PMS.Cliptest

Manual

Version 21.1.1.0

MAR Rostock GmbH

23.02.2021

Software version management

Version	Date	Status	Range of Validity	Creator	Note
1.0.0.0	17.05.2016	Release	MAR / Global	Schuster / Seidel	Software release V1.0.0.0
1.0.1.0	31.05.2016	Release	MAR / Global	Schuster / Seidel	small changes, for details see changelog
1.1.0.1	18.10.2016	Release	MAR / Global	Schuster / Seidel	Improve MasterDataManager user experience, Add Production Server Interface
1.1.1.0	24.10.2016	Release	MAR / Global	Schuster / Seidel	Add SAP Import to MasterDataManager
1.2.0.0	04.11.2016	Release	MAR / Global	Schuster / Seidel	Add Backup feature
1.2.9.0	12.05.2017	Release	MAR / Global	Schuster / Seidel	Refactoring Scanner Interface
1.3.0.0	01.06.2017	Release	MAR / Global	Schuster / Seidel	Adding module list scan
2.0.0.0	21.01.2019	Release	MAR / Global	Seidel	Currently in release process
2.1.0.0	19.07.2019	Release	MAR / Global	Schuster / Seidel	Adding Clip Groups, Extended operator header, Localization
2.2.2.0	15.01.2020	Release	MAR / Global	Schuster / Seidel	Improve user experience
20.1.1.0	11.03.2020	Release	MAR / Global	Schuster / Seidel	Different colors for cliptest and emptytest, New backends, performance and usibility improvements
20.2.1.0	06.05.2020	Release	MAR / Global	Schuster / Seidel	PMS.Cliptest.Eventing extension
20.3.1.0	14.07.2020	Release	MAR / Global	Schuster / Seidel	Added PMS.Ret device for Eventing. Communication is based on CAN Bus, Switch Outputs and receive Input Signals
20.3.2.0	26.08.2020	Release	MAR / Global	Schuster / Seidel	Improved PMS.Ret eventing device
20.4.1.4	18.12.2020	Release	MAR / Global	Schuster / Seidel	Minor Bugfixes
21.1.1.0	14.01.2021	Release	MAR / Global	Schuster / Seidel	Added Opc.Ua.Client device for Eventing, Added BrainBox device, Improve Eventing configuration, Added Eventing Conditions

Document version management

Version	Date	Status	Range of Validity	Creator	Document
1.0.0.0	17.05.2016	Release	MAR / Global	Schuster / Seidel	Publishing of Document V1.0.0.0
1.0.1.0	31.05.2016	Release	MAR / Global	Schuster	Documentation for menu option *Manage Master Data* added
1.0.2.0	21.07.2016	Release	MAR / Global	Schuster	Dongle Licensing modified
1.0.2.1	24.08.2016	Release	MAR / Global	Schuster	Hardware requirements modified
1.1.0.1	18.10.2016	Release	MAR / Global	Schuster	MasterDataManager and ServiceConfigurator changes
1.1.1.0	24.10.2016	Release	MAR / Global	Schuster	MasterDataManager SAP Import
1.2.0.0	04.11.2016	Release	MAR / Global	Schuster / Seidel	Add Backup feature
1.2.9.0	12.05.2017	Release	MAR / Global	Schuster	Refactoring Scanner Interface
1.3.0.0	01.06.2017	Release	MAR / Global	Schuster	Adding module list scan
1.3.0.1	06.06.2017	Release	MAR / Global	Schuster	Adding version upgrade instructions
1.3.0.2	15.06.2017	Release	MAR / Global	Schuster	MasterData – PinStatus file add hint for LED number and address combinations
1.3.0.3	21.01.2019	Release	MAR / Global	Seidel	Refactored references according to production interfaces
1.4.0.0	27.02.2019	Release	MAR / Global	Seidel	Added chapter dialogs, refreshed screenshots
2.1.0.0	19.07.2019	Release	MAR / Global	Schuster	Added chapter clip groups and localization
2.2.2.0	15.01.2020	Release	MAR / Global	Schuster	Improved user experience
2.2.3.0	10.03.2020	Release	MAR / Global	Seidel	Added chapter Label confirmation and dialog section printing confirmation
20.1.1.0	11.03.2020	Release	MAR / Global	Schuster	Different colors for cliptest and emptytest, New backends, extend printing interface
20.2.1.0	06.05.2020	Release	MAR / Global	Schuster / Seidel	PMS.Cliptest.Eventing extension
20.3.1.0	14.07.2020	Release	MAR / Global	Schuster / Seidel	Added chapter 6.5 PMS.Ret eventing configuration
20.3.2.0	26.08.2020	Release	MAR / Global	Schuster / Seidel	Refreshed chapter 6.5
21.1.1.0	14.01.2020	Release	MAR / Global	Schuster / Seidel	Added Opc.Ua.Client device, Added BrainBox device

Referenced Documents

ID	Document
1	PMS_PrintProcessor_Documentation_v1.5.3.0.pdf
2	PMS_PrintProcessor_SamplesDoc_v1.1.0.0.pdf

Content

1   Preamble

1.1   License Royalty

PMS is a registered trademark of the MAR Rostock GmbH. To use the full functionality of **PMS.Cliptest** you'll have to purchase a license. Fill out the registration document and send it to your local distributor to obtain a license.

1.2   License Activation

There are two possibilities to add new licenses to the **PMS.Cliptest.Service**. You are able to add a license to the local machine or use an usb license dongle. To use a dongle license you have only to plug in the dongle on the machine where the **PMS.Cliptest.Service** should be executed. The process to add new local licenses is described in chapter 1.2.1.

1.2.1   License Key

To add licenses with a license key you have to use the **PMS.Cliptest.ServiceConfigurator**. If the security option is enabled, you will be asked for the password to proceed starting the **ServiceConfigurator** (For further information see chapter 5.3). At the tab page **Licensing** you will find the currently installed licenses and are able to add new local licenses.
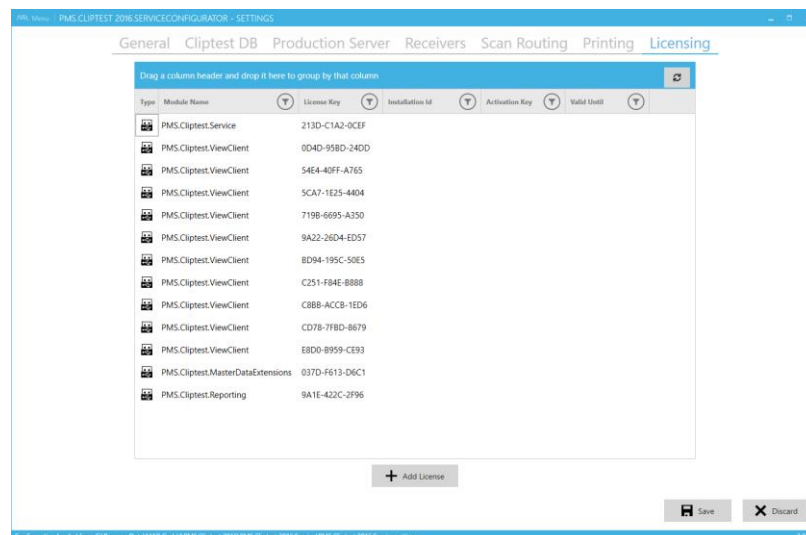
figure 1: PMS.Cliptest.ServiceConfigurator Licensing

The button **Add New Licenses** starts a wizard to add new licenses. The following screenshot shows the initial dialog of the wizard.

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

figure 2: Licensing Wizard - step welcome

Click next to proceed the wizard. In the next step you are asked to read carefully the license terms and conditions and agree to proceed.
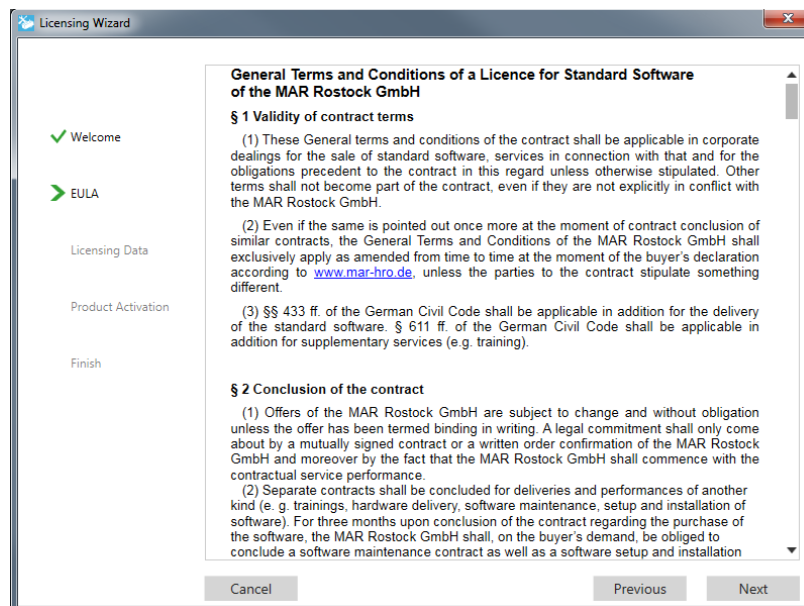
figure 3: Licensing Wizard - step terms and conditions

In the following step, you can add your license keys. Select the corresponding software product (module) to your license key, enter the license key and add the module to the list.

After adding all your license keys for this **PMS.Cliptest.Service** instance click **Next** to proceed.
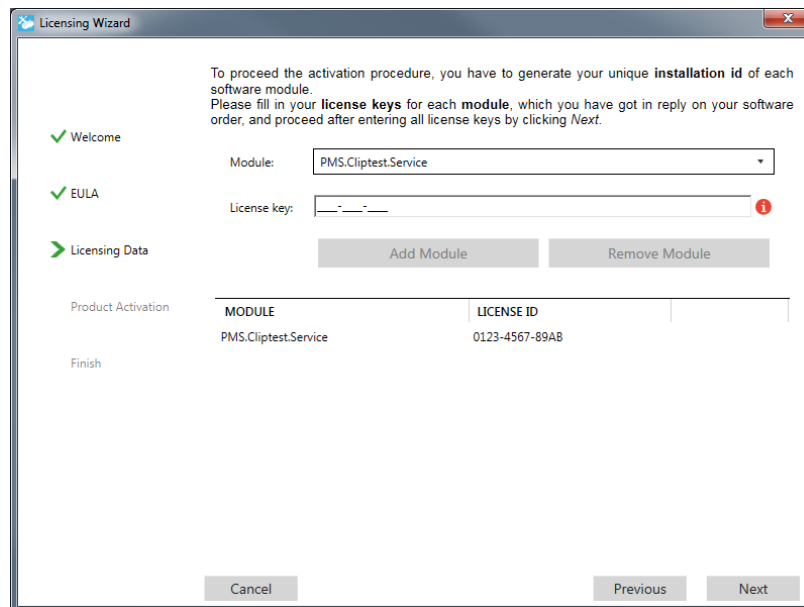
figure 4: Licensing Wizard – step licensing data

All added license keys have to be activated in the following step. There are two possibilities to activate a license key. If your computer is connected to the internet, you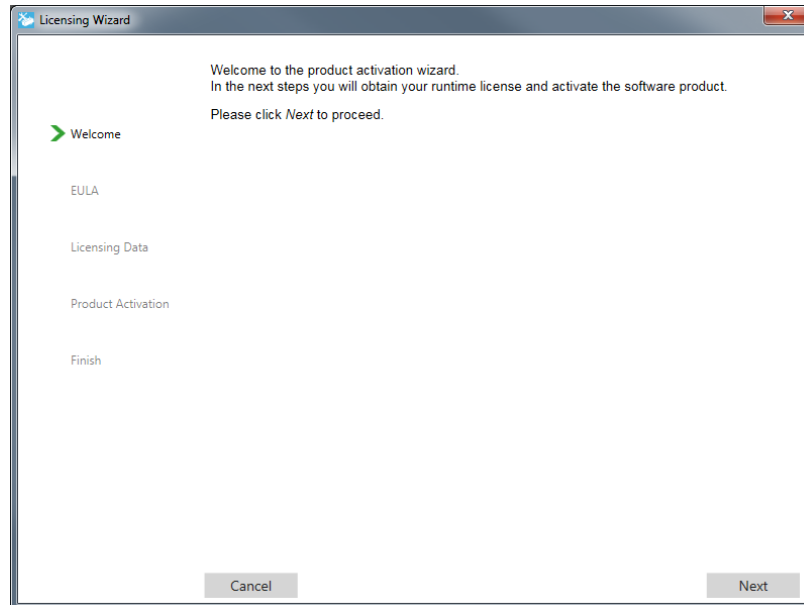 can use the automatic activation. Otherwise, visit the Website for the activation (*http://pac.mar-hro.de*). Enter there the license key and the generated Installation id, check the confirmation box and click submit. You will get the activation Key that has to be inserted manually. Repeat this procedure for each entered license key by clicking the **Next license** button. If you have a problem during the activation process of a license key, you can skip it.
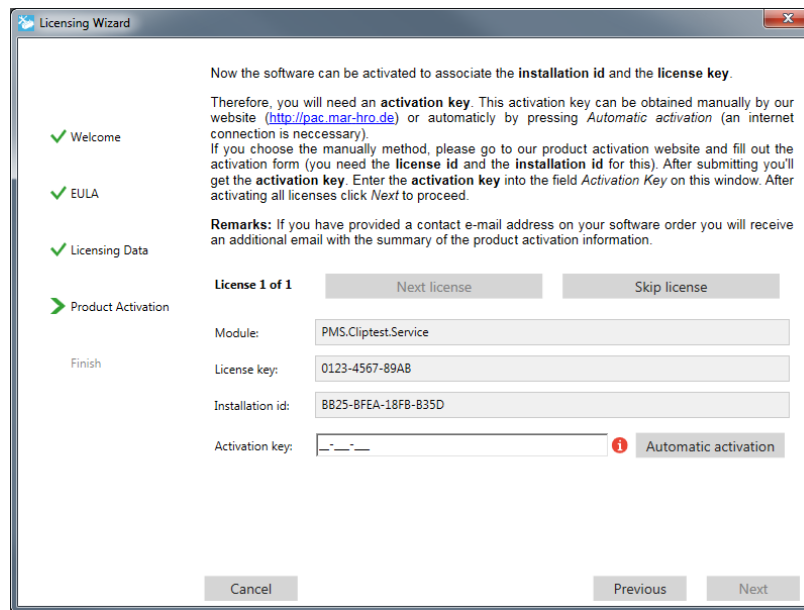
figure 5: Licensing Wizard – step product activation

If you have activated (or skipped) all licenses, click **Next** to proceed. Finish the last step of the activation process with a click on the **Finish** button.
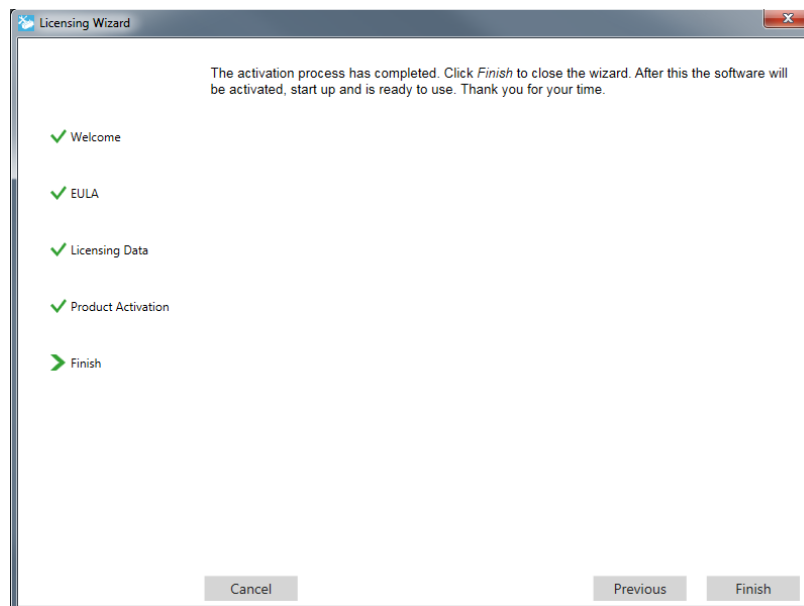
figure 6: Licensing Wizard - step finish

1.2.2 USB Dongle

The second possibility to add new licenses to the **PMS.Cliptest.Service** is the usb dongle license system. You are able to add licenses from multiple usb dongles. The **PMS.Cliptest.Service** collect the licenses from all plugged in dongles. The available licenses are shown in the **PMS.Cliptest.ServiceConfigurator** on the tab page **Licensing**.

Attention:

The license file (pms.lic) must remain on the dongle.
Do not modify, copy or override this file. Any modifications of the license file can invalidate all stored licenses.

In the following chapter you will find information about activating upgraded licenses.

1.2.3 Upgrade existing licenses

If you have purchased upgrade licenses, it is necessary to activate the new licenses with the **PMS.Licensing.UpgradeTool** (short: UpgradeTool) before you can use them. You will find the **PMS.Licensing.UpgradeTool** on your license dongle.
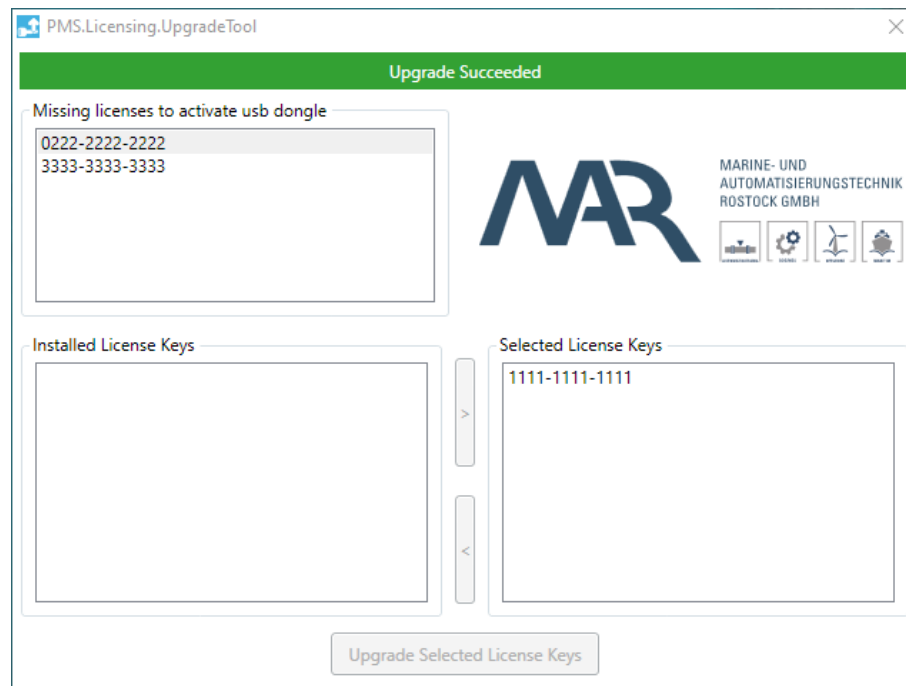
figure 7: PMS.Licensing.UpgradeTool

After starting the **PMS.Licensing.UpgradeTool** you are able to see which license keys are required to activate the new upgrade licenses. You have to run the **UpgradeTool** on **each** computer, which uses a license key you have selected to upgrade. If a required license key is installed on the computer where the **UpgradeTool** is running, you will find the license in the list on the left bottom side. To continue the upgrade process, select the licenses on the left side and move them to the selected license keys. Click **Upgrade Selected License Keys** to proceed. In the next step, you have to confirm the upgrade process.

After upgrading all required licenses, you can use the usb dongle for licensing **PMS.Cliptest**.

1.2.3.1 License Upgrade Examples

The following chapter shows two examples of the upgrade process from **PMS.Cliptest** licenses to **PMS.Cliptest licenses**. The examples are independent of the version of **PMS.Cliptest**.

Example 1 – Single license PMS.Cliptest to PMS.Cliptest

The first scenario describes the upgrade process from one PMS.Cliptest license to a *PMS.Cliptest* license. At the ordering procedure, you have to select the license key, which should be upgraded. With your order, you will get an usb license dongle, which contains the new *PMS.Cliptest* licenses. These licenses are locked. To unlock the licenses, run the *PMS.Licensing.UpgradeTool* on the computer where the specified upgrade license key is currently installed. Follow the instructions to activate the license. After the upgrade process, the new *PMS.Cliptest* licenses are active. To use the activated licenses, plug the usb dongle to the computer where *PMS.Cliptest* should run.

Example 2 – Two licenses of PMS.Cliptest to PMS.Cliptest with additional ViewClient

The second example describes the upgrade process from two PMS.Cliptest licenses to a *PMS.Cliptest* license with an additional ViewClient. At the ordering procedure, you have to select two license keys that should be upgraded. With your order, you will get a usb license dongle which contains the new *PMS.Cliptest* licenses. These licenses are locked until all upgraded license keys are activated with the *PMS.Licensing.UpgradeTool*. To unlock the licenses, run the *PMS.Licensing.UpgradeTool* on each computer where the specified upgrade license keys are currently installed. Follow the instructions to activate the license. After the upgrade process the new *PMS.Cliptest* licenses are active. To use the activated licenses, plug the usb dongle to the computer where *PMS.Cliptest* should run.

1.2.4 PMS.Cliptest DEMO

The *PMS.Cliptest.Service* can be started in a DEMO mode, by adding a demo license with the license wizard. This license expires after 50 days. The DEMO mode has no limitation regarding the functionality.

1.3 Designated use of PMS.Cliptest

PMS.Cliptest was developed to test the correct assembling of orders or order types on assembly boards. For further descriptions, imagine an assembly line where products are manufactured by assembling modules. **PMS.Cliptest** can now finally check if all modules are correctly installed to enhance the quality of the produced products.

Therefore, the assembly boards must have been equipped with radio receiver clips which are going to be pressed when a module is installed. **PMS.Cliptest** can uses multiple radio receivers to obtain the current states of all clips (pressed / unpressed) and is now able to check these data against the configured master data. If all necessary clips are pressed (depending on master data) the result of the test will be ok. If the test time is over and not all clips are pressed correctly the test result will be not ok.

Mainly there are two types of tests that can be processed by **PMS.Cliptest**:

Clip Test: A Clip Test is the main test of **PMS.Cliptest**. The test can run some time (configurable) and shows on a panel, which clips should be pressed and which clips are already pressed to obtain a positive test result. If all clips are correctly pressed while a running test, this test is automatically finished with a positive test result. If the test time is over and not all clips are pressed correctly, the test result will be negative.

Empty Test: An Empty Test can be used to ensure that all clips on a board are currently unpressed. This may be used to grant an initial state of the board before start assembling a new product on there. This test can also run some time (configurable) and shows all clips of the board and their states on a display. If all clips are unpressed while a running empty test, this test is automatically finished with a positive test result. If the test time is over and not all clips are unpressed, the test result will be negative.

1.4 System structure

The PMS.Cliptest application is divided into several parts. The main part is the **PMS.Cliptest.Service** (short: Service). It handles the test logic, the master data import and the communication to the **PMS.PrintProcessor.Service** (short: PrintProcessor). The configuration of the Service can be changed by usage of the **PMS.Cliptest.ServiceConfigurator** (short: ServiceConfigurator). For the interaction with the service you can use the **PMS.Cliptest.ViewClient** (short: ViewClient) or the **PMS.Cliptest.MasterDataManager** (short: MasterDataManager). All print jobs will be handled by the **PMS.PrintProcessor.Service**.
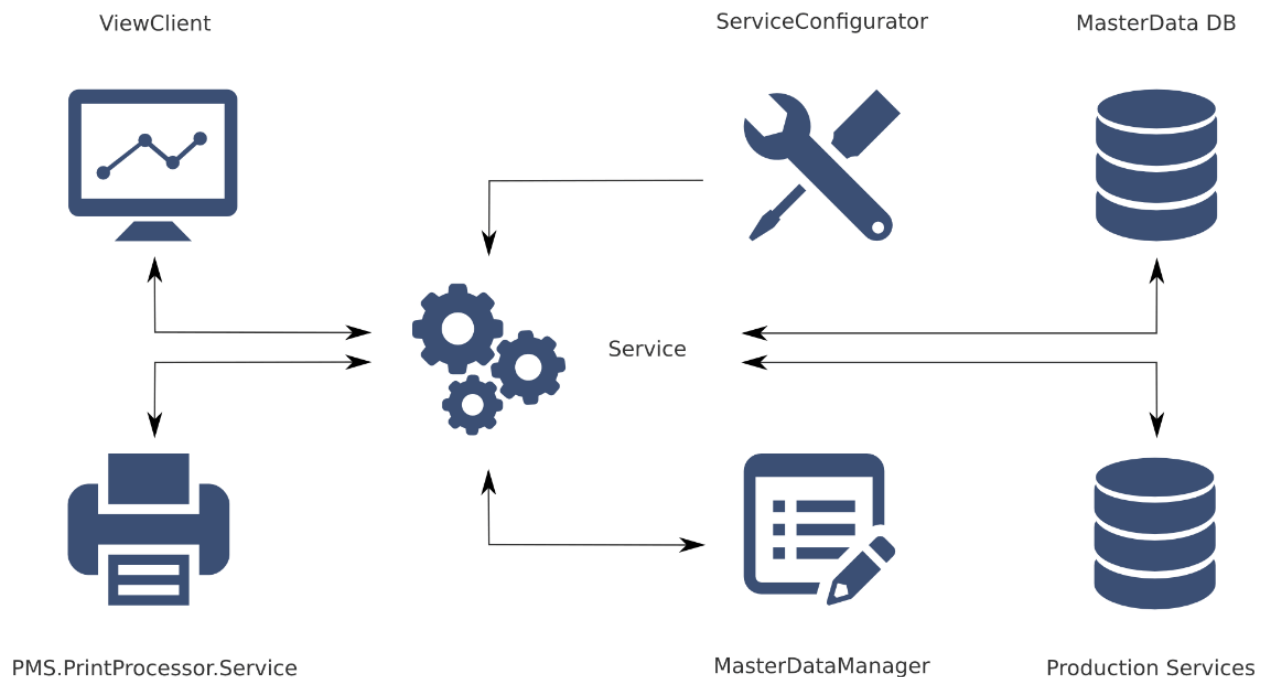

figure 8: system structure overview

The radio receivers, which transmit the clip press events to the **Service,** will be directly connected to it.

1.5 Program Modes

PMS.Cliptest can be used in various combinations.

1.5.1 Without Production Server

In this configuration, **PMS.Cliptest** acts as a single workplace instance. Start commands for clip- or empty-tests are received from a scanner connected to the **ViewClient**. The scanned barcodes are identifying the assembly boards which you want to use for the test.

1.5.2 With Production Server

The production server can be a higher level manufacturing execution system (e.g. MES) to which **PMS.Cliptest** is able to communicate with.

In this configuration, **PMS.Cliptest** uses the local database to handle master data but uses also the configured production server interface for the communication to any supported production system. Start commands for clip tests or empty tests as well as the list of clips for a test can be pre-generated and received from the production server.

2 Program Runtime Requirements

2.1 Hardware Requirements

If all application not running on the same computer a 100/1000MBit Network card is required.

