SmartButtons Triggers

User Guide

Version 2

Contents

Introduction

Triggers feature in SmartButtons are designed to add triggering mechanism that trigger the launch of certain buttons based on agent’s actions in Smartpoint.

Triggers are controlled by trigger rules which define:

- what agent action should launch the usage of a button
- the launch of which button should be triggered
- should the button be launched before or after completion of agent’s action (the trigger)
- in case button is triggered before agent’s action is performed: what should happen with the original agent action after the button has finished running
- in case agent clicks on 'cancel' (on the triggered button): what should happen with the original agent action

An example usage of triggers

Task: user’s email address must be placed into MT field before ticketing.

Solution with SmartButtons:

- Button MT is configured to check PNR for existing MT field, and if there is no MT field yet, alert agent to collect e-mail address of passengers. Button places the e-mail address in MT field of the PNR.
- Triggering is configured for button MT to trigger the button usage for any agent entry that starts with ‘TKP’.
- Button MT is triggered to run ‘before’ TKP entry is performed in Smartpoint.
- TKP entry is put on hold while button MT is running.
- After user enters e-mail address and clicks on ‘submit’, the button runs MT entry with e-mail address (as defined in the button setup).
- Original TKP entry (that was put on hold when triggering the button MT) is released from hold and sent to host – therefore performing the ticketing.

