

IPE TRONIK



IPEMOTION
Software

IPEmotionSettings COM Manual

Version: 2014.0

February 2014

Table of contents

1	Important and general information	1
1.1	Important information	1
1.2	General information	5
2	IPEmotionSettings	7
2.1	IPEmotionSettings Interfaces Overview	8
3	IPEmotionSettings Namespace Index	9
3.1	IPEmotionSettings Namespace List	9
4	IPEmotionSettings Interface Index	11
4.1	IPEmotionSettings Interface List	11
5	IPEmotionSettings Namespace Documentation	13
5.1	IPEmotionSettings Namespace Reference	13
6	IPEmotionSettings Interface Documentation	23
6.1	ICloudDrive Interface Reference	23
6.2	ICloudDrives Interface Reference	25
6.3	IDataPlugIn Interface Reference	27
6.4	IDataPlugIns Interface Reference	29
6.5	IPlugIn Interface Reference	31
6.6	IPlugInOption Interface Reference	33
6.7	IPlugInOptions Interface Reference	34
6.8	IPlugIns Interface Reference	35
6.9	IReferenceObject Interface Reference	36
6.10	IReferenceObjects Interface Reference	37
6.11	ISettingsHandler Interface Reference	38
6.12	IUserAccount Interface Reference	53
6.13	IUserAccounts Interface Reference	55

Chapter 1

Important and general information

1.1 Important information

Please follow these instructions before and during the use and application on any IPETRONIK product!

1.1.1 Safety and Warning instructions

Please follow the instructions **and** information as contained in the user manual!

1. The user can **influence an electronic system by applying the IPETRONIK product**. This might cause risk of personal injury or property damages.
2. The **use and application of the IPETRONIK product is permitted only to qualified professional staff**, as well as, only in appropriate manner and in the designated use.
3. **Before using an IPETRONIK measurement system** in the vehicle it **has to be verified that no function of the vehicle, which is relevant for secure operation, might be influenced**:
 - by the installation of the IPETRONIK measurement system in the vehicle,
 - by an potential malfunction of the IPETRONIK system during the test drive.

In order to avoid possible danger or personal injury and property damages, appropriate actions are to be taken; such actions have to bring the entire system into a secured condition (e.g. by using a system for emergency stop, an emergency operation, monitoring of critical values). Please check the following points to avoid errors:

- Adaption of sensors to components of the electrical system / electronics, brake system, engine and transmission control, chassis, body.
 - Tap of one or several bus systems (CAN, LIN, ETHERNET) including the required electrical connection(s) for data acquisition.
 - Communication with the vehicle's control units (ECUs), especially with such of the brake system and/or of the engine and transmission control (power train control system).
 - Installation of components for remote data transmission (mobiles, GSM/GPRS modems, WiFi and Bluetooth components).
4. **Before** directly or indirectly using **the data acquired by an IPETRONIK measurement system to calibrate control units, please review the data regarding to plausibility**.
 5. With regard to the application of IPETRONIK products in vehicles during use on public roads the manufacturer and/or registered user of the vehicle **has to ensure that all changes/modifications have no influence concerning the license of the vehicle or its license of operation**.
 6. **User does agree to the instructions and regulations as mentioned above**. In case the user does not agree with the instructions and regulations as mentioned above, he has to notify this expressly and immediately in writing to IPETRONIK before confirming the sales contract.

1.1.2 Liability, Warranty, Copyright, License agreement

Limitation of liability

Any liability of IPETRONIK, its representatives, agents and the like, especially with regard to personal injury or damage to property of any kind, shall be excluded (within the legally admissible framework), as far as, the instructions and warnings, as mentioned below, have not been followed.

Warranty

Products, accessories and services have a 24 months warranty.

All product data, specifications, drawings, etc., correspond to the current condition of the indicated creation date. For the purpose of optimizing technical processes and production, some details of our modules and accessory components may be modified at any time without prior notification.

Although the present document has been prepared with the utmost attention to detail, it may not be exempt of misprints, typing or transcription errors. These errors are not covered by any warranty.

Copyright and Duplication

All rights reserved to IPETRONIK GmbH & Co. KG, in particular those of property, copyright and trademarks. The rights related to any third party trademarks mentioned in the present document remain unaffected.

This document may not be duplicated, partially or entirely without the prior approval from IPETRONIK GmbH & Co. KG. All graphics and explanations are copyright protected. Any use beyond the scope of the document is prohibited.

Software license agreement

This software is property of IPETRONIK GmbH & Co. KG, and is protected by copyright laws. Its total or partial reproduction is strictly forbidden.

A valid software license is required to use the software.

IMPORTANT - READ CAREFULLY! THIS IS A LEGAL AGREEMENT BETWEEN YOU, LICENSEE, AND IPETRONIK GMBH & CO. KG/IPETRONIK INC. ("IPETRONIK"). BY CHECKING "I ACCEPT ALL OF THE TERMS CONTAINED IN THE ABOVE AGREEMENT" DURING INSTALLATION, COPYING OR USING THIS PRODUCT IN ANY WAY YOU ACKNOWLEDGE THAT YOU HAVE READ THIS LICENSE AND THAT YOU UNDERSTAND AND EXPRESSLY AGREE TO BE BOUND BY THE TERMS AND CONDITIONS SET FORTH BELOW.

1. Definitions

- a. SOFTWARE is defined as computer program in object code or machine-readable format, together with any and all modifications, enhancements, updates, and improvements provided by IPETRONIK as well as any subsequent versions, corrections, bug fixes, enhancements, updates or other modifications, regardless of the source. The term "Licensed Software" shall not include the source code version of the Licensed Software.
- b. EQUIPMENT is defined as automotive measuring equipment produced by IPETRONIK as well as other parties.
- c. LICENSEE is defined as the recipient of this SOFTWARE and any of its employees, agents or representatives.
- d. MODIFY or MODIFICATION is defined as change to the SOFTWARE by LICENSEE in order to customize the SOFTWARE for use solely by LICENSEE.

2. License Terms

- a. As long as LICENSEE complies with all terms in this Software License Agreement IPETRONIK grants LICENSEE a non-exclusive, non-transferable license to load and use the SOFTWARE upon the terms and conditions set forth below.
- b. LICENSEE has the right to load the SOFTWARE for use on any internal computer or piece of EQUIPMENT, as long as it is only on one computer or piece of EQUIPMENT at any given time.
- c. LICENSEE will notify all of its employees, agents or representatives permitted access to the SOFTWARE of the duties and obligations under this Software License Agreement.

- d. LICENSEE has the right to MODIFY the SOFTWARE for use on any internal computer or of EQUIPMENT, as long as it is only on one computer or piece of EQUIPMENT at any given time.
- e. Any MODIFICATION(S) to the SOFTWARE are subject to the terms and conditions of this Agreement.
- f. LICENSEE may not:
 - i. Loan, rent, lease, give, sublicense, distribute, transmit or otherwise transfer the SOFTWARE, or otherwise exercise any of IPETRONIK's legal rights in and to the SOFTWARE, or any derivative works of the SOFTWARE, in whole or in part, except with the prior written agreement of IPETRONIK.
 - ii. Copy, translate, reverse engineer, decompile, disassemble the SOFTWARE, in whole or in part.
 - iii. Except as provided in Section 2(d), create derivative works based on the SOFTWARE, in whole or in part.
 - iv. Remove, modify or cause not to be displayed any copyright or trademark notices, license agreements, or startup messages contained in the programs or documentation.
 - v. Transmit or otherwise export outside of the United States any of the SOFTWARE or technology in violation of United States or other applicable laws or regulations.

3. Ownership of Intellectual Property

LICENSEE agrees and acknowledges that the SOFTWARE is being provided to it only for use in EQUIPMENT in the ordinary course of business and that LICENSEE agrees and acknowledges that IPETRONIK is the owner of all title and proprietary rights in the SOFTWARE, including, without limitation, any and all patents, copyrights, trademarks or any other intellectual property rights associated with it under the laws of the United States or any jurisdiction throughout the world. No right, title or interest in the SOFTWARE or any IPETRONIK patent, copyright, trademark, or any other intellectual property right is transferred to LICENSEE or any other party through this Software License Agreement.

4. Disclaimer of Warranties; Liability Limitations

- a. THE SOFTWARE IS PROVIDED TO YOU "AS IS". THERE ARE NO WARRANTIES OF ANY KIND, WHETHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND COMPATIBILITY, AND TITLE.
- b. LICENSEE ASSUMES ALL RISK AS TO THE SELECTION, USE, PERFORMANCE AND QUALITY OF THE SOFTWARE. IN NO EVENT WILL IPETRONIK OR ANY OTHER PARTY WHO HAS BEEN INVOLVED IN THE CREATION, PRODUCTION OR DELIVERY OF THE SOFTWARE BE LIABLE FOR SPECIAL, DIRECT, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING LOSS OF PROFITS OR INABILITY TO USE THE LICENSED MATERIAL. IN NO EVENT SHALL IPETRONIK'S LIABILITY FOR ANY DAMAGES OR LOSS TO LICENSEE OR TO ANY THIRD PARTY EXCEED ANY LICENSE FEE ACTUALLY PAID BY THE LICENSEE TO IPETRONIK FOR THE SOFTWARE.
- c. Since some states or jurisdictions do not permit the exclusion of implied warranties or limitation of liability for consequential damages, in such states or jurisdictions, the liability is limited to the fullest extent permitted by law.

5. Intellectual Property Infringement Indemnification

- a. IPETRONIK shall defend, indemnify, and hold LICENSEE harmless from and against any claims and fees (including attorneys' fees), damage awards arising in connection with a claim that the licensed SOFTWARE or documentation, when properly used, infringes upon any presently existing, valid and enforceable United States patent, trademark, or other intellectual property right, provided that:
 - i. such claim of infringement is not based on any Modification or action taken or suffered by LICENSEE other than the use of the licensed SOFTWARE and documentation in accordance with the terms and conditions of this agreement;
 - ii. such claim of infringement is not based on any action by LICENSEE in modifying the SOFTWARE pursuant to the terms of Section 2(d).
 - iii. LICENSEE promptly notifies IPETRONIK of such claim in writing at support@ipetronik.com, and gives IPETRONIK exclusive control over the defense and settlement of such claim;
 - iv. LICENSEE provides such cooperation and assistance, at IPETRONIK'S expense, as IPETRONIK may reasonably request to settle or oppose any such claim; and
 - v. such claim of infringement is based only on the licensed SOFTWARE and documentation as provided to LICENSEE.

- b. In the event of any infringement claim for which IPETRONIK is liable pursuant to section 5 (a), IPETRONIK may, at its sole option and expense:
 - i. procure for LICENSEE the right to continue using the licensed SOFTWARE or documentation;
 - ii. modify or amend the licensed SOFTWARE or documentation so that it becomes non-infringing;
 - iii. replace the licensed SOFTWARE or documentation with a non-infringing substitute; or
 - iv. recover the infringing licensed software and documentation from LICENSEE and repay to LICENSEE all license fees paid to IPETRONIK in connection therewith, less a reasonable amount based on LICENSEE's use prior to such recovery and refund.
- c. This Article 5 sets forth IPETRONIK's sole obligations and liability for intellectual property infringement. These indemnity provisions only apply to the SOFTWARE as originally licensed to LICENSEE and do not cover any MODIFICATIONS made by LICENSEE or any other third party.