- X86/X64-Platform with 1,0 Ghz   (min. 2,0 Ghz recommended)
- 2 GB RAM   (4 GB recommended)
- 500MB free disk space for PMS.Cliptest and distributed Database
- 1000MB free disk space for PMS.Cliptest
- recommended screen resolution 1920 x 1080

2.1.1 PMS.Cliptest.Service

- If running with RS232 receivers at least one free COM-port (RS232) for each receiver

2.1.2 PMS.Cliptest.ViewClient

- If running with a RS232 barcode scanner an additional free COM-Port (RS232)
- If running with a USB barcode scanner a free USB Port

2.1.3 PMS.Cliptest.MasterDataManager

- For scanning addresses from clips is a free COM-Port (RS232) for barcode-scanner required

2.2 Software Requirements

- Microsoft Windows 10 (Version 1903) (PMS.Cliptest was tested to work with this operating system)

- Microsoft .net Framework 4.7.2

  (If the required version of the .net Framework is not installed, the setup shows an error. You will find the setup for the .net Framework at the installation medium)

- Access to the default communication ports (for further information see *chapter 5.3.2.1*)

  - o Basic communication: 50181   TCP   **required**
  - o Http communication: 50182   TCP   optional
  - o Client Interface: 50183   TCP   optional

3   Installation guide

3.3   *Installing PMS.Cliptest*

This chapter describes how to install PMS.Cliptest with the delivered PMS.Cliptest Setup. Make sure that a Microsoft SQL Server (e.g. Express version) is installed.

3.3.1 Deployment options

This section describes possible installation scenarios with an explanation of the required licenses.

3.3.1.1 Classic use case

The typical installation scenario has two computers, one for the Empty Test and one for the Clip Test.On one machine you have to install all applications (Service, ServiceConfigurator, MasterDataManager,ViewClient) and on the second machine you have to install only the ViewClient. For this scenario you need at least one PMS.Cliptest.Service license and one PMS.Cliptest.ViewClient license.

3.3.1.2 Standalone use case

In the standalone use case, you have to install all applications on a single machine. You need at least one PMS.Cliptest.Service license to use this configuration.
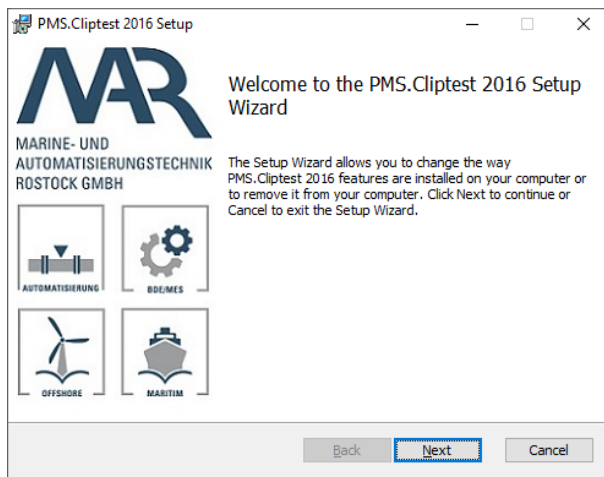
3.3.2 Setup Steps

Click on "PMS.Cliptest Setup.exe" to start the setup wizard. The setup wizard automatically detects whether you have installed the Microsoft .net Framework. If it could not be found, you have to install it manually.
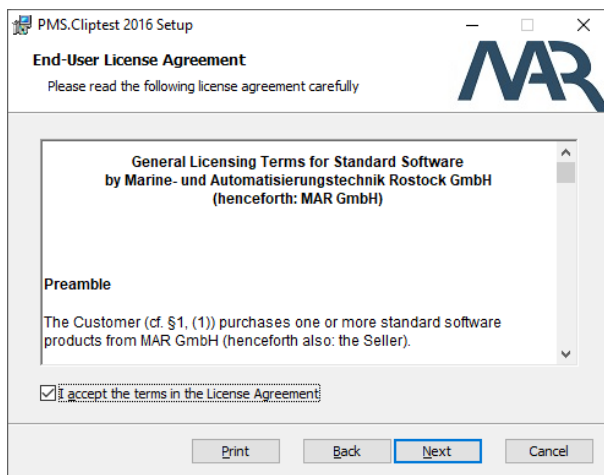
Follow the steps above:

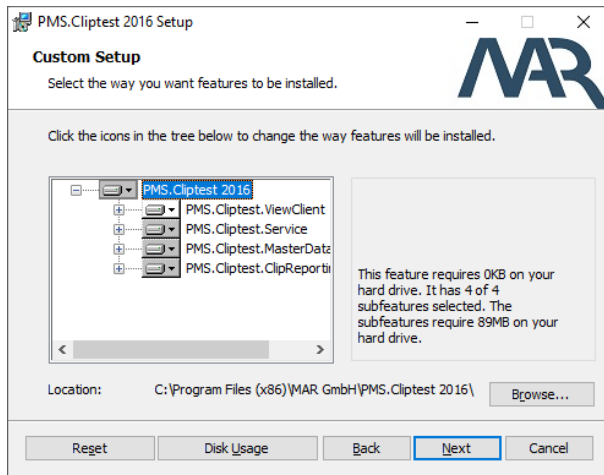<u>dialog</u>   <u>steps</u>


Click Next to proceed to the license agreement

Accept the License agreement and press Next to proceed to the feature dialog

Choose in this step the all components which you want to install.

By Default, the setup installs the PMS.Cliptest.Service, PMS.Cliptest.ViewClient and the PMS.Cliptest.MasterDataManager.

You can customize the installation by selecting single components. The install location can be changed by selecting PMS.Cliptest.
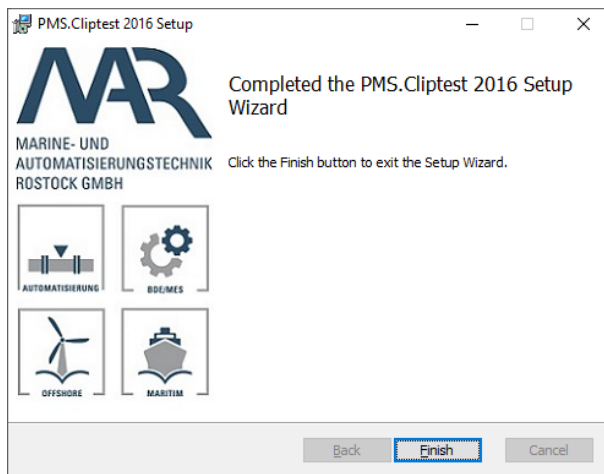
After the installation is completed close the dialog by a click on the finish button. The PMS.Cliptest.Service is not started automatically without a reboot. The PMS.Cliptest.ServiceConfigurator can be used to start the PMS.Cliptest.Service (See chapter 5.3).

3.4 Upgrading PMS.Cliptest

During the upgrade progress to newer versions of PMS.Cliptest it can be possible that a database upgrade is required. If you have an older version of PMS.Cliptest installed, you have to upgrade the database before you can use the new version. Please process the following steps to upgrade the database:

1. Create a backup of the current master data with the MasterDataManager (ref. see 5.2.4.1 Backup)
2. Uninstall the current version of PMS.Cliptest
3. Install the new version of PMS.Cliptest
4. Create a new database with the ServiceConfigurator (the old database can be deleted if update was successful). Save the changes and restart the service.
5. Restore the backup with the MasterDataManager

In some cases it is also possible to reconfigure the service application.

4   Program Master Data Requirements

This chapter describes the necessary master data to run the program effectively. The master data have to be prepared as CSV-files and can be imported by using the master data import. Refer to the following sub-chapters for detailed descriptions about the necessary CSV-File data.

The structure of the files is defined as CSV (comma separated variables). You may use comma or semicolon to separate the data of columns inside the file. Default separator is a comma. To set a new separator, refer to chapter 5.2.3.3.

You may also write the column headers into file as the first line. Default comment token is #. To set a new comment token, Refer to chapter 5.2.3.3 for detailed information.

The following master data must be prepared when using *PMS.Cliptest* without any production server (Production Server Interface NONE):

PinStatus   file
Cliplist   file
Variants   file
Coordinates file   (optionally)

The following master data must be prepared when using *PMS.Cliptest* in combination with an external production server:

PinStatus   file
Coordinates file   (optionally)
Modules   file   (depending on production server interface)

4.1 PinStatus file

The PinStatus file defines the structure of every board. It defines all LEDs (Clips) and their addresses for every board. Therefore, it defines the correlation between the physical Clip (Clip-address) on the board and the virtual Clip-Number defined in the master data for a module of an order (CAD data of the order). It is also used in runtime to retentively save the states of all radio receiver clips (ON/OFF). This file must contain all clip IDs of all assembly boards to be processed with the used instance of **PMS.Cliptest**. The combination of LED, Address and Board have to be unique, but it is possible that you can have the same LED number with multiple addresses or the same address with multiple LED numbers on the same board. You are able to split the information into multiple files, e.g. one file for each board.

Necessary CSV-File structure

Column name	Column type	Comment
LED	Short	Unique Clip position on Board (LED)
Address	String	Unique ID of radio receiver clip (e.g. 001099BB)
Board	String	Assembly board where the clip is located on
ForceTest	Boolean	**Optionally:** True/False; True if clip must be tested in mode "Unpressed clips", even if the clip was already pressed before.
KSKType	String	**Optionally:** The KSKType is an additional identifier to differentiate same LED/Addresses on the same Board (This option can be used for specific types of harnesses). Leave it empty if there are no duplicate LED/Addresses on the same Board. This feature can only be used in combination with the ForceTest.
Description	String	**Optionally:** The Description is an optionally property to get more information about a Clip. Leave it empty if there is no description. This feature can only be used in combination with the KSKType.
ClipGroup	String	**Optionally:** This field configures the ClipGroup. Leave it empty if the Clip is not a member of a group. This

feature can only be used in combination with the option Description.

The file can contain from three columns up to seven. If the file contains no "ForceTest" column, the related value is initialized with "False". If the file contains no "KSKType" column, the value will be empty. The "KSKType" value can only be used in combination with "ForceTest". The option "Description" can only be used in combination with "KSKType" and the option "ClipGroup" can only be used in combination with "Description".

4.1.1 Clip Groups

The Clip Groups feature allows the user to assign clips to different groups. Each clip can only be member of one group. If during a running test all clips of a group are in the state *IO*, the status of the clips switch to *GRPIO* and the status will not change until the end of the test (Clips that are in the state *NLIO* will be not considered for this calculation). Changes of the clip status will be not considered in the clip state calculation. The behavior is similar to the test mode *clips must be pressed once* for all clips in a group. The clip groups can be configured in the master data files.
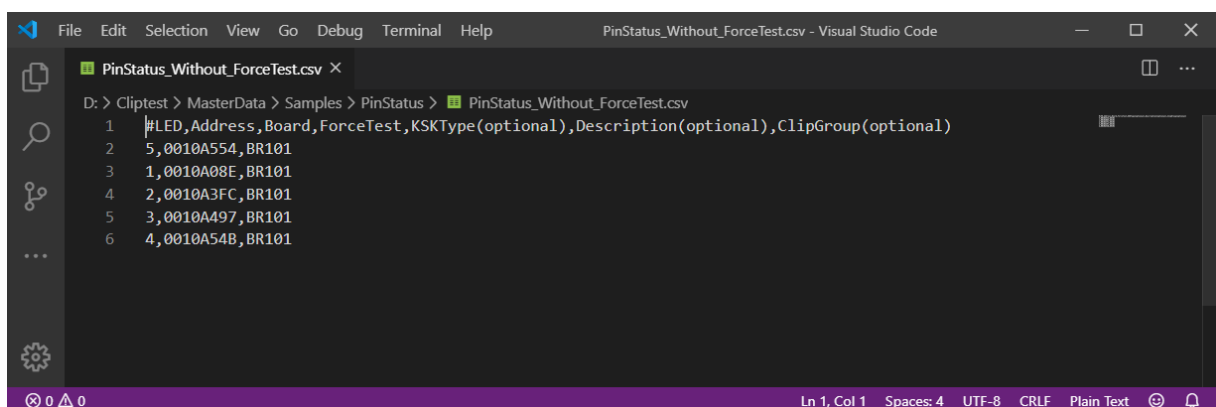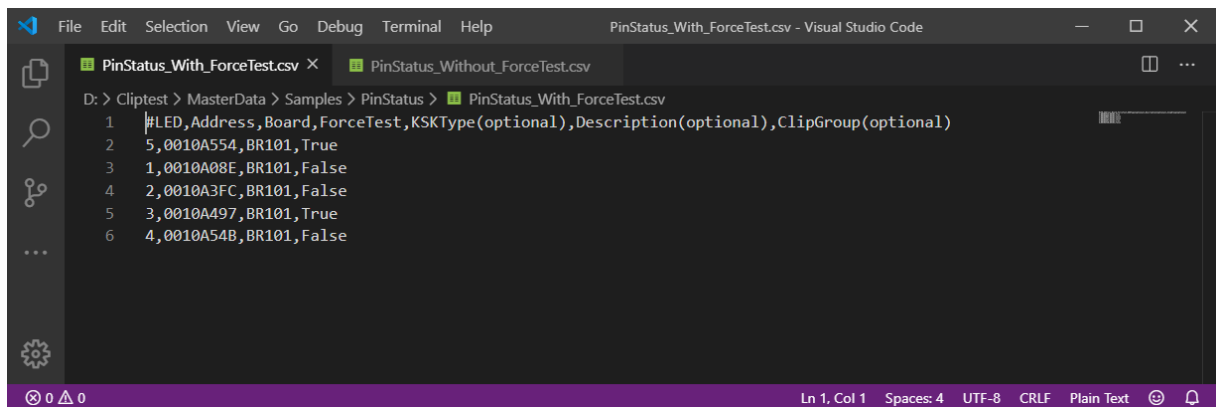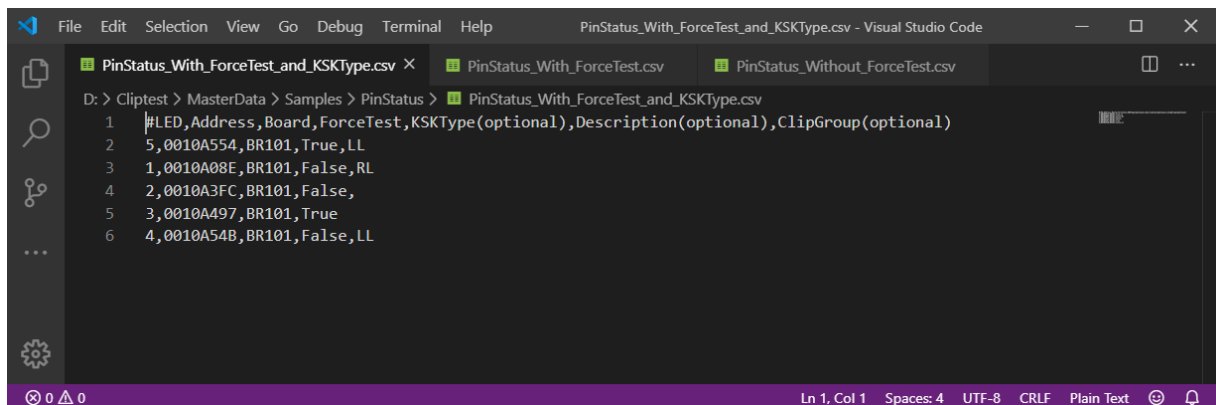
4.1.2 Example CSV-File structures

figure 9: PinStatus file without additional information

figure 10: PinStatus file with ForceTest Flag

figure 11: PinStatus File with ForceTest Flag and KSK Type

The optional columns Description and ClipGroup can be also appended after the KSKType column.

4.2 Coordinates file

The coordinates file can optionally be used to store additional coordinate information for the clip positions (LEDs) on a board. This data is additionally displayed on the panel of PMS.Cliptest. You may use it to define an X/Y coordinate system to simplify the search for a physical clip position on a board. The configured values are not case sensitive. All letters will be converted to upper letters. You are able to split the information into multiple files, e.g. one file for each board.

If you want to use the Matrix layout for the PMS.Cliptest.ViewClient check the requirements for the coordinates (see 5.1.4.3 Matrix layout).

Necessary CSV-File structure

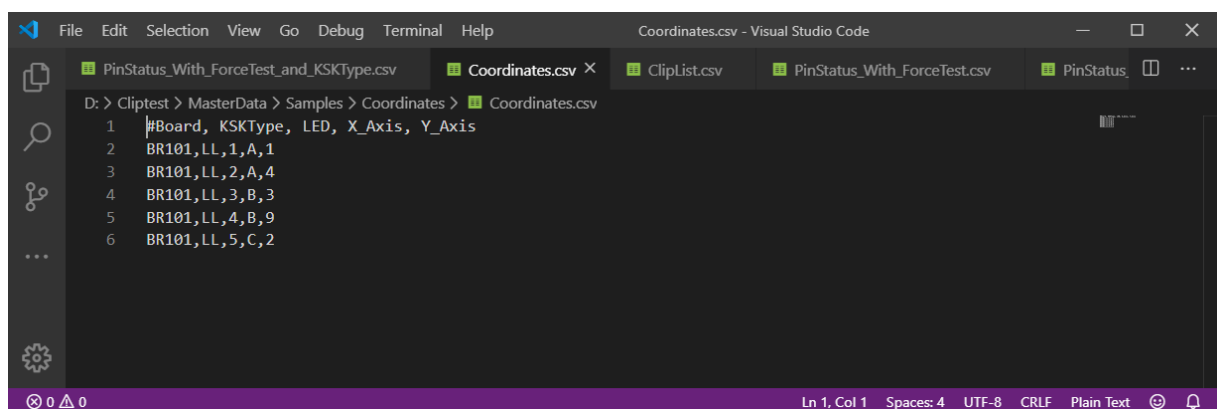| Column name	Column type	Comment
Board	String	Assembly board
KSKType	String	Reserved for future usage (you can enter arbitrary data)
LED	Integer	Clip position on assembly board wherefore the X/Y information is for
X_Axis	String	X- Information for clip position
Y_Axis	String	Y- Information for clip position

figure 12: example csv-file structure

4.3 Variants file

The variants file is used to define the different order types, which are produced with **PMS.Cliptest**. You can also define additional parameters, which can be used for the label. Every variant must have related entries in the Clip list, defining the necessary LEDs in this variant. You are able to split the information into multiple files, e.g. one file for each board.

Necessary CSV-File structure

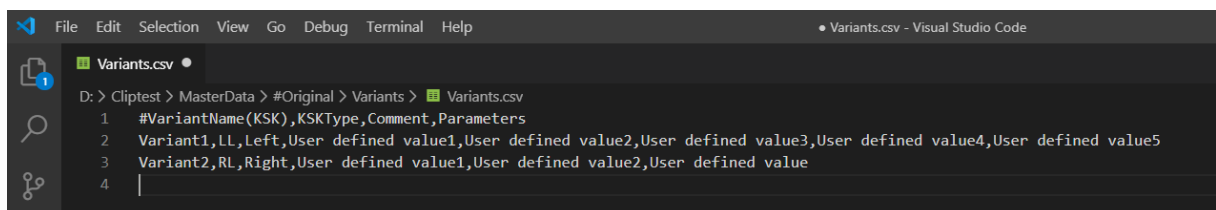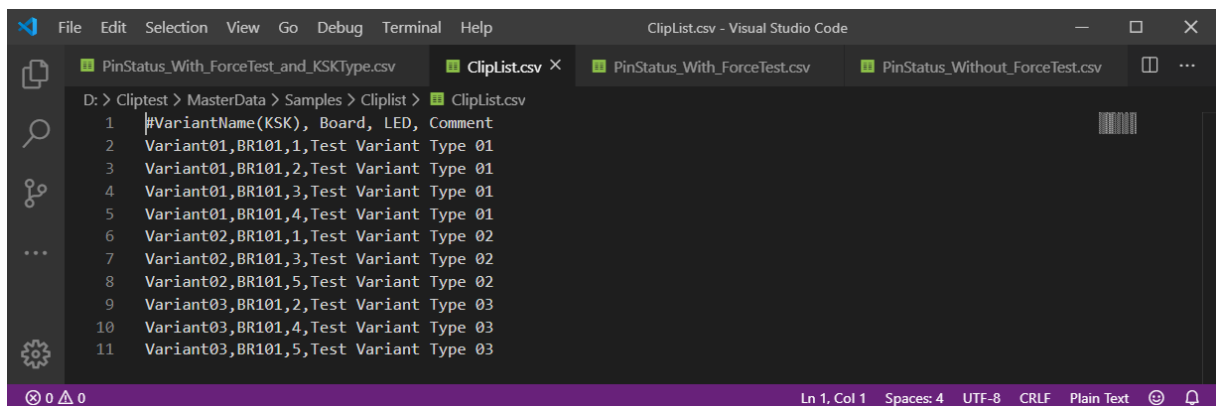| Column name	Column type	Comment
Variant	String	Name of the variant
KSKType	String	KSK Type of the variant
Comment	String	**Optionally:** Comment for the variant
Parameter_1	String	**Optionally:** User defined value for the variant
Parameter_2	String	**Optionally:** User defined value for the variant
Parameter_N	String	**Optionally:** User defined value for the variant

figure 13: example csv-file structure

4.4 ClipList file

The ClipList defines necessary clips (LEDs) for the defined variants and the given assembly boards. This file must contain all clips (LEDs) on the assembly board, which must exist (must be pressed) while the given variant is tested. You are able to split the information into multiple files, e.g. one file for each board.

Necessary CSV-File structure

Column name	Column type	Comment
KSK	String	Name of the Variant (see variants table)
Board	String	Assembly board where the order / order type should be produced on
LED	Integer	Clip position which must be pressed to correctly assembly the order / order type
Comment	String	Description for an order / order type to be displayed in PMS.Cliptest panel (should always be the same for the same order / order type)

figure 14: example csv-file structure

4.5 Modules files

The Modules defines necessary clips (LEDs) for a module. This file must contain all module with their corresponding clips for a harness. You are able to split the information into multiple files, e.g. one file for type of harness.
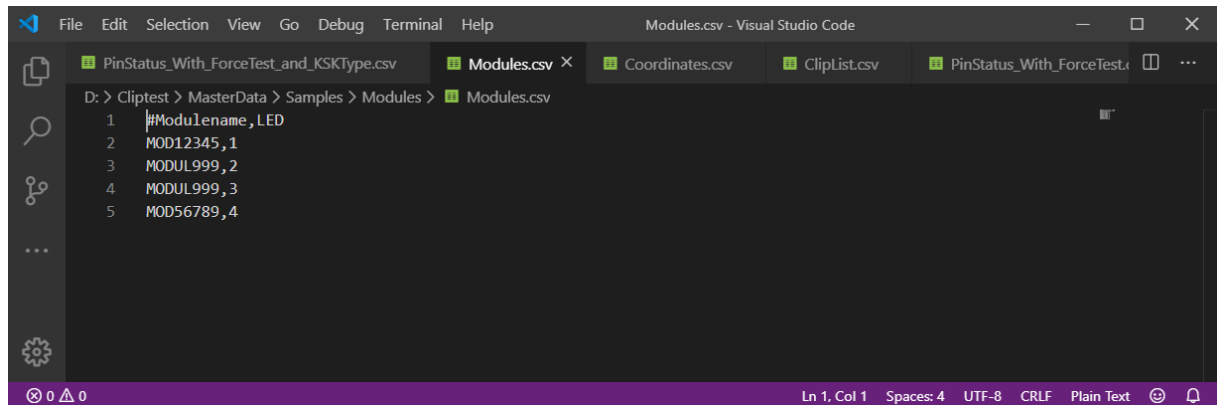
figure 15: example csv-file structure

5 User Interface Overview

5.1 PMS.Cliptest.ViewClient

The main window of **PMS.Cliptest.ViewClient** (short: ViewClient) contains different views. By default, not all tab pages are displayed (simple mode). The expert mode with all configuration options can be activated in the **Menu** with a toggle switch on the right side. In the documentation are all options described, which are maybe not displayed in the simple mode.

5.1.1 Menu

In the menu you have access to all features of the **PMS.Cliptest.ViewClient**. To open the main menu, you have to click in the left top corner on **MAR Menu**.
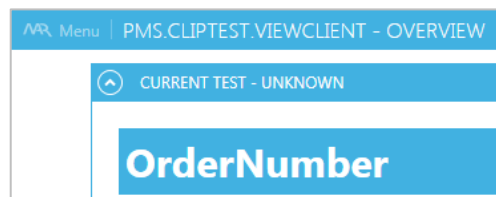

figure 16: PMS.Cliptest.ViewClient – open MAR Menu

The click opens the following menu bar. To switch between different views click on the requested button. To close the menu you can switch to another view or use the arrow on the left side of the menu.
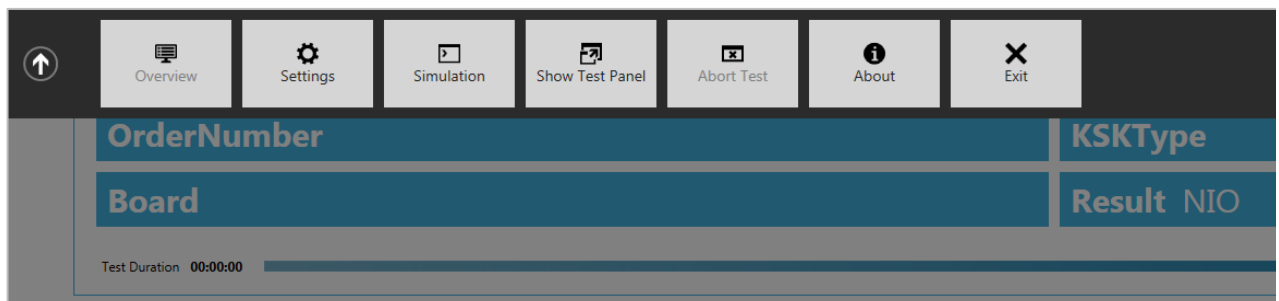

figure 17: PMS.Cliptest.ViewClient - MAR Menu

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

Button	Action
Overview	Use this button to navigate to the overview.
Settings	The settings button allows the administrator to modify the current configuration.
Login / Logout	If the option "Using Operators" is checked, with these buttons you can switch the logged in user.
Debug Mode	In the debug mode you are able to simulate scans and clip press events. By default the debug mode is hidden. You have to be an administrator to use this mode.
Show Test Panel	To open the configured Test Panels click on this button.
Abort Test	If a test is running you can cancel it with a scan of the configured abort barcode or use this button.
About	In the about view you can find information about the product and licensing.
Exit	To close the *PMS.Cliptest.ViewClient* administrator privileges are required.

