2024 Feature Overview

Typhoon Test Hub 1.0.0

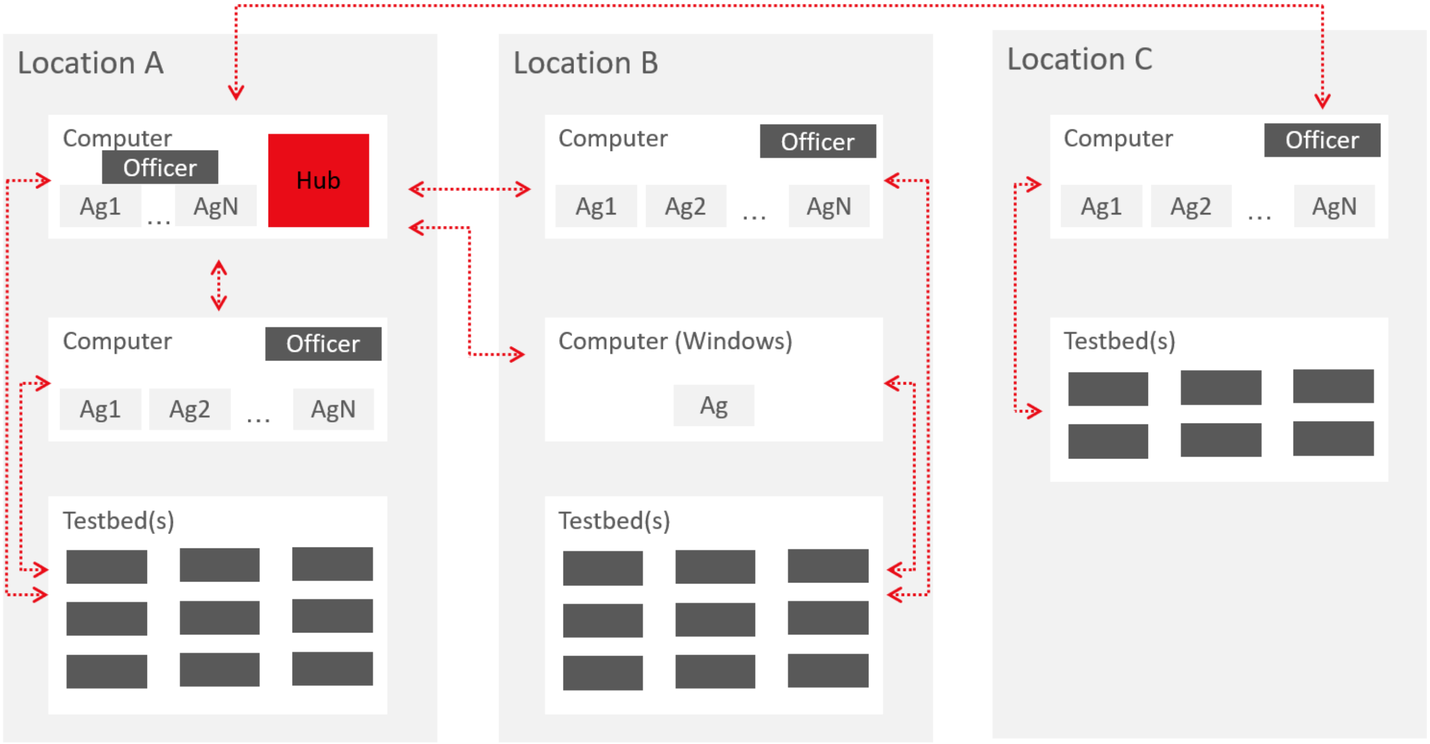
Contents

Typhoon Test Hub

Typhoon Test Hub (TTH) orchestrates test execution and organizes the test results. It can be deployed on-premises or in the cloud and manage distributed computers to run tests. It is first and foremost a continuous integration and testing tool, built on the principles of ease-of-use, visibility, scalability, and reproducibility.

TTH has a web interface, making it easily accessible. It is designed for convenience, removing the complexity of integrating Controller Hardware-in-the-Loop (C-HIL) into your automated testing process. It also allows you to quickly extract relevant information from a large amount of test data. Using the web user interface (UI), it is possible to easily configure which tests should be executed when and where.