Triggers configuration

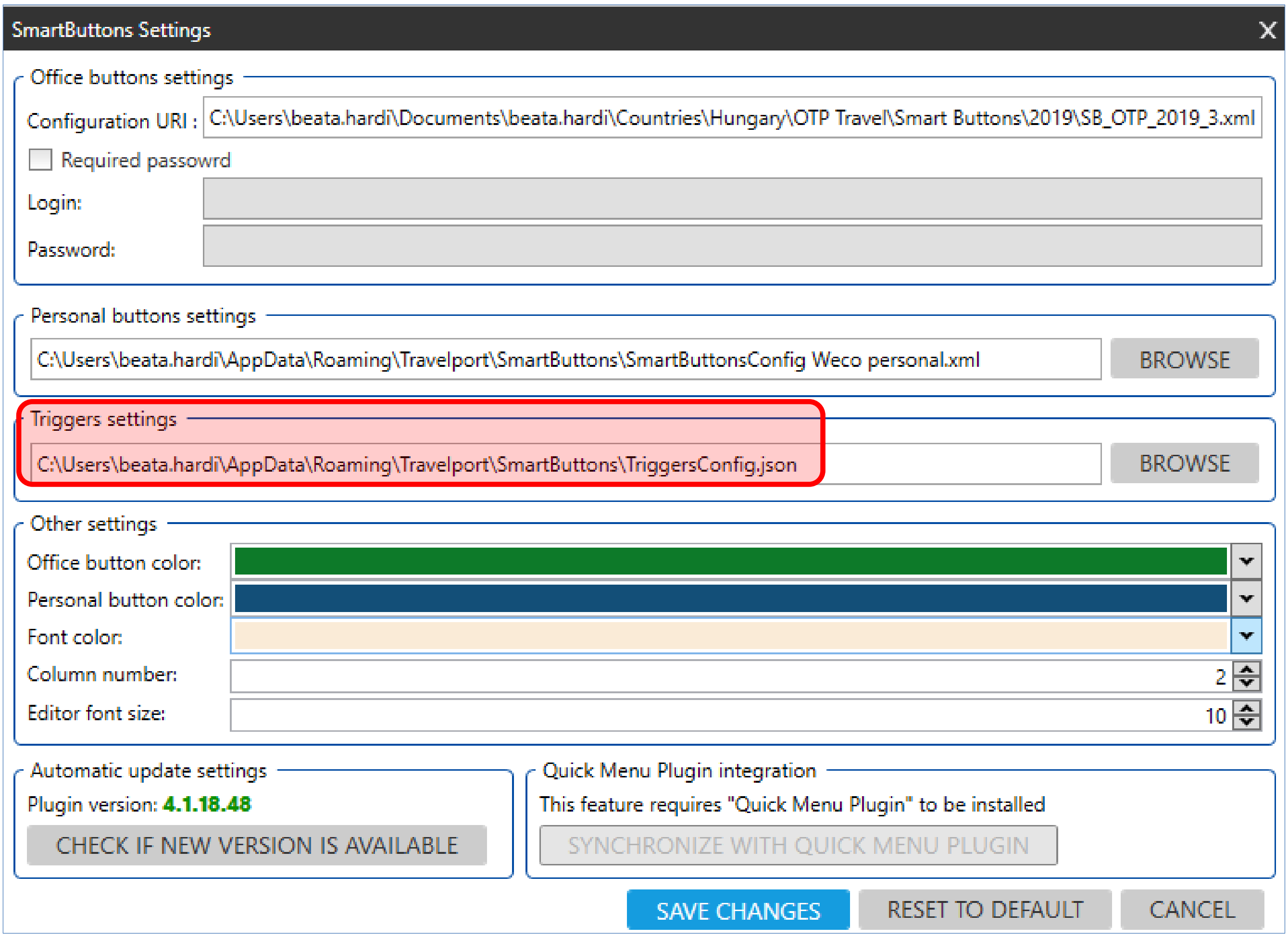
Configuration file

The triggering mechanism of buttons is controlled by a ‘Trigger settings’ configuration file.

It can be stored on each user’s PC (local drive), a common place (network drive) or a web location.

<u>WARNING!</u> If the triggers configuration file is stored on a network location, user will be able to see the triggers setup, but will NOT be able to edit it. This way, the usage of triggers is mandatory for users and they are unable to bypass it.

Configuration of file path is done in SmartButtons settings:



SmartButtons Settings

Office buttons settings

Configuration URI : C:\Users\beata.hardi\Documents\beata.hardi\Countries\Hungary\OTP Travel\Smart Buttons\2019\SB_OTP_2019_3.xml

□ Required passowrd

Login:

Password:

Personal buttons settings

C:\Users\beata.hardi\AppData\Roaming\Travelport\SmartButtons\SmartButtonsConfig Weco personal.xml

Triggers settings

C:\Users\beata.hardi\AppData\Roaming\Travelport\SmartButtons\TriggersConfig.json

Other settings

Office button color: 🟩

Personal button color: 🟦

Font color:

Column number: 2

Editor font size: 10

Automatic update settings

Plugin version: **4.1.18.48**

Quick Menu Plugin integration

This feature requires "Quick Menu Plugin" to be installed

X
BROWSE
BROWSE
CHECK IF NEW VERSION IS AVAILABLE
SYNCHRONIZE WITH QUICK MENU PLUGIN
SAVE CHANGES
RESET TO DEFAULT
CANCEL

Trigger configuration file path in SmartButtons settings

Configuration file contents

The triggers that are defined in the configuration file can be created in 2 ways:

- via SmartButtons user interface (recommended)
- by manually editing the configuration file (not recommended)