5.1.2 Overview View

This is the initial view of **ViewClient**. Here you will see the current state of the application.
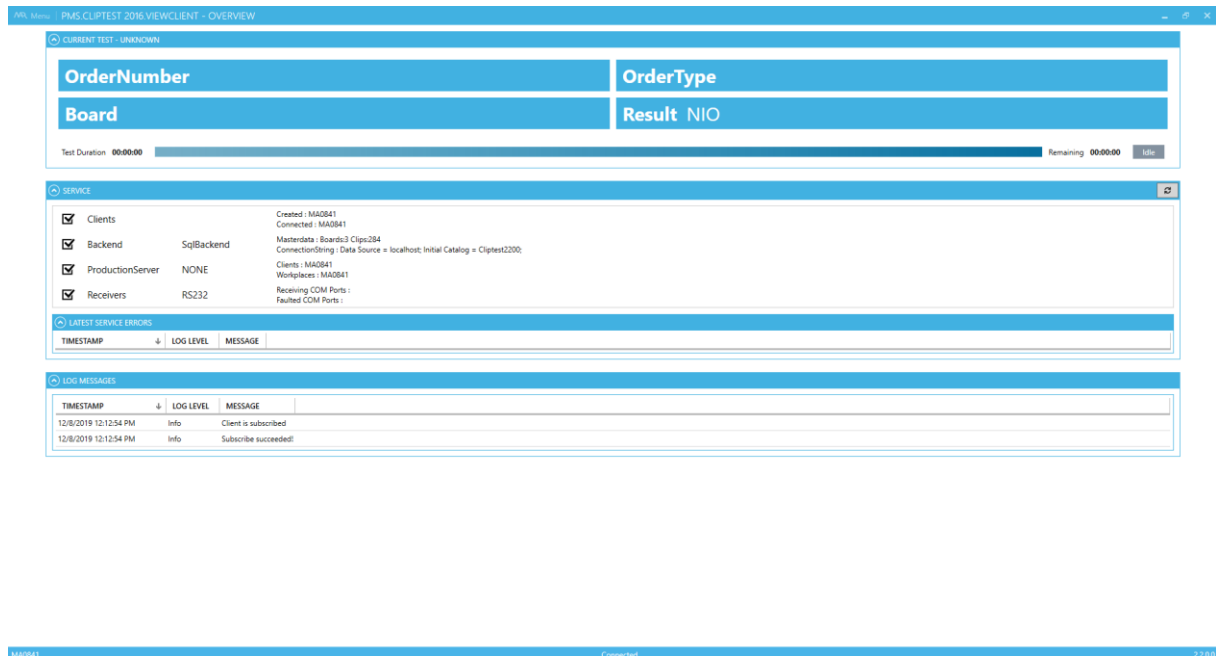

figure 18: PMS.Cliptest.ViewClient - Overview

The top of the view is informing you about the state of the current test. You can see information like order / order type, board and test result of the currently running or the last completed test.

In the middle of the overview, you can see information about the status of the PMS.Cliptest.Service. There you will find information about the connected clients, the internal PMS.Cliptest database, the configured production server, the clip receiver and the latest server errors.

The log window on the bottom is informing you about events and errors occurred in **PMS.Cliptest** workflow. It also shows information messages, depending on the configured log level.

5.1.3 Settings View

You can open the configuration by usage of the settings menu option. By default, the password protection for the settings is disabled (for further information see chapter 5.1.3.3). When this option is enabled, you will be asked for the configured password. Default user after installation of **PMS.Cliptest** is "Admin" with the password "0". If

On the bottom of the settings view you are able to save or discard current changes of the settings. Outstanding changes will be signalized with a * behind to program name on top of the settings window.

To Transfer the configuration from one client to another, you can use the **Import** and **Export** features. The export feature enables to you, to export the settings as an xml file, which can be imported on another client with the import Feature. The option **Workplace to Act for /Name By facility** will be not transferred because this setting should be unique for each instance of PMS.Cliptest
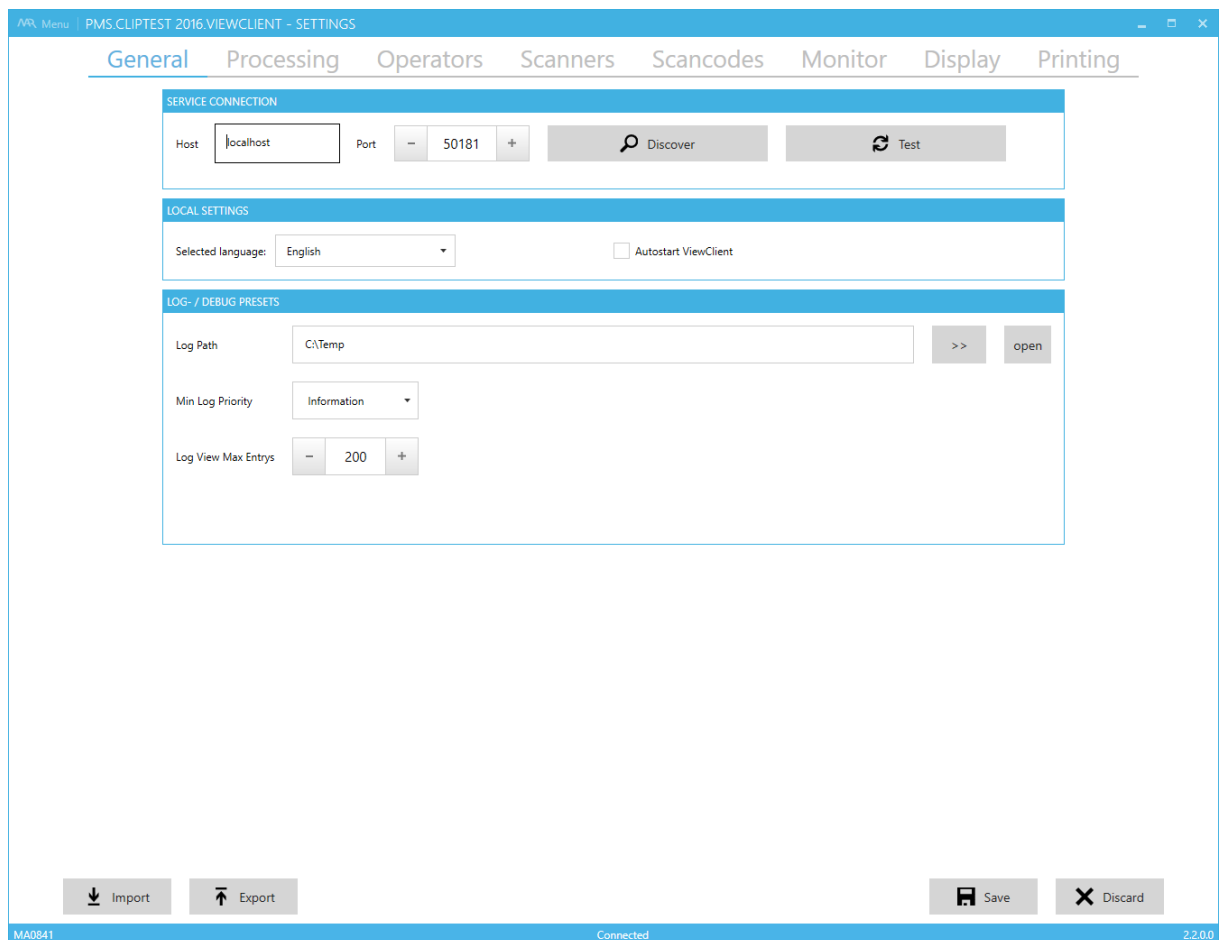

figure 19: PMS.Cliptest.ViewClient – Settings: Overview

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

5.1.3.1 General

The **General** page is used to configure the connection to the service, the language of the user interface and the log- / debug-presets. For the Service connection is only the hostname (or IP-Address) and the port required. If you are not sure what you have to setup, you can use the **Discover** button, which searches for all PMS.Cliptest.Service applications in the network. You can check your input with the **Test** button. The language of the user interface can be selected in the second group box. The changes will be applied on saving the settings. You will get additional information about the localization in the chapter 11 (Appendix III – Localization). If the PMS.Cliptest.ViewClient application should start automatically on computer start, check the option **Autostart ViewClient**.
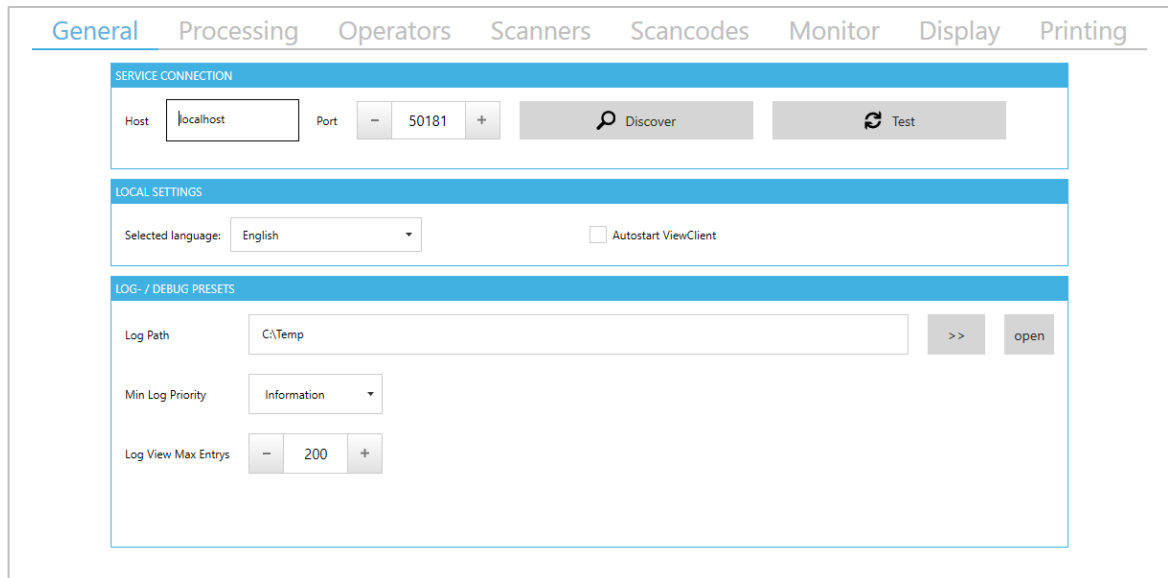

figure 20: PMS.Cliptest.ViewClient – Settings: General

The following table describes each configurable option:

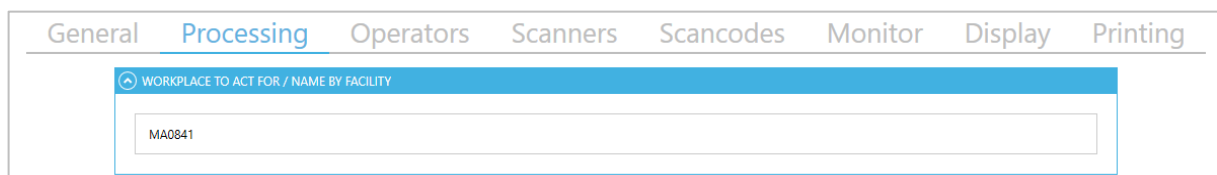| Log Debug Settings	
Log Path	Enter a Path where *PMS.Cliptest.ViewClient* should save Log-Files. The default value is "{InstallationPath}/Logs".
Min Log Priority	Select the minimum priority of Log-Events.
Log View Max Entries	Enter the count of maximum entries to be shown in log-window. If log-window contains more entries than entered count the oldest entries will be removed from log-window. The default value is 200.

5.1.3.2 Processing

The **Processing** tab page is used to configure various settings concerning the process flow of **PMS.Cliptest**. On this page you can also set the program modes **ViewClient** is running with. This page is divided into three sections.

5.1.3.2.1 Workplaces to act for / Name by Facility

The section Workplaces to act for configure the internal workplace name of **PMS.Cliptest.ViewClient**. For example, use "ASS01" as assembly line 1. The default value is the same as the ViewClientId. If you're using different instances of **PMS.Cliptest.ViewClient** which are using the same database, you'll have to configure different names for each instance to avoid interactions.

If the program is connected to a production server, this name must correspond to machine names configured in the server. The server will request the tests for the corresponding work places.

figure 21: PMS.Cliptest.ViewClient – Settings: Processing - Workplaces to act for

5.1.3.2.2 Test Parameter

In the **Test Parameter** section, you are able to configure the clip conditions and the timings.

figure 22: PMS.Cliptest.ViewClient – Settings: Processing - Test Parameter

The following table describes each configurable option:

Test mode

Configures the mode, which defines under which condition a clip is recognized as OK while running a clip test. The mode is described in the corresponding chapter. The check boxes below the combo box are displaying the three conditions which must be fulfilled to recognize a clip as OK corresponding to the chosen test mode.

Min Clip Press Time

The Min Clip Press Time defines, how long a clip must be pressed until it is recognized as pressed. This parameter affects the sub status of a clip during the pre-process (if clip is pressed before the test is started). The sub states are combined to one final status when the test is finished.

Maximum ClipTest time

Configures the maximum time a clip test is running. If this time has elapsed after starting a clip test the test is automatically completed. Remind that a clip test will also be automatically completed if all necessary clips to be pressed for current processing order / order type (Cliplist) are ok. The default value is 600 seconds. Set this option to 0 if the test should be not completed automatically. If this option and the option Maximum EmptyTest time are configured with 0, the status strip on the operator window will be hidden.

Maximum EmptyTest time

Configures the maximum time an empty test is running. If this time has elapsed after starting an empty test the test is automatically completed. Remind that an empty test will also be automatically completed if all configured clips of the assembly board are not pressed. The default value is 20 seconds. Set this option to 0 if the test should be not completed automatically. If this option and the option Maximum ClipTest time are configured with 0, the status strip on the operator window will be hidden.

Enable export of test data

The detailed status of a clip when the test was completed can be exported to a database server or to a XML file. Therefore the corresponding check box must be activated and the database server respectively the path must be setup.

5.1.3.2.2.1 Test Modes

These modes affect the test process of the clip test. The empty test is not affected. By this mode it can be differentiated, under which conditions a clip is recognized as IO while running a clip test.

Each clip has two different sub states, which are combined to one final state, when the clip test is finished: status pre-process (clip pressed before the test is started) and status while the running clip test.

	Status pre-process	Status clip test	clip pressed while test is completed
Test Mode 1 Unpressed clips	X	X	X
Test Mode 2 All clips pressed once	O	X	X
Test Mode 3 All clips must remain pressed	O	O	X

As shown in this table, one or several conditions can cause an IO status depending on the chosen test mode. In the first test mode an IO status from the pre-process OR an IO status from the clip test OR if the clip is pressed in the moment when the test is finished are resulting in a final IO. The sub states (status pre-process, status clip test) are depending on the latency period. Following conditions are causing an IO status.

Pre-process

If the clip is pressed before the test is started and remains pressed until the latency period exceeded, this sub status will be recognized as IO. When the test is started, **PMS.Cliptest** loads the sub states of the pre-process for all contained clips. Only clip events, which were raised after the generation of the clip list, will be taken into consideration.

ClipTest

If the clip is pressed while the test is running, this sub status will be recognized as IO. The latency period is not taken into consideration.

Following the three modes will be described and how the mentioned sub states are combined to calculate the final status of a clip:

1. Un-pressed clips

   PMS.Cliptest.Service collects every clip event in the local database, even for those clips which are not contained by an actual running test and even if no test is actually running. This log provides the information if a clip was pressed before the clip test was started (pre-process). In this mode the log is taken into consideration to calculate the final status of a clip during the running test. If a clip was pressed before starting the test and it fulfills also the configured latency period (duration, how long the clip must remain pressed during the pre-process), the final clip status during the test is IO, even if the clip is not pressed during the test. The final clip status will be IO under the following conditions:

   - ✓ Clip was pressed in the preprocess (before clip test) and remained pressed until the latency period was IO
   - ✓ <u>OR</u> this clip was pressed during the running clip test without taking into consideration the latency period. *If the clip was pressed in the pre-process and remained pressed until the test was started, it will not be recognized as IO until it was released and pressed again.*
   - ✓ <u>OR</u> the clip is pressed, in the moment when the clip test completes (pressed while or until the end of the test)

2. All clips must be pressed once (during the running clip test)

   In this mode the clip must be pressed once during the running clip test and must remain pressed until the latency period was fulfilled to achieve an IO status for the clip. The final clip status will be IO under the following conditions:

   - ✓ The clip was pressed during the running clip test without taking into consideration the latency period. *If the clip was pressed in the pre-process and remained pressed until the test was started, it will not be recognized as IO until it was released and pressed again.*
   - ✓ <u>Or</u> the clip is pressed, in the moment when the clip test completes (pressed while or until the end of the test)

3. All clips must remain pressed

In this mode the clip must be pressed until the end of the test to achieve an OK status for the clip. The clip will be missing, if the clip was pressed during the test but gets un-pressed before the test was finished.

Furthermore, the first two modes are taking into consideration the ForceTest flag from the master data: Single clips can be forced to be tested even if they were already pressed before the test or during the test in these two modes. Therefore, an additionally flag "ForceTest" must be added to the imported master data PinStatus.csv. If this flag is activated (True), the clip is required to be pressed until the test was finished.

For a better understanding the following table shows the final status for a clip differentiated for all three modes:

In the following table the latency period is set to 10 seconds

Process				Resulting Sub-States			Config	Final Status in Testmodes		
Configured	Requested	PreProcess Duration	ClipTest Duration	PreProcess Status	ClipTest Status	Clip pressed until end of te	Force Test	1	2	3
true	true	7	0	Missing	Missing	false	true	Missing	Missing	Missing
true	true	12	0	OK	Missing	false	true	Missing	Missing	Missing
true	true	7	15	Missing	OK	false	true	Missing	Missing	Missing
true	true	12	22	OK	OK	false	true	Missing	Missing	Missing
true	true	7	0	Missing	Missing	true	true	OK	OK	OK
true	true	12	0	OK	Missing	true	true	OK	OK	OK
true	true	7	23	Missing	OK	true	true	OK	OK	OK
true	true	12	23	OK	OK	true	true	OK	OK	OK
true	true	7	0	Missing	Missing	false	false	Missing	Missing	Missing
true	true	12	0	OK	Missing	false	false	OK	Missing	Missing
true	true	7	1	Missing	OK	false	false	OK	OK	Missing
true	true	12	2	OK	OK	false	false	OK	OK	Missing
true	true	7	0	Missing	Missing	true	false	OK	OK	OK
true	true	12	0	OK	Missing	true	false	OK	OK	OK
true	true	7	12	Missing	OK	true	false	OK	OK	OK
true	true	12	12	OK	OK	true	false	OK	OK	OK
true	true	12	0	OK	OK	true	false	OK	OK	OK
false	true	-	-	unconfigured	unconfigured	unconfigured	-	unconfigured	unconfigured	unconfigured
true	false	12	22	unnecessary	unnecessary	true	-	unnecessary	unnecessary	unnecessary
true	false	7	1	OK	OK	true	false	OK	OK	unnecessary
true	false	12	12	unnecessary	unnecessary	false	false	unnecessary	unnecessary	OK
true	false	7	15	OK	unnecessary	true	false	OK	unnecessary	unnecessary

figure 23: test modes examples

5.1.3.2.3 Test Processing Settings

In the **Test Processing Settings** section, you can enable the communication with a production server system and configure the communication parameters.
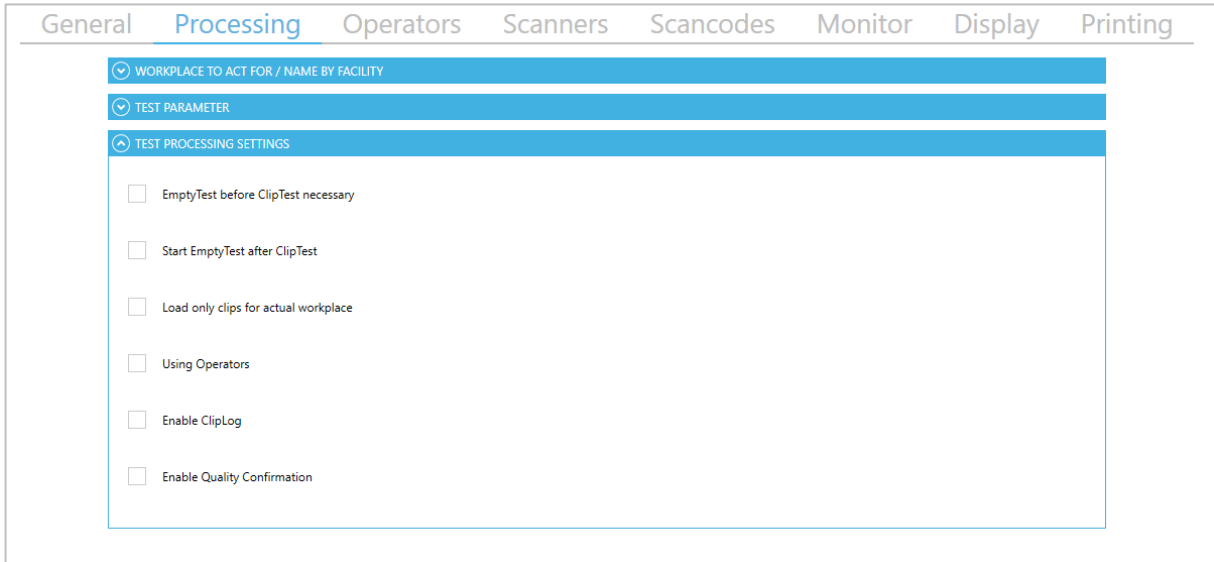

figure 24: PMS.Cliptest.ViewClient – Settings: Processing - Test Processing Settings

The following table describes each configurable option:

EmptyTest before ClipTest necessary

Configures that first an empty test with result IO has to be performed for the given board, before a clip test can be started. After every ClipTest, a new empty test has to be done.

Start EmptyTest after Cliptest

If this option is enabled, the EmptyTest will be started directly, when the ClipTest is finished, respectively starts automatically the EmptyTest if the corresponding process status was set by the production server.

Load only clips for actual workplace

If this option is enabled, *ViewClient* will only load the clip list for the actual running work place. The work place must be provided by the server. With help of this option, it is possible to use several clip tests on one line, so that each work place can test a different range of clips.

Using Operators

If this option is checked, a scan can only be performed if an operator is logged in to the program. A change of the variant is only possible with a logged in operator.

Enable ClipLog

This function is only available if PMS.Cliptest.ClipReporting is licensed. All clip events related to the running clip test are being collected and traced. After the clip test is finished or aborted, these clip events are stored permanently. The information contains a unique ID for the performed test, including the number of presses per clip involved in this test.

Enable Quality Confirmation

This option is used to require a qualified user to confirm unusual critical events. A dialog at the end of each test will ask for confirmation or the test will be finished with NIO.

5.1.3.3 Operators

The use of **Operators** can be activated in the settings tab **Processing**. With the use of operators, the operator must perform a log in to start a process by scanning or to change the order variant.

On this settings section you can also enable or disable the authorization features. By default, these options are disabled. With the option **Require login to access settings** you are able to protect the settings against unauthorized changes. If it is required to do not allow closing the application without privileges, enable the option **Require login to close application**.
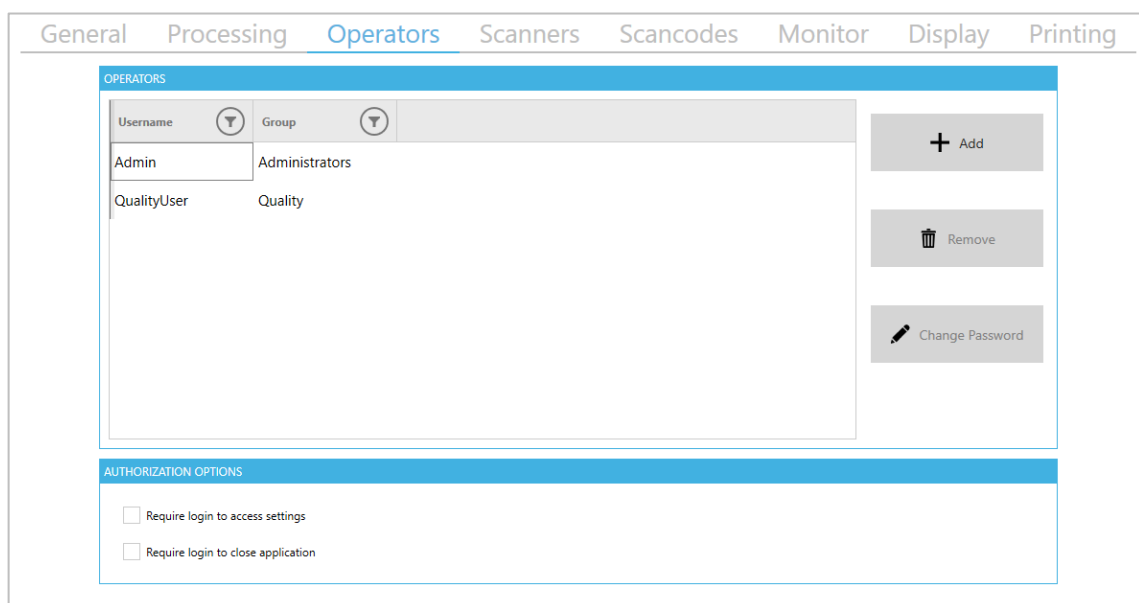
figure 25: PMS.Cliptest.ViewClient – Settings: Operators

To login an operator, you have to press the log in button in the menu. If an operator is logged on, the operator name is displayed in the status bar on the bottom of the main windows. The Logoff button is located in the menu and will log off the logged in operator. After a logout you are able to log in a new operator.

If no operator was logged in and a scan was performed, the program will automatically open the log in box. If cancel is pressed and no operator is logged in, the scan is canceled.

5.1.3.3.1 Manage Operators

To Add a new operator, enter a name and a password and click the add button. To delete an existing operator, select it from the list or enter the name and press the delete button. To change the password for an operator select it or enter the name, enter a new password and click the change password button.

figure 26: PMS.Cliptest.ViewClient – Settings: Operators – Add new Operator

5.1.3.4 Scanners

The **Scanner** tab page is used to configure the scanner settings. This scanner is used to scan the assembly board barcodes to start a ClipTest or an EmptyTest for a special board or any other data.

figure 27: PMS.Cliptest.ViewClient – Settings: Scanners

The *PMS.Cliptest ViewClient* application supports two different types of scanner. The scanner can be connected via the COM-interface (physical or virtual COM-Port) or via USB as a keyboard. The COM-Interface have to be configured with specific options (see chapter **Fehler! Verweisquelle konnte nicht gefunden werden.** for further information). The USB scanner uses an input box on the operator window to handling the data. This option can also be used for testing scenarios.

5.1.3.4.1 COM (RS232) Scanner

This section describes all required configuration options for a COM-port scanner which is connected via RS232.

figure 28: PMS.Cliptest.ViewClient – Settings: Scanners RS232

Name

A unique name that describes the scanner.

Prefix Length

Number of characters which are used for the address stamping. The default value is 0 (no address stamping enabled).

COM port

Enter the COM-Port you have connected your scanner to e.g. COM1.

Baudrate

Select the baudrate the scanner works with. The default values is 9600.

Parity

Enter the parity the scanner works with. The default value is NONE.

Stopbits

Enter the count of stopbits the scanner works with. The default value is one.

Databits

Enter the count of databits the scanner works with. The default value is eight data bits.

Scan Terminator

Scanners are sending the scanned value with a terminator in data stream. This terminator marks the end of the scanned data. To make PMS.Cliptest.ViewClient correctly evaluate scans of the connected scanner this terminator must be set. Normal scanners do use a CRLF (Carriage Return Line Feed) as terminator so this is the default value.

Since you cannot enter special chars like CR (carriage return) with your keyboard you can use the ASCII-list. Enter the decimal values of the chars you want to use as terminator, e.g. "13" for CR or "10" for LF. Use an ASCII-table if you're not sure. The default value is CRLF corresponding to ASCII{13} + ASCII{10}.

