_handler (void)

  Handle connection to host.

Public Member Functions

- TcpClient (QTcpSocket ∗parent=0)

  Builder of the TcpClient class.
- TcpClient (QHostAddress host_address, quint16 port)

  Builder of the TcpClient class.
- ∼TcpClient ()

  Destroyer of the TcpClient class.

Private Attributes

- quint16 m_port
- QHostAddress m_host_address

4.5.1 Detailed Description

The TcpClient class allow to make usage of QTcpSocket more suitable for development.

4.5.2 Constructor & Destructor Documentation

4.5.2.1 TcpClient() [1/2]

```
TcpClient::TcpClient (
              QTcpSocket * parent = 0 )
```

Builder of the TcpClient class.

Parameters

parent	The parent of the TcpClient

4.5.2.2 TcpClient() [2/2]

```
TcpClient::TcpClient (
              QHostAddress host_address,
              quint16 port )
```

Builder of the TcpClient class.

Parameters

host_address	The address of the server
port	The port of the server

4.5.2.3 ∼TcpClient()

```
TcpClient::∼TcpClient ( )
```

Destroyer of the TcpClient class.

4.5.3 Member Function Documentation

4.5.3.1 connected_handler

```
void TcpClient::connected_handler (
            void  )  [slot]
```

Handle connection to host.

4.5.3.2 host_found_handler

```
void TcpClient::host_found_handler (
            void  )  [slot]
```

Handle host found.

4.5.4 Member Data Documentation

4.5.4.1 m_host_address

```
QHostAddress TcpClient::m_host_address  [private]
```

server host address

4.5.4.2 m_port

```
quint16 TcpClient::m_port  [private]
```

server port

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

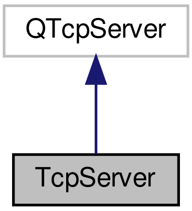
- src/network/tcp_client.hpp
- src/network/tcp_client.cpp

4.6 TcpServer Class Reference

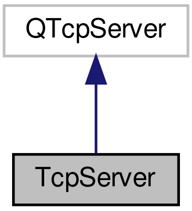
The TcpServer class allow to make usage of QTcpServer more suitable for development.

```
#include <tcp_server.hpp>
```

Inheritance diagram for TcpServer:



Collaboration diagram for TcpServer:



Public Slots

- void create_socket (void)

  create_socket

Public Member Functions

- TcpServer ()
- TcpServer (bool autostart=0)
- TcpServer (QString &host_address, quint16 port=4444, bool autostart=0)
- ∼TcpServer ()
- void start (void)

  start

- void set_port (quint16 port)

    set_port
- void set_address (const char ∗address)

    set_address
- QTcpSocket ∗ get_client (void)

    get_client

Private Attributes

- quint16 m_port
- QString m_host_address
- QTcpSocket ∗ m_tcp_client

4.6.1 Detailed Description

The TcpServer class allow to make usage of QTcpServer more suitable for development.

4.6.2 Constructor & Destructor Documentation

4.6.2.1 TcpServer() [1/3]

```
TcpServer::TcpServer ( )  [explicit]
```

4.6.2.2 TcpServer() [2/3]

```
TcpServer::TcpServer (
              bool autostart = 0 )
```

4.6.2.3 TcpServer() [3/3]

```
TcpServer::TcpServer (
              QString & host_address,
              quint16 port = 4444,
              bool autostart = 0 )
```

4.6.2.4 ∼TcpServer()

```
TcpServer::∼TcpServer ( )
```

4.6.3 Member Function Documentation

4.6.3.1 create_socket

```
void TcpServer::create_socket (
             void ) [slot]
```

create_socket

4.6.3.2 get_client()

```
QTcpSocket * TcpServer::get_client (
             void )
```

get_client

Returns

client address for interacting with the client connected to the server

4.6.3.3 set_address()

```
void TcpServer::set_address (
             const char * address )
```

set_address

Parameters

address	

4.6.3.4 set_port()

```
void TcpServer::set_port (
             quint16 port )
```