When it comes to output, TTH generates overviews for quick result checks and has the ability to present detailed test results for debugging. All test results are collected and available at the same location, simplifying the process of sharing test results. You can create any number of users, free of charge, and make the results available and useful to everyone in your company – or even to customers, suppliers, and partners. From a single place (here depicted in red as “Hub”), you can manage, trigger, and collect data from several locations.



Typhoon Test Hub can be installed on a Computer on-premises, or in the cloud. It runs on Linux (Debian). The Computer can be provided by Typhoon HIL (recommended) or sourced by you. When provided by Typhoon HIL, the Computer will come pre-configured with the optimal hardware configuration. It is highly recommended the Computer is used solely for Test Automation in Linux OS, where the Hub, Agents, and additional applications are run in docker containers. This means your tests must be “Linux compatible”, including applications that interface with the device under test (DUT) – such as those that update device firmware, parameters, and communicate with it. If that is not possible, Agents can also run Typhoon Test Hub on a Windows Computer (one Agent per Computer), with limited capabilities and unavailable features.

TTH can run a wide range of tests for different products and applications – not only for HIL testing, but also for software only tests or running/collecting results from manual or laboratory tests. All the results will be available in the same location. You can narrow down the search results in the Hub, so it is easy to find the specific type of test you are looking for.

Definitions

Here are the explanations for some of the components and nomenclature used in this document:

- Hub: Software which orchestrates test execution and collects test results. It has a web interface for ease of access.

Agent: “Component” that executes the Job. When running with Linux, a single Computer can have several Agents running independently and in parallel.

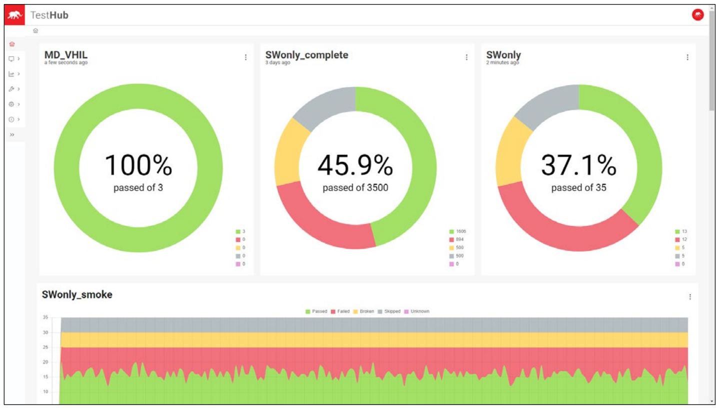
- Computer: Refers to the Linux computer that is running the Hub + Agents or just Agents.
- Officer: Application running on the Computer which allows the Hub to control and monitor it, including controlling the Agents.
- Job: Describes what should be executed, which resources to use, where to connect to.
- Setup: Combination of HILs and DUTs which will be used to run a Job.
- Execution: Once a Job is started, it generates one Execution.
- Report: When the Execution runs Typhoon Test, it generates an Allure Report.

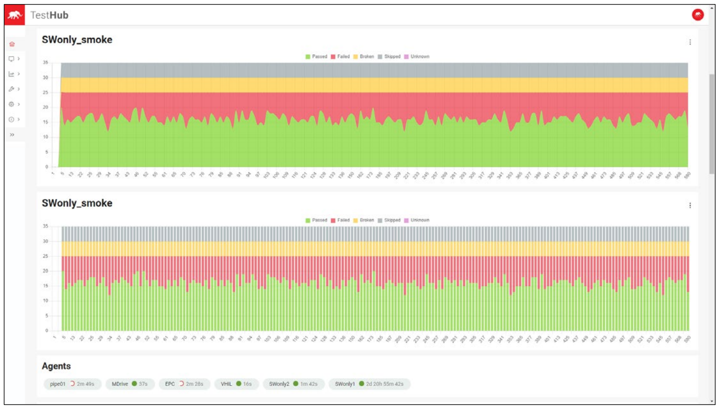
Visualization

One of the main features of Typhoon Test Hub is improved visibility of both test results and system performance. You can have an overall performance view at a glance or dive into details. Executions carry important information about how a test was executed and what was used, so you can reproduce tests and have traceability.