5.1.3.4.2 USB Scanner

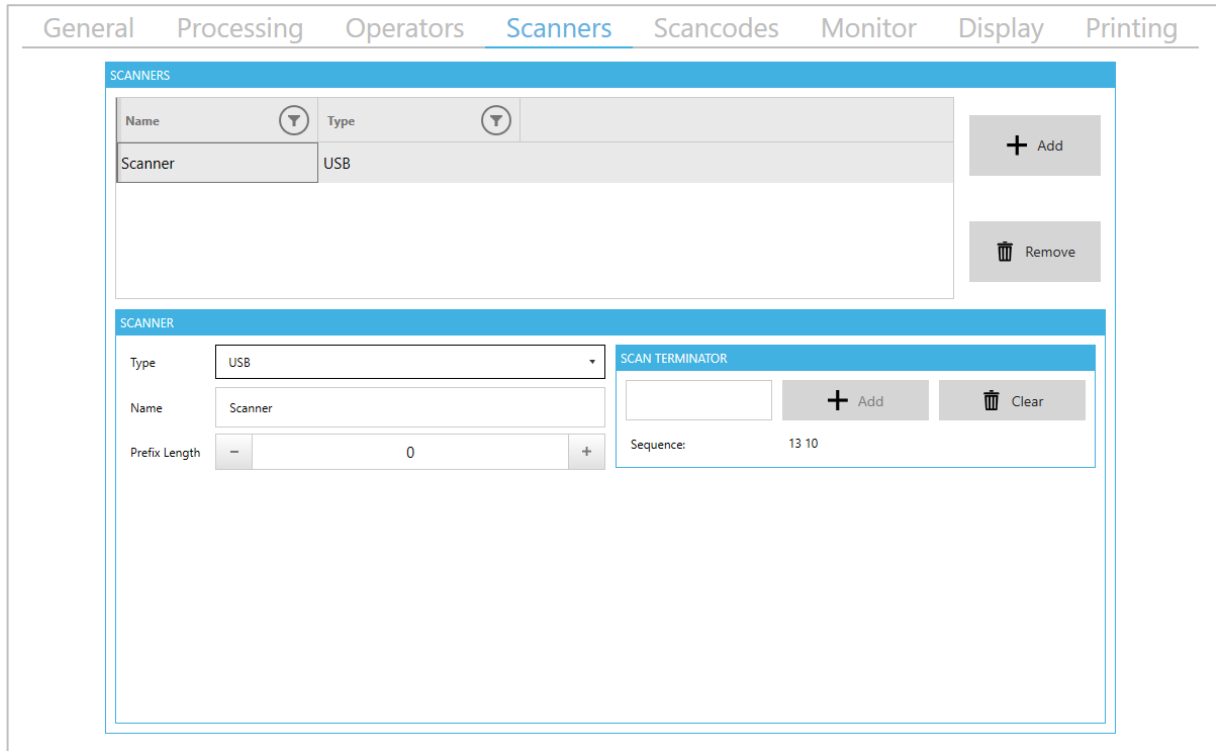This section describes all required configuration options for an usb scanner.

figure 29: PMS.Cliptest.ViewClient – Settings: Scanners USB

Name

A unique name that describes the scanner.

Prefix Length

Number of characters, which are used for the address stamping. The default value is 0 (no address stamping enabled).

Scan Terminator

Scanners are sending the scanned value with a terminator in data stream. This terminator marks the end of the scanned data. To make PMS.Cliptest.ViewClient correctly evaluate scans of the connected scanner this terminator must be set. Normal scanners do use a CRLF (Carriage Return Line Feed) as terminator so this is the default value.

Since you cannot enter special chars like CR (carriage return) with your keyboard you can use the ASCII-list. Enter the decimal values of the chars you want to use as terminator, e.g.

"13" for CR or "10" for LF. Use an ASCII-table if you're not sure. The default value is CRLF corresponding to ASCII{13} + ASCII{10}.

5.1.3.5 Scancodes

In the section **scan code configuration**, you are able to configure the format of the scanned messages and the corresponding attributes. The scan codes are grouped in sequences. A scan sequence contains a type, the name of the sequence and the scan codes. Based on the type of the sequence Cliptest decides, which action should be performed when all scans were fulfilled. It is possible to have multiple sequences with the same type. The following table explain the available types and give some example data.

Cliptest

This type defines a sequence of scans, which are required to start a cliptest. By default, a board scan and an order scan are required in this sequence.

Emptytest

This type defines a sequence of scans, which are required to start an emptytest. By default only a board scan with a prefix is require

Optional

This type is for scans, which are not required to start a test. For example, the ABORT scan or you can provide additional scan data.

figure 30: PMS.Cliptest.ViewClient – Settings: scan code sequences

The scan code sequences can be managed with the buttons on the right side of the sequences list. After selecting a scan code sequence, the configuration view for the sequence will appear below the list of sequences. On top of the sequence the type of the sequence can be change. The name property is only used to allow the user to differentiate between the sequences. The scan codes of a sequence can be managed right next to the scan code list. There you have to option to add or remove a scan code and you sort the scan codes. The sort feature is required for customs scans, which should be accessible in the printing feature. To configure a scan code, select it from the list and below the list the configuration view will appear.

The **Type** of a scan code defines how cliptest should handle the scanned data. There are several types, which will be described in the following section.

Board

This type will be used to identify a board scan independent from the requested test type. If your assembly boards may be called BR001 to BR999, enter "^(BR)" for the **Scan Pattern**. If the connected scanner scans for example a "BR002"-code the board property for a test will be "BR002". The default value for the sequence type Cliptest is "^(BR)". In case of sequence type Emptytest it is "^(LE)". Additionally you have to configure a **Prefix** for the Emptytest to differentiate between the scan sequences.

Order

With scanning an order the current order is changed to the scanned one. The scanned order must be configured in the master data tables Variants and Cliplist. If the order is not configured, *PMS.Cliptest.ViewClient* shows an error in in the main program window. *If the option Prefix is configured, it is cut from the scan code before the variant is processed. In this case, the master data must be provided without the prefix!*

Examples:

- If Prefix "OR" is configured and an order scan starts with "OR", then the "OR" is cut from the scan code. Scanning "OR123456" would be interpreted as an order scan referencing to order "123456".
- If Prefix is empty and an order scan starts with "OR", then the "OR" is <u>not</u> cut from the scan code. Scanning "OR123456" would be interpreted as an order scan referencing to order "OR123456".

Abort

Enter a special code to abort a running test. If this field is set to "ABORT" and an "ABORT"-code is scanned from the connected scanner the currently running Cliptest will be completed. The default value is "ABORT".

Modules

The modules scan contains a list of modules, which should be tested during a ClipTest. The Format of a single module can be described with the scan pattern. The module list needs additional master data, which can be managed with the MasterDataManager. For further information about the configuration see 5.2.7 Module Management

Custom

To store additionally information during a test, you can use the Custom scan code. The additional information can be used for printing. The order of the scan codes described the order of the bookmarks for printing.

The *Scan Pattern* describes with a regular expression the format of the barcode. If it is required that the scanned data should be trimmed at the beginning, you can configure the *Prefix*. The configured value will be removed at the start of scanned data. The *Separator* is only used for the type *Modules*. The configured *Separator* will be used to split the single data items from the scanned data. The option *Keep* allows the user to scan a scan code once and use the data until the next scan of this type. Do not use this option for all scan codes in a sequence this can cause unintended behaviors. You will find a short description of regular expressions in the help dialog. You can open this dialog with the question mark symbol in the header of the scan code configuration.

5.1.3.6 Monitor

The **Monitor** tab page is used to configure the monitors where each panel of **PMS.Cliptest.ViewClient** should be shown on. The user can also select the header of the panels.

5.1.3.6.1 Header

There are two templates for the header available. The Default header contains information about the **OrderNumber**, the **OrderType**, the **Board** and the **result** of the current test.
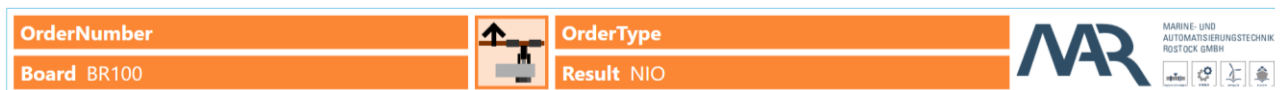
figure 31: PMS.Cliptest.ViewClient – Settings: Monitor – Default header

The Extended header displays information about the current test (**OrderNumber**, **Board**, **#Wrong**), client specific information (**#All**, **#User**, **User**) and general information (**Project Name**, **DateTime**). The field **#Wrong** displays during a ClipTest the number of clips, which are in the state **Needless pressed** (*NLNIO*), **Needless unknown** (*NLUKN*) or **Unconfigured** (*UNCNF*). During an EmptyTest the field displays the number of clips, which are in the state **Missing** (*NIO*) or **Unconfigured** (*UNCNF*). The field **#All** displays the number of all completed tests of the selected Client Id. Depending on the selected Production Server, the name of the current project can be displayed. If the feature *using operators* (see 0 for further information) is activated, the logged on operator name will be displayed in the field **User**. The field **#User** counts the completed tests of the current operator. On changing the user, this counter will be reset. If no operator is logged on, the counter displays the number of completed test since the last start of the *PMS.Cliptest.Service*.

figure 32: PMS.Cliptest.ViewClient – Settings: Monitor – Extended header

5.1.3.6.2 Windows

The panels will automatically be shown on the selected monitor. Additionally, you can choose between three different layouts types for each screen. The first type is the column layout. It displays a list of all clips and can be used without any preparation. Each column enables separate filter options to you. The matrix and the image layout requires some additional configuration to use. The configuration of these layout types have to be done in the **PMS.Cliptest.MasterDataManager**. For more information, see chapter 5.2.10.

To setup a layout for a monitor, select at first the monitor and after that you can choose one of the possible layouts. After saving the settings, the new layout will be applied. If you setup up a layout for a wrong monitor, use the clear button to remove a layout from a monitor.
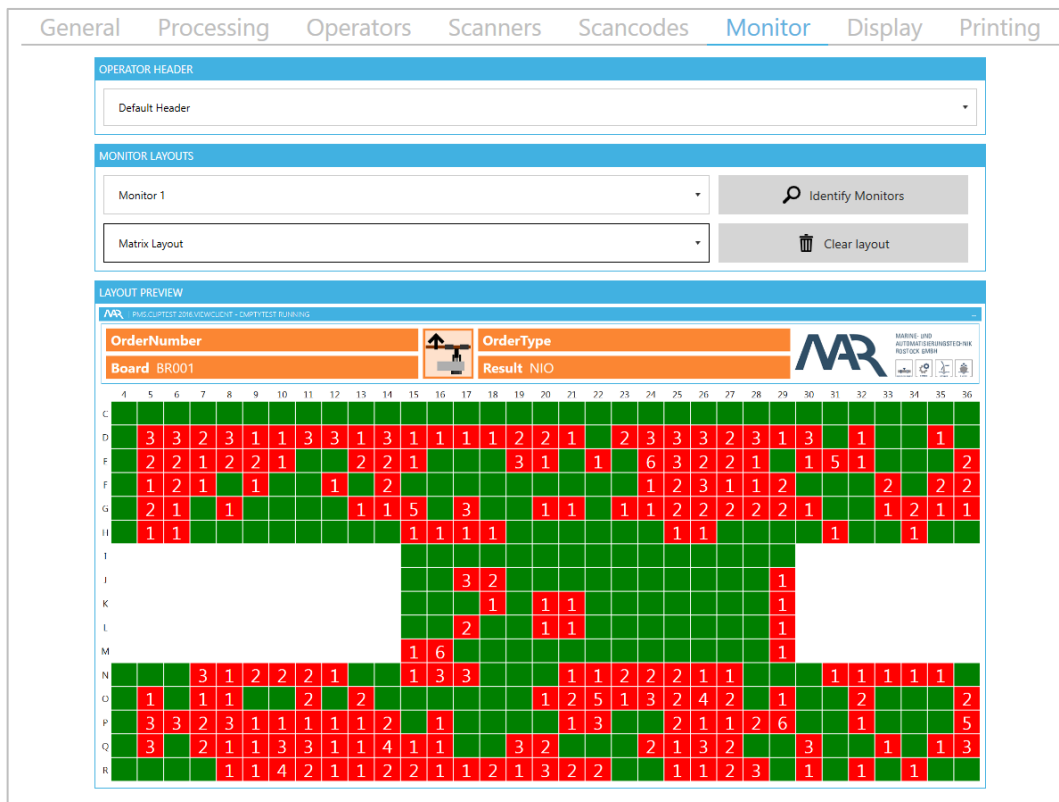
figure 33: PMS.Cliptest.ViewClient – Settings: Monitor

If you are not sure to configure the requested monitor, use the Identify Monitors button. By default, the first monitor shows a test panel with only one column.

5.1.3.7 Display

The **Display** tab page is used to configure the settings of the test panel of **PMS.Cliptest.ViewClient**.

figure 34: PMS.Cliptest.ViewClient – Settings: Display

5.1.3.7.1 Style Configuration

In the Style Configuration section, you change the display style of the clips on the operator windows. To configure a style, select first one of the clip state on top of the configuration. You are able to configure the clip states **OK** (*IO*), **Missing** (*NIO*), **Needless** (*NLNIO*), **Unconfigured** (*UNCNF*) and **Group Ok** (*GRPIO*). For each of them, you can configure the font size, the diameter for the image layout and the foreground and background colors. If required, you can choose different style for Cliptests and Emptytests. In the preview you can see the configured values for each test type. If you disable the option **Use different styles**

for Cliptest and Emptytest the configured values for cliptests will be also used for emptytests. The emptytest style will be not deleted when you disable the this option.

5.1.3.7.2 Test Panel Configuration

In this section you can configure all options regarding the test panel. The options *Test Information Font Size* and *Clip Animation Size* configures the sizes for the test header (see 5.1.3.6.1).

If you have boards with many clips, which changes their state, very quickly, the grouping and filtering options can slow down the display performance. To avoid these display issues, you can configure the *Clip change interval*. If this option is set to 0, all clip updates will be performed immediately including filtering and grouping. With a higher value the grouping and filtering actions will be delayed and a group of updates are displayed at one time. The color changes of the clips states will be not affected by this option.

If you do not want to see the clip updates in the live mode check the option *Disable clip updates in live mode*.

The option *Enable filter and zoom modification* allows everyone to change the filtering, grouping and zooming settings in the test panel.

5.1.3.7.3 Desktop alerts

The PMS.Cliptest ViewClient application can notify the user if there are any errors or issues during working with the application e.g. invalid scan data. In the right bottom corner a desktop alert will appear on any issue. You can configure how long this box should be displayed. If you set the option to 0, the desktop alert is disabled.
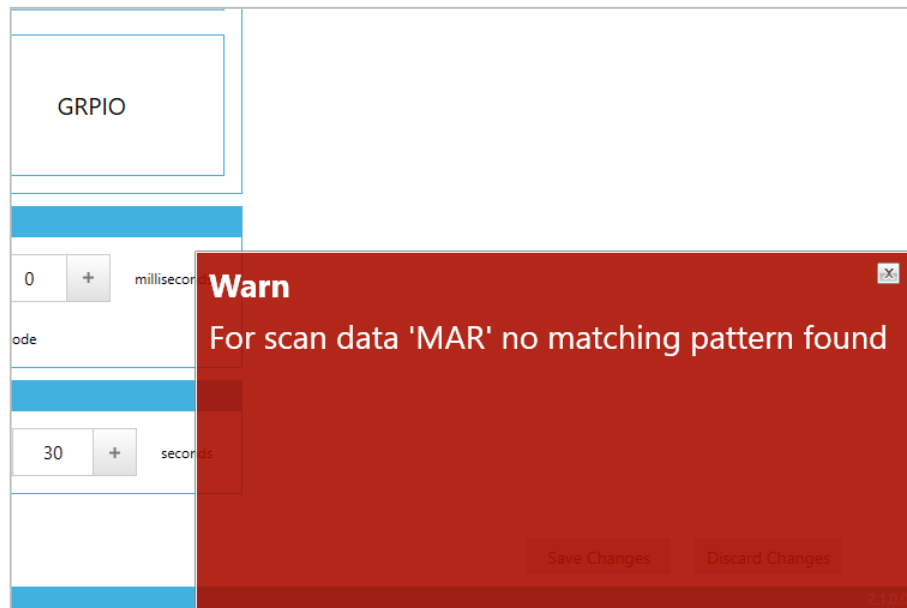
figure 35: PMS.Cliptest.ViewClient – Example desktop alert

5.1.3.8 Printing

The **Printing** tab page is used to configure the print options of **PMS.Cliptest.ViewClient**. You may want to print labels after completing a test or when starting a test. Refer to chapter 9 for further descriptions.

figure 36: PMS.Cliptest.ViewClient – Settings: Printing

Printer settings

Is Enabled	Enables the Print Option for the defined printing event. The default value is disabled.
Name	You can define a name for better differentiation between your printing events.
PrintProcessor Name	Choose the PrintProcessor you wish to use for the print job. The list of available PrintProcessors have to be configured in the service using the ServiceConfigurator.
Printer Name	Select a printer you wish to use for label printing. This is a list of known printers from the selected PrintProcessor.
Label Name	Select the template for printing the data. *PMS.Cliptest* is using labels designed with TFormer. This list contains all available labels for the selected PrintProcessor.
Test Type	Select the test type for the printing event. Possible values are ClipTest, EmptyTest and All.
Test Status	Select the test status for the printing event. Possible values are Idle, Created, Finished and All.
Test Result	Select the test result for the printing event. Possible values are IO, NIO and ALL.
Copies	This option is used for the count of copies. The default value is 1.
Combined Test Result	The flag "CombinedTestResult" is used only for empty test labels (EmptyTest_Finished). If the flag is set, then the result of the empty test is combined with the last clip test result. That means, if you configure a result OK for the empty test label and check the "CombinedTestResult" flag, then the label is only printed, if the previous clip test <u>and</u> the empty test were OK for the given board. If you choose the result NIO, the label will be printed if the previous clip test <u>or</u> the empty test were not OK for board.

Printer settings

Confirm Printout	Enable this option if you need to perform a scan-validation of your printed barcode. After this print event a dialog will appear (see chapter 5.1.5.2) asking for a value to scan. You can use valid bookmarks in the value configuration surrounded by square brackets. For example the value "[Order]" will ask for the order number of the test. You can also use combinations of multiple bookmarks and static text without brackets for example: "[Board] finished with result [Result]".

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

5.1.4 Test Panel

A **PMS.Cliptest.ViewClient** can display multiple test panels. The amount of panels depends on the count of monitors are connected to the computer. The layout and the position of the test panels can be configured in the settings tab **Monitor**.
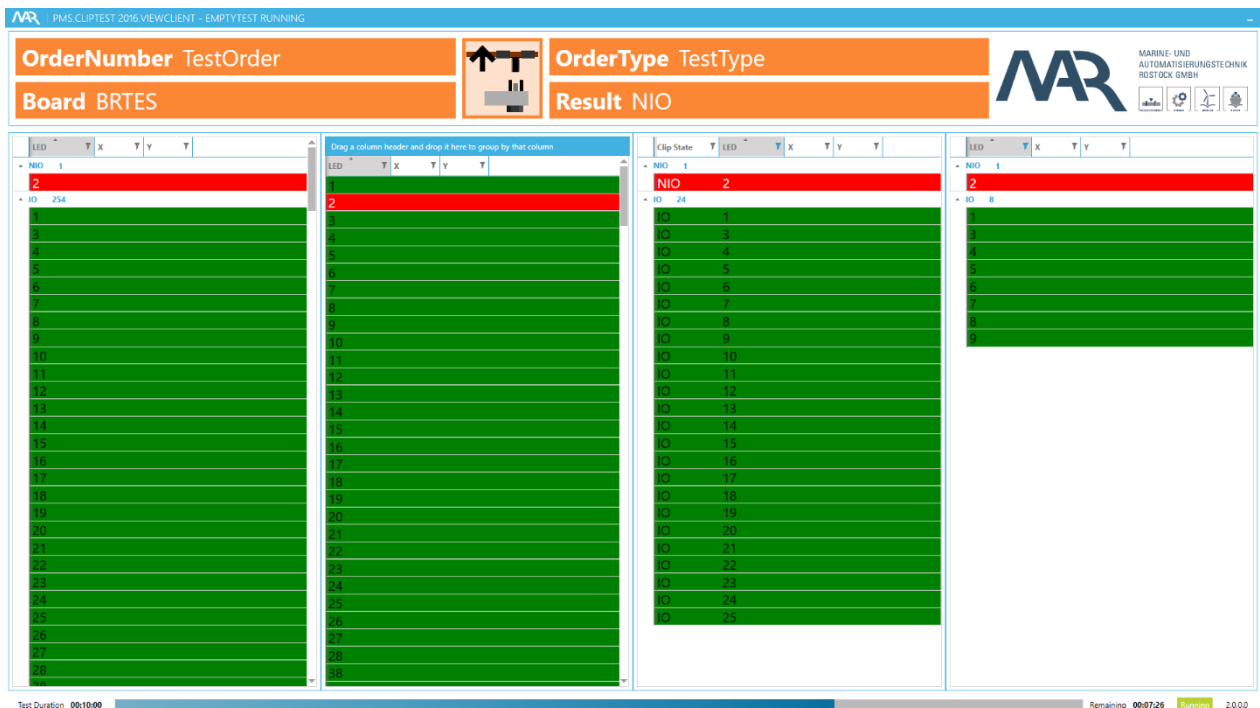
figure 37: PMS.Cliptest.ViewClient – Test Panel

The test panel contains three main parts. On the top of each test panel are information about the current test displayed. In the middle you can see selected layout and on the bottom is the remaining time for the current test.

5.1.4.1 List layout

The list layout displays all clips in a list with a different number of columns. Each column of this layout can be configured separately. To change the displayed clips, the grid view headers enable sorting, filtering and grouping to you. To save or discard the changes or reset to defaults use the context menu (right mouse click) of the grid view header. The settings for each grid view will be handled separately (one save action is required for each changed grid view or you can use the save option for the window). Attention: the save option will save all unsaved changes from the settings. E.g. if you change the color of the clips, don't save this operation and press the save changes button in the context menu, all other changed settings will be applied too. If you choose the Load Defaults option, the configuration will be reset, but not automatically saved. In the context menu you can also show the grouping panel and edit the displayed columns. The context menu will be opened with a right mouse button click on the grid view column headers. To disable the context menu, uncheck the option **Enable filter and zoom modification** in the settings tab **Display**.
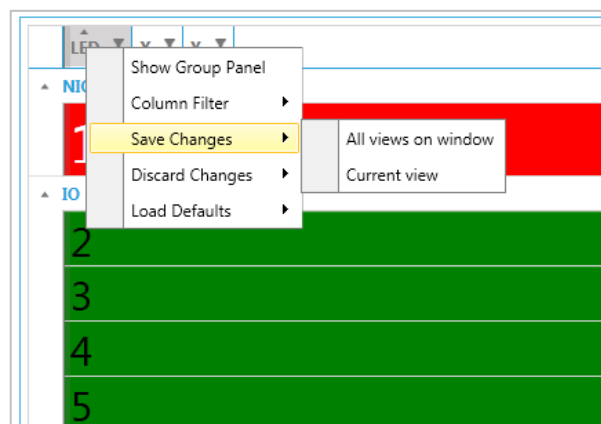
figure 38: PMS.Cliptest.ViewClient – Test Panel: List Layout - Filtering

5.1.4.2 Image layout

The image layout displays all clips on a configurable board layout. The style of the clips is configurable in the settings.
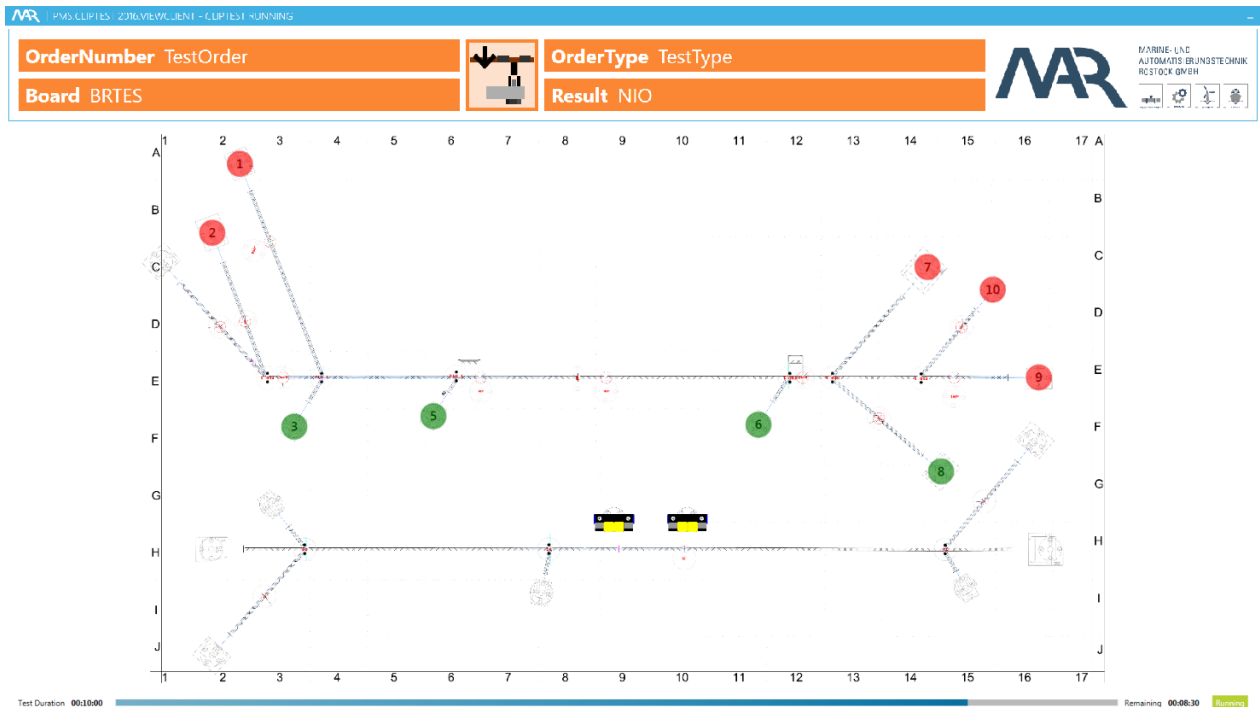
figure 39: PMS.Cliptest.ViewClient – Test Panel: Image Layout

If you want to see only a part of the layout are able to zoom and scroll the layout. To open the zoom control, click double on the background. The zoom control is placed on the right bottom corner. Set up the requested section and **Save** the configuration for the next test.
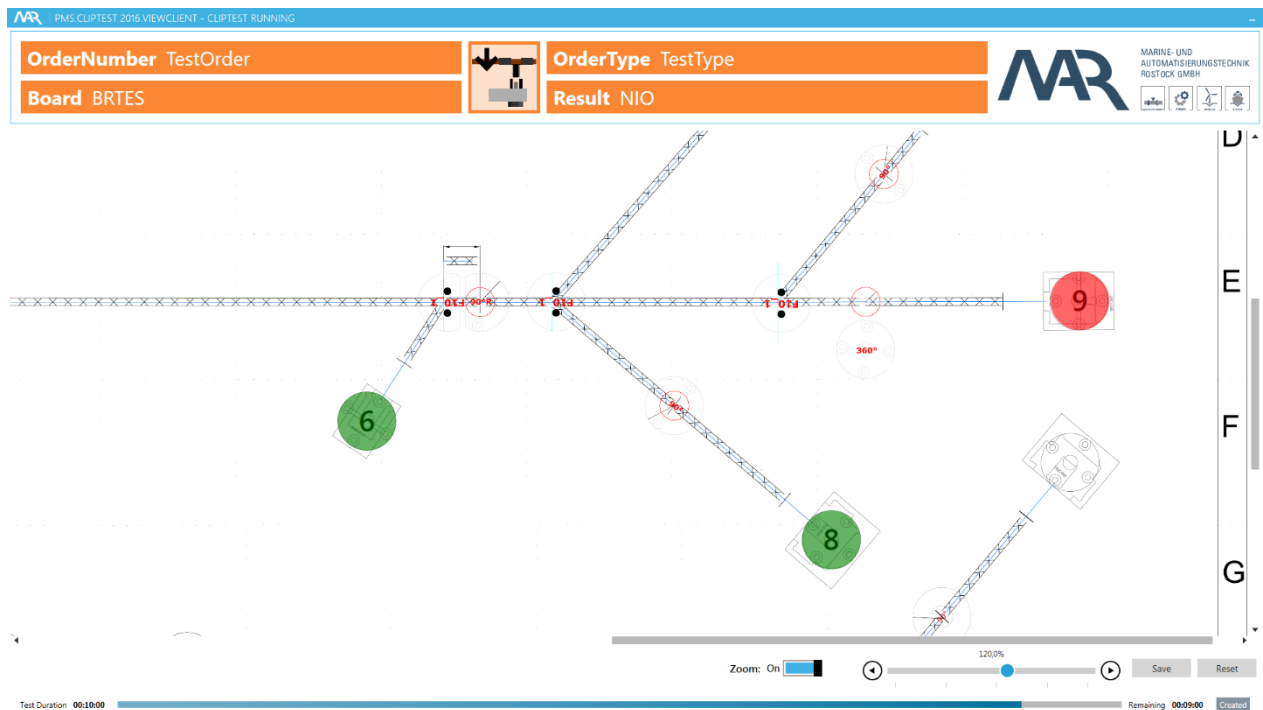
figure 40: PMS.Cliptest.ViewClient – Test Panel: Image Layout – Zoom Control

Important

To use this layout, you have to prepare each board with die *PMS.Cliptest.MasterDataManager*.