6. Limitation of Liability

- a. EXCEPT WITH RESPECT TO ITS INTELLECTUAL PROPERTY INDEMNIFICATION OBLIGATIONS, AS SET FORTH IN ARTICLE 5, IN NO EVENT SHALL IPETRONIK BE LIABLE FOR SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, LOST DATA, OR LOST SAVINGS), EVEN IF IPETRONIK WAS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. FURTHERMORE, IPETRONIK'S LIABILITY (WHETHER IN CONTRACT, TORT, OR OTHERWISE) ARISING OUT OF, OR CONNECTED WITH, THIS AGREEMENT OR THE LICENSED SOFTWARE OR DOCUMENTATION SHALL IN NO CASE EXCEED THE PAYMENTS RECEIVED BY IPETRONIK FROM LICENSEE FOR THE LICENSED SOFTWARE AND DOCUMENTATION.
- b. EXCEPT IN CONNECTION WITH ITS OBLIGATIONS UNDER ARTICLE 5:
 - i. IN NO EVENT SHALL LICENSEE BE LIABLE FOR SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, LOST DATA, OR LOST SAVINGS), EVEN IF LICENSEE WAS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES; AND
 - ii. LICENSEE'S LIABILITY (WHETHER IN CONTRACT, TORT, OR OTHERWISE) ARISING OUT OF, OR CONNECTED WITH, THIS AGREEMENT OR THE LICENSED SOFTWARE OR DOCUMENTATION SHALL IN NO CASE EXCEED THE PAYMENTS OWED TO LICENSOR FOR THE LICENSED SOFTWARE AND DOCUMENTATION.

7. Indemnification Obligations of LICENSEE

- a. LICENSEE shall defend, indemnify, and hold IPETRONIK harmless from any claims, losses, expenses, fees (including attorneys' fees), costs or damages arising in connection with a MODIFICATION or LICENSEE'S unauthorized use of the Licensed Software or Documentation.

8. Merger Clause

- a. LICENSEE agrees that this Software License Agreement is the complete and exclusive agreement between LICENSEE and IPETRONIK governing the SOFTWARE. This Software License Agreement supersedes and merges all prior agreements with IPETRONIK concerning the SOFTWARE and can only be modified by a subsequent written agreement signed by IPETRONIK. To the extent that there is any conflict between this Software License Agreement and any IPETRONIK purchase order or other written agreement for the purchase of IPETRONIK parts or products, the terms of the purchase order or written agreement control.

9. General

- a. If any provision or portion of a provision of this Software License Agreement is determined to be invalid or unenforceable, it shall be deemed omitted and the remaining provisions of this Software License Agreement shall remain in full force and effect to the fullest extent permitted by law.
- b. LICENSEE may not assign or transfer all or part of this Software License Agreement to any third party without the express written approval of IPETRONIK.
- c. This Software License Agreement will be governed by the laws of the State of Michigan without regard to its conflict of laws provisions.
- d. All disputes arising out of, or in connection with, the present contract shall be finally settled under the Rules of Arbitration of the International Chamber of Commerce by one or more arbitrators appointed in accordance with the said Rules.

1.2 General information

1.2.1 About this manual

This manual describes the COM interface provided by the software IPEmotionSettings.

1.2.2 Version

This manual has the version number 2014.0, released February 2014 © **All rights reserved !**

1.2.3 Support

Headquarter:

IPETRONIK GmbH & Co. KG

Im Rollfeld 28

76532 Baden-Baden, Germany

Phone +49 7221 9922 0

Fax +49 7221 9922 100

info@ipetronik.com

www.ipetronik.com

Limited commercial partnership with its head office in Baden-Baden, registry court HRA No. 201313

IPETRONIK Verwaltungs-GmbH Baden-Baden is an individually liable society, registry court Mannheim HRB No. 202089

CEOs: Erich Rudolf, Andreas Wocke

Technical support and product information

www.ipetronik.com

e-mail: support@ipetronik.com

Chapter 2

IPEmotionSettings

This document describes the COM interfaces of the IPEmotionSettings program.

The IPEmotionSettings program is meant to be used to modify the IPEmotion options.

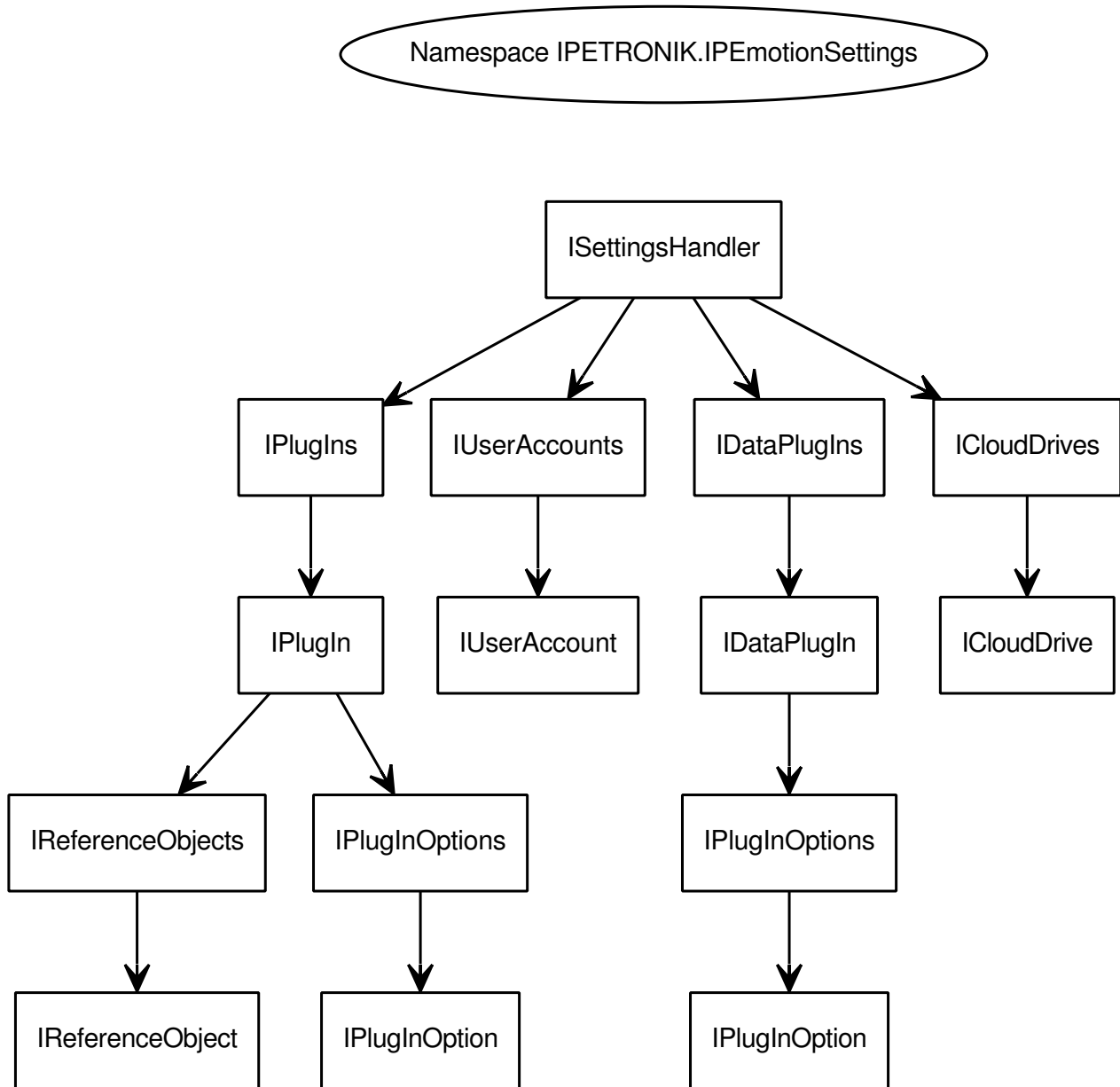
The program can only be used, while no instance of IPEmotion is running. Only one instance of this program can be run simultaneously. It will only change the options of the latest installed version of IPEmotion.

To change the IPEmotion options, you need to write a script or program, that uses the COM interface, provided by the IPEmotionSettings program.

The actual options are loaded at program startup and will be changed only if the interface function SaveSettings() is called.

The IPEmotionSettings program must be terminated by calling the interface function Quit(), before IPEmotion will be started.

2.1 IPEmotionSettings Interfaces Overview



Chapter 3

IPEmotionSettings Namespace Index

3.1 IPEmotionSettings Namespace List

The following chapter lists all namespaces with brief descriptions:

IPEmotionSettings (Namespace of the IPEmotion settings interface)	13
---	----

Chapter 4

IPEmotionSettings Interface Index

4.1 IPEmotionSettings Interface List

The following chapter lists all interfaces with brief descriptions:

ICloudDrive (Cloud drive interface)	23
ICloudDrives (Cloud drive list interface)	25
IDataPlugIn (Data plug-in interface)	27
IDataPlugIns (Data plug-in list interface)	29
IPlugIn (Plug-in interface)	31
IPlugInOption (Plug-in options interface)	33
IPlugInOptions (Plug-in options list interface)	34
IPlugIns (Plug-in list interface)	35
IReferenceObject (Reference object interface)	36
IReferenceObjects (Reference object list interface)	37
ISettingsHandler (IPEmotion settings handler interface)	38
IUserAccount (User account interface)	53
IUserAccounts (User account list interface)	55

Chapter 5

IPEmotionSettings Namespace Documentation

5.1 IPEmotionSettings Namespace Reference

Namespace of the IPEmotion settings interface.

Interfaces

- interface [ICloudDrive](#)
Cloud drive interface.
- interface [ICloudDrives](#)
Cloud drive list interface.
- interface [IDataPlugIn](#)
Data plug-in interface.
- interface [IDataPlugIns](#)
Data plug-in list interface.
- interface [IPlugIn](#)
Plug-in interface.
- interface [IPlugInOption](#)
Plug-in options interface.
- interface [IPlugInOptions](#)
Plug-in options list interface.
- interface [IPlugIns](#)
Plug-in list interface.
- interface [IReferenceObject](#)
Reference object interface.
- interface [IReferenceObjects](#)
Reference object list interface.
- interface [ISettingsHandler](#)

IPEmotion settings handler interface.

- interface [IUserAccount](#)
User account interface.
- interface [IUserAccounts](#)
User account list interface.

Enumerations

- enum [CharacteristicSupport](#)
Enumeration of available characteristic support types.
- enum [ConfigurationPriority](#)
Enumeration of available usage priorities.
- enum [HardwareDetectionAction](#)
Enumeration of available actions to be performed after the automatic hardware detection.
- enum [MAP_IMAGERY_SOURCE](#)
Enumeration of supported sources of map imagery.
- enum [PhysicalDimensionTypes](#)
Enumeration of supported physical dimension types.
- enum [SelectedPlotHighlightType](#)
Enumeration of supported highlighting types used to highlight selected curves in yt-charts.
- enum [SignalConfigurationMode](#)
Enumeration of available configuration modes.
- enum [TimeZoneDisplayType](#)
Enumeration of supported time zones used to display absolute times in the analysis.
- enum [UiCommand](#)
Enumeration of available shortcut commands.
- enum [UserLevel](#)
Enumeration of available access levels.

5.1.1 Detailed Description

Namespace of the IPEmotion settings interface.