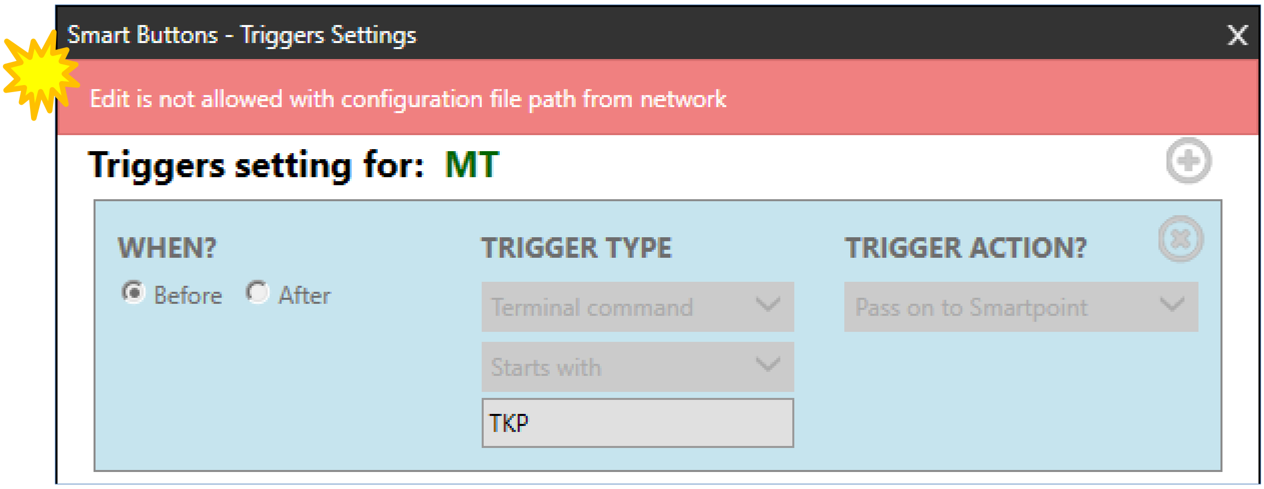
By default (at SmartButtons installation), the triggers configuration file is empty – there are no triggers automatically added to any buttons.

Recommended configuration workflow

To introduce a workflow with triggers at a travel agency, following steps are the most effective and recommended:

1. Create triggers as desired for certain buttons on 1 specific PC on the SmartButtons user interface (method described in details below).
2. As the triggers are created on the user interface, the configuration file is automatically filled with the trigger rules.
3. Test your triggers on the initial PC, make sure they work correctly.
4. When trigger creation is ready, browse your PC for the file that is defined in the Trigger settings. In our example: C:\Users\beata.hardi\AppData\Roaming\Travelport\SmartButtons\TriggersConfig.json
5. Copy the file (TriggersConfig.json) and place it on a network drive location (or web location) in the travel agency.
6. Setup SmartButtons on each travel agents' PC to read the Trigger configuration file from the network drive location where it was placed in step 5. This can be done in 2 ways:
   a. Modify the file path on each agent's PC: paste the correct file location path in SmartButtons settings / Trigger settings.
   b. Install SmartButtons in one step for the entire organization with the .msi installer file, using options to define 'triggers config file path' along with 'office buttons file path'. (method explained at the end of this document)

If Triggers configuration file is stored in a network location, the user can still access the trigger settings of any button, but they will not be able to modify it:



Smart Buttons - Triggers Settings

Edit is not allowed with configuration file path from network

Triggers setting for: MT

WHEN? ● Before ○ After

TRIGGER TYPE: Terminal command / Starts with / TKP

TRIGGER ACTION?: Pass on to Smartpoint

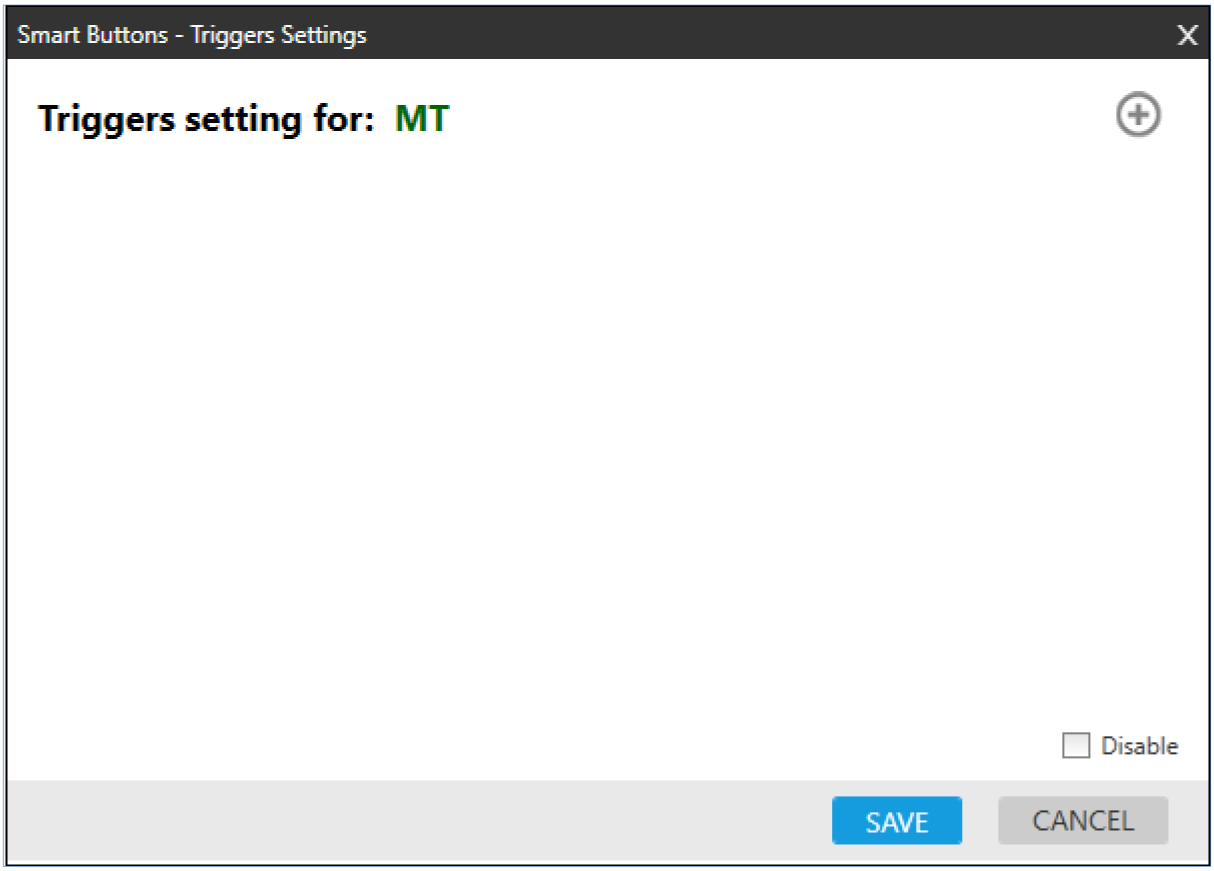
User can see trigger configuration for the button called 'MT' but unable to modify it

How to setup a trigger

You can use the SmartButtons user interface to setup triggers for any existing button.

1. **Trigger settings of each button can be accessed by right-clicking on the button and choosing the 'Triggers' option.**

   Initially, the Triggers setting for each button is empty.