5.1.4.3 Matrix layout

The matrix layout displays all clips in a grid. Each cell shows the count of the missing clips. The style of the cells is configurable in the settings.

figure 41: PMS.Cliptest.ViewClient – Test Panel: Matrix Layout

Important

To use this layout, you have to prepare each board with die **PMS.Cliptest.MasterDataManager**.

5.1.5   Dialogs

This chapter provides an overview about possible dialogs which might appear in different szenarios.

5.1.5.1   Quality Confirmation

This dialog appears in the case if there happened quality related issues during a test. Currently the state change of any clip from state NeedlessPressed to NeedlessNotPressed during a test will be treated as quality issue. PMS.Cliptest will track such events and if the option "Enable Quality Confirmation" is enabled this dialog will appear on all application windows. The dialog will ask for a user of group "Quality" to login and confirm all issues forcing this user to also add a comment to each issue. If the user doesn´t confirm all of the issues, the test will be finished with result NIO.

figure 42: PMS.Cliptest.ViewClient – Dialog: Quality Confirmation

5.1.5.2 Label confirmation

This dialog appears if you enabled the option "Confirm Printout" in a printing event. It is supposed to verify the printed barcode. Therefore you should set the value in the configuration to the same value which is your barcode content. If the scan doesn´t match the configured value the dialog won´t close and the barcode can be scanned again. The print button will print the label again in case the printer has any problem.

Please scan printed barcode to confirm
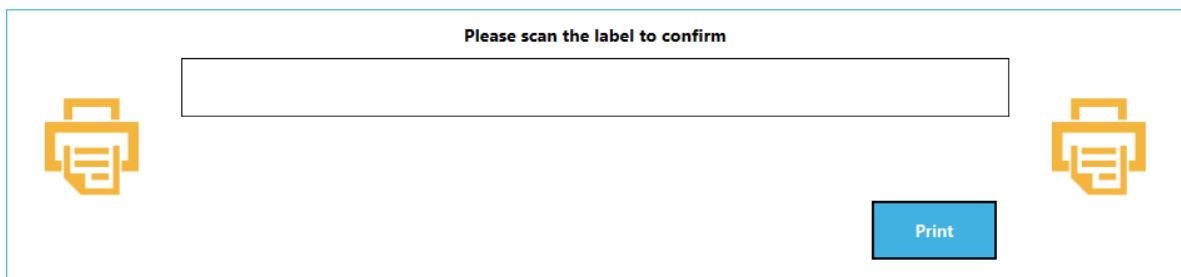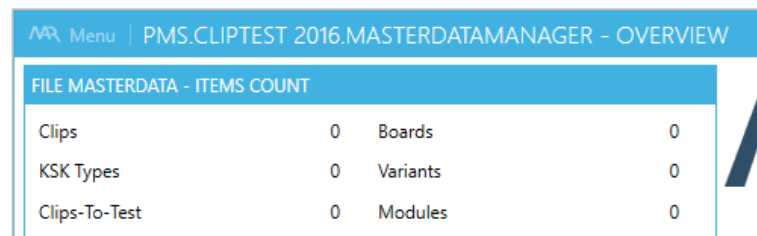
figure 43: PMS.Cliptest.ViewClient – Dialog: Confirm label printout

5.2 PMS.Cliptest.MasterDataManager

The main window of **PMS.Cliptest.MasterDataManager** (short: MasterDataManager) contains different views. The following chapters describe all views.

5.2.1 Menu

In the menu you have access to all features of the **MasterDataManager**. To open the main menu, you have to click in the left top corner on **MAR Menu**.

figure 44: PMS.Cliptest.MasterDataManager – open MAR Menu

The click opens the following menu bar. To switch between different views click on the requested button. To close the menu you can switch to another view or use the arrow on the left side of the menu.
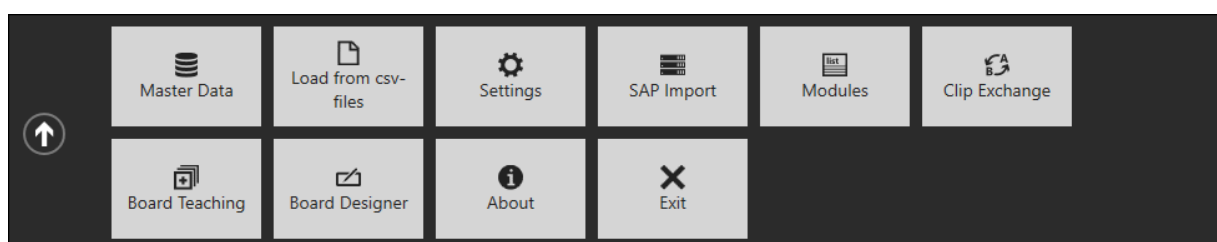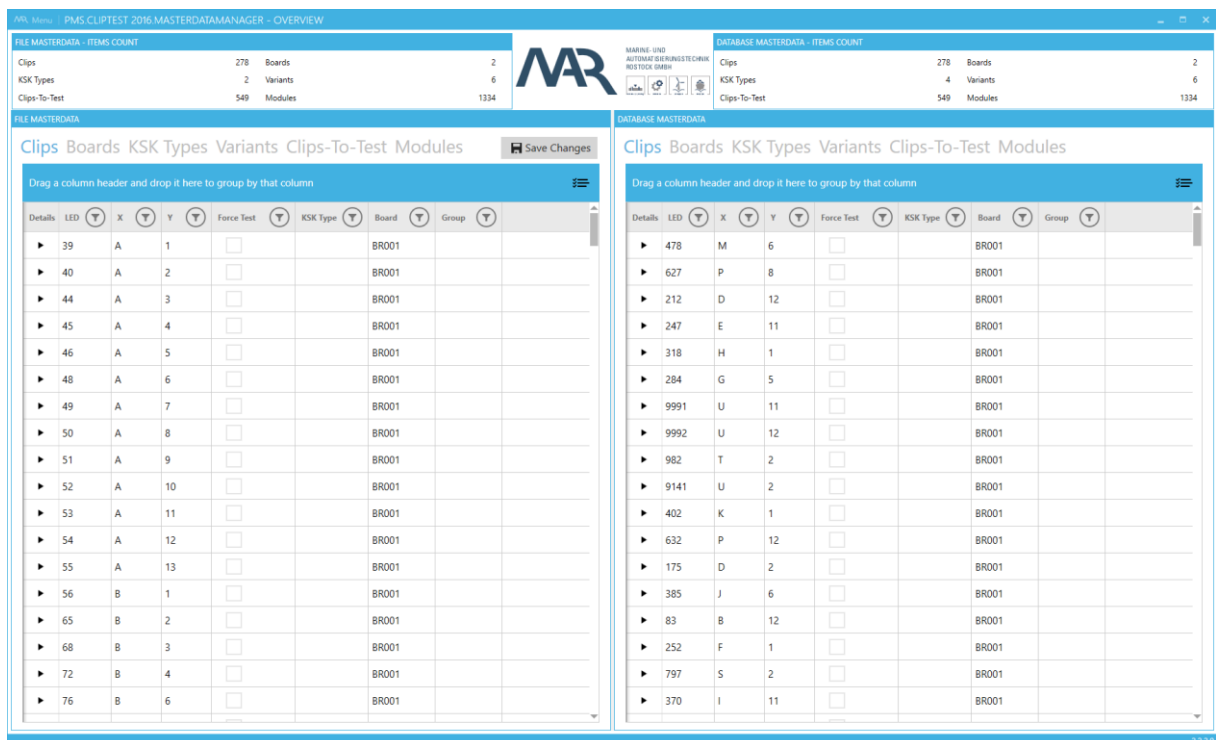
figure 45: PMS.Cliptest.MasterDataManager - MAR Menu

The master data can be modified with the button **Master Data**. A click on it opens a submenu. In this submenu you are able to go back to the Overview, save or discard outstanding changes, delete the master data or create / restore a backup from the complete master data (including configurations and traceability data). To import new master data from csv-files use the menu option **Load from csv-files**. The features **Clip Exchange** and **Board**

Teaching are only available if you have a license for the MasterDataManager Extensions. The ***Board Designer*** is used to configure the image and the matrix layouts of the test panel.

5.2.2 Overview View

This is the initial view of ***PMS.Cliptest.MasterDataManager***. Here you will see all information about the file and database master data. The left side shows the file master data which can be modified with the user interface or by importing data from csv-files. All changes are temporary and will be not submitted automatically. On the right side is the current master data in the database visualized.

figure 46: PMS.Cliptest.MasterDataManager - Overview

The user interface allows you to edit the coordinates and the force test flag for the clips. It is also possible to delete unused data (*delete key on the keyboard*). Some elements of the master data can only be deleted, if the depending element will be deleted too. Before you delete an element, a message shows you how many depending elements are impacted by this operation. All changes are temporary and will be not saved automatically. In the menu option Master Data, you will find the save master data feature. Attention: the save process cannot be undone.

Before the save operation will be performed, the MasterDataManager create a backup for the existing csv-files in the import folders. After finishing the save operation, new csv-file with the current data from the database will be generated. The following table describes the generated files for each file type (***auto-save mechanism***).

File Type	Result
Cliplist	- a separate file for each variant
Coordinates	- a separate file for each board
PinStatus	- a separate file for each board
Variants	- one file which contains all variants with their parameters
Modules	- one files which contains all modules

5.2.2.1 Board operations

The database master data allows you to remove unused board at the tab page for the board. Please consider this operation cannot be undone. After the import of new master data all clips on a board are initialized as unknown because the current state is not available. If you try to run an Empty Test for a new imported board, it will fail because all clips are unknown. To solve this problem, you can press **Initialize Board** (otherwise you have to press every clip at least one time to initialize it). The initialize process can only be performed if no test is running on the selected board. These operations are only possible when the file master data is empty.

figure 47: PMS.Cliptest.MasterDataManager – master data – board operations

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

5.2.3 Settings View

You can open the configuration by usage of the settings menu option. The settings view has five main configuration sections which are organized in tab pages.

5.2.3.1 General

On tab page **General,** you are able to configure the connection of the service, enable the authorization and can configure the logging behavior. For the Service connection is only the hostname (or IP-Address) and the port required. If you are not sure what you have to setup, you can use the **Discover** button, which searches for all PMS.Cliptest.Service applications in the network. You can check your input with the **Test** button.
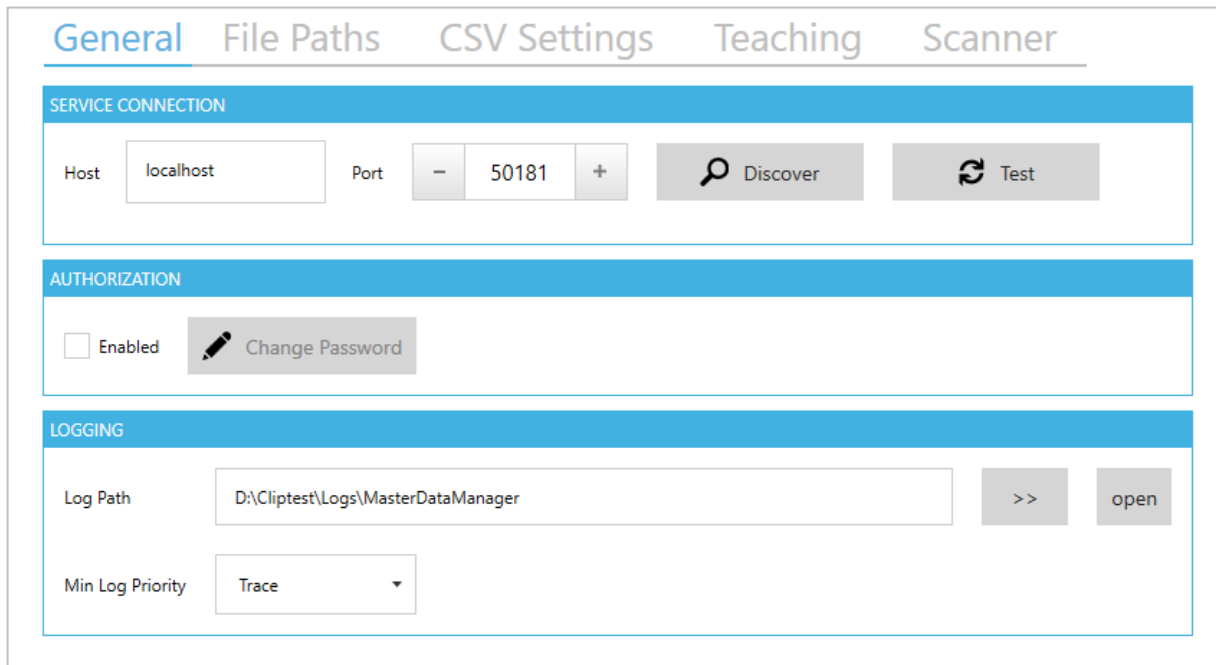

figure 48: PMS.Cliptest.MasterDataManager – Settings: General

The log- / debug-presets can be configured in the Logging group box on this tab page. Select the Log Path and the Log Min Priority.

Log Debug Settings	
Log Path	Enter a Path where *PMS.Cliptest.MasterDataManager* should save Log-Files. The default value is "{InstallationPath}/Logs".
Min Log Priority	Select the priority of Log-Events to be shown in Log-window and will be written to the log file.

5.2.3.2 File Paths

On the second tab page (**File Paths**) you can configure the folders which contains the master data files. For each file type (Cliplist, Coordinates, PinStatus, Variants, Modules) you have to configure a separate folder. All files in the configured folder will be used for the import of the master data. Files in subfolders will be not considered. These folders will be also used for the export of the saved master data.

figure 49: PMS.Cliptest.MasterDataManager – Settings: file paths

5.2.3.3 CSV Settings

To configure the file separators and comment tokens for the import and export of master data use the tab page **CSV Settings**. The configured values in the group **Import** will be used for the import of master data. You can select multiple separators if you need more than one separator. The comment token can be used for comments in the csv-files (these lines will be ignored).

The values in the group **Export** will be used for the files which will be generated by the auto-save mechanism and the Board Teaching view.

figure 50: PMS.Cliptest.MasterDataManager – Settings: csv-file separators and comment tokens

5.2.3.4 Teaching

The page **Teaching** is only relevant for the **PMS.Cliptest.MasterDataManager** Extensions. These four configurable filters can be used to get the filtered clip press events. The option **Min Press Time** defines how long a clip must be pressed to be displayed in the **Board Teaching** mode (0 means no Min Press Time). The option **Max Press Time** can be used to get only clip which are no longer pressed than the configured value. The value Press Timespan describes the timespan in which the number of press events (**Min Press Count**) have to be recognized.

figure 51: PMS.Cliptest.MasterDataManager – Settings: Teaching

5.2.3.5 Scanner

In the Clip Exchange process, you are able to select clips by scanning a barcode. At the tab page **Scanner,** you can set up the configuration of the scanner.
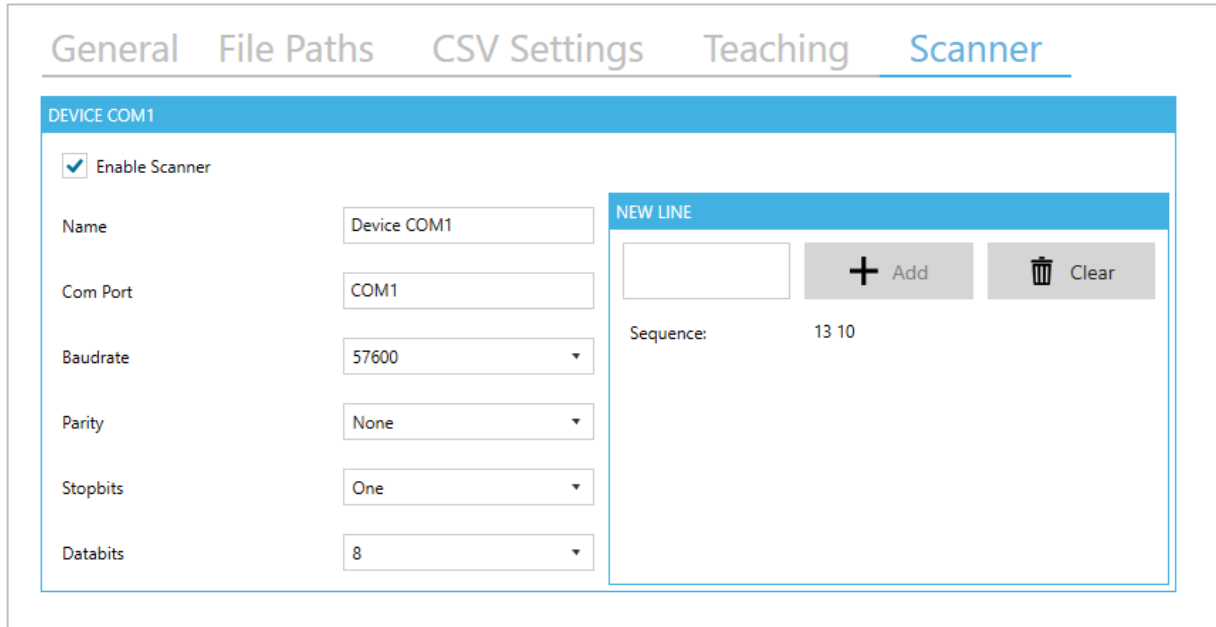
figure 52: PMS.Cliptest.MasterDataManager – Settings: Scanner

The following table describes all settings of the connected RS232 scanner.

Com Port settings	
Enable Scanner	Enables the Scanner.
Name	A unique name which describes the scanner.
COM port	Enter the COM-Port you have connected your scanner to.
Baudrate	Select the baudrate the scanner works with. The default value is 9600.
Parity	Enter the parity the scanner works with. The default value is NONE.
Stopbits	Enter the count of stopbits the scanner works with. The default value is one.
Databits	Enter the count of databits the scanner works with. The default value is eight data bits.

Prefix Length	Number of characters which are used for the address stamping. The default value is 0 (no address stamping enabled).
New Line	Scanners are sending the scanned value with a terminator in data stream. This terminator marks the end of the scanned data. To make *PMS.Cliptest.ViewClient* correctly evaluate scans of the connected scanner this terminator must be set. Normal scanners do use a CRLF (Carriage Return Line Feed) as terminator so this is the default value. Since you cannot enter special chars like CR (carriage return) with your keyboard you can use the ASCII-list. Enter the decimal values of the chars you want to use as terminator, e.g. "13" for CR or "10" for LF. Use an ASCII-table if you're not sure. The default value is CRLF corresponding to ASCII{13} + ASCII{10}.

5.2.4 Manage Master Data

The menu option **Master Data** can be used to manage the currently used data. The first button **Overview** navigates to the Overview view and shows you the current state of the master data.
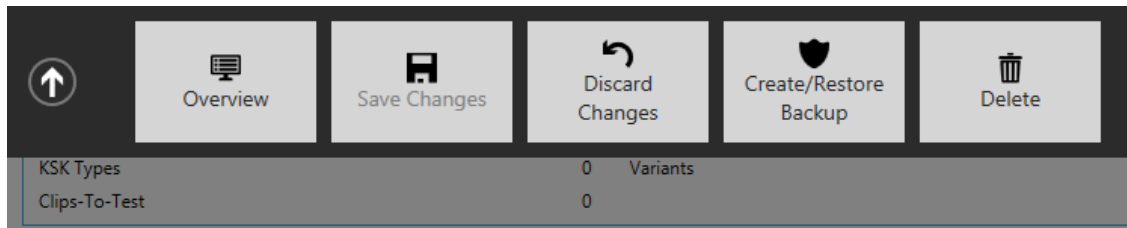

figure 53: PMS.Cliptest.MasterDataManager – Menu Master Data

After the import process you will be asked to save the imported data, if you decline the save process at this moment, you can do it manually with the **Save Changes** button. The **Discard changes** button allows you to discard all changes since the last save action. With the **Create/Restore Backup** button you can use the backup mechanism of PMS.Cliptest (for further information see 5.2.4.1 Backup). Corrupt or old master data can be removed with the **Delete** button (*Attention: the complete master data will be deleted – this process should be not performed if any test is running – create a backup of the database before you run this operation*).

5.2.4.1 Backup

The backup feature allows you to create or restore backups. You can select which components of the master data should be included in the backup. The following table describes each option:

Option	Description
Board data	The board data is composed of the data which was imported from the csv-files and additionally the configured clip positions from image layout.
Configurations	The configurations can only be added to the backup if the board data is selected. This backup option saves the settings of all ViewClients and the configured matrix- and image-layouts.
Traceability data	The traceability data option adds all data which are required for the clip reporting and the traceability of harnesses to the backup. Attention: If you enable this option, the backup process may takes a long period of time and takes a lot of disk space.

To create or restore a backup, select the components which should be included and press **Create Backup** or **Restore Backup** button. After clicking on the create button a new backup archive will be created in the destination folder. To avoid data loss, copy the backup archive to another device. Created backups will be not removed automatically. Please do a manual cleanup of the configured destination folder. In the restore case the latest backup will be restored.

The restore operation can only be performed if the database is empty (deleting only the master data is not enough).

figure 54: PMS.Cliptest.MasterDataManager – Menu Master Data

During the backup process it is possible that PMS.Cliptest needs more free disk space than the created backup archive. The directory for the backups is configurable in the PMS.Cliptest.ServiceConfigurator (see 5.3.2.1 General).

5.2.4.1.1 Automatic Backup

On shutting down, the service application tries to create a backup of the **Board data** and the **Configuration**. The **Traceability data** will be not included in the automatic backup. Only if the backup path is configured, the backup will be created. Further information for the backup back you will get in chapter 5.3.2.1. On creating a new automatic backup the previous automatic backup will be deleted to avoid increasing disk usage.

5.2.5 Import Master Data

To import new master data from csv-files use the menu option **Load from csv-files** (The requirements for these files are described in section *4 Program Master Data Requirements*). By clicking this menu option, the import process from the files will be started. After these process is finished you can save the imported data to the database master data. You have two options to save the outstanding changes, use question dialog after the import or if you want to check the imported data use the save changes action in the submenu of Master Data. New imported clips are initialized as unknown because the current state cannot be determined. All other clips where the state is known will remain in their state.

After saving the imported data to the database, the MasterDataManager creates for all existing import csv-files a backup. If you have added invalid data, you can restore the backup and run the import process again. These backup files remain in the import folder until the user delete them. Please Cleanup the directory manually if you import the master data many times.

The import process will be performed per board. If a board doesn't exist in the database master data, all information from the csv files will be imported into the database. If a board already exists in the database, the information from the csv-files will be used. That means if the csv-files contain more or less clips than the database the information from the file will be used. Boards which are in the database but not in the files will be not changed.

If you add a new board to the master data, all clips are marked as unknown. You have to press each clip at least one time to set it to the right state. To avoid this process, you can use the initialize board feature. This operation set all clip states as unpressed. Please use this option only if you are sure that all clips are unpressed, otherwise you can get invalid test results. For further information see chapter 5.2.2.1.

5.2.6   SAP Import

The SAP Import feature can be used to import harnesses from SAP and run a clip- or empty-test for the specific harness without a connection to a production server. The required information for the tests will be exchanged via csv files. The traceability data for succeeded tests will be exported as csv-files (see 5.2.6.3 Traceability data export). These data have to be imported via other tools into the corresponding tables in the production server.

You are only able to use this feature if there are no unsaved changes in the master data.

5.2.6.1 Import files

For the import are three files required. In the following chapters are all files described and their required columns. To be independent for the column order, the first line in each file have to be the column headers.

ZCrea file

The ZCrea file contains entries for each single harness. Usually the export file from SAP contains a couple of columns, but for the import are only two columns relevant.

Column name	Column type	Comment
KSK	String	This column is used to identify a single harness.
LIUMF	String	This column describes the KSK Type of the harness.

ZCrea Module file

In the ZCrea Module file are all required modules for a harness specified. This file contains all modules for all harnesses of the ZCrea file. The KSK column is used to assign the modules to a harness.

Column name	Column type	Comment
KSK	String	This column is used to assign a module to a single harness.
MODULE	String	This column describes the modules which are required for the harness. The module will be assigned to a specific clip with the OGC Master Data file.

OGC Master Data file

The OGC Master Data file describe the relationship between the module and the LED number of a clip. It is possible that the same LED is used for multiple modules.

Column name	Column type	Comment
LED	Short	This column describes the LED of a clip.
Module	String	This column is used to assign the modules to a LED number.

5.2.6.2 Workflow

The following section describes the general workflow of the import process. At first you have to export the required data from SAP as csv files. After the export you can start the import process by selecting the files and their separators.
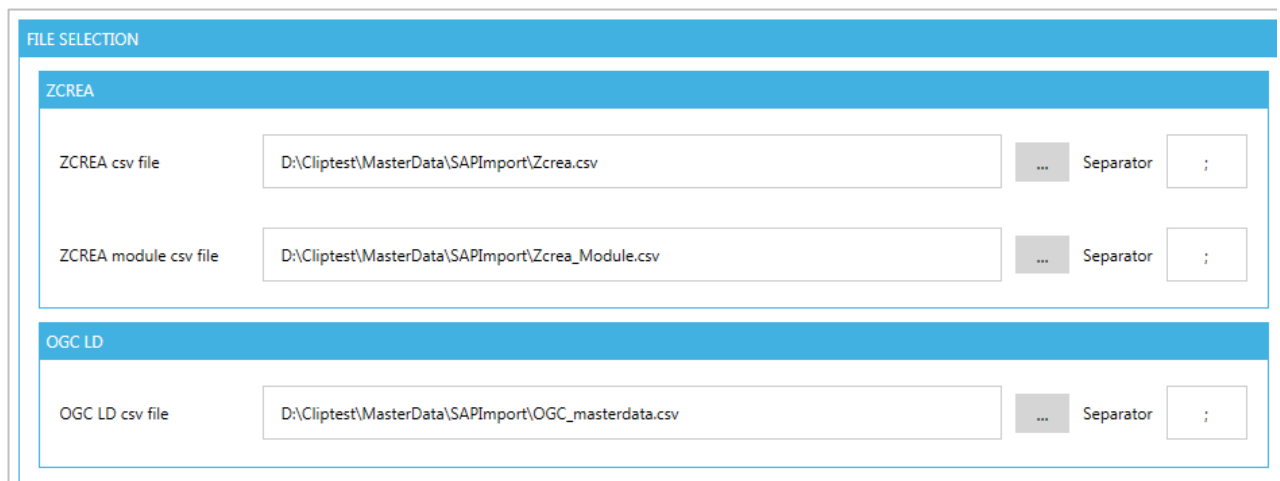

figure 55: PMS.Cliptest.MasterDataManager – SAP Import – File Selection

You can select the file by typing the path or using the file selection dialog behind the input (Button with caption "…"). By default, the separator for the files is a semicolon. If you use another separator you have to configure it to continue. The path selection checks the specified file and tries to find the required columns depending on the type of the file (*ZCrea, ZCrea Module, OGC Master Data*). If the file validation fails, you are not able to continue the import process. Please check the error messages in the path input fields, to find out what is the reason for the error is. If you get an error that a column is missing, the first line of the document doesn't contain the column names or the columns name are not valid. The default values for the column headers are described in the Import files section.