5.1.2 Enumeration Type Documentation

5.1.2.1 enum CharacteristicSupport

Enumeration of available characteristic support types.

Remarks:

The following enumeration values are available:

- 0: `AsCharacteristic` - import as characteristic
- 1: `AsMeasurement` - conversion to measurement
- 2: `Ignore` - no import at all

5.1.2.2 enum ConfigurationPriority

Enumeration of available usage priorities.

Remarks:

The following enumeration values are available:

- 0: `High` - Use this setting for your preferred hardware
- 1: `Standard` - The default setting
- 2: `Low` - Use this for hardware you own but don't want to be used
- 3: `NotUsed` - Removes this type of hardware from the list of available hardware

5.1.2.3 enum HardwareDetectionAction

Enumeration of available actions to be performed after the automatic hardware detection.

Remarks:

The following enumeration values are available:

- 0: `ACTION_USER_GUIDANCE` - Guided configuration
- 1: `ACTION_AUTO_CONFIGURATION` - Automatic creation of a default configuration
- 2: `ACTION_MANUAL_CONFIGURATION` - Manual configuration

5.1.2.4 enum MAP_IMAGERY_SOURCE

Enumeration of supported sources of map imagery.

Remarks:

The following enumeration values are available:

- 0x00010001: `InternalCacheWithInternet` - Tiles are retrieved from the internal cache if contained within, otherwise loaded from the internet. Tiles loaded from the internet are stored in the cache.
- 0x00020001: `LocalDatabaseWithInternet` - Tiles are retrieved from the specified local database if contained within, otherwise loaded from the internet. Tiles loaded from the internet are not stored.
- 0x00020000: `LocalDatabase` - Tiles are retrieved from the specified local database only. Map regions requiring imagery which is not contained in the database are not displayed.

5.1.2.5 enum PhysicalDimensionTypes

Enumeration of supported physical dimension types.

Remarks:

The following enumeration values are available:

- 0: `NONE` - Used for values without a physical dimension
- 1: `Length`

- 2: Mass
- 3: Time
- 4: Current
- 5: Temperature
- 6: MolarAmount
- 7: LuminousIntensity
- 8: Voltage
- 9: Resistance
- 10: Frequency
- 11: Force
- 12: Energy
- 13: Power
- 14: Pressure
- 15: Speed
- 16: Rpm
- 17: MassFlowRate
- 18: Angle
- 19: Strain
- 20: BaudeRate
- 21: Percent
- 22: ByteSpace
- 23: Acceleration
- 24: Torque
- 25: VolumetricFlowRate
- 26: SpecificEnthalpy

5.1.2.6 enum SelectedPlotHighlightType

Enumeration of supported highlightning types used to highlight selected curves in yt-charts.

Remarks:

The following enumeration values are available:

- 0: NoHighlightning - no highlightning
- 1: Bold - the curve is drawn with an increased line width
- 2: Flash - the seelcted curve is flashing

5.1.2.7 enum SignalConfigurationMode

Enumeration of available configuration modes.

Remarks:

The following enumeration values are available:

- 0: CONFIGURATION_MODE_HARDWARE - Hardware centric configuration
- 1: CONFIGURATION_MODE_MEASURE_POINT - Configuration based on a measurement point plan

5.1.2.8 enum TimeZoneDisplayType

Enumeration of supported time zones used to display absolute times in the analysis.

Remarks:

The following enumeration values are available:

- 0: `Utc` - UTC
- 1: `Local` - local time zone of the analysis PC
- 2: `Source` - local time zone of the measurement file

5.1.2.9 enum UiCommand

Enumeration of available shortcut commands.

Remarks:

The following enumeration values are available:

- 0: `COMMAND_FILE_NEW`
- 1: `COMMAND_FILE_SAVE`
- 2: `COMMAND_FILE_SAVE_AS`
- 3: `COMMAND_FILE_OPEN`
- 4: `COMMAND_PRINT`
- 5: `COMMAND_EDIT_CUT`
- 6: `COMMAND_EDIT_COPY`
- 7: `COMMAND_EDIT_PASTE`
- 8: `COMMAND_EDIT_UNDO`
- 9: `COMMAND_EDIT_REDO`
- 10: `COMMAND_SCRIPTING_NEW_SCRIPT`
- 11: `COMMAND_SCRIPTING_IMPORT`
- 12: `COMMAND_ABOUT`
- 13: `COMMAND_NEW_SCRIPT_GROUP`
- 14: `COMMAND_PRINT_ONCE`
- 15: `COMMAND_PRINT_AS_PDF`
- 16: `COMMAND_PRINT_PREVIEW`
- 17: `COMMAND_CREATE_RUNTIME_VERSION`
- 18: `COMMAND_PRINT_EXPORT_EXCEL`
- 19: `COMMAND_PRINT_EXPORT_HTML`
- 20: `COMMAND_PRINT_EXPORT_RTF`
- 21: `COMMAND_PRINT_EXPORT_IMAGE`
- 22: `COMMAND_PRINT_EXPORT_V`
- 23: `COMMAND_SHOW_MESSAGE_WINDOW`
- 24: `COMMAND_RESET_USER_SETTINGS`
- 25: `COMMAND_SHOW_OPTIONS_DIALOG`
- 26: `COMMAND_CREATE_SUPPORT_FILE`
- 27: `COMMAND_QUIT_PROGRAM`
- 28: `COMMAND_EDIT_DELETE`
- 29: `COMMAND_EDIT_CLEAR`
- 30: `COMMAND_SHOW_ITEM_PROPERTIES`

- 31: COMMAND_SHOW_HELP
- 32: COMMAND_BUILD_CONFIGURATION
- 33: COMMAND_USE_USER_GUIDE
- 34: COMMAND_ADJUST_WITH_SENSORDATABASE
- 35: COMMAND_DETECT_HARDWARE
- 36: COMMAND_INIT_HARDWARE
- 37: COMMAND_RESET_HARDWARE
- 38: COMMAND_SYNC_HARDWARE
- 39: COMMAND_SHOW_HARDWARE_DETAILS
- 40: COMMAND_ADD_CALCULATION
- 41: COMMAND_ADD_STORAGE_GROUP
- 42: COMMAND_ADD_LIMIT
- 43: COMMAND_SHOW_AQUISITION_DETAILS
- 44: COMMAND_START_AQUISITION
- 45: COMMAND_START_RECORD_DATA
- 46: COMMAND_PAUSE_AQUISITION
- 47: COMMAND_ONLINEVIEW_ADD_SCREEN
- 48: COMMAND_ONLINEVIEW_ADD_GROUP
- 49: COMMAND_ONLINEVIEW_LAYOUT_FIXED
- 50: COMMAND_ONLINEVIEW_LAYOUT_FREE
- 51: COMMAND_ONLINEVIEW_LAYOUT_1_1
- 52: COMMAND_ONLINEVIEW_LAYOUT_2_1
- 53: COMMAND_ONLINEVIEW_LAYOUT_3_1
- 54: COMMAND_ONLINEVIEW_LAYOUT_2_2_1
- 55: COMMAND_SHOW_SCREEN_LIST
- 56: COMMAND_ONLINEVIEW_ADD_TRACKING_DIAGRAM
- 57: COMMAND_ONLINEVIEW_ADD_XY_DIAGRAM
- 58: COMMAND_ONLINEVIEW_ADD_YT_OSCILSCOPE
- 59: COMMAND_ONLINEVIEW_ADD_ALPHANUMERIC_DISPLAY
- 60: COMMAND_ONLINEVIEW_ADD_BAR_INSTRUMENT
- 61: COMMAND_ONLINEVIEW_ADD_TACHO_INSTRUMENT
- 62: COMMAND_ONLINEVIEW_ADD_POINTER_INSTRUMENT
- 63: COMMAND_ONLINEVIEW_ADD_SLIDER_INSTRUMENT
- 64: COMMAND_ONLINEVIEW_ADD_DIGITAL_INSTRUMENT
- 65: COMMAND_ONLINEVIEW_ADD_SWITCH_INSTRUMENT
- 66: COMMAND_ONLINEVIEW_ADD_VIDEO_DISPLAY
- 67: COMMAND_LOAD_DATA_FILE
- 68: COMMAND_CLEAR_DATA_POOL
- 69: COMMAND_EXPORT_DATA_FILE
- 70: COMMAND_ANALYZE_IN_EXCEL
- 71: COMMAND_SHOW_DATA_FILES
- 72: COMMAND_ANALYSIS_ADD_SCREEN
- 73: COMMAND_ANALYSIS_ADD_GROUP
- 74: COMMAND_ANALYSIS_LAYOUT_FIXED
- 75: COMMAND_ANALYSIS_LAYOUT_FREE
- 76: COMMAND_ANALYSIS_LAYOUT_1_1

- 77: COMMAND_ANALYSIS_LAYOUT_2_1
- 78: COMMAND_ANALYSIS_LAYOUT_3_1
- 79: COMMAND_ANALYSIS_LAYOUT_2_2_1
- 80: COMMAND_ANALYSIS_ADD_LINE_DIAGRAM
- 81: COMMAND_ANALYSIS_ADD_VIDEO_DISPLAY
- 82: COMMAND_ANALYSIS_ZOOM_TYPE_PANNING
- 83: COMMAND_ANALYSIS_ZOOM_TYPE_STRETCHING
- 84: COMMAND_ANALYSIS_ZOOM_TYPE_RUBBERBAND
- 85: COMMAND_ANALYSIS_ZOOM_UNDO
- 86: COMMAND_ANALYSIS_SYNCHRONIZED_ZOOM
- 87: COMMAND_ANALYSIS_OPTIMAL_RANGE
- 88: COMMAND_ANALYSIS_ORIGINAL_RANGE
- 89: COMMAND_ANALYSIS_FIRST_CURSOR
- 90: COMMAND_ANALYSIS_SECOND_CURSOR
- 91: COMMAND_ANALYSIS_START_REPLAY
- 92: COMMAND_ANALYSIS_STOP_REPLAY
- 93: COMMAND_ANALYSIS_SHOW_DATA_FILES
- 94: COMMAND_SCRIPTING_EXPORT
- 95: COMMAND_SCRIPTING_START_RECORDING
- 96: COMMAND_SCRIPTING_STOP_RECORDING
- 97: COMMAND_SCRIPTING_SHOW_SCRIPT_LIBRARY
- 98: COMMAND_SCRIPTING_SHOW_SCRIPT_DETAILS
- 99: COMMAND_ADD_FFT
- 100: COMMAND_ADD_VARIABLE_DOUBLE
- 101: COMMAND_ADD_VARIABLE_STATUS
- 102: COMMAND_ADD_VARIABLE_TEXT
- 103: COMMAND_ADD_FUNCTION_GENERATOR_FILE
- 104: COMMAND_ADD_FUNCTION_GENERATOR_RAMP
- 105: COMMAND_ADD_FUNCTION_GENERATOR_RECTANGLE
- 106: COMMAND_ADD_FUNCTION_GENERATOR_SAWTOOTH
- 107: COMMAND_ADD_FUNCTION_GENERATOR_SINE
- 108: COMMAND_ADD_PID_CONTROLLER
- 109: COMMAND_ONLINEVIEW_ADD_FFT_OSCILSCOPE
- 110: COMMAND_ONLINEVIEW_ADD_BUTTON
- 111: COMMAND_ANALYSIS_ADD_XY_DIAGRAM
- 112: COMMAND_ONLINEVIEW_ADD_LOG_PH_DIAGRAM
- 113: COMMAND_ANALYSIS_ADD_LOG_PH_DIAGRAM
- 114: COMMAND_FULLSCREEN_MODE
- 115: COMMAND_PRINT_REPORT
- 116: COMMAND_PRINT_REPORT_ONCE
- 117: COMMAND_EXPORT_REPORT_AS_PDF
- 118: COMMAND_EXPORT_REPORT_AS_HTML
- 119: COMMAND_EXPORT_REPORT_AS_IMAGE
- 120: COMMAND_EXPORT_REPORT_AS_EXCEL
- 121: COMMAND_ZOOM_REPORT_800
- 122: COMMAND_ZOOM_REPORT_400