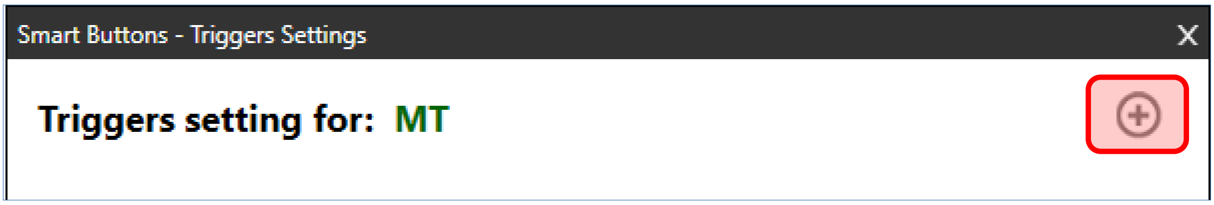


Smart Buttons - Triggers Settings

Triggers setting for: MT

Example: Triggers settings panel for the 'MT' button

2. **To add a trigger, click on the + sign in the top right corner:**

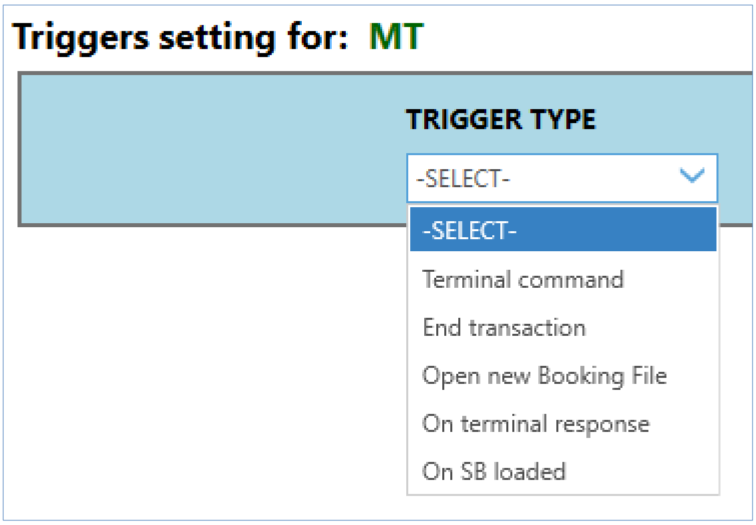


Smart Buttons - Triggers Settings

Triggers setting for: MT

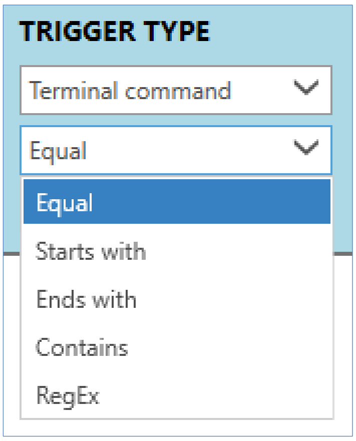
Example: Add new trigger for the 'MT' button

3. **Select a 'trigger type':**
    a. Terminal command: button should be triggered when agent sends an entry that fits the criteria that is defined in Triggers setting
    b. End transaction: button should be triggered for actions 'End transact and retrieve' (ER), 'Normal end transact' (E) and 'Send to queue' (QEP)
    c. Open new Booking File: button should be triggered whenever a new booking file is opened
    d. On terminal response: button should be triggered when terminal response fits the criteria that is defined in Triggers setting
    e. On SB loaded: button should be triggered when SmartButtons is loaded (=at Smartpoint launch)



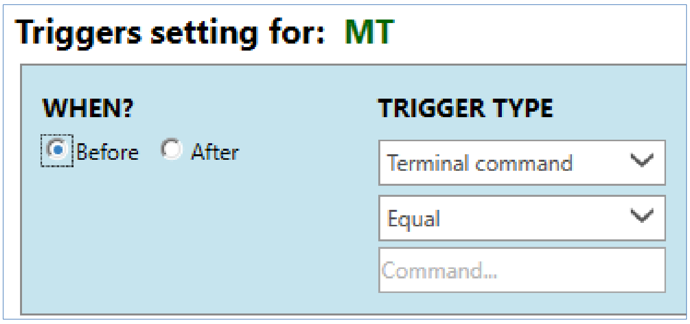
Example: Selecting trigger type

In case of 'Terminal command' and 'On terminal response' triggers, the triggering rule is looking for an entry by the user that is equal, starts with, ends with or contains text that is setup in the Triggers setting. Alternatively, user entry can be also matched with a 'regular expression'.



Example: Matching terminal command text

4. **Choose when the button should be launched:**
    a. before completing the triggering command
    b. after completing the triggering command