After you have finished the file selection, you have to select the board for that you want to generate the list of Clips-To-Test. If it is required that you are able to run test for the specified harness, you have to import the data for each board separately.

During reading the data from the files the application checks the data. If there are any errors, you will be notified by a desktop alert and in the section for the import data you will find detailed messages for the modules.

figure 56: PMS.Cliptest.MasterDataManager – SAP Import – Import Data

You have two options to save the imported data to the database master data. The first option is to open the details section of a single harness and press the import button or you can select all harnesses that you want to import and use Import selected button below the data grid (selected harnesses which are not valid will be not imported). The import process will create a Variant for each harness with a KSK Type for the LIUMF. The list of Clips-To-Test will be generated by searching all clips on the selected board which have a LED from the import data.

Remarks:

1. If there are any clips in the database which only available for a specific KSK Type and their LED numbers is used during the SAP Import, you will get an error because the clip cannot be assigned.
2. If an LED is used multiple times on the selected board, all clips with these LED have to be pressed to complete a cliptest.

5.2.6.3 Traceability data export

To ensure the traceability of completed tests you are able to export the data into csv-files. By default, the export of these files is not enabled. You have to enable this option in the PMS.Cliptest.ServiceConfigurator (ref.: 5.3.2.1 General).

Remarks:

1. Please ensure that the generated files will be included in a backup.
2. If you enable the export option over a long time, ensure that you have enough free disk space for the files.

5.2.7 Module Management

There are three different types to decide which clips should be tested during a cliptest. You can use the configured file master data with a variant, a higher-level production system or a barcode scan, which contains the cliplist. For the last two options you may need additional master data. This data can be managed with the MasterDataManager or by an import via modules master data files. You can open the module management with the menu option **Modules**.

figure 57: PMS.Cliptest.MasterDataManager – Module Management

You will find a list of all configured modules. Each module has a list of LED numbers in the details section. Select a module to see the details. You can change the number of the module by selecting the module and click in the number column. The column will change to an editor field. Please consider that each module number have to be unique, otherwise you will be not able to save the changes. All changes will be performed temporarily, press the Save changes button to submit the changes to the database or Discard changes to reject the changes. Attention: the actions cannot be undone.

If you want to add a new module, use the **Add** button in the bottom left corner. A module, which is not used anymore, can be deleted with the button **Delete Selected**. Open the details section to manage the LED numbers of a module. You can add new LED numbers with the **Add** button and remove them by select a LED number and click on the **Delete selected** button. Similar to the module number you can change the LED number with a double click on the value in the LED column.

The column validation informs you if the configuration is not valid or there are some remarks. Remarks will be shown as orange exclamation marks with a tooltip message. There will be also red exclamation marks; these are displayed if the configuration is not valid. In that case, you will be not able to save the configuration. Fix all error to save the configuration.

5.2.8 Clip Exchange View

This feature is only available if the **MasterDataManager Extensions** are licensed.

A wizard helps you to exchange the address of an existing clip. In the first step of the wizard you are asked to select the modes, how to select the clips or addresses you want to replace. The following table describes the available clip selection modes.

Mode	Old Clip	New Clip
Manual Input	Type the address of the clip, which already exists in the MasterData.	Type the address of new the clip. You can select addresses, which are already in use or not match with the physical address, but this could cause unintended behaviors.
Press Selection	Press the clip, which already exists in the MasterData. and select it from the list. (To configure the press selection use the tab page **Teaching** in the settings)	Press the new clip. You can press clips which are already in use, but this could cause unintended behaviors.
Scan Selection	Scan the address of the clip, which already exists in the MasterData.	Scan the address of the clip. You can scan clips which are already in use, but this could cause unintended behaviors.
MasterData Selection	Select a clip from the list of the MasterData.	Select a clip from the list of the MasterData.

figure 58: PMS.Cliptest.MasterDataManager – Clip Exchange – mode selection

After selecting a clip, you can choose the exchange mode. Use Press if you want to replace a (not damaged) clip or use Manual for a complete clip exchange. If you choose Press, start in the next step listening process and press the new clip. The clips which fulfill the configured

values in the teaching section of the settings will be displayed in a list. Stop the listening process and select an address to proceed. Before the clip exchange will be executed, you can check the selected clip. Start the exchange process with pushing the Exchange button.

5.2.9 Board Teaching View

This feature is only available if the **MasterDataManager** Extensions are licensed.

figure 59: PMS.Cliptest.MasterDataManager – Board Teaching

To start the **Board Teaching** process, you have to enter a **Board** and set it active with the **Change** Button. If these requirements are fulfilled, you can start the teaching with the **Start** button. Similar to the board you can change the **KSK Type**. The configured KSK Type will only be used if the option **Use current KSK Type** is checked. Otherwise the KSK Type will be empty.

Information about the last clip will be displayed in the center of the screen. The coordinates and the LED of the next clip can be set below to the last clip. On the bottom of this view you can find a list of the current teached clips. With a click on the **Save** button the coordinates and cliplist files will be generated and saved in the configured input folders. The **Reset** button resets all settings to the default.

5.2.10 Board Designer View

To configure the board layouts, use the menu option **Board Designer**. In the submenu of board designer, you can select between to modes.
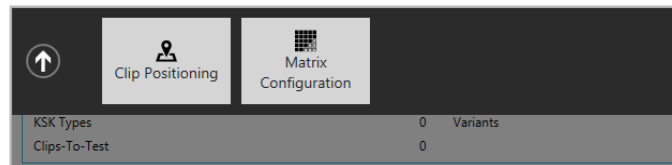

figure 60: PMS.Cliptest.MasterDataManager – Menu Board Designer

The first mode is **Clip Positioning.** Use this to configure the image layouts. The other mode is **Matrix Configuration**. It enables the configuration of the matrix layout to you.

5.2.10.1   Clip Positioning

To configure the image layout are three steps required. At first select the board you want to configure. Then load the background image, if the current image doesn't match. After selecting the board layout place all clips by drag and drop. If you want to increase the accuracy of the positions, you can use the zoom control and the diameter setting. To display or hide the zoom control click double on the background image. To complete the positioning process save the positions.

If another board has the same positions and same LEDs, you can clone them by using **use settings for other board**. The clone feature will only work if the source and destination board have the same LED numbers and each LED number is unique on the boards.
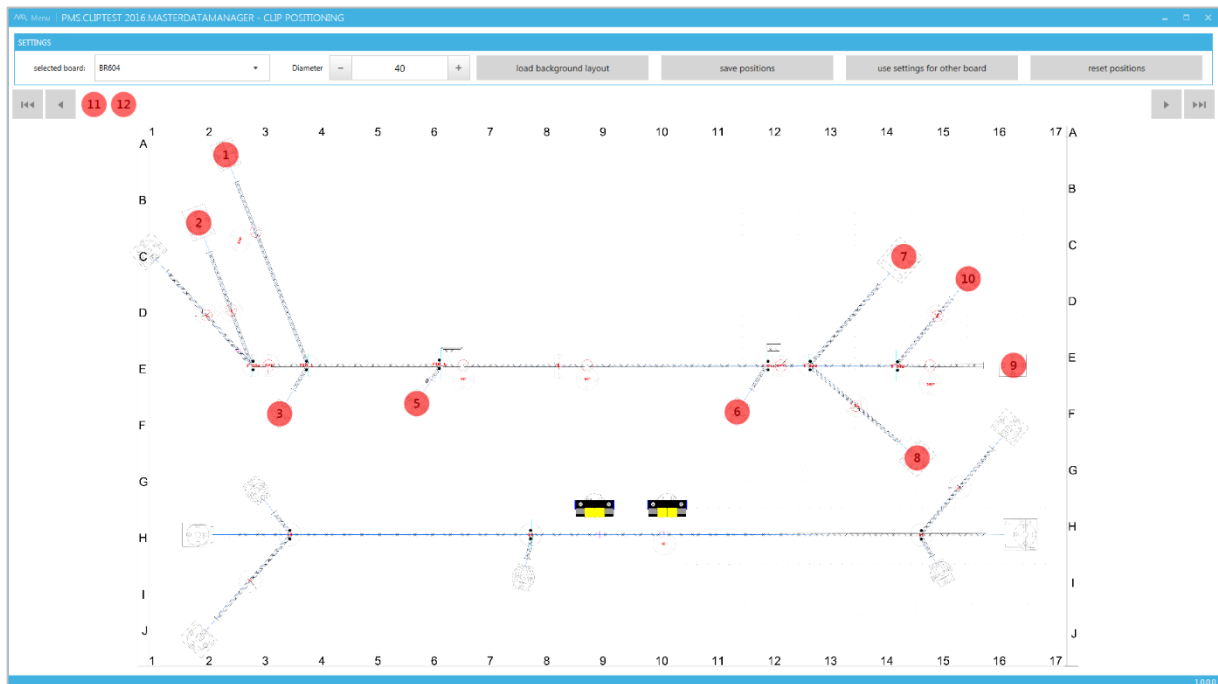
Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

figure 61: PMS.Cliptest.MasterDataManager – Board Designer: Clip Positioning

5.2.10.2 Matrix Configuration

To configure the matrix layout are three steps required. At first select the board you want to configure. After selecting the board, you can add columns and rows for sections were no clips are placed. Adding columns and rows is required if the area of physical coordinates is greater than the area of the master data coordinates.
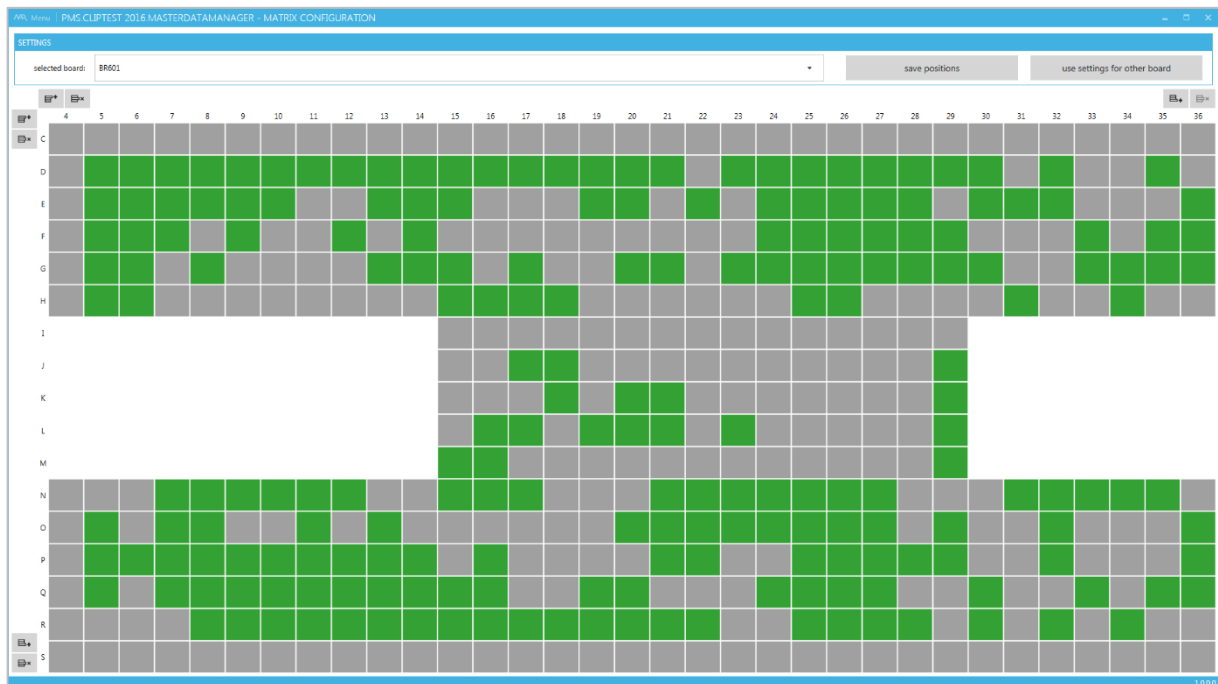

figure 62: PMS.Cliptest.MasterDataManager – Board Designer: Matrix Configuration

For example, a board has physical coordinates for the columns from 1 to 25 and for the rows from A to J. The imported master data contains only clips with column coordinates between 5 and 25. In This case you can add the missing columns by clicking the items in the corners.
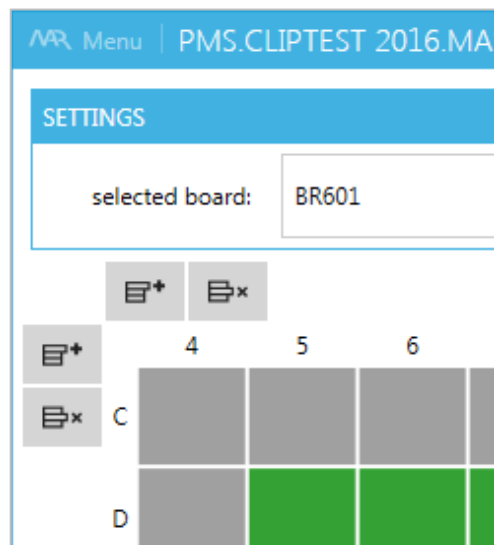
figure 63: PMS.Cliptest.MasterDataManager – Board Designer: Matrix Configuration – Add/remove columns and rows

The green cells contain clips and the gray cells are empty. If some cells are not on the board, you can exclude them by clicking on it. Excluded cells are displayed with a white brush. Excluding cells is not required for using this test panel layout. It's only a feature to visualize the board with its physical dimensions.

Important:

If you want to configure the layout for a specific board, the coordinates of these board have to fulfill the following two requirements:

1. All coordinates are configured
2. The coordinates for each direction (X or Y) are of the same type
    a. only letters (between A and Z) or
    b. only numbers (between 0 and 99))

5.3 PMS.Cliptest.ServiceConfigurator

With the **PMS.Cliptest.ServiceConfigurator** (short: ServiceConfigurator) you are able to modify the settings of the **PMS.Cliptest.Service** on the machine where the **ServiceConfigurator** is running. If the security option is enabled, the start screen of **ServiceConfigurator** is the **Login** view. Enter the configured password to proceed. The password protection of the service configuration can be changed via the **Service Connection** menu option.

figure 64: PMS.Cliptest.ServiceConfigurator – Login

5.3.1 ServiceConfigurator options

The ServiceConfigurator has a few options. To configure them, open the options view of the ServiceConfigurator by opening the menu in the left top corner and select **Service Connection**.

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

figure 65: PMS.Cliptest.ServiceConfigurator – options

In the first section of the options view you can enable or disable the service notification. If the feature is enabled, the server configurator notifies the service, that the settings were changed. By default this option is enabled. The configuration of host and port is similar to the other applications. Please note that you can only configure the settings of a service on which the service configurator is running.

You can activate in the options of the ServiceConfigurator the authorization. If this option is checked, you have to enter a password for change the settings. Here you can also configure the password, which is required to open the configuration.

5.3.2 Configuration

The configuration view is divided into several tab pages. The following chapter describes all pages. By default, not all tab pages are displayed (simple mode). The expert mode with all configuration options can be activated in the *Menu* with a toggle switch on the right side. In the documentation are all options described, which are maybe not displayed in the simple mode. When you are logged in to modify the settings, you can access the service connection settings via the menu. There you can change the authorization feature.

5.3.2.1 General

At the tab page *General* you can configure the *log- / debug presets*, the **traceability data export**, the master data backup and the communication settings (**Interfaces and Ports**)

figure 66: PMS.Cliptest.ServiceConfigurator – Settings: General

The log- / debug-presets can be configured in the first section in this view. Select the Log Path and the Log Min Priority. The log files will be automatically moved into an archive subfolder in the configured path, if the application will be restarted or the file is bigger than 1 megabyte. The archive folder contains the last 30 log files.

Log Debug Settings	
Log Path	Enter a Path where *PMS.Cliptest.Service* should save Log-Files. The default value is "{InstallationPath}/Logs".
Min Log Priority	Select the Priority of Log-Events to be shown in Log-window.

The option **Export Traceability Data** is only relevant if you use the SAP Import feature in the MasterDataManager. Enable the export to csv-files to ensure the traceability. Select a directory where the export files should be stored. Consider that PMS.Cliptest.Service needs write access to the directory and you have enough free disk space. For further information see 5.2.6.

In the group box **Master Data Backup** you have to select a folder where the backups of the master data should be stored. Ensure that there is enough free disk space, the service hat write access and the backups are safe in case of data loss.
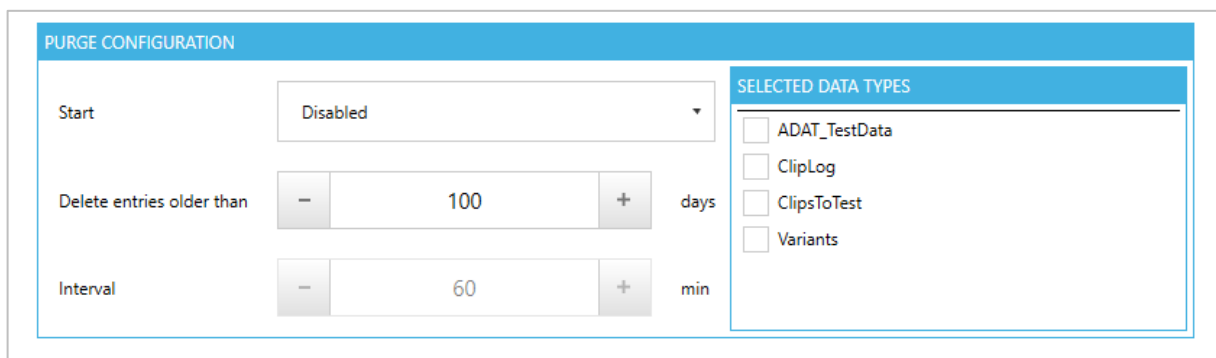
In the last group box of the general tab page you can expand the port configuration for the PMS.Cliptest Service. The default port for the WCF communication is 50181. This port is required for the basic communication between the client applications and the service. You can enable or disable additional communication interfaces and configure the used port. Use for each option a unique port which will be not used from other applications and ensure that this port will be not blocked by the firewall.

In the group box **Updates** you can configure a folder, which will be used for minor version updates. Use this option only if you are sure that the update will not affect the existing installation.

5.3.2.2 Backend

The **PMS.Cliptest.Service** needs a backend for its own data (masterdata storage). To configure it use the tab page **Backend.** There are different types of backends, which you can use to store the master data. Depending on the selected backend you have different configuration options. The available backends and their corresponding configuration will be explained in the following chapters.

PMS.Cliptest write data to its backend, which will be not cleaned automatically, e.g. the table *COM_OGC.ClipLog*. If you want to clean these tables, you can configure the cleaning process with the **Purge Configuration**. The setting **Interval** is only used if the start condition is configured to *Run purge periodical*. You are able to set the duration between two purge processes in minutes. By default, the value is 60 minutes. The purge process delete only entries, which are older than the configured duration in days (default value 100 days). In the list of the **Selected Tables,** you can configure on which tables the purge process should be performed.

figure 67: PMS.Cliptest.ServiceConfigurator – Settings: Backend – Purge configuration

Attention: The purge process can only be performed while no tests are active to avoid inconsistent data.

There are different start conditions for the purge process. By default, is the purge process disabled. The following table describes the available types.

Start Condition	behavior
Disabled	The purge process will be not performed.
Run purge on next start	The purge process will be performed one time at the next start of the PMS.Cliptest.Service.
Run purge on every start	The purge process will be performed on every start of the PMS.Cliptest.Service.
Run purge on every close	The purge process will be performed on every close of the PMS.Cliptest.Service.
Run purge periodical	The purge process will be performed periodical.

5.3.2.2.1 SQL Backend

This backend is based on a SQL server database. You are able to set the server address and the database. For the authentication you can use the *Windows Integrated Security* or a combination of username and password (*Attention: if you choose Windows Integrated Security, you have to ensure that the local system account has access to the database or you have modify the user account which starts the service*). To verify the input, use the Test Connection feature. If the database does not exist, you can create it at the configured SQL server.

figure 68: PMS.Cliptest.ServiceConfigurator – Settings: Backend – SQL Backend

5.3.2.2.2 File Backend / Demo Backend

These backends are based on files. The file and demo backends have similar behaviors regarding functionality. The demo backend has additional example data for testing scenarios.

For this backend is in the default case no configuration required. If you want to change the storage path for these backends, uncheck the option **Use default path** and select the path. The option **Write Runtime Data** configures the interval how often the runtime data will be saved from memory to the files.

figure 69: PMS.Cliptest.ServiceConfigurator – Settings: Backend – File Backend

5.3.2.3 Production Server

PMS.Cliptest has the ability to be controlled by a higher level manufacturing execution system (e.g. MES). If you want to use **PMS.Cliptest** in combination with a production server, you have to select the communication interface on the **Production Server** tab page. Depending on the interface of the production server you have to enter different settings. BY default the production server NONE is selected, which will be used for the standalone mode.

Please contact support@cliptest.de for detailed information about available interfaces.

5.3.2.4 Receivers

If your receivers are directly connected to the **Service,** you can configure their parameters at the **Receivers** tab page. Select a receiver from the top list and configure it below.
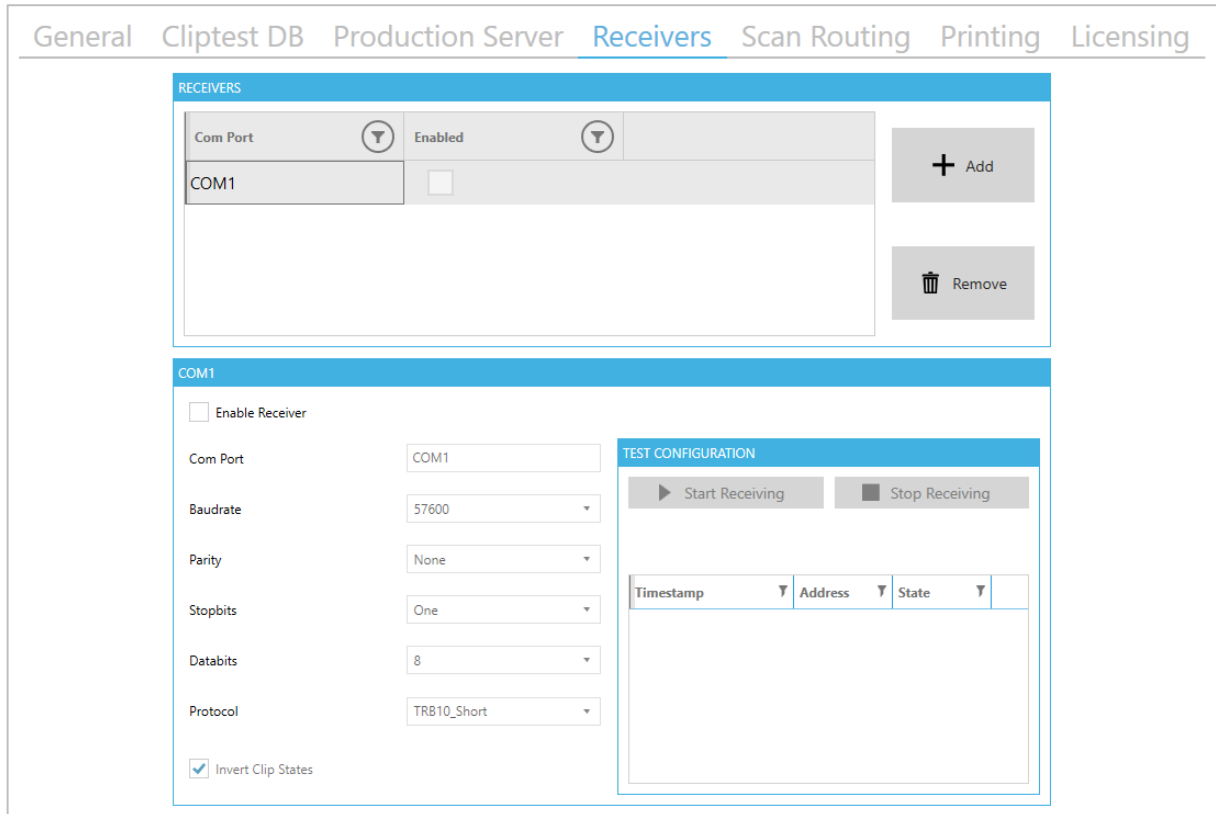
figure 70: PMS.Cliptest.ServiceConfigurator – Settings: Receivers

The following table explains all configurable parameters of a receiver.

Com port settings	
Enable Receiver	Enables the receiver.
Name	The Name is used to identify the receiver.
COM port	Enter the COM-Port you have connected your radio receiver to.
Baudrate	Select the baudrate the receiver works with. Short protocol receivers (TRB10_Short) should work with 57600 baud, long protocol receivers (TRB10_Long) should work with 9600 baud. The default value is 57600 baud.
Parity	Select the parity the receiver works with. Short protocol receivers as well as long protocol receivers should work with Parity NONE. The default value is NONE.

Stopbits	Select the count of stopbits the receiver works with. Short protocol receivers as well as long protocol receivers should work with one Stopbit. The default value is one.
Databits	Select the count of databits the receiver works with. Short protocol receivers as well as long protocol receivers should work with eight data bits. The default value is eight data bits.
Protocol	Select the protocol the receiver works with. Short protocol receivers are working with TRB10_Short. Long protocol receivers are working with TRB10_Long. The default value is TRB10_Short.
Invert Clip Stats	Some radio receivers may invert all received clip states. Use this option to reinvert the clip states for correct processing. Use Clip debug window to obtain if this option must be activated. The default value is deactivated.

If you want to check the receiver configuration, you can use the test view next to the configuration. Press **Start Receiving** to test the current setting. **Attention:** the test mechanism works only if the selected COM-port is not used. For example, the receiver is connected to the COM-port COM1 and is enabled, you cannot test the settings form port COM1. You have to stop the service or disable the receiver, save the settings and after that you can test your receiver settings.