- 123: COMMAND_ZOOM_REPORT_300
- 124: COMMAND_ZOOM_REPORT_200
- 125: COMMAND_ZOOM_REPORT_150
- 126: COMMAND_ZOOM_REPORT_100
- 127: COMMAND_ZOOM_REPORT_50
- 128: COMMAND_ADD_REPORT_ELEMENT_ARROW
- 129: COMMAND_ADD_REPORT_ELEMENT_LINE
- 130: COMMAND_ADD_REPORT_ELEMENT_PICTURE
- 131: COMMAND_ADD_REPORT_ELEMENT_TEXT
- 132: COMMAND_ADD_REPORT_ELEMENT_RICH_TEXT
- 133: COMMAND_CHECK_CONFIGURATION
- 134: COMMAND_MAP_HARDWARE
- 135: COMMAND_ADD_RANGE
- 136: COMMAND_ADD_SEQUENCE_CONTROL
- 137: COMMAND_ADD_SEQUENCE_CONTROL_BLOCK
- 138: COMMAND_ADJUST_WITH_TEDS
- 139: COMMAND_ADJUST_OFFSETS
- 140: COMMAND_REPORTING_ELEMENT_LINE_BACK
- 141: COMMAND_REPORTING_ELEMENT_LINE_VERT
- 142: COMMAND_REPORTING_ELEMENT_LINE_HORZ
- 143: COMMAND_REPORTING_ELEMENT_ELLIPSE
- 144: COMMAND_REPORTING_ELEMENT_RECTANGLE
- 145: COMMAND_REPORTING_LAYOUT_DESIGNER
- 146: COMMAND_REPORTING_ZOOM_WHOLE_PAGE
- 147: COMMAND_COMPARE_CONFIGURATIONS
- 148: COMMAND_SHOW_HARDWARE_STATUS_WINDOW
- 149: COMMAND_SHOW_PC_STATUS_WINDOW
- 150: COMMAND_SHOW_STORAGE_STATUS_WINDOW
- 151: COMMAND_SHOW_OUTPUT_WINDOW
- 152: COMMAND_RESET_TEMPLATES
- 153: COMMAND_RESET_FORMULA_POOL
- 154: COMMAND_LOGIN
- 155: COMMAND_LOGOUT
- 156: COMMAND_SIGNALS_SHUNT_CHECK
- 157: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_SAMPLE_COUNT
- 158: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_TIME_AT_LEVEL
- 159: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_FROM_TO_COUNT
- 160: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_LEVEL_CROSSING
- 161: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_TRANSITION_MATRIX
- 162: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_RAINFLOW
- 163: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_SAMPLE_COUNT_COMPOUND
- 164: COMMAND_MEASUREMENT_ADD_CLASSIFICATION_TIME_AT_LEVEL_COMPOUND
- 165: COMMAND_MEASUREMENT_ADD_ROUTER
- 166: COMMAND_ONLINEVIEW_ADD_HISTOGRAM
- 167: COMMAND_ONLINEVIEW_ADD_TABLE
- 168: COMMAND_ONLINEVIEW_ADD_MAP

- 169: COMMAND_ANALYSIS_ADD_MAP
- 170: COMMAND_REPORTING_ADD_ARROW_BACK
- 171: COMMAND_REPORTING_ADD_ARROW_HORIZONTAL
- 172: COMMAND_REPORTING_ADD_ARROW_VERTICAL
- 173: COMMAND_REPORTING_ADD_ARROW_START
- 174: COMMAND_REPORTING_ADD_ARROW_BACK_START
- 175: COMMAND_REPORTING_ADD_ARROW_VERTICAL_START
- 176: COMMAND_REPORTING_ADD_ARROW_HORIZONTAL_START
- 177: COMMAND_REPORTING_ADD_TABLE
- 178: COMMAND_EDIT_PASTE_BEHIND
- 179: COMMAND_EDIT_COPY_TO_FILE
- 180: COMMAND_EDIT_PASTE_FROM_FILE
- 181: COMMAND_GENERAL_IMPORT_WORKING_ENVIRONMENT
- 182: COMMAND_GENERAL_EXPORT_WORKING_ENVIRONMENT
- 183: COMMAND_MEASUREMENT_ADD_SCALING
- 184: COMMAND_ANALYSIS_CURSORS_FREE
- 185: COMMAND_GENERAL_RESET_OBJECT_POOL
- 186: COMMAND_GENERAL_EMPTY_MAP_CACHE
- 187: COMMAND_ANALYSIS_OPTIMAL_X_RANGE
- 188: COMMAND_ONLINEVIEW_ADD_TRAFFIC_ANALYSER
- 189: COMMAND_ONLINEVIEW_ADD_CHANNEL_TABLE
- 190: COMMAND_GENERAL_APP_EXPORT

The following commands need a parameter:

- 666666: COMMAND_SCRIPTING_RUN_SCRIPT
- 666667: COMMAND_SHOW_SCREEN
- 666668: COMMAND_USER_BUTTON
- 666669: COMMAND_ANALYSIS_SHOW_SCREEN
- 666670: COMMAND_MARKER

5.1.2.10 enum UserLevel

Enumeration of available access levels.

Remarks:

The following enumeration values are available:

- 0: Level0 - Access level with the least privileges
- 1: Level1
- 2: Level2
- 3: Level3 - Access level with the highest privileges

Chapter 6

IPEmotionSettings Interface Documentation

6.1 iCloudDrive Interface Reference

Cloud drive interface.

Properties

- bool **Active** [get, set]
Specifies whether the cloud drive will be connected.
- string **LoginName** [get, set]
Returns or sets the login name needed to access the cloud drive.
- string **MappedDrive** [get, set]
Returns or sets the drive letter that should be assigned to the cloud drive.
- string **Name** [get, set]
Returns or sets the name of the cloud drive.
- string **Password** [get, set]
Returns or sets the password.
- string **Url** [get, set]
Returns or sets the URL of the cloud drive.

6.1.1 Detailed Description

Cloud drive interface.

6.1.2 Property Documentation

6.1.2.1 bool Active [get, set]

Specifies whether the cloud drive will be connected.

6.1.2.2 string LoginName [get, set]

Returns or sets the login name needed to access the cloud drive.

6.1.2.3 string MappedDrive [get, set]

Returns or sets the drive letter that should be assigned to the cloud drive.

6.1.2.4 string Name [get, set]

Returns or sets the name of the cloud drive.

The name should be unique.

6.1.2.5 string Password [get, set]

Returns or sets the password.

6.1.2.6 string Url [get, set]

Returns or sets the URL of the cloud drive.

6.2 iCloudDrives Interface Reference

Cloud drive list interface.

Public Member Functions

- [ICloudDrive AddCloudDrive](#) (string nameOfCloudDrive, string url, string loginName, string password, bool activate, string driveLetter)
Adds a new cloud drive to the cloud drives list.
- [IEnumerator GetEnumerator](#) ()
Enumerator for the cloud drives list.
- void [RemoveCloudDrive](#) ([ICloudDrive](#) cloudDriveToRemove)
Removes a cloud drive from the cloud drives list.

Properties

- int [Count](#) [get]
Count of cloud drives in this list.
- [ICloudDrive Item](#) [int index] [get]
Returns the cloud drive with the given (zero based) index from this list.

6.2.1 Detailed Description

Cloud drive list interface.

6.2.2 Member Function Documentation

6.2.2.1 ICloudDrive AddCloudDrive (string *nameOfCloudDrive*, string *url*, string *loginName*, string *password*, bool *activate*, string *driveLetter*)

Adds a new cloud drive to the cloud drives list.

Parameters:

- nameOfCloudDrive*** Name of the new cloud drive.
- url*** URL of the new cloud drive.
- loginName*** Log-in name of the new cloud drive.
- password*** Password of the new cloud drive.
- activate*** Connect cloud drive.
- driveLetter*** Drive letter that should be assigned to the cloud drive.

The name, the URL and the log-in name are required parameters.

Using an empty password is not recommended.

In case the drive letter is not specified the operating system will choose a drive letter when connecting the cloud drive.

Returns:

Interface of the new cloud drive

6.2.2.2 IEnumerator GetEnumerator ()

Enumerator for the cloud drives list.

Returns:

Interface of the cloud drives list enumerator

6.2.2.3 void RemoveCloudDrive (ICloudDrive *cloudDriveToRemove*)

Removes a cloud drive from the cloud drives list.

Parameters:

cloudDriveToRemove Interface of the cloud drive to be removed from the list.

6.2.3 Property Documentation

6.2.3.1 int Count [get]

Count of cloud drives in this list.

6.2.3.2 ICloudDrive Item[int index] [get]

Returns the cloud drive with the given (zero based) index from this list.

Parameters:

index Index of the cloud drive in this list.

Returns:

Interface of the cloud drive.

6.3 IDataPlugIn Interface Reference

Data plug-in interface.

Public Member Functions

- object [GetOptionValue](#) (string optionName)
Returns the value assigned to the named data plug-in option.
- object [GetOptionValue](#) (string optionName)
Returns the value assigned to the named data plug-in option.
- void [SetOptionValue](#) (string optionName, object valueToSet)
Sets the value of the named data plug-in option.
- void [SetOptionValue](#) (string optionName, object valueToSet)
Sets the value of the named data plug-in option.

Properties

- string [FormatId](#) [get]
Returns the data plug-in format identifier.
- [IPlugInOptions Options](#) [get]
Returns the options list of the data plug-in.

6.3.1 Detailed Description

Data plug-in interface.

6.3.2 Member Function Documentation

6.3.2.1 object [GetOptionValue](#) (string *optionName*)

Returns the value assigned to the named data plug-in option.

Parameters:

optionName Name of the data plug-in option.

6.3.2.2 object [GetOptionValue](#) (string *optionName*)

Returns the value assigned to the named data plug-in option.

Parameters:

optionName Name of the data plug-in option.

6.3.2.3 void SetOptionValue (string *optionName*, object *valueToSet*)

Sets the value of the named data plug-in option.

Parameters:

optionName Name of the data plug-in option

valueToSet Value to be set for the named data plug-in option.

6.3.2.4 void SetOptionValue (string *optionName*, object *valueToSet*)

Sets the value of the named data plug-in option.

Parameters:

optionName Name of the data plug-in option

valueToSet Value to be set for the named data plug-in option.

6.3.3 Property Documentation**6.3.3.1 string FormatId [get]**

Returns the data plug-in format identifier.

6.3.3.2 IPlugInOptions Options [get]

Returns the options list of the data plug-in.

6.4 IDataPlugIns Interface Reference

Data plug-in list interface.

Public Member Functions

- IEnumerator [GetEnumerator](#) ()
Enumerator for the data plug-in list.
- IEnumerator [GetEnumerator](#) ()
Enumerator for the data plug-in list.