set_port

Parameters

port	

4.6.3.5 start()

```
void TcpServer::start (
            void  )
```

start

4.6.4 Member Data Documentation

4.6.4.1 m_host_address

```
QString TcpServer::m_host_address  [private]
```

Client object used to communicate with the client connected to the server

4.6.4.2 m_port

```
quint16 TcpServer::m_port  [private]
```

< Server port Server address

4.6.4.3 m_tcp_client

```
QTcpSocket* TcpServer::m_tcp_client  [private]
```

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

- src/network/tcp_server.hpp
- src/network/tcp_server.cpp

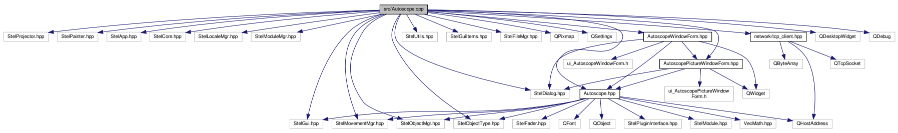
Chapter 5

File Documentation

5.1 src/Autoscope.cpp File Reference

```
#include "StelProjector.hpp"
#include "StelPainter.hpp"
#include "StelApp.hpp"
#include "StelCore.hpp"
#include "StelLocaleMgr.hpp"
#include "StelModuleMgr.hpp"
#include "Autoscope.hpp"
#include "StelUtils.hpp"
#include "StelObjectType.hpp"
#include "StelMovementMgr.hpp"
#include "StelObjectMgr.hpp"
#include "StelGui.hpp"
#include "StelGuiItems.hpp"
#include "StelFileMgr.hpp"
#include <QPixmap>
#include <QSettings>
#include "StelDialog.hpp"
#include "AutoscopeWindowForm.hpp"
#include "AutoscopePictureWindowForm.hpp"
#include "network/tcp_client.hpp"
#include <QDesktopWidget>
#include <QDebug>
```

Include dependency graph for Autoscope.cpp:

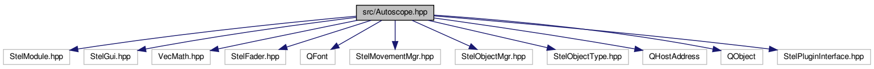


5.2 src/Autoscope.hpp File Reference

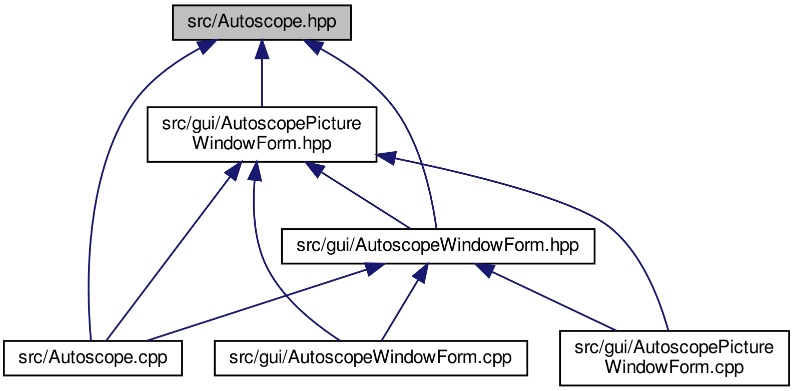
Main plugin class.

```
#include "StelModule.hpp"
#include "StelGui.hpp"
#include "VecMath.hpp"
#include "StelFader.hpp"
#include <QFont>
#include "StelMovementMgr.hpp"
#include "StelObjectMgr.hpp"
#include "StelObjectType.hpp"
#include <QHostAddress>
#include <QObject>
#include "StelPluginInterface.hpp"
```

Include dependency graph for Autoscope.hpp:



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


src/Autoscope.hpp
src/gui/AutoscopePicture WindowForm.hpp
src/gui/AutoscopeWindowForm.hpp
src/Autoscope.cpp
src/gui/AutoscopeWindowForm.cpp
src/gui/AutoscopePicture WindowForm.cpp

Classes

- class Autoscope

  This class is use to retrieve and compute data for sending to the telescope, and call user interfaces windows.
- class AutoscopeStelPluginInterface

  This class is used by Qt to manage a plug-in interface.

5.2.1 Detailed Description

Main plugin class.

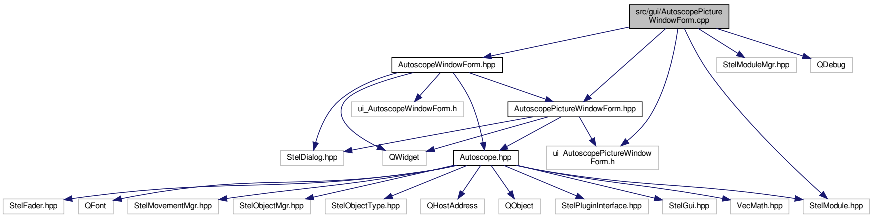
Author

thibaud-ledoledec

5.3 src/gui/AutoscopePictureWindowForm.cpp File Reference

```
#include "AutoscopePictureWindowForm.hpp"
#include "ui_AutoscopePictureWindowForm.h"
#include "AutoscopeWindowForm.hpp"
#include "StelModule.hpp"
#include "StelModuleMgr.hpp"
#include <QDebug>
```