5.3.2.5 Scan Routing

The **PMS.Cliptest.ViewClient** provides the ability to use multiple barcode scanners with only one cradle. Each scanner needs a prefix for identification. The Address Stamp Mapping tab page is used to assign the scanner to the correct **ViewClient** (Ref.: 7.1 Service Configuration). The Address Stamp Mapping also allows to use one scanner for multiple clients. You can define a mapping between the source (from where the scan is performed) and one or many destinations (clients which use the scan information). This setting is also required if scanners are connected directly to the service.
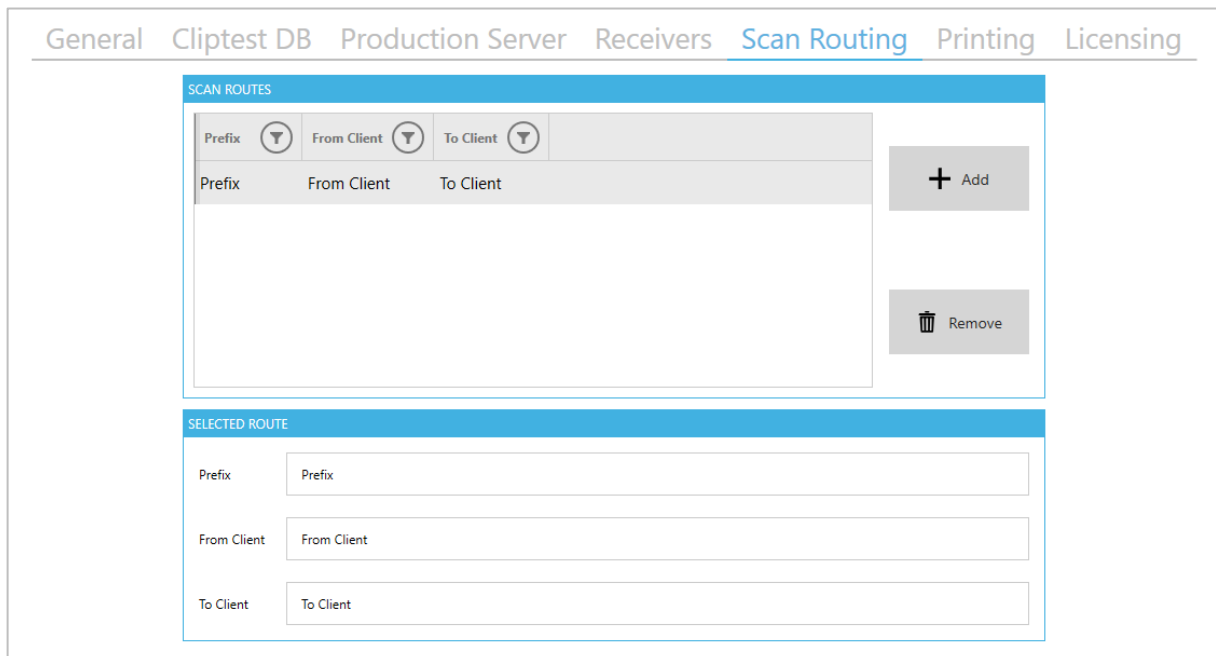
figure 71: PMS.Cliptest.ServiceConfigurator – Settings: Scan Routing

5.3.2.5.1 Example Configurations

Example 1

In the first example is one scanner directly connected to the Service. To assign the scanner to a specific ViewClient, you have to configure only one mapping. The prefix of these mapping can be optionally configured, if the scanner uses a prefix. If the scanner is connected to the Service, you have to use for the option ***From Client*** the constant ***Service*** (case sensitive). The option ***To Client*** is used for the ***ViewClient*** which should be responsible for the scans. Enter the configured ***ClientId*** of the responsible ***ViewClient*** (attention: the ***ClientId*** is case sensitive). The following table shows an example configuration.

Configuration for ClientId ASS01_01	
Setting	Value
Prefix	
From Client	Service
To Client	ASS01_01

Example 2

The second example describes a configuration, which have two scanners directly connected to the Service. To identify the scanners prefixes are used. The first scanner has the prefix ***A*** and the other the prefix ***B***. To assign the scanners to a specific workplace, you have to configure two mappings. Caused by the directly to the Service connected scanners, you have to use for the option ***From Client*** the constant ***Service*** (case sensitive). The following table shows an example configuration.

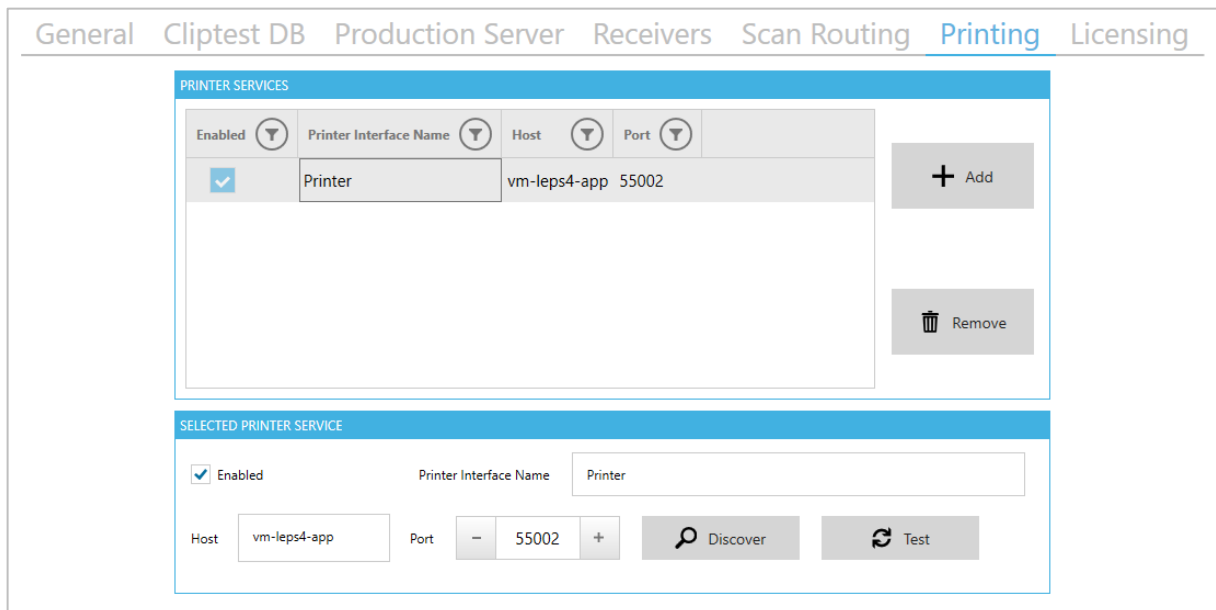| Configuration for ClientId ASS01_01		Configuration for ClientId ASS01_02	
Setting	Value	Setting	Value
Prefix	A	Prefix	B
From Client	Service	From Client	Service
To Client	ASS01_01	To Client	ASS01_02

Example 3

In this example are two scanners connected to the ***ViewClient*** with the ***ClientId*** *ASS01_01* and the ***ViewClient*** with the ***ClientId*** *ASS01_02* has no connected scanner. Prefixes are

used to identify the scanners. The first scanner has the prefix **A** and the other the prefix **B**. To assign the scanners to a specific workplace, you have to configure two mappings. The scanner with the prefix **A** is used for **ClientId** *ASS01_01* and the other with the prefix **B** for the **ClientId** *ASS01_02*. The following table shows an example configuration.

Configuration for ClientId ASS01_01		Configuration for ClientId ASS01_02	
Setting	Value	Setting	Value
Prefix	A	Prefix	B
From Client	ASS01_01	From Client	ASS01_01
To Client	ASS01_01	To Client	ASS01_02

5.3.2.6 Printing

The **Printing** tab is used to manage the PMS.PrintProcessor instances. You can define multiple instance and enable or disable each separately. The **PrintProcessor Name** and the **Host** and **Port** combination have to been unique in the list.

figure 72: PMS.Cliptest.ServiceConfigurator – Settings: Printing

5.3.2.7 Eventing

For using this feature is a **PMS.Cliptest.Eventing** license required. The configuration of the *Eventing* is possible, but it will be not processed while PMS.Cliptest is running.

The *Eventing* feature enables to the user many possibilities to automatize workflows or manage interactions with other applications / devices. Bunches of possibilities are grouped into a device that has actions and events. All devices have a main configuration, which will be used to enable or disabled a device, set up a name to identify it and the type of the device (all available device types are explained in chapter 6). Some devices have additionally a specific configuration, e.g. connection information for a PLC, etc. Each action or event can have its own configuration, which will be used to raise the event or handle the action.

figure 73: PMS.Cliptest.ServiceConfigurator – Settings: Eventing

On the left side are the configured devices displayed as a tree view. To manage the devices, use the add- and remove-buttons. The device configuration can be opened via selecting the device node in the tree view. A device node has up to two sub nodes, one for the implemented events and one for the possible actions. To access the event or action configuration select the request node in the tree view. For further information about the configuration of actions and events, check the explanations for each type (see chapter 6).
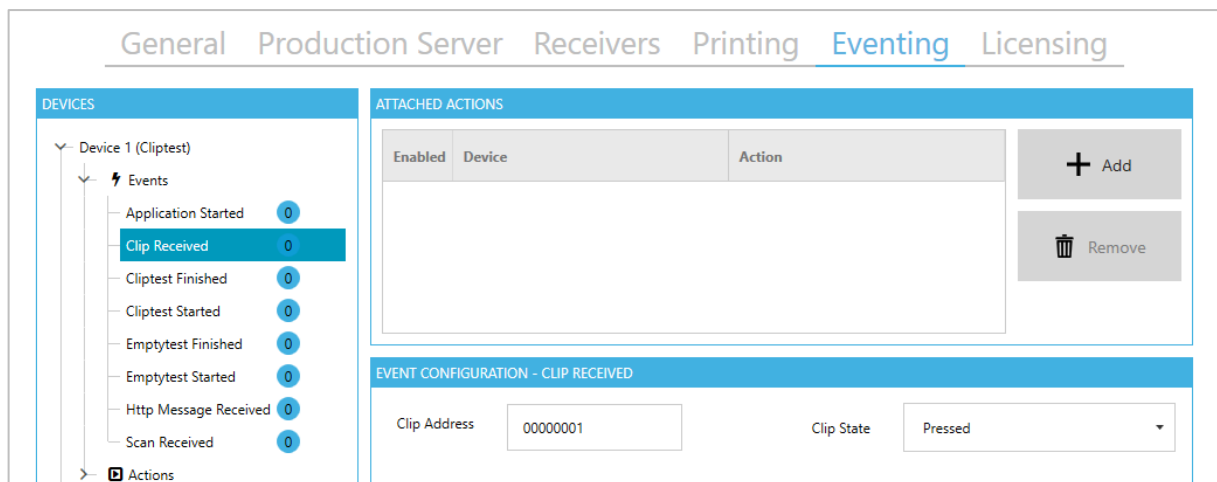
figure 74: PMS.Cliptest.ServiceConfigurator – Settings: Eventing – event configuration

Each event can execute multiple actions, which will be executed asynchronously. Please note, that there is no guarantee that the actions of an event will be handled in the configured order. To attach an action to an event, select the event node and use the ***Attached Actions*** list. The configuration of an attachment can be change in the grid line. Select at first the device, which should handle the event and after that, select the action of the device. Caused by the specific data format from an event, it is not possible to handle each event with all actions. The incompatible actions will be displayed in a gray style. If an incompatible action is attached to an event, the actions will be not executed and the issue will be logged.

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

5.3.2.7.1 Device Context Menu

The device tree view has a context menu which enables several options to the user. The available options depending on the selected node in the tree view. Some of the events and actions can be duplicated, to have the same items with different configurations. For example it is possible to duplicate the Test Changed event to have single events, for changes in a cliptest and changes in an emptytest.
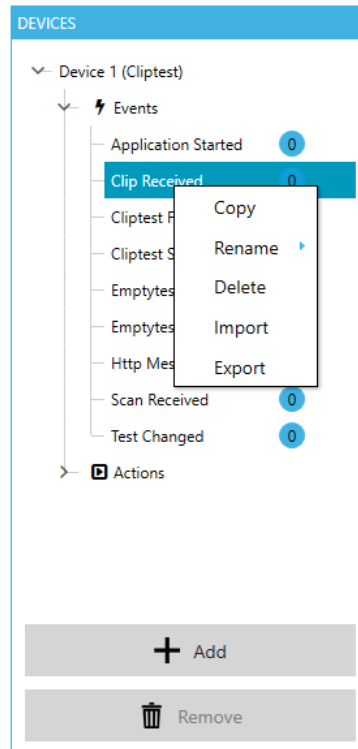
figure 75: PMS.Cliptest.ServiceConfigurator – Settings: Eventing – context menu

option	Description
Copy	This button duplicates the selected item. This option is not available for all actions and events of a device.
Rename	This button renames the selected item. This option is only available if the copy feature is enabled for an item.
Delete	This button deletes the selected item. This option is only available if the copy feature is enabled for an item. The last remaining copy of an item cannot be deleted.

Import	This button imports the eventing configuration from a created export. Therefor the existing configuration will be dropped and the data from the import will be applied.
Export	This button creates an export of the complete eventing configuration. With this option the user can distribute the eventing configuration between different PMS.Cliptest instances or create backups.

5.3.2.8 Licensing

The **Licensing** tab is used to manage the currently used licenses. You can see all active licenses and their types. If the license is stored on an usb dongle, the **Type** column shows a small icon. To add a new license, start the license wizard by clicking **Add New Licenses**.

figure 76: PMS.Cliptest.ServiceConfigurator – Settings: Licensing

6   PMS.Cliptest.Eventing

PMS.Cliptest.Eventing is a powerful feature to automate various tasks to help operators, enhance product quality and reduce testing time. It is based on logical or physical devices that provide information as events and consume information to perform an action. Available devices can be combined by attaching one or more actions to an event. Furthermore, each action can consider one or more conditions which need to be fulfilled to execute an action. Conditions are to be used to check any case and result in true or false. Only when all configured conditions result in true then this action will be executed.

This feature is implemented as part of the PMS.Cliptest.Service and can be configured in PMS.Cliptest.ServiceConfigurator. It was introduced in version 20.2.1.0 and is continuously extended with more devices.

This section describes all currently available devices with their events and actions. The events have some example actions which can be triggered from the event. This list will be ongoing extended by more devices and can be easily added to existing installations.

6.1   Data Types

The following tables describes the available data types, which will be provided from the events. The description of an event contains the provided type and the description of an action contains the accepted types.

Data Type	Description
None	No specific data will be provided
Clip Status	information about a clip with its address and status
Test Information	All information about a test
Text	Raw text data
Scanner Message	Scan data from a view client
All	All Data Types are allowed and will be process if possible.

6.2 Cliptest

PMS.Cliptest acts as a device for the Eventing feature. The following chapter describes all events and actions of PMS.Cliptest.
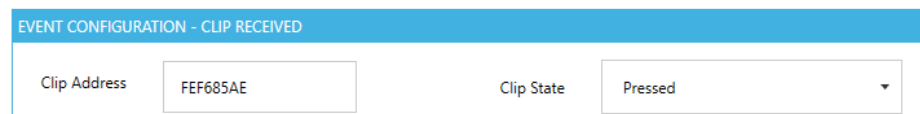
6.2.1 Events

6.2.1.1 Application Started

Trigger	only one time when the PMS.Cliptest.Service application starts
Data Type	None
Configuration	no specific configuration

6.2.1.2 Clip Received

Trigger	Clip changes to configured address and the configured state
Data Type	Clip Status
Configuration	

figure 77: PMS.Cliptest.ServiceConfigurator – Eventing: Clip Received

filter options for **Clip Address** and **Clip State** for which clip the event should be raised

Example actions	- Start Cliptest on pressing a specific clip - Start Emptytest on releasing a specific clip

6.2.1.3 Cliptest Finished

Trigger	a Cliptest was finished (independent from the test result)
Data Type	Test Information
Configuration	no specific configuration

6.2.1.4 Cliptest Started

Trigger	a Cliptest was started
Data Type	Test Information
Configuration	no specific configuration

**Marine- und Automatisierungstechnik
Rostock GmbH**

**Hundsburgallee 9 c
18106 Rostock**

**Date:
2021-02-23**

6.2.1.5 Emptytest Finished

Trigger	an Emptytest was finished (independent from the test result)
Data Type	Test Information
Configuration	no specific configuration

6.2.1.6 Emptytest Started
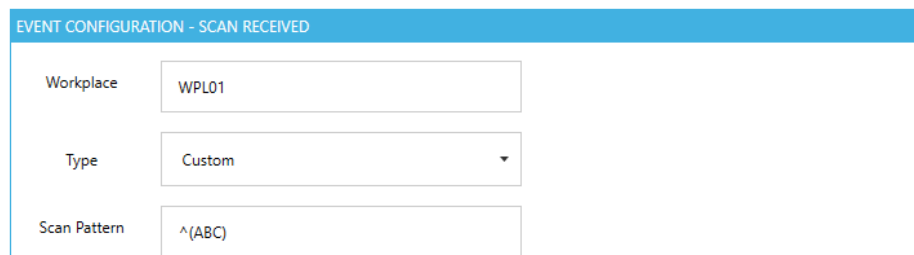
Trigger	an Emptytest was started
Data Type	Test Information
Configuration	no specific configuration

6.2.1.7 Http Message Received

Trigger	cliptest receives a message from the web interface
Data Type	Text
Configuration	no specific configuration

6.2.1.8 Scan Received

Trigger	a ViewClient sends scan data to the Service
Data Type	Scanner Message
Configuration	

EVENT CONFIGURATION - SCAN RECEIVED

Workplace	WPL01
Type	Custom ▼
Scan Pattern	^(ABC)

figure 78: PMS.Cliptest.ServiceConfigurator – Eventing: Scan Received

- filter for a workplace which sends the scanned data (optional)

- filter for the type of the scanner message

- filter for the scanned data with regex pattern matching (optional)
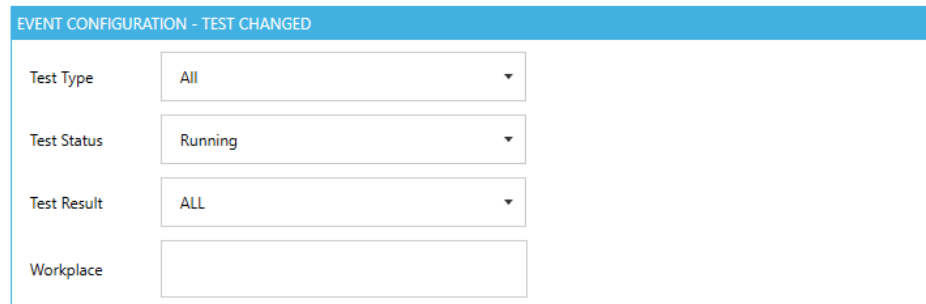
6.2.1.9 Test Changed

Trigger	Occurs when the state a test changed.
Data Type	Test Information
Configuration	

EVENT CONFIGURATION - TEST CHANGED

Test Type	All ▼
Test Status	Running ▼
Test Result	ALL ▼
Workplace	

figure 79: PMS.Cliptest.ServiceConfigurator – Eventing: Scan Received

Test Type: optional test type filter

Test Status: optional test status filter

Test Result: optional test result filter

Workplace: optional Workplace filter

6.2.2 Actions

6.2.2.1 Change Clip Status

Description	Updates the status of a clip, e.g. triggered by a hardware device
Data Types	Clip Status
Configuration	no specific configuration

6.2.2.2 Handle Scan

Description	Handles the provided data as a scan.
Data Types	Text, ScannerMessage
Configuration	This action has no specific configuration.

6.2.2.3 Import Master Data
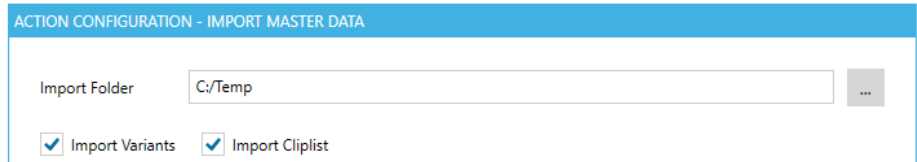
Description	Cliptest tries to start the import of the master data from the configured path.
Data Types	All
Configuration	

figure 80: PMS.Cliptest.ServiceConfigurator – Eventing: Import Masterdata

- Import folder where the master data files are located
- option to enable the Variants import (requires csv files with the header **#VariantName; KSKType; Comment; Parameters**)
- option to enabled the Cliplist import (requires csv files with the header **#VariantName; Board; LED; Comment**)

6.2.2.4 Log Event data

Description	Writes the provides event data to the eventing log file with log level **Info**
Data Types	All
Configuration	no specific configuration

6.2.2.5 Show Message

Description	Shows a popup on all ViewClients with the provided data
Data Types	Text or Scanner Message
Configuration	no specific configuration

6.2.2.6 Start Cliptest

Description	Starts a Cliptest
Data Types	Test or Clip Status
Configuration	
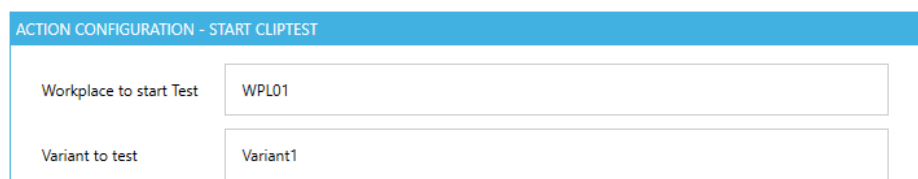
figure 81: PMS.Cliptest.ServiceConfigurator – Eventing: Start Cliptest

- The workplace (for the ViewClient) to start the test for. Only required in case Data Type is Clip Status.
- The Variant to be tested.

6.2.2.7 Start Emptytest

Description	Starts an Emptytest
Data Types	Test or Clip Status
Configuration	

figure 82: PMS.Cliptest.ServiceConfigurator – Eventing: Start Emptytest

- The workplace (for the ViewClient) to start the test for. Only required in case Data Type is Clip Status.
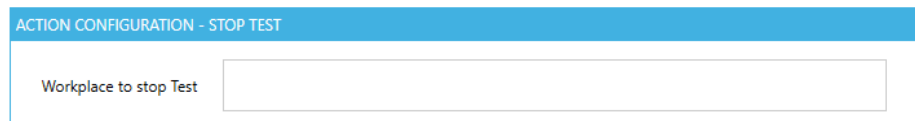
6.2.2.8 Stop Test

Description	Stops a test
Data Types	Text, ScannerMessage, All (value from static configuration)
Configuration	

figure 83: PMS.Cliptest.ServiceConfigurator – Eventing: Stop Test

- The workplace (for the ViewClient) to stop the test for. If this value is configured, the event data will be not used.

6.2.3 Conditions

6.2.3.1 IsTestRunningForBoard

Description	Checks if a test is running for a provided board
Data Types	Text, ScannerMessage, All (value from static configuration)
Configuration	This condition has no specific configuration.

6.3 Timer

6.3.1 Events

6.3.1.1 Scheduler

Trigger	Triggered by configured schedule. Can be used to perform cyclic tasks at defined times like for example every day at midnight.
Data Types	None
Configuration	
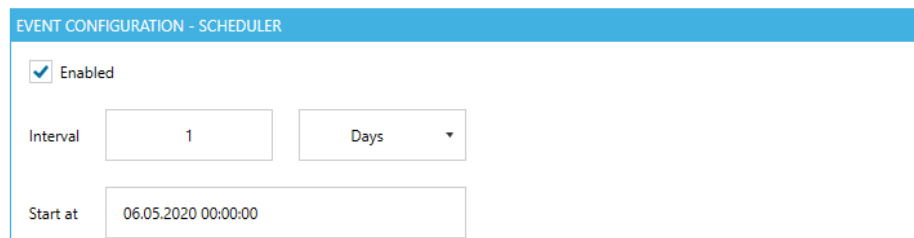
figure 84: PMS.Cliptest.ServiceConfigurator – Eventing: Scheduler

- Interval is the time between events
- "Start at" defines the date and time which is used as first occurrence of cyclic events. If the date or time is in the past, all further events will be relative to the start date and time.

6.3.1.2 Timer Elapsed

Trigger	Not external triggered. A timer will raise events considering the configured interval
Data Types	None
Configuration	
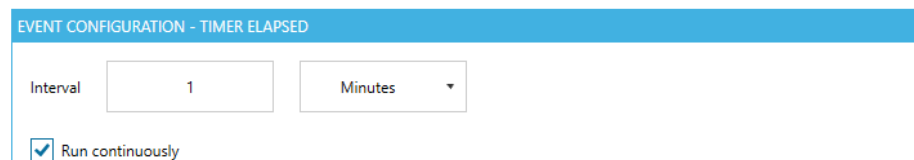
figure 85: PMS.Cliptest.ServiceConfigurator – Eventing: Timer Elapsed

- Configure the interval between the events of this timer
- Check "Run continuously" if events should be raised cyclic, if unchecked the event is raised once

6.3.2   Actions

6.3.2.1 Start Timer

Description	Starts the timer
Data Types	All
Configuration	no specific configuration

6.3.2.2 Stop Timer

Description	Stops the timer and therefore the occurrence of events of this device
Required data	All
Configuration	no specific configuration

6.4 Audio Player

6.4.1 Actions

6.4.1.1 Play File

Description   Play a wave file

Required data   All
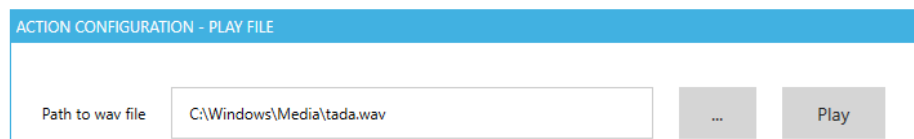
Configuration

figure 86: PMS.Cliptest.ServiceConfigurator – Eventing: Play File

- Select your *.wav file which should be played
- Play immediately and test your output using the Play - Button

6.4.1.2 Play System Sound

Description   Plays random system sounds

Required data   All

Configuration   no specific configuration

6.4.1.3 Text To Speech

Description   Speaks a provided text using the default output device (windows default audio device)

Required data   Text, Scanner Message or All (configured static text will be used)

Configuration
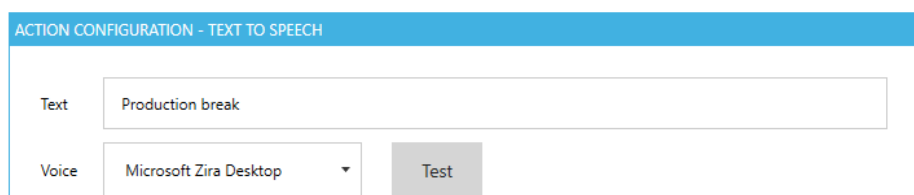
figure 87: PMS.Cliptest.ServiceConfigurator – Eventing: Text To Speech
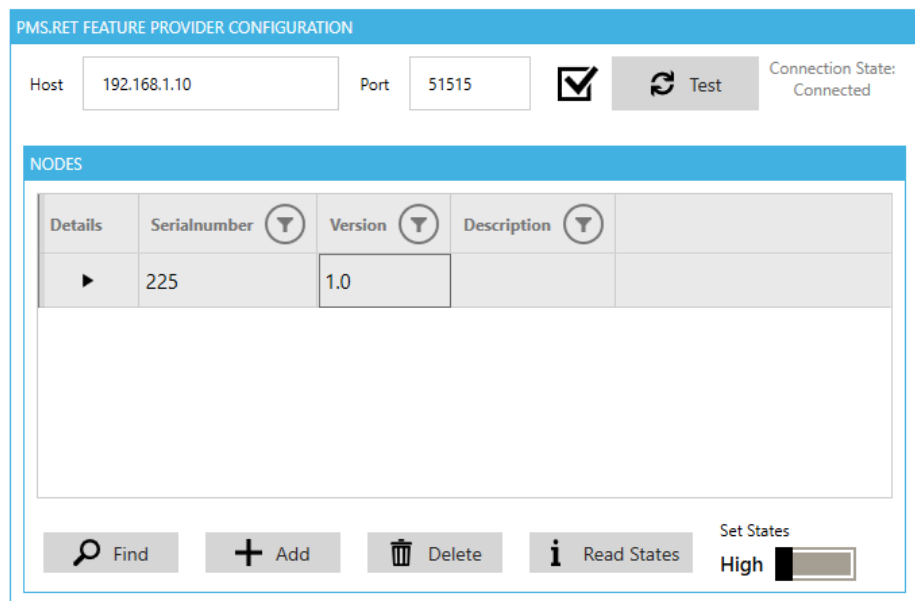
- Text will be used when configured. Leave empty if the text is provided by the corresponding event
- Select the preferred voice (needs to be installed in Windows)
- Use the Test – Button to try the output

6.5 PMS.Ret

The PMS.Ret device is to be used to connect hardware that is able to communicate over CAN bus. The implementation assumes that PMS.Cliptest is not connected to the CAN bus directly, instead an Ethernet-CAN Gateway is to be used.