Properties

- int [Count](#) [get]
Count of data plug-ins in this list.
- [IDataPlugIn Item](#) [int index] [get]
Returns the data plug-in with the given (zero based) index from this list.

6.4.1 Detailed Description

Data plug-in list interface.

6.4.2 Member Function Documentation

6.4.2.1 IEnumerator GetEnumerator ()

Enumerator for the data plug-in list.

Returns:

Interface of the data plug-in list enumerator.

6.4.2.2 IEnumerator GetEnumerator ()

Enumerator for the data plug-in list.

Returns:

Interface of the data plug-in list enumerator.

6.4.3 Property Documentation

6.4.3.1 int Count [get]

Count of data plug-ins in this list.

6.4.3.2 IDataPlugin Item [get]

Returns the data plug-in with the given (zero based) index from this list.

Parameters:

index Index of the data plug-in in this list.

Returns:

Interface of the data plug-in.

6.5 IPlugIn Interface Reference

Plug-in interface.

Public Member Functions

- object [GetOptionValue](#) (string *optionName*)
Returns the value assigned to the named plug-in option.
- void [SetOptionValue](#) (string *optionName*, object *valueToSet*)
Sets the value of the named plug-in option.

Properties

- bool [Active](#) [get, set]
Specifies whether the plug-in is active.
- [IPlugInOptions Options](#) [get]
Returns the options list of the plug-in.
- string [PreferredVersion](#) [get, set]
Specifies the preferred plug-in version.
- [IReferenceObjects ReferenceObjectList](#) [get]
Returns the reference object list of the plug-in.
- string [TypeName](#) [get]
Returns the plug-in type name.

6.5.1 Detailed Description

Plug-in interface.

6.5.2 Member Function Documentation

6.5.2.1 object [GetOptionValue](#) (string *optionName*)

Returns the value assigned to the named plug-in option.

Parameters:

optionName Name of the plug-in option.

6.5.2.2 void [SetOptionValue](#) (string *optionName*, object *valueToSet*)

Sets the value of the named plug-in option.

Parameters:

optionName Name of the plug-in option

valueToSet Value to be set for the named plug-in option.

6.5.3 Property Documentation

6.5.3.1 **bool Active** [get, set]

Specifies whether the plug-in is active.

6.5.3.2 **IPlugInOptions Options** [get]

Returns the options list of the plug-in.

6.5.3.3 **string PreferredVersion** [get, set]

Specifies the preferred plug-in version.

Remarks:

In case the latest installed plug-in version should be used an empty string should be set.

6.5.3.4 **IReferenceObjects ReferenceObjectList** [get]

Returns the reference object list of the plug-in.

6.5.3.5 **string TypeName** [get]

Returns the plug-in type name.

6.6 IPlugInOption Interface Reference

Plug-in options interface.

Properties

- string **Name** [get]
Name of the plug-in option.
- object **Value** [get, set]
Value of the plug-in option.

6.6.1 Detailed Description

Plug-in options interface.

6.6.2 Property Documentation

6.6.2.1 string Name [get]

Name of the plug-in option.

6.6.2.2 object Value [get, set]

Value of the plug-in option.

6.7 IPlugInOptions Interface Reference

Plug-in options list interface.

Public Member Functions

- `IEnumerator GetEnumerator ()`
Enumerator for the plug-in-option list.

Properties

- `int Count [get]`
Count of plug-in-options in this list.
- `IPlugInOption Item [int index] [get]`
Returns the plug-in-option with the given (zero based) index from this list.

6.7.1 Detailed Description

Plug-in options list interface.

6.7.2 Member Function Documentation

6.7.2.1 IEnumerator GetEnumerator ()

Enumerator for the plug-in-option list.

Returns:

Interface of the plug-in-option list enumerator.

6.7.3 Property Documentation

6.7.3.1 int Count [get]

Count of plug-in-options in this list.

6.7.3.2 IPlugInOption Item[int index] [get]

Returns the plug-in-option with the given (zero based) index from this list.

Parameters:

index Index of the plug-in-option in this list.

Returns:

Interface of the plug-in-option.

6.8 IPlugIns Interface Reference

Plug-in list interface.

Public Member Functions

- IEnumeration [GetEnumerator](#) ()
Enumerator for the plug-in list.

Properties

- int [Count](#) [get]
Count of plug-ins in this list.
- [IPlugIn Item](#) [int index] [get]
Returns the plug-in with the given (zero based) index from this list.

6.8.1 Detailed Description

Plug-in list interface.

6.8.2 Member Function Documentation

6.8.2.1 IEnumeration GetEnumerator ()

Enumerator for the plug-in list.

Returns:

Interface of the plug-in list enumerator.

6.8.3 Property Documentation

6.8.3.1 int Count [get]

Count of plug-ins in this list.

6.8.3.2 IPlugIn Item[int index] [get]

Returns the plug-in with the given (zero based) index from this list.

Parameters:

index Index of the plug-in in this list.

Returns:

Interface of the plug-in.

6.9 IReferenceObject Interface Reference

Reference object interface.

Properties

- bool **IsReferencePart** [get, set]
Specifies whether the object will be used in the reference.
- **ConfigurationPriority Priority** [get, set]
Specifies the usage priority of the object.
- string **TypeName** [get]
Reference object type name.

6.9.1 Detailed Description

Reference object interface.

6.9.2 Property Documentation

6.9.2.1 bool IsReferencePart [get, set]

Specifies whether the object will be used in the reference.

6.9.2.2 ConfigurationPriority Priority [get, set]

Specifies the usage priority of the object.

6.9.2.3 string TypeName [get]

Reference object type name.

6.10 IReferenceObjects Interface Reference

Reference object list interface.

Public Member Functions

- IEnumerator [GetEnumerator](#) ()
Enumerator for the reference object list.

Properties

- int [Count](#) [get]
Count of reference objects in this list.
- [IReferenceObject Item](#) [int index] [get]
Returns the reference object with the given (zero based) index from this list.

6.10.1 Detailed Description

Reference object list interface.

6.10.2 Member Function Documentation

6.10.2.1 IEnumerator GetEnumerator ()

Enumerator for the reference object list.

Returns:

Interface of the reference object list enumerator.

6.10.3 Property Documentation

6.10.3.1 int Count [get]

Count of reference objects in this list.

6.10.3.2 IReferenceObject Item[int index] [get]

Returns the reference object with the given (zero based) index from this list.

Parameters:

index Index of the reference object in this list.

Returns:

Interface of the reference object.

6.11 ISettingsHandler Interface Reference

IPEmotion settings handler interface.

Public Member Functions

- void [ActivatePlugIn](#) (string pluginName)
Activates a specific plug-in.
- void [ActivatePlugInVersion](#) (string pluginName, string pluginVersion)
Activates a specific plug-in version.
- void [AddHotKey](#) (UiCommand commandId, Keys key, int parameter)
Adds a hot key to a command.
- bool [ChangeAdminPassword](#) (string oldAdminPassword, string newAdminPassword)
Changes the password of the administrator account.
- void [DeactivatePlugIn](#) (string pluginName)
Deactivates a plug-in.
- void [DisableUserAdministration](#) (string adminPassword)
Disables the user administration.
- void [EnableUserAdministration](#) (string newAdminPassword)
Enables the user administration and sets the administrator password.
- void [ExportEnvironment](#) (string fileName)
Creates a working environment file using the actual settings.
- string [GetPreferredUnit](#) ([PhysicalDimensionTypes](#) physicalDimension)
Returns the preferred unit for a given physical dimension.
- [IUserAccounts](#) [GetUserAccounts](#) (string adminPassword)
Returns the list of registered user accounts in the user administration.
- void [ImportEnvironment](#) (string fileName)
Imports a working environment file, overwriting the actual settings.
- void [Quit](#) ()
Terminates the program.
- void [RemoveHotKey](#) (UiCommand commandId, int parameter)
Removes a hot key from a command.
- void [SaveSettings](#) ()
Saves the modified settings.
- void [SetPreferredUnit](#) ([PhysicalDimensionTypes](#) physicalDimension, string preferredUnit)
Sets the preferred Unit for a given physical dimension.

Properties

- int [AnalysisDiagramCurveQuality](#) [get, set]
Specifies whether the analysis diagrams uses all measured values or only samples when drawing the curves.
- bool [AutoAdministration](#) [get, set]
Returns or sets whether the autotomatic service administration is used.
- [CharacteristicSupport](#) [CharacteristicSupport](#) [get, set]
Returns or sets the way characteristics are handled during import.
- [ICloudDrives](#) [CloudDrives](#) [get]
Returns the list of registered cloud drives.
- string [ConfigDefaultDirectory](#) [get, set]
Specifies the default directory for loading and saving configuration files.
- string [Culture](#) [get, set]
Returns or sets the localization of the IPEmotion UI.
- string [CustomDataDefaultDirectory](#) [get, set]
Specifies the default directory where user specific data is loaded from and stored to.
- string [DatabaseDefaultDirectory](#) [get, set]
Specifies the default directory for database files.
- [IDataPlugIns](#) [DataExportPlugInList](#) [get]
Returns the list of available data export plug-ins.
- [IDataPlugIns](#) [DataImportPlugInList](#) [get]
Returns the list of available data import plug-ins.
- bool [DisplayExtendedTabs](#) [get, set]
Returns or sets whether extended dialog tabs like the format tab of channels are displayed.
- bool [EnableAdditionalWarnings](#) [get, set]
Returns or sets whether additional warnings are displayed in popup-windows when expert mode is active.
- bool [EnableImporterLogging](#) [get, set]
Returns or sets whether warnings and errors appearing during import are written to a log file.
- bool [EnableMapCache](#) [get, set]
Specifies whether caching of map tiles is enabled.
- bool [EnableProtocolChannelConfiguration](#) [get, set]
Returns or sets whether the scaling of channels, imported from a description file, can be edited.
- bool [EnableProtocolConfiguration](#) [get, set]
Returns or sets whether protocol nodes are visible in the hardware tree when the expert mode is active.
- bool [EnableReferenceConfiguration](#) [get, set]
Returns or sets whether the channel references can be configured in the plug-in options when expert mode is active.
- bool [EnableVariableConfiguration](#) [get, set]
Returns or sets whether variable channels are available when the expert mode is active.