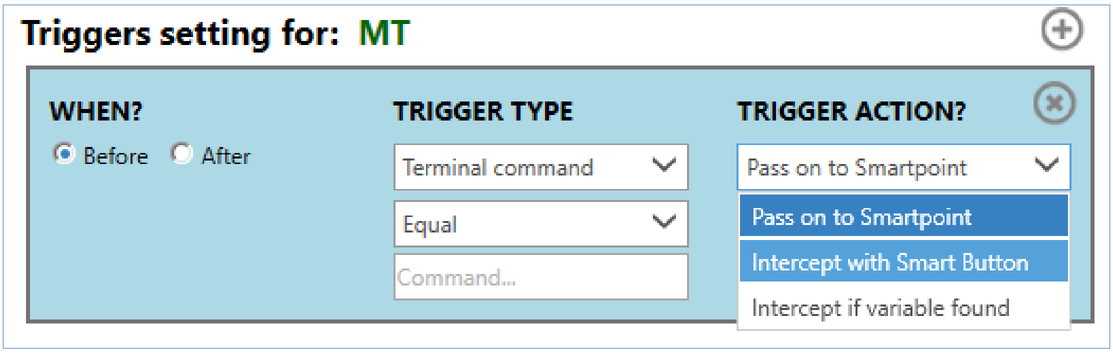
Triggers setting for: MT

WHEN? ● Before ○ After

TRIGGER TYPE: Terminal command / Equal / Command...

Example: Selecting trigger time

5. **Select trigger action:**
    a. Pass on to Smartpoint
    b. Intercept with Smart Button
    c. Intercept if variable found



Triggers setting for: MT

WHEN? ● Before ○ After

TRIGGER TYPE: Terminal command / Equal / Command...

TRIGGER ACTION?: Pass on to Smartpoint

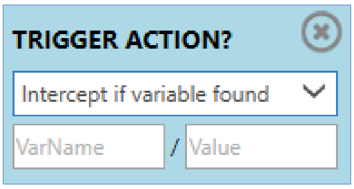
Example: Selecting trigger action

<u>Pass on to Smartpoint:</u> original triggering command is ‘put on hold’ while button is running. After button is finished, the original command is ‘released’, sent to Smartpoint (host) – and so the original command is performed.

<u>Intercept with Smart Button:</u> original triggering command is taken to launch the button, but after button is finished, it is NOT sent to Smartpoint. Original command is simply discarded after button finished running.

<u>Intercept if variable found:</u> the decision if original command should be sent to Smartpoint or not is depending on a variable in the button that is launched.

- If variable and its value from the running button matches the variable and value that is setup in the Trigger settings, original command will be intercepted (discarded).
- If variable is not found (or value does not match), original command will be sent to Smartpoint (host) – and so the original command is performed.

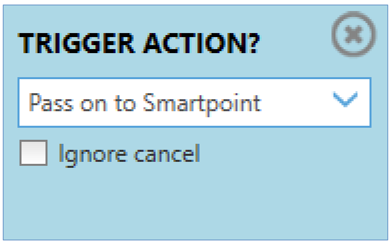


TRIGGER ACTION?: Intercept if variable found

VarName / Value

Example: Trigger action with variable

6. **Select what should happen in case user clicks on ‘cancel’ after the button is triggered:**
   - Ignore cancel on (ticked): user is allowed to click on ‘cancel’, the trigger action will be performed.
   - Ignore cancel off (un-ticked): user is NOT allowed to click on ‘cancel’ – if they do, the trigger action will not be performed.