The following configuration is required:

Configuration

figure 88: PMS.Cliptest.ServiceConfigurator – Eventing: PMS.Ret

- Specify the IP address and port of the Ethernet-CAN gateway. Use the Test-Button to verify the connection to the gateway.

- Use the "Find"-Button to scan the CAN bus for devices. You can also use Add and Remove buttons to specify nodes and their serial numbers. The "Read States" button can be used to read the current channel states and the firmware version of the selected node.

- The "Set States" toggle switch button is used to set all output channels to high or low signal level state.

- You can find the serial numbers on the hardware (marked yellow):

figure 89: PMS.Cliptest.ServiceConfigurator – Eventing: PMS.Ret Serialnumber

- Configure the IO-Channels of each CAN bus device by pressing the down-arrow in the first column:

figure 90: PMS.Cliptest.ServiceConfigurator – Eventing: PMS.Ret Channels

- Configure for each of the 16 IO channels if it is output or input, assign the corresponding Led number from PMS.Cliptest Masterdata and specify the Output-Signal level using the "Invert Output" option. The default is outputs are low active. The current state column shows the result of the "Read States" button.

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

6.5.1 Events

6.5.1.1 Clip Received
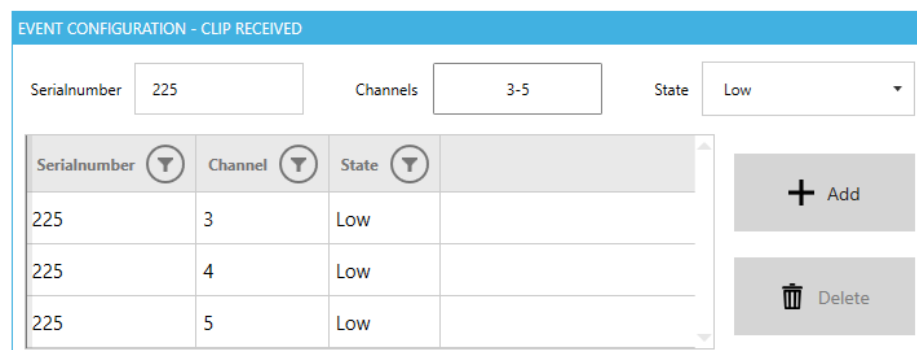
Trigger Changes of input signals will be treated as changed clips. The changed channel is being mapped to the led by PMS.Ret device configuration.

Data Type Clip Status

Configuration

figure 91: PMS.Cliptest.ServiceConfigurator – Eventing: PMS.Ret Clip Received Configuration

- Optional filter for input changes on serialnumber, specified channels and input state. Add and Delete buttons modify the filter table.

- If no filter is used (default) all input changes from all nodes will raise this event

6.5.2 Actions

6.5.2.1 Set Output
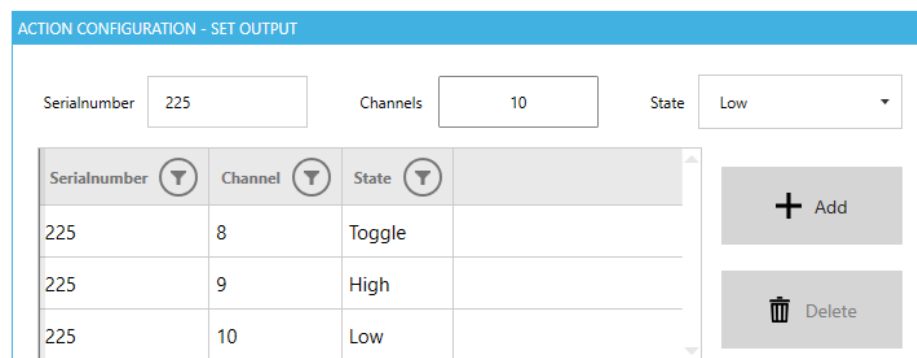
Description Change the signal level of output channels.

Required data Test, Clip Status, Text, None (configuration is required)

Configuration

figure 92: PMS.Cliptest.ServiceConfigurator – Eventing: PMS.Ret Set Output Configuration

- Optional definition of the nodes, channels and states to be set. Add and Delete buttons modify the filter table.
- If no filter is used (default) all output channels from all nodes will be changed corresponding to the connected event´s datatype

6.6 Opc Ua Client

Configuration
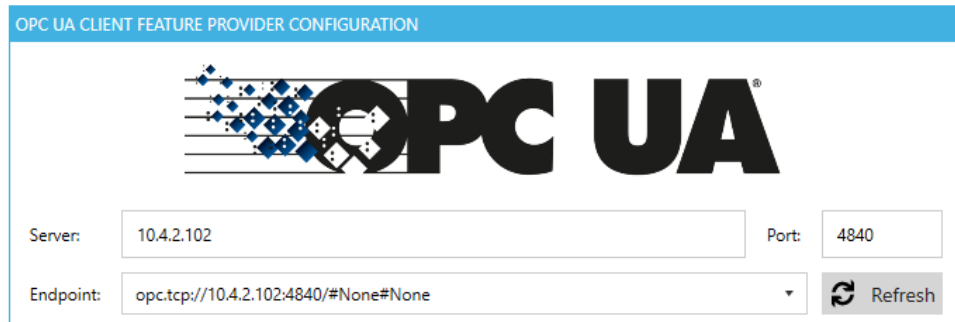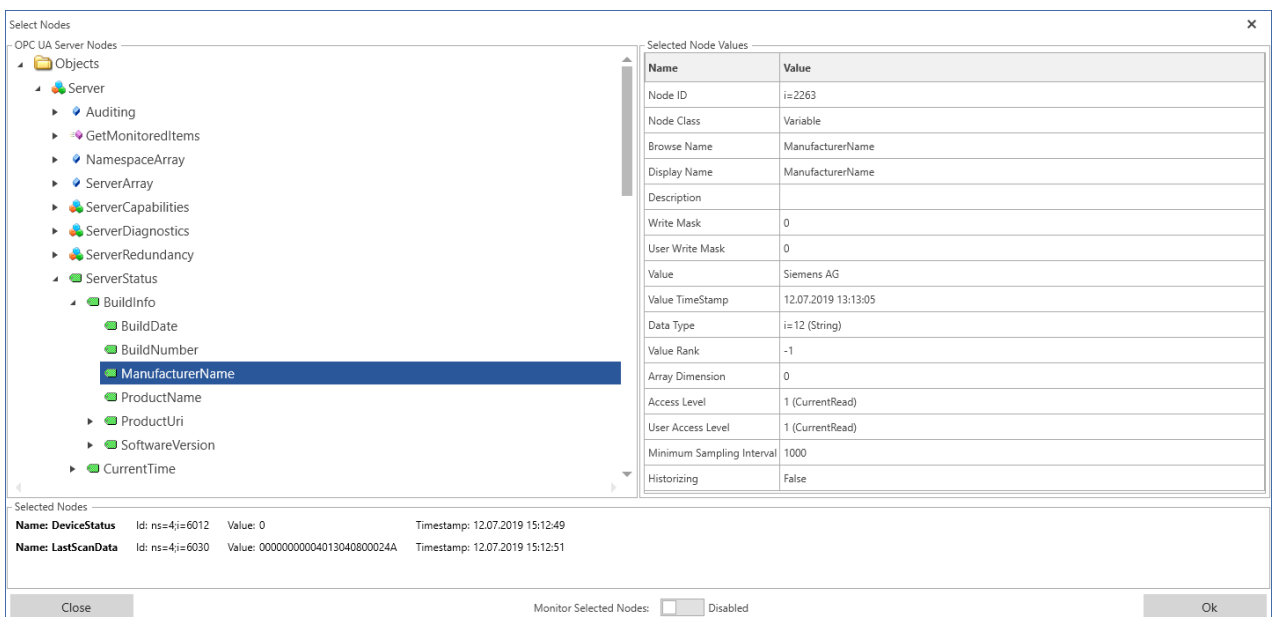
figure 93: PMS.Cliptest.ServiceConfigurator – Eventing: Opc Ua Client Configuration

- Specify the IP address and port of the Opc Ua Server
- Use the Refresh-Button to discover the endpoints of the configured server. If there are multiple entries for the same server, the entries have different authentication types and use different communication encryption. "#None#None" means no authentication and no encryption.

6.6.1 Opc Ua Server Browser

In the configuration of the actions and events it is possible to specify Opc Ua Items. To reduce the risk of invalid item paths, the item can be selected via the browse button next to the item input fields.

figure 94: PMS.Cliptest.ServiceConfigurator – Eventing: Opc Ua Client Server Browser

The top left area "OPC UA Server Nodes" shows the structure of the server as a tree view. When an item is selected, it loads its sub-elements and expands automatically if there are sub items. There are different icons for different types of nodes.

The type and other details of the selected node are visualized on the top right area "Selected Node Values". This area contains information about the selected node like the data type, the current value and the access permissions. If the selected item is writeable, you can change the value directly in the corresponding cell, which contains the current value.

The area below "Selected Nodes" holds the list of already selected nodes. To select a node drag and drop a node from the "OPC UA Server Nodes" to this area "Selected Nodes". A row for the dropped item will be added to the list with its current value. To remove items from this list just select the item and press the delete-key on the keyboard. You can also just close the browser and open again to clear all selections. If you need to select only one node then the first node (top) of this list will be returned by this browser dialog.

The switch-button below "Monitor Selected Nodes" enables a live-view of the current values of all items, which are in the selected nodes list. It might be very helpful to verify node value changes already during the configuration of this eventing device.

6.6.2 Events

6.6.2.1 Item Changed

Trigger	Occurs when a subscribed item changed.
Data Type	Text
Configuration	
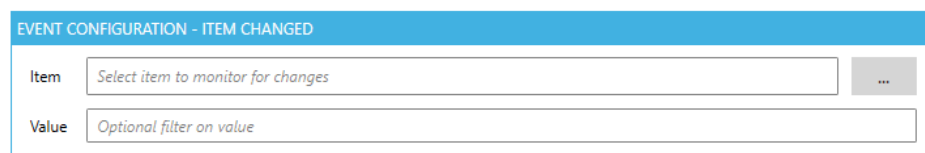
figure 95: PMS.Cliptest.ServiceConfigurator – Eventing: Opc Ua Client – Item Changed Configuration

Item: Path to the OPC UA item which should be monitored

Value: optional filter for a specific value. Use regular expressions like "([0-9]{10})\w+" for example to extract numbers with at least 10 digits. See https://regexr.com/ for demo and explanation.

6.6.2.2 Item Read

Trigger Occurs when action *Read Value* completes.

Data Type Text

Configuration
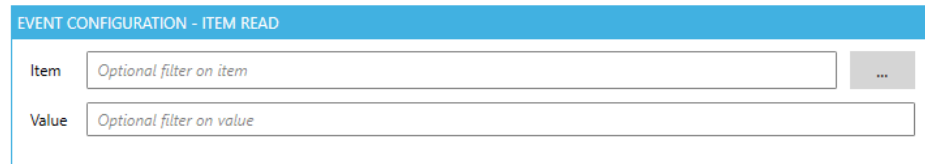
figure 96: PMS.Cliptest.ServiceConfigurator – Eventing: Opc Ua Client – Item Read Configuration

Item: optional path to the OPC UA item for which the read action was executed.

Value: optional filter for a specific value.

6.6.3 Actions

6.6.3.1 Read Value

Description Reads a value to a specified item.

Required data All (configuration will be used)

Configuration

figure 97: PMS.Cliptest.ServiceConfigurator – Eventing: Opc Ua Client – Read Value Configuration

Item: path to the OPC UA item which should be read.

6.6.3.2 Write Value

Description Writes a value to a specified item.

Required data Text, ScannerMessage, All (configuration will be used)
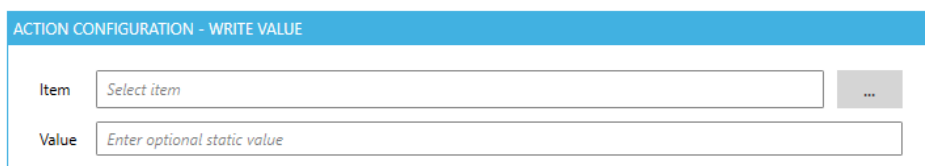
Configuration

figure 98: PMS.Cliptest.ServiceConfigurator – Eventing: Opc Ua Client – Write Value Configuration

Item: path to the OPC UA item which should be written.

Value: optional field for writing a static value if the required data is not a Text or a ScannerMessage

6.7 Brainbox

Configuration
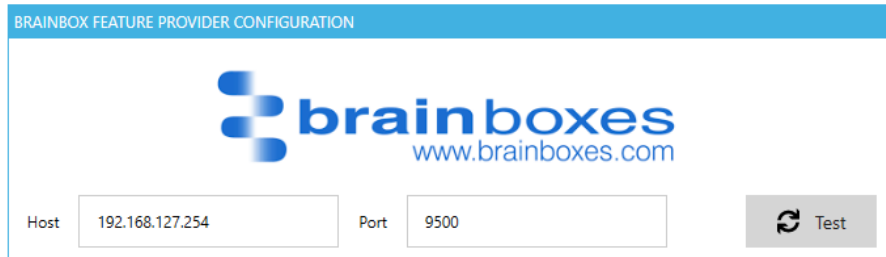
figure 99: PMS.Cliptest.ServiceConfigurator – Eventing: Brainbox Configuration

- Specify the IP address and port of the brainbox. Use the Test-Button to verify the connection to the device.

6.7.1 Events

6.7.1.1 Input Changed

Trigger   Change the signal level of the channels.

Data Type   Clip Status
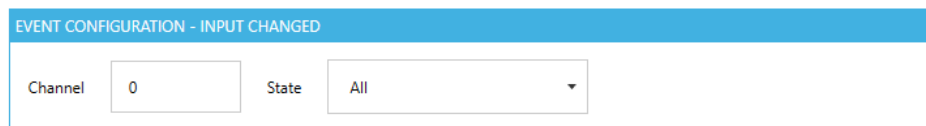
Configuration

figure 100: PMS.Cliptest.ServiceConfigurator – Eventing: Brainbox – Input Changed Configuration

Channel: number of the channel (required information)

State: Configures the condition when the event will be raised (input changed from LowToHigh, HighToLow or both)

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

6.7.2   Actions

6.7.2.1 Pulse Output

Description   Writes an impulse to the output.

Required data   All (configuration will be used)

Configuration

figure 101: PMS.Cliptest.ServiceConfigurator – Eventing: Brainbox – Pulse Output Configuration

Channel: number of the channel to which the value should be written

Level: Signal level of the impulse. (High or Low)

Duration: Duration of the impulse in milliseconds.

6.7.2.2 Set Output

Description   Change the signal level of an output channel. The level can be inverted by configuration.

Required data   All (configuration will be used)
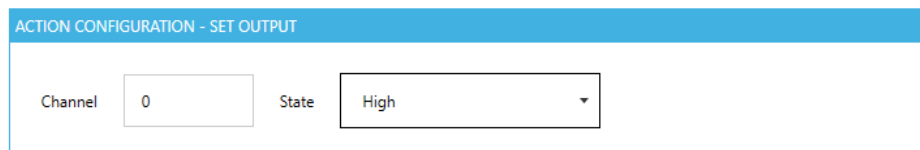
Configuration

figure 102: PMS.Cliptest.ServiceConfigurator – Eventing: Brainbox – Set Output Configuration

Channel: number of the channel to which the value should be written

State: value which should be written to the output. (High, Low or Toggle)

7   First Program Start

Before the first use of **PMS.Cliptest** you have to configure the **Service**, import the master data und configure the **ViewClient**.

7.1   Service Configuration

Before you can use **PMS.Cliptest** you have to add licenses with the ServiceConfigurator or usb dongles. For further information see 1.2 License Activation. **PMS.Cliptest** need its own local database to store the imported master data, the test data and the current clip states. Configure the connection settings and test it. If no tables exist in the selected database, the **PMS.Cliptest.Service** creates the required tables at the first start.

The radio receivers for the clip events have to be directly connected to the **Service**. **PMS.Cliptest** enables to you to connect multiple scanners to one **ViewClient**. The **ViewClient** transmit the scans to the Service and there the requested action will be performed. To assign the scan from a source **ViewClient** to the destination 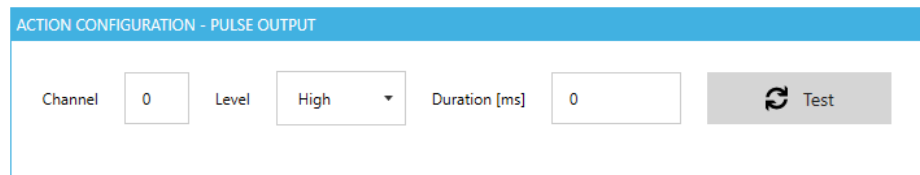**ViewClient** you can configure a scan routing between the prefix of the scanner, the source **ViewClient** and the destination **ViewClient**. If only one scanner is connected to each **ViewClient** this setting is not relevant.

The **PMS.Cliptest.Service** is responsible to handle the defined print events. Print events can be configured in the **ViewClient**. On demand the **Service** transmit the print jobs to the **PMS.PrintProcessor.Service**. You have to configure at least a minimum of one **PMS.PrintProcessor.Service** to use the printing feature. By default the communication URI to PMS.PrintProcessor is **localhost** on port **55002** where localhost is the machine on which the PMS.PrintProcessor is installed.

7.2   Master Data Import

The first step is to configure the host and port to enable the communication with the **PMS.Cliptest.Service**. After that set up the paths for the csv files. Each file type needs its dedicated folder. If you do not use the default separators and the default comment token, configure them before starting the import.

7.3 ViewClient

The each ViewClient application has a unique id to differentiate between the clients. By default for this id, the machine name will be used. If it is required to start two instances of the ViewClient application on the same computer, you have to use different client ids. You can set up the client id with a startup parameter for the ViewClient application. Create a new shortcut for the ViewClient application and open the properties. On the tab page **Shortcut** add in the **Target** field the id for the second client. You have to enter the id after the last quote separated with a whitespace.

a
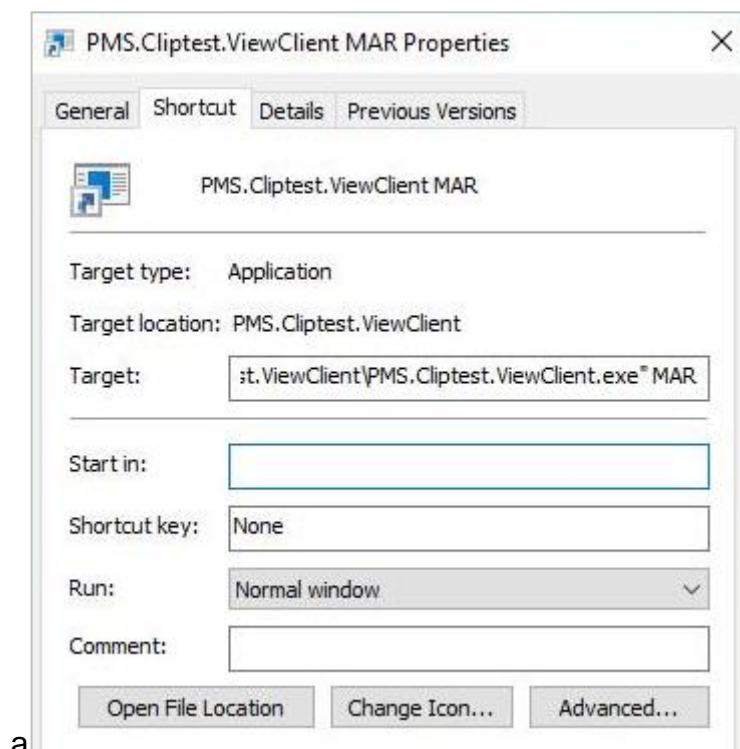
figure 103: PMS.Cliptest.ViewClient – Different Client Ids

If you use different computers for the **Service** and the **ViewClient**, it is necessary to configure the host (and port) in the **ViewClient** settings. By default, the host is **localhost**. To change the host set the machine name or IP-address where the **PMS.Cliptest.Service** is running. If you have change the port of the service application, configure here the same value. For further information, see chapter 5.1.3.1.

8   Troubleshooting

In this section are common issues with possible solutions described. In the ViewClient application, you can get further information about the service status on the Overview page in the main window. The status of the connected clients, the backend, the production server and the receivers will be displayed. You can also see the latest log information of the service.

Application	Issue	Possible solution
All applications	Save settings is not possible	Check all tab pages for errors. If there is an error on any configuration section you will be not able to save the settings.
ViewClient	ViewClient cannot connect to the service	- Check service logs of categories higher than warning - Check the firewall settings. The required communication ports are described in chapter 0 - Check that the PMS.Cliptest has a valid license
ViewClient	Test immediately finished with result NIO	- Check service logs for the message "Clip (LED=XXX) is not configured for board (BoardTitle)" - If the production server interface requests Clips which are not configured in the master data you will get these error. These Clips have the state UNCNF
ViewClient **MasterDataManager**	Demo Backend selected	Check the connection settings to the database in the ServiceConfigurator.
Printing	Multiple pages printed	Check the printing condition in the template. If there is no printing condition is configured, for each entry of the list booksmarks a single page will be generated.

9   Appendix   I   -   Printing Interface

The printing interface of **PMS.Cliptest** raises an Event on each change of internal process state (e.g. Empty test started or Clip test completed). If a corresponding event can be found in events configuration the specified label is printed on a printer via the PMS.PrintProcessor.Service. The following data fields (Key-Values) are provided with this interface and can be used in the TFORMer as bookmarks.

The printing data contains two main parts. The first part is the general data, which provides information of a test. These data items are grouped by the TableName **General**. The second part is the list data. Each group of the list data has its own table name. The available groups with their corresponding bookmarks will be described in chapter 9.2. For further information about the printing interface, use the corresponding documentation.

9.1   General Data

Bookmark	Description
TableName	Has the value **General** to identify this group.
Order	Providing the name of the order / order type proceeded with the actual test.
OrderType	Providing the type of the order / order type proceeded with the actual test.
Board	Providing the name of the assembly board on which the actual test was performed.
Steering	Providing the Steering of the order / order type proceeded with the actual test.
Workplace	Providing the internal configured name of the instance of PMS.Cliptest.
ANLIE	Providing the ANLIE of the order / order type proceeded with the actual test.
ProcessState	Providing the process state of the actual test in clear text (e.g. Finished).
TestType	Providing the type of the actual test in clear text (ClipTest or EmptyTest).
Result	Providing the result of the actual test (e.g. IO or NIO)

ID	Providing a unique ID for the test with a format of a GUID.
NOW	Providing the current date and time.
VariantComment	Providing the Comment of the selected Order variant, which was configured in the Cliplist.
VariantParameter	Providing the User defined parameters for the selected order variant, which were imported with master data or inserted from a scan.
Operator	Providing the actual logged in operator performing the actual printed test.

9.2 List Data

9.2.1 Variant Parameters

Bookmark	Description
TableName	Has the value *VariantParameters* to identify this group.
VariantParameter	The parameters of a variant are accessible via a table. The data of each row is also available as a bookmark in the general data. e.g. **{VariantParameter3}** for the third parameter.

9.2.2 Scans

Bookmark	Description
TableName	Has the value *Scans* to identify this group.
Scan	The custom scans of a test are available via this table. The data of each row is also available as a bookmark in the general data. e.g. **{Scan3}** for the third scan.

9.2.3 Critical Events

Bookmark	Description
TableName	Has the value **CriticalEvents** to identify this group.
CriticalEventMessage	Automatic description of the event
CriticalEventComment	Comment of the operator
CriticalEventIsAcknowledged	Status of the acknowledgement
CriticalEventOperator	Operator who has acknowledged the event
CriticalEventTimestamp	Timestamp of event
CriticalEventAckTimestamp	Timestamp of acknowledgement

9.2.4 Clips

Bookmark	Description
TableName	Has the value **Clips** to identify this group.
ClipLED	LED number
ClipState	Final clip state e.g. IO, NIO, ...
ClipX	X coordinate
ClipY	Y coordinate
ClipAddress	Clip address
ClipPressCount	Count of clip press events
ClipPressTime	Timestamp of the last clip press event
ClipReleaseTime	Timestamp of the last clip release event
ClipDescription	Clip description

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

10 Appendix II - Available Clip States

There are seven clipstates available in **PMS.Cliptest**. They have different meanings depending what kind of test is running

Clip Test

Clipstate	meaning
IO	This state is shown for all ***pressed*** clips which are requested in Cliplist table to be pressed for the order / order type and board.
NIO	This state is shown for all ***unpressed*** clips which are requested in Cliplist table to be pressed for the order / order type and board.
NLIO	This state is shown for all ***unpressed*** clips which are ***NOT*** requested in Cliplist table.
NLNIO	This state is shown for all ***pressed*** clips which are NOT requested in Cliplist table.
NLUKN	This state is shown for all ***unknown*** clips which are ***NOT*** requested in Cliplist table.
UNCNF	This state is shown for all clips which are requested in Cliplist table to be pressed for the order / order type and board but cannot be found in PinStatus table.
GRPIO	If all clips of a clip group are in the state IO, the clips switches to this state and will not change to NIO on any changes during the test.

Marine- und Automatisierungstechnik
Rostock GmbH

Hundsburgallee 9 c
18106 Rostock

Date:
2021-02-23

Empty Test

Clipstate	meaning
IO	This state is shown for all *unpressed* clips defined for the requested Board in PinStatus table.
NIO	This state is shown for all *pressed* clips defined for the requested Board in PinStatus table.
NLIO	This state is not relevant for an empty test.
NLNIO	This state is not relevant for an empty test.
NLUKN	This state is not relevant for an empty test.
UNCNF	This state is not relevant for an empty test.
GRPIO	If all clips of a clip group are in the state IO, the clips switches to this state and will not change to NIO on any changes during the test.

11   Appendix III – Localization

For the localization in **PMS.Cliptest** are xliff files used (version 2.0). Xliff is an XML-based standardized format for the localization process. The files are located in the directory *C:\ProgramData\MAR GmbH\PMS.Cliptest\Localization\Translations*. For each application and language is a separate file. On the application start, a file with the default language will be generated. If the application has new elements which are not in the existing files, the missing elements will be added with the default language in each language file.

11.1 Adding a new language

If a new language should be added, open the directory with the translations and create a copy of the default language file. Rename the file to the new language, update the translations and change target language during the update process. After an application restart, the new language will be available.

11.2 Xliff tools

The Language **PMS.Cliptest** files in can be edited with tools, which supports the xliff standard in version 2.0.

Freeware tool: https://poedit.net/