- bool [ErrorFreeMeasurementChainAtInit](#) [get, set]
Returns or sets whether an error free configuration is required before any hardware initialization can be done.
- bool [ExpertMode](#) [get, set]
Returns or sets the expert mode.
- string [ExportDefaultDirectory](#) [get, set]
Specifies the default directory where description files should be exported to.
- int [FontSize](#) [get, set]
Returns or sets the font size used by the display elements.
- [HardwareDetectionAction](#) [HardwareDetectionAction](#) [get, set]
Returns or sets which kind of configuration action should be performed after a successful automatic hardware detection on IPEmotion start.
- bool [HardwareDetectionOnStart](#) [get, set]
Returns or sets whether IPEmotion starts a hardware detection on start.
- double [HudTransparency](#) [get, set]
Returns or sets the transparency used by the HUD dialogs.
- bool [IgnoreVerbalTable](#) [get, set]
Returns or sets whether verbal tables are ignored during import of description files.
- string [ImportDefaultDirectory](#) [get, set]
Specifies the default directory where import description files can be found.
- bool [IncludeExternalFilesInConfiguration](#) [get, set]
Returns or sets whether external files are added to the configuration file.
- bool [IncludeOptionsInConfiguration](#) [get, set]
Returns or sets whether option settings that influence the configuration are saved in the configuration file.
- bool [InitViewPagesAtFirstVisibility](#) [get, set]
Specifies whether the online view pages are initialized during configuration loading or delayed until the page is made visible for the first time.
- bool [IsUserAdministrationActive](#) [get]
Returns true in case the user administration is active.
- bool [JobConfiguration](#) [get, set]
Returns or sets whether diagnostic job sequences are displayed in the IPEmotion setup tree and diagnostic jobs in the grid.
- string [LayoutDefaultDirectory](#) [get, set]
Specifies the default directory for report layout files.
- double [LimitMessageDisplayTime](#) [get, set]
Returns or sets the display time of limit exceeded messages in seconds.
- bool [LoadLastConfigurationOnStart](#) [get, set]
Returns or sets whether IPEmotion loads the last used configuration on start.
- int [MapCacheSize](#) [get, set]
Specifies the maximum size of the cache for map tiles in MByte.

- [MAP_IMAGERY_SOURCE](#) [MapImagerySource](#) [get, set]
Specifies the data source from which to load map imagery.
- int [MapTilesProxyTcpPortNumber](#) [get, set]
Specifies the TCP port number used by the map imagery processing unit.
- int [MaxPollingListCount](#) [get, set]
Returns or sets the number of polling lists that should be created when importing CCP/XCP signals.
- bool [MergeDataFilesOnImport](#) [get, set]
Specifies whether data files are to be connected when loading a data set.
- double [NoValueTimeout](#) [get, set]
Returns or sets the NoValue timeout in seconds.
- int [OnlineYtDiagramHistoryBufferDepth](#) [get, set]
Specifies the online yt charts history buffer depth.
- int [OnlineYtDiagramHistoryBufferResolution](#) [get, set]
Specifies the online yt charts history buffer resolution factor.
- bool [PerformanceOptimization](#) [get, set]
Specifies whether the performance of the display elements should be optimized at the expense of the design.
- [IPlugIns](#) [PluginList](#) [get]
Returns the list of available plug-ins.
- string [ProjectDataDefaultDirectory](#) [get, set]
Specifies the default directory for loading and saving project data files.
- string [RawDataDefaultDirectory](#) [get, set]
Specifies the default directory where temporary measurement data files will be saved.
- string [ScriptingDefaultDirectory](#) [get, set]
Specifies the default directory where IPEmotion loads and stores cript files.
- [SelectedPlotHighlightType](#) [SelectedPlotHighlightType](#) [get, set]
Specifies the highlightning type of selected curves in yt-charts.
- bool [SharedTimeChannel](#) [get, set]
Indicates whether cyclical time channels, that have the same sampling rate and the same data set, should be combined in the display and while they are exported.
- bool [ShowToolTips](#) [get, set]
Returns or sets whether tool tips are shown.
- [SignalConfigurationMode](#) [SignalConfigurationMode](#) [get, set]
Returns or sets the configuration mode.
- string [Skin](#) [get, set]
Returns or sets the name of the skin to be used by the IPEmotion UI.
- int [SplitFileSize](#) [get, set]
Returns or sets the maximum size of an acquisition file before it is split.
- string [StorageDataDefaultDirectory](#) [get, set]

Specifies the default directory where measurement data files can be loaded from.

- bool [TimeChannellsAbsolut](#) [get, set]

Specifies whether the IPEmotion time channels are displayed in relative time or absolute time.

- [TimeZoneDisplayType](#) [TimeZoneDisplayType](#) [get, set]

Specifies whether absolute times are displayed referring to the local time zone of the analysys system, the local time zone of the measurement system or as UTC time.

- bool [UseDisplayNames](#) [get, set]

Specifies whether the display instruments use the display name instead of the channel name.

- bool [UseExtendedCursorValues](#) [get, set]

Specifies whether the values of the cursors should be displayed in a separate window instead of directly beside the crosshair.

- bool [UseHardwareDetectionAction](#) [get, set]

Returns or sets if a specific configuration action should be performed after a succesfull automatic hardware detection on IPEmotion start.

- bool [UseStandardDialogs](#) [get, set]

Specifies whether the Windows standard file open/save dialog is user or the IPEmotion specific variant.

6.11.1 Detailed Description

IPEmotion settings handler interface.

6.11.2 Member Function Documentation

6.11.2.1 void ActivatePlugIn (string *pluginName*)

Activates a specific plug-in.

Parameters:

pluginName Name of the plug-in to be activated

6.11.2.2 void ActivatePlugInVersion (string *pluginName*, string *pluginVersion*)

Activates a specific plug-in version.

Parameters:

pluginName Name of the plug-in to be activated

pluginVersion Version of the plug-in to be activated

Remarks:

In case no version is given the default version will be used.

6.11.2.3 void AddHotKey (UiCommand *commandId*, Keys *key*, int *parameter*)

Adds a hot key to a command.

Parameters:

commandId Command id

key Hotkeys key code

parameter Optional parameter (index of script, screen, ...)

Remarks:

In case the command does not support a parameter the parameter value must be set to -1.

6.11.2.4 bool ChangeAdminPassword (string *oldAdminPassword*, string *newAdminPassword*)

Changes the password of the administrator account.

Parameters:

oldAdminPassword Old password of the administrator account

newAdminPassword New password of the administrator account

Remarks:

The old password must match the password of the administrator account, otherwise no change will happen.

It is recommended to assign a non-empty password for the administrator account.

Returns:

True in case the password was changed, otherwise false.

6.11.2.5 void DeactivatePlugin (string *pluginName*)

Deactivates a plug-in.

Parameters:

pluginName Name of the plug-in to be deactivated

6.11.2.6 void DisableUserAdministration (string *adminPassword*)

Disables the user administration.

Parameters:

adminPassword Password of the administrator account

Remarks:

The password must match the password of the administrator account, otherwise no change will happen.

6.11.2.7 void EnableUserAdministration (string *newAdminPassword*)

Enables the user administration and sets the administrator password.

Parameters:

newAdminPassword Password to be set for the administrator account

Remarks:

It is recommended to assign a non-empty password for the administrator account.

6.11.2.8 void ExportEnvironment (string *fileName*)

Creates a working environment file using the actual settings.

Parameters:

fileName Name of the working environment file to be saved (*.ief)

Remarks:

The actual settings are automatically saved before the environment file is created.

6.11.2.9 string GetPreferredUnit (PhysicalDimensionTypes *physicalDimension*)

Returns the preferred unit for a given physical dimension.

Parameters:

physicalDimension Typ of the physical dimension

6.11.2.10 IUserAccounts GetUserAccounts (string *adminPassword*)

Returns the list of registered user accounts in the user administration.

Parameters:

adminPassword Password of the administrator account

Remarks:

The password must match the password of the administrator account, otherwise no user account list will be returned.

6.11.2.11 void ImportEnvironment (string *fileName*)

Imports a working environment file, overwriting the actual settings.

Parameters:

fileName Name of the working environment file to be imported (*.ief)

6.11.2.12 void Quit ()

Terminates the program.

Remarks:

Modified settings will not be saved automatically. Call SaveSettings to save the modifications.

6.11.2.13 void RemoveHotKey (UiCommand *commandId*, int *parameter*)

Removes a hot key from a command.

Parameters:

commandId Command id

parameter Optional parameter (index of script, screen, ...)

Remarks:

In case the command does not support a parameter the parameter value must be set to -1.

6.11.2.14 void SaveSettings ()

Saves the modified settings.

6.11.2.15 void SetPreferredUnit (PhysicalDimensionTypes *physicalDimension*, string *preferredUnit*)

Sets the preferred Unit for a given physical dimension.

Parameters:

physicalDimension Type of the physical dimension

preferredUnit Symbol of the preferred unit

6.11.3 Property Documentation

6.11.3.1 int AnalysisDiagramCurveQuality [get, set]

Specifies whether the analysis diagrams uses all measured values or only samples when drawing the curves.

Remarks:

Only values from 0 up to 100 are allowed.

The value 100 means that all measured values are used to draw the curves.

A value of 0 specifies that only a minimum number of samples is used.

6.11.3.2 bool AutoAdministration [get, set]

Returns or sets whether the autotomatic service administration is used.

6.11.3.3 CharacteristicSupport CharacteristicSupport [get, set]

Returns or sets the way characteristics are handled during import.

6.11.3.4 ICloudDrives CloudDrives [get]

Returns the list of registered cloud drives.

6.11.3.5 string ConfigDefaultDirectory [get, set]

Specifies the default directory for loading and saving configuration files.

6.11.3.6 string Culture [get, set]

Returns or sets the localization of the IPEmotion UI.

Remarks:

The localization is specified by a language tag that conforms to IETF RFC 3066. The language part is a two letter subtag according to ISO standard 639 and the country part is a two letter subtag according to the country code specified in ISO 3166.

6.11.3.7 string CustomDataDefaultDirectory [get, set]

Specifies the default directory where user specific data is loaded from and stored to.

6.11.3.8 string DatabaseDefaultDirectory [get, set]

Specifies the default directory for database files.

6.11.3.9 IDataPlugins DataExportPluginList [get]

Returns the list of available data export plug-ins.

6.11.3.10 IDataPlugins DataImportPluginList [get]

Returns the list of available data import plug-ins.

6.11.3.11 bool DisplayExtendedTabs [get, set]

Returns or sets whether extended dialog tabs like the format tab of channels are displayed.

6.11.3.12 bool EnableAdditionalWarnings [get, set]

Returns or sets whether additional warnings are displayed in popup-windows when expert mode is active.

6.11.3.13 bool EnableImporterLogging [get, set]

Returns or sets whether warnings and errors appearing during import are written to a log file.

6.11.3.14 bool EnableMapCache [get, set]

Specifies whether caching of map tiles is enabled.

Remarks:

Obsolete. This configuration option is no longer available. Please use property 'MapImagerySource' if applicable.

6.11.3.15 bool EnableProtocolChannelConfiguration [get, set]

Returns or sets whether the scaling of channels, imported from a description file, can be edited.

6.11.3.16 bool EnableProtocolConfiguration [get, set]

Returns or sets whether protocol nodes are visible in the hardware tree when the expert mode is active.

6.11.3.17 bool EnableReferenceConfiguration [get, set]

Returns or sets whether the channel references can be configured in the plug-in options when expert mode is active.

Remarks:

No longer supported.

6.11.3.18 bool EnableVariableConfiguration [get, set]

Returns or sets whether variable channels are available when the expert mode is active.

6.11.3.19 bool ErrorFreeMeasurementChainAtInit [get, set]

Returns or sets whether an error free configuration is required before any hardware initialization can be done.

6.11.3.20 bool ExpertMode [get, set]

Returns or sets the expert mode.

6.11.3.21 string ExportDefaultDirectory [get, set]

Specifies the default directory where description files should be exported to.

6.11.3.22 int FontSize [get, set]

Returns or sets the font size used by the display elements.

Remarks:

The font size can be set to 7 (S), 9 (M), 11 (L) or 13 (XL) point.

6.11.3.23 HardwareDetectionAction HardwareDetectionAction [get, set]

Returns or sets which kind of configuration action should be performed after a successful automatic hardware detection on IPEmotion start.