Include dependency graph for AutoscopePictureWindowForm.cpp:

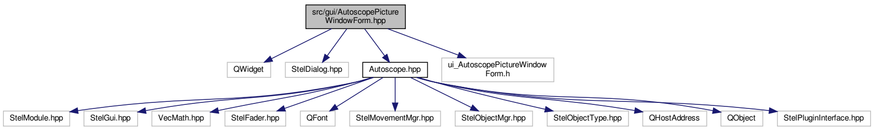


5.4 src/gui/AutoscopePictureWindowForm.hpp File Reference

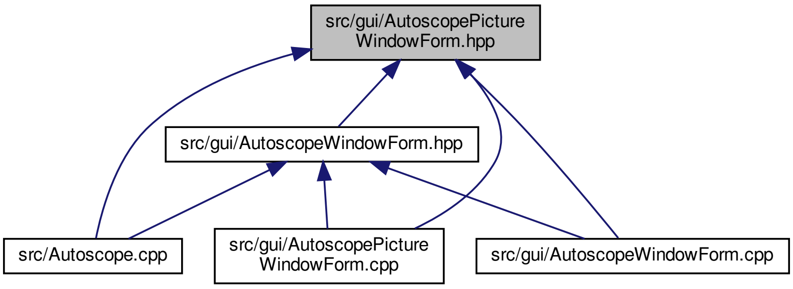
Header file including the picture display window definition.

```
#include <QWidget>
#include "StelDialog.hpp"
#include "Autoscope.hpp"
#include "ui_AutoscopePictureWindowForm.h"
```

Include dependency graph for AutoscopePictureWindowForm.hpp:



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


src/gui/AutoscopePicture WindowForm.hpp
src/gui/AutoscopeWindowForm.hpp
src/Autoscope.cpp
src/gui/AutoscopePicture WindowForm.cpp
src/gui/AutoscopeWindowForm.cpp

Classes

- class AutoscopePictureWindowForm

  This class is use to build a picture display interface to show to the user the last taken picture.

5.4.1 Detailed Description

Header file including the picture display window definition.

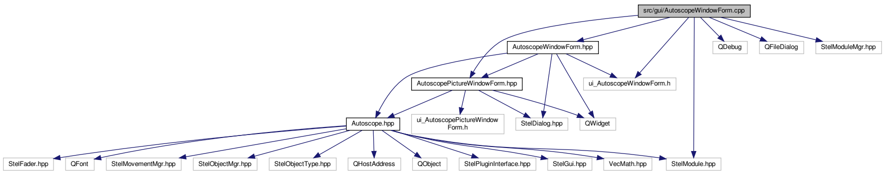
Author

thibaud-ledoledec

5.5 src/gui/AutoscopeWindowForm.cpp File Reference

```
#include "AutoscopeWindowForm.hpp"
#include "ui_AutoscopeWindowForm.h"
#include <QDebug>
#include <QFileDialog>
#include "AutoscopePictureWindowForm.hpp"
#include "StelModule.hpp"
#include "StelModuleMgr.hpp"
```

Include dependency graph for AutoscopeWindowForm.cpp:

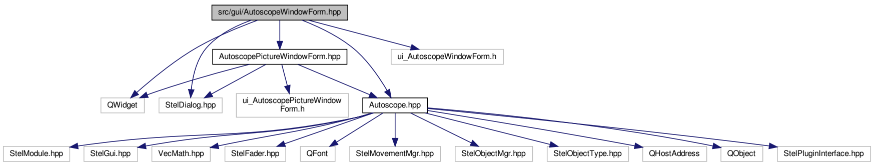


5.6 src/gui/AutoscopeWindowForm.hpp File Reference

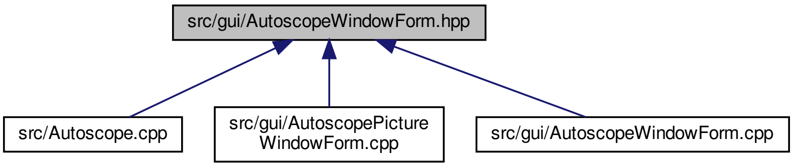
Header file including configuration window definition.

```
#include <QWidget>
#include "StelDialog.hpp"
#include "Autoscope.hpp"
#include "ui_AutoscopeWindowForm.h"
#include "AutoscopePictureWindowForm.hpp"
```

Include dependency graph for AutoscopeWindowForm.hpp:



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


src/gui/AutoscopeWindowForm.hpp
src/Autoscope.cpp
src/gui/AutoscopePicture WindowForm.cpp
src/gui/AutoscopeWindowForm.cpp

Classes

- class AutoscopeWindowForm

  This class is use to build a configuration interface between the user and the plugin.