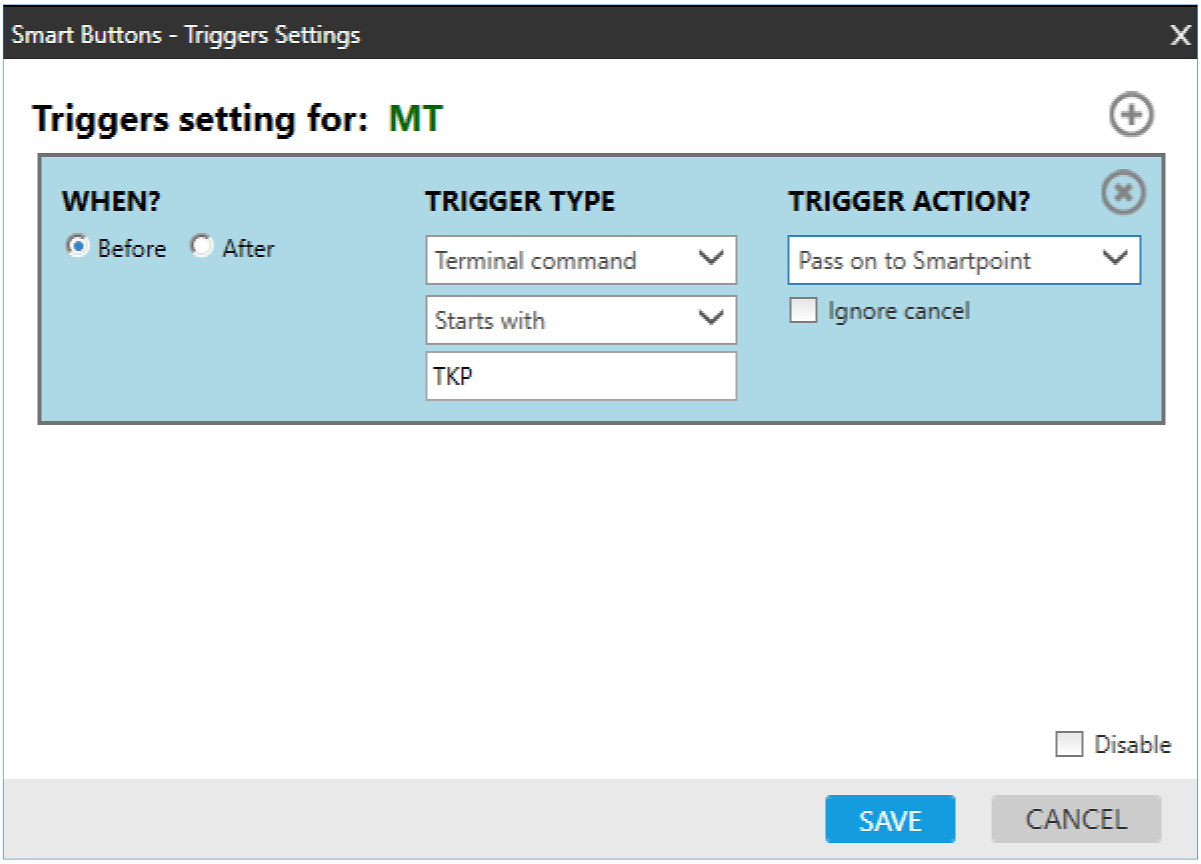
TRIGGER ACTION?

Pass on to Smartpoint

□ Ignore cancel

Example: Setting up ‘cancel’ action

7. **Save the settings with clicking at ‘SAVE’ in the bottom right corner:**



Smart Buttons - Triggers Settings

Triggers setting for: MT

WHEN?

● Before ○ After

TRIGGER TYPE

Terminal command

Starts with

TKP

TRIGGER ACTION?

Pass on to Smartpoint

□ Ignore cancel

□ Disable

SAVE CANCEL

Example triggers settings for button MT

The above example triggers the usage of button MT when agent enters any entry that starts with TKP. Button is launched before the TKP action is performed. After button has finished running, the original entry is sent to Smartpoint (host), so the original action is performed.

When the MT button is triggered and user clicks on ‘cancel’, TKP is action is not performed.

Other options

With the ‘disable’ option it is possible to disable a trigger temporarily, without having to build it again later when it’s required again.

Buttons with pure Snippet content

In case you have some buttons which contain only Snippets (C# scripts) and display a pop-up dialog to the user, please pay extra attention to the ‘Ignore cancel’ option.

Witch such a button, if you have ‘Ignore cancel’ OFF, you might need to modify your Snippet.

The general rule is: return "FALSE" (boolean) in your snipped code if you wish to BREAK triggers chain.
In case of snippets that displays dialog(s) return "FALSE" (if required) should be done after last dialog closed.

With this completed, if user clicks on ‘cancel’ on the triggered button, the original trigger action will not be sent to host. This way, you can make it truly mandatory to fulfill the requirements of the triggered button.