6.11.3.24 bool HardwareDetectionOnStart [get, set]

Returns or sets whether IPEmotion starts a hardware detection on start.

6.11.3.25 double HudTransparency [get, set]

Returns or sets the transparency used by the HUD dialogs.

Remarks:

The transparency can be set to values between 0 % (totally opaque) and 30 %.

6.11.3.26 bool IgnoreVerbalTable [get, set]

Returns or sets whether verbal tables are ignored during import of description files.

6.11.3.27 string ImportDefaultDirectory [get, set]

Specifies the default directory where import description files can be found.

6.11.3.28 bool IncludeExternalFilesInConfiguration [get, set]

Returns or sets whether external files are added to the configuration file.

6.11.3.29 bool IncludeOptionsInConfiguration [get, set]

Returns or sets whether option settings that influence the configuration are saved in the configuration file.

6.11.3.30 bool InitViewPagesAtFirstVisibility [get, set]

Specifies whether the online view pages are initialized during configuration loading or delayed until the page is made visible for the first time.

6.11.3.31 bool IsUserAdministrationActive [get]

Returns true in case the user administration is active.

6.11.3.32 bool JobConfiguration [get, set]

Returns or sets whether diagnostic job sequences are displayed in the IPEmotion setup tree and diagnostic jobs in the grid.

6.11.3.33 string LayoutDefaultDirectory [get, set]

Specifies the default directory for report layout files.

6.11.3.34 double LimitMessageDisplayTime [get, set]

Returns or sets the display time of limit exceeded messages in seconds.

Remarks:

Only positive values are allowed. In case 0 is set the messages stay visible until closed by the user.

6.11.3.35 bool LoadLastConfigurationOnStart [get, set]

Returns or sets whether IPEmotion loads the last used configuration on start.

6.11.3.36 int MapCacheSize [get, set]

Specifies the maximum size of the cache for map tiles in MByte.

Remarks:

Obsolete. This configuration option is no longer available.

6.11.3.37 MAP_IMAGERY_SOURCE MapImagerySource [get, set]

Specifies the data source from which to load map imagery.

6.11.3.38 int MapTilesProxyTcpPortNumber [get, set]

Specifies the TCP port number used by the map imagery processing unit.

6.11.3.39 int MaxPollingListCount [get, set]

Returns or sets the number of polling lists that should be created when importing CCP/XCP signals.

Remarks:

Values between 0 and 4 are supported, which means the support of polling lists can be disabled or up to 4 polling lists can be used. The usage of multiple DAQ lists requires a plug-in that supports this feature.

6.11.3.40 bool MergeDataFilesOnImport [get, set]

Specifies whether data files are to be connected when loading a data set.

6.11.3.41 double NoValueTimeout [get, set]

Returns or sets the NoValue timeout in seconds.

Remarks:

Only values from 0 up to 5.0 are allowed.

6.11.3.42 int OnlineYtDiagramHistoryBufferDepth [get, set]

Specifies the online yt charts history buffer depth.

Remarks:

Allowed values are in the range from 1 to 50.

6.11.3.43 int OnlineYtDiagramHistoryBufferResolution [get, set]

Specifies the online yt charts history buffer resolution factor.

Remarks:

Allowed values are in the range from 1 to 10.

6.11.3.44 bool PerformanceOptimization [get, set]

Specifies whether the performance of the display elements should be optimized at the expense of the design.

6.11.3.45 IPlugIns PluginList [get]

Returns the list of available plug-ins.

6.11.3.46 string ProjectDataDefaultDirectory [get, set]

Specifies the default directory for loading and saving project data files.

6.11.3.47 string RawDataDefaultDirectory [get, set]

Specifies the default directory where temporary measurement data files will be saved.

6.11.3.48 string ScriptingDefaultDirectory [get, set]

Specifies the default directory where IPEmotion loads and stores cript files.

6.11.3.49 SelectedPlotHighlightType SelectedPlotHighlightType [get, set]

Specifies the highlightning type of selected curves in yt-charts.

6.11.3.50 bool SharedTimeChannel [get, set]

Indicates whether cyclical time channels, that have the same sampling rate and the same data set, should be combined in the display and while they are exported.

6.11.3.51 bool ShowToolTips [get, set]

Returns or sets whether tool tips are shown.

6.11.3.52 SignalConfigurationMode SignalConfigurationMode [get, set]

Returns or sets the configuration mode.

6.11.3.53 string Skin [get, set]

Returns or sets the name of the skin to be used by the IPEmotion UI.

Remarks:

The following skins are available:

- "IPETRONIK2": Blue
- "IPETRONIK-DarkBlue-Skin": Dark blue
- "IPETRONIK-Black-Skin": Black
- "IPETRONIK-LightBlue-Skin": Light blue
- "IPETRONIK-Grey-Skin": Grey
- "IPETRONIK-Bright-Skin": Bright

6.11.3.54 int SplitFileSize [get, set]

Returns or sets the maximum size of an acquisition file before it is split.

Remarks:

Only values from 1048576 (1 MByte) up to 1073741824 (1 GByte) are allowed.

6.11.3.55 string StorageDataDefaultDirectory [get, set]

Specifies the default directory where measurement data files can be loaded from.

6.11.3.56 bool TimeChannellsAbsolut [get, set]

Specifies whether the IPEmotion time channels are displayed in relative time or absolute time.

6.11.3.57 TimeZoneDisplayType TimeZoneDisplayType [get, set]

Specifies whether absolute times are displayed referring to the local time zone of the analysys system, the local time zone of the measurement system or as UTC time.

Remarks:

This setting affects the display of absolute times only if the loaded data contains the necessary informations.

6.11.3.58 bool UseDisplayNames [get, set]

Specifies whether the display instruments use the display name instead of the channel name.

6.11.3.59 bool UseExtendedCursorValues [get, set]

Specifies whether the values of the cursors should be displayed in a separate window instead of directly beside the crosshair.

6.11.3.60 bool UseHardwareDetectionAction [get, set]

Returns or sets if a specific configuration action should be performed after a successful automatic hardware detection on IPEmotion start.

6.11.3.61 bool UseStandardDialogs [get, set]

Specifies whether the Windows standard file open/save dialog is used or the IPEmotion specific variant.

6.12 IUserAccount Interface Reference

User account interface.

Properties

- bool **Active** [get, set]
Specifies whether the user can log in.
- bool **CanConfigure** [get, set]
Specifies whether the user can change the configuration.
- bool **CanModifyStatus** [get, set]
Specifies whether the user can modify the IPEmotion status.
- string **Password** [get, set]
Returns or sets the user password.
- **UserLevel UserLevel** [get, set]
Specifies the access level of the user.
- string **Username** [get]
Returns the name of the user.

6.12.1 Detailed Description

User account interface.

6.12.2 Property Documentation

6.12.2.1 bool Active [get, set]

Specifies whether the user can log in.

6.12.2.2 bool CanConfigure [get, set]

Specifies whether the user can change the configuration.

6.12.2.3 bool CanModifyStatus [get, set]

Specifies whether the user can modify the IPEmotion status.

6.12.2.4 string Password [get, set]

Returns or sets the user password.

6.12.2.5 UserLevel UserLevel [get, set]

Specifies the access level of the user.

6.12.2.6 string Username [get]

Returns the name of the user.

6.13 IUserAccounts Interface Reference

User account list interface.

Public Member Functions

- [IUserAccount AddUser](#) (string nameOfUser, string userPassword, [UserLevel](#) accessLevel, bool enableConfig, bool enableStatusModification)
Adds a new user to the user account list.
- IEnumerator [GetEnumerator](#) ()
Enumerator for the user account list.
- void [RemoveUser](#) (string nameOfUser)
Removes a user from the user account list.

Properties

- int [Count](#) [get]
Count of user accounts in this list.
- [IUserAccount Item](#) [int index] [get]
Returns the user account with the given (zero based) index from this list.

6.13.1 Detailed Description

User account list interface.

6.13.2 Member Function Documentation

6.13.2.1 IUserAccount AddUser (string *nameOfUser*, string *userPassword*, UserLevel *accessLevel*, bool *enableConfig*, bool *enableStatusModification*)

Adds a new user to the user account list.

Parameters:

nameOfUser Name of the new user.

userPassword Password of the new user.

accessLevel Access level of the new user.

enableConfig Specifies whether the new user may modify configurations.

enableStatusModification Specifies whether the new user may change the IPEmotion status.

Returns:

Interface of the new user account

6.13.2.2 IEnumerator GetEnumerator ()

Enumerator for the user account list.

Returns:

Interface of the user account list enumerator

6.13.2.3 void RemoveUser (string *nameOfUser*)

Removes a user from the user account list.

Parameters:

nameOfUser Name of the user.

6.13.3 Property Documentation

6.13.3.1 int Count [get]

Count of user accounts in this list.

6.13.3.2 IUserAccount Item[int index] [get]

Returns the user account with the given (zero based) index from this list.

Parameters:

index Index of the user account in this list.

Returns:

Interface of the user account.

Index

- ActivatePlugIn
 - ISettingsHandler, [42](#)
- ActivatePlugInVersion
 - ISettingsHandler, [42](#)
- Active
 - ICloudDrive, [23](#)
 - IPlugIn, [32](#)
 - IUserAccount, [53](#)
- AddCloudDrive
 - ICloudDrives, [25](#)
- AddHotKey
 - ISettingsHandler, [42](#)
- AddUser
 - IUserAccounts, [55](#)
- AnalysisDiagramCurveQuality
 - ISettingsHandler, [45](#)
- AutoAdministration
 - ISettingsHandler, [45](#)
- CanConfigure
 - IUserAccount, [53](#)
- CanModifyStatus
 - IUserAccount, [53](#)
- ChangeAdminPassword
 - ISettingsHandler, [43](#)
- CharacteristicSupport
 - IPEmotionSettings, [14](#)
 - ISettingsHandler, [45](#)
- CloudDrives
 - ISettingsHandler, [45](#)
- ConfigDefaultDirectory
 - ISettingsHandler, [46](#)
- ConfigurationPriority
 - IPEmotionSettings, [15](#)
- Count
 - ICloudDrives, [26](#)
 - IDataPlugIns, [29](#)
 - IPlugInOptions, [34](#)
 - IPlugIns, [35](#)
 - IReferenceObjects, [37](#)
 - IUserAccounts, [56](#)
- Culture
 - ISettingsHandler, [46](#)
- CustomDataDefaultDirectory
 - ISettingsHandler, [46](#)
- DatabaseDefaultDirectory
 - ISettingsHandler, [46](#)
- DataExportPlugInList
 - ISettingsHandler, [46](#)
- DataImportPlugInList
 - ISettingsHandler, [46](#)
- DeactivatePlugIn
 - ISettingsHandler, [43](#)
- DisableUserAdministration
 - ISettingsHandler, [43](#)
- DisplayExtendedTabs
 - ISettingsHandler, [46](#)
- EnableAdditionalWarnings
 - ISettingsHandler, [46](#)
- EnableImporterLogging
 - ISettingsHandler, [46](#)
- EnableMapCache
 - ISettingsHandler, [46](#)
- EnableProtocolChannelConfiguration
 - ISettingsHandler, [47](#)
- EnableProtocolConfiguration
 - ISettingsHandler, [47](#)
- EnableReferenceConfiguration
 - ISettingsHandler, [47](#)
- EnableUserAdministration
 - ISettingsHandler, [43](#)
- EnableVariableConfiguration
 - ISettingsHandler, [47](#)
- ErrorFreeMeasurementChainAtInit
 - ISettingsHandler, [47](#)
- ExpertMode
 - ISettingsHandler, [47](#)
- ExportDefaultDirectory
 - ISettingsHandler, [47](#)
- ExportEnvironment
 - ISettingsHandler, [44](#)
- FontSize
 - ISettingsHandler, [47](#)
- FormatId
 - IDataPlugIn, [28](#)
- GetEnumerator
 - ICloudDrives, [25](#)
 - IDataPlugIns, [29](#)
 - IPlugInOptions, [34](#)
 - IPlugIns, [35](#)
 - IReferenceObjects, [37](#)
 - IUserAccounts, [55](#)
- GetOptionValue
 - IDataPlugIn, [27](#)
 - IPlugIn, [31](#)
- GetPreferredUnit
 - ISettingsHandler, [44](#)
- GetUserAccounts
 - ISettingsHandler, [44](#)
- HardwareDetectionAction