5.6.1 Detailed Description

Header file including configuration window definition.

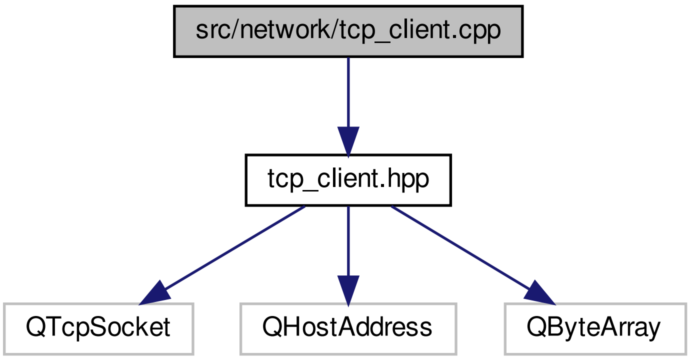
Author

thibaud-ledoledec

5.7 src/network/tcp_client.cpp File Reference

```
#include "tcp_client.hpp"
```

Include dependency graph for tcp_client.cpp:

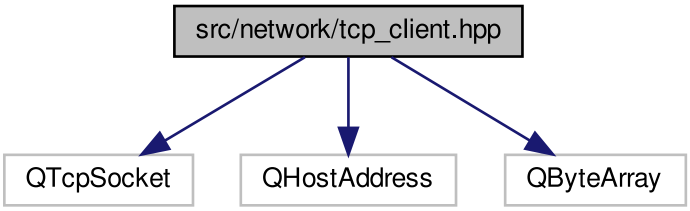


5.8 src/network/tcp_client.hpp File Reference

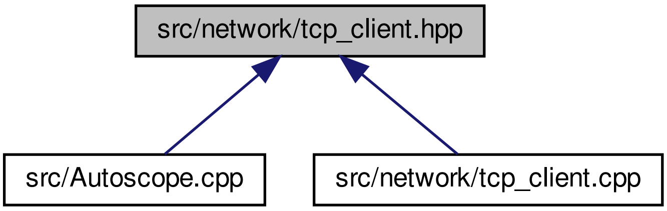
Header file including TCP client definition.

```
#include <QTcpSocket>
#include <QHostAddress>
#include <QByteArray>
```

Include dependency graph for tcp_client.hpp:



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



Classes

- class TcpClient

  The TcpClient class allow to make usage of QTcpSocket more suitable for development.

5.8.1 Detailed Description

Header file including TCP client definition.

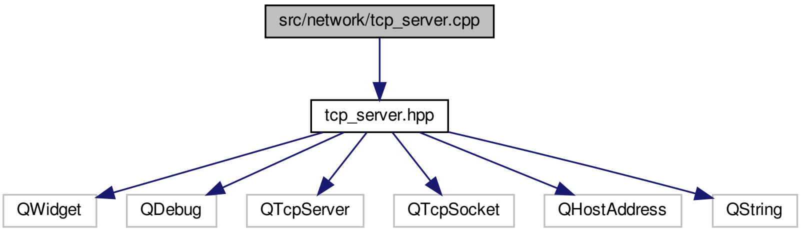
Author

clement-ailloud

5.9 src/network/tcp_server.cpp File Reference

```
#include "tcp_server.hpp"
```

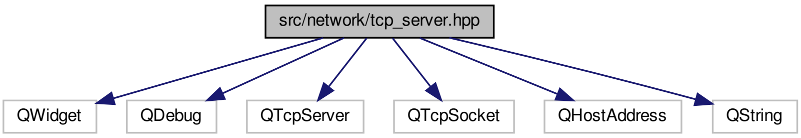
Include dependency graph for tcp_server.cpp:



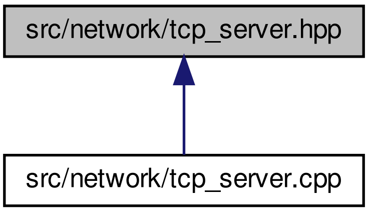
5.10 src/network/tcp_server.hpp File Reference

```
#include <QWidget>
#include <QDebug>
#include <QTcpServer>
#include <QTcpSocket>
#include <QHostAddress>
#include <QString>
```

Include dependency graph for tcp_server.hpp:



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



Classes

- class TcpServer

    The TcpServer class allow to make usage of QTcpServer more suitable for development.