- IPeMotionSettings, 15
- ISettingsHandler, 47
- HardwareDetectionOnStart
 - ISettingsHandler, 48
- HudTransparency
 - ISettingsHandler, 48
- ICloudDrive, 23
 - Active, 23
 - LoginName, 23
 - MappedDrive, 24
 - Name, 24
 - Password, 24
 - Url, 24
- ICloudDrives, 25
 - AddCloudDrive, 25
 - Count, 26
 - GetEnumerator, 25
 - Item, 26
 - RemoveCloudDrive, 26
- IDataPlugIn, 27
 - FormatId, 28
 - GetOptionValue, 27
 - Options, 28
 - SetOptionValue, 27, 28
- IDataPlugIns, 29
 - Count, 29
 - GetEnumerator, 29
 - Item, 29
- IgnoreVerbalTable
 - ISettingsHandler, 48
- ImportDefaultDirectory
 - ISettingsHandler, 48
- ImportEnvironment
 - ISettingsHandler, 44
- IncludeExternalFilesInConfiguration
 - ISettingsHandler, 48
- IncludeOptionsInConfiguration
 - ISettingsHandler, 48
- InitViewPagesAtFirstVisibility
 - ISettingsHandler, 48
- IPeMotionSettings, 13
 - CharacteristicSupport, 14
 - ConfigurationPriority, 15
 - HardwareDetectionAction, 15
 - MAP_IMAGERY_SOURCE, 15
 - PhysicalDimensionTypes, 15
 - SelectedPlotHighlightType, 16
 - SignalConfigurationMode, 16
 - TimeZoneDisplayType, 16
 - UiCommand, 17
 - UserLevel, 21
- IPlugIn, 31
 - Active, 32
 - GetOptionValue, 31
 - Options, 32
 - PreferredVersion, 32
 - ReferenceObjectList, 32
 - SetOptionValue, 31
 - TypeName, 32
- IPlugInOption, 33
 - Name, 33
 - Value, 33
- IPlugInOptions, 34
 - Count, 34
 - GetEnumerator, 34
 - Item, 34
- IPlugIns, 35
 - Count, 35
 - GetEnumerator, 35
 - Item, 35
- IReferenceObject, 36
 - IsReferencePart, 36
 - Priority, 36
 - TypeName, 36
- IReferenceObjects, 37
 - Count, 37
 - GetEnumerator, 37
 - Item, 37
- ISettingsHandler, 38
 - ActivatePlugIn, 42
 - ActivatePlugInVersion, 42
 - AddHotKey, 42
 - AnalysisDiagramCurveQuality, 45
 - AutoAdministration, 45
 - ChangeAdminPassword, 43
 - CharacteristicSupport, 45
 - CloudDrives, 45
 - ConfigDefaultDirectory, 46
 - Culture, 46
 - CustomDataDefaultDirectory, 46
 - DatabaseDefaultDirectory, 46
 - DataExportPlugInList, 46
 - DataImportPlugInList, 46
 - DeactivatePlugIn, 43
 - DisableUserAdministration, 43
 - DisplayExtendedTabs, 46
 - EnableAdditionalWarnings, 46
 - EnableImporterLogging, 46
 - EnableMapCache, 46
 - EnableProtocolChannelConfiguration, 47
 - EnableProtocolConfiguration, 47
 - EnableReferenceConfiguration, 47
 - EnableUserAdministration, 43
 - EnableVariableConfiguration, 47
 - ErrorFreeMeasurementChainAtInit, 47
 - ExpertMode, 47
 - ExportDefaultDirectory, 47
 - ExportEnvironment, 44
 - FontSize, 47
 - GetPreferredUnit, 44
 - GetUserAccounts, 44
 - HardwareDetectionAction, 47
 - HardwareDetectionOnStart, 48
 - HudTransparency, 48
 - IgnoreVerbalTable, 48
 - ImportDefaultDirectory, 48
 - ImportEnvironment, 44
 - IncludeExternalFilesInConfiguration, 48
 - IncludeOptionsInConfiguration, 48
 - InitViewPagesAtFirstVisibility, 48

- IsUserAdministrationActive, [48](#)
- JobConfiguration, [48](#)
- LayoutDefaultDirectory, [48](#)
- LimitMessageDisplayTime, [48](#)
- LoadLastConfigurationOnStart, [49](#)
- MapCacheSize, [49](#)
- MapImagerySource, [49](#)
- MapTilesProxyTcpPortNumber, [49](#)
- MaxPollingListCount, [49](#)
- MergeDataFilesOnImport, [49](#)
- NoValueTimeout, [49](#)
- OnlineYtDiagramHistoryBufferDepth, [49](#)
- OnlineYtDiagramHistoryBufferResolution, [50](#)
- PerformanceOptimization, [50](#)
- PlugInList, [50](#)
- ProjectDataDefaultDirectory, [50](#)
- Quit, [44](#)
- RawDataDefaultDirectory, [50](#)
- RemoveHotKey, [45](#)
- SaveSettings, [45](#)
- ScriptingDefaultDirectory, [50](#)
- SelectedPlotHighlightType, [50](#)
- SetPreferredUnit, [45](#)
- SharedTimeChannel, [50](#)
- ShowToolTips, [50](#)
- SignalConfigurationMode, [50](#)
- Skin, [51](#)
- SplitFileSize, [51](#)
- StorageDataDefaultDirectory, [51](#)
- TimeChannellsIsAbsolut, [51](#)
- TimeZoneDisplayType, [51](#)
- UseDisplayNames, [51](#)
- UseExtendedCursorValues, [51](#)
- UseHardwareDetectionAction, [51](#)
- UseStandardDialogs, [52](#)
- IsReferencePart
 - IReferenceObject, [36](#)
- IsUserAdministrationActive
 - ISettingsHandler, [48](#)
- Item
 - ICloudDrives, [26](#)
 - IDataPlugIns, [29](#)
 - IPlugInOptions, [34](#)
 - IPlugIns, [35](#)
 - IReferenceObjects, [37](#)
 - IUserAccounts, [56](#)
- IUserAccount, [53](#)
 - Active, [53](#)
 - CanConfigurate, [53](#)
 - CanModifyStatus, [53](#)
 - Password, [53](#)
 - UserLevel, [53](#)
 - Username, [53](#)
- IUserAccounts, [55](#)
 - AddUser, [55](#)
 - Count, [56](#)
 - GetEnumerator, [55](#)
 - Item, [56](#)
 - RemoveUser, [56](#)
- JobConfiguration
 - ISettingsHandler, [48](#)
 - LayoutDefaultDirectory
 - ISettingsHandler, [48](#)
 - LimitMessageDisplayTime
 - ISettingsHandler, [48](#)
 - LoadLastConfigurationOnStart
 - ISettingsHandler, [49](#)
 - LoginName
 - ICloudDrive, [23](#)
 - MAP_IMAGERY_SOURCE
 - IPeemotionSettings, [15](#)
 - MapCacheSize
 - ISettingsHandler, [49](#)
 - MapImagerySource
 - ISettingsHandler, [49](#)
 - MappedDrive
 - ICloudDrive, [24](#)
 - MapTilesProxyTcpPortNumber
 - ISettingsHandler, [49](#)
 - MaxPollingListCount
 - ISettingsHandler, [49](#)
 - MergeDataFilesOnImport
 - ISettingsHandler, [49](#)
 - Name
 - ICloudDrive, [24](#)
 - IPlugInOption, [33](#)
 - NoValueTimeout
 - ISettingsHandler, [49](#)
 - OnlineYtDiagramHistoryBufferDepth
 - ISettingsHandler, [49](#)
 - OnlineYtDiagramHistoryBufferResolution
 - ISettingsHandler, [50](#)
 - Options
 - IDataPlugIn, [28](#)
 - IPlugIn, [32](#)
 - Password
 - ICloudDrive, [24](#)
 - IUserAccount, [53](#)
 - PerformanceOptimization
 - ISettingsHandler, [50](#)
 - PhysicalDimensionTypes
 - IPeemotionSettings, [15](#)
 - PlugInList
 - ISettingsHandler, [50](#)
 - PreferredVersion
 - IPlugIn, [32](#)
 - Priority
 - IReferenceObject, [36](#)
 - ProjectDataDefaultDirectory
 - ISettingsHandler, [50](#)
 - Quit
 - ISettingsHandler, [44](#)
 - RawDataDefaultDirectory
 - ISettingsHandler, [50](#)
 - ReferenceObjectList

- IPlugIn, [32](#)
- RemoveCloudDrive
 - ICloudDrives, [26](#)
- RemoveHotKey
 - ISettingsHandler, [45](#)
- RemoveUser
 - IUserAccounts, [56](#)
- SaveSettings
 - ISettingsHandler, [45](#)
- ScriptingDefaultDirectory
 - ISettingsHandler, [50](#)
- SelectedPlotHighlightType
 - IPEmotionSettings, [16](#)
 - ISettingsHandler, [50](#)
- SetOptionValue
 - IDataPlugIn, [27, 28](#)
 - IPlugIn, [31](#)
- SetPreferredUnit
 - ISettingsHandler, [45](#)
- SharedTimeChannel
 - ISettingsHandler, [50](#)
- ShowToolTips
 - ISettingsHandler, [50](#)
- SignalConfigurationMode
 - IPEmotionSettings, [16](#)
 - ISettingsHandler, [50](#)
- Skin
 - ISettingsHandler, [51](#)
- SplitFileSize
 - ISettingsHandler, [51](#)
- StorageDataDefaultDirectory
 - ISettingsHandler, [51](#)
- TimeChannellsAbsolut
 - ISettingsHandler, [51](#)
- TimeZoneDisplayType
 - IPEmotionSettings, [16](#)
 - ISettingsHandler, [51](#)
- TypeName
 - IPlugIn, [32](#)
 - IReferenceObject, [36](#)
- UiCommand
 - IPEmotionSettings, [17](#)
- Url
 - ICloudDrive, [24](#)
- UseDisplayNames
 - ISettingsHandler, [51](#)
- UseExtendedCursorValues
 - ISettingsHandler, [51](#)
- UseHardwareDetectionAction
 - ISettingsHandler, [51](#)
- UserLevel
 - IPEmotionSettings, [21](#)
 - IUserAccount, [53](#)
- Username
 - IUserAccount, [53](#)
- UseStandardDialogs
 - ISettingsHandler, [52](#)
- Value
 - IPlugInOption, [33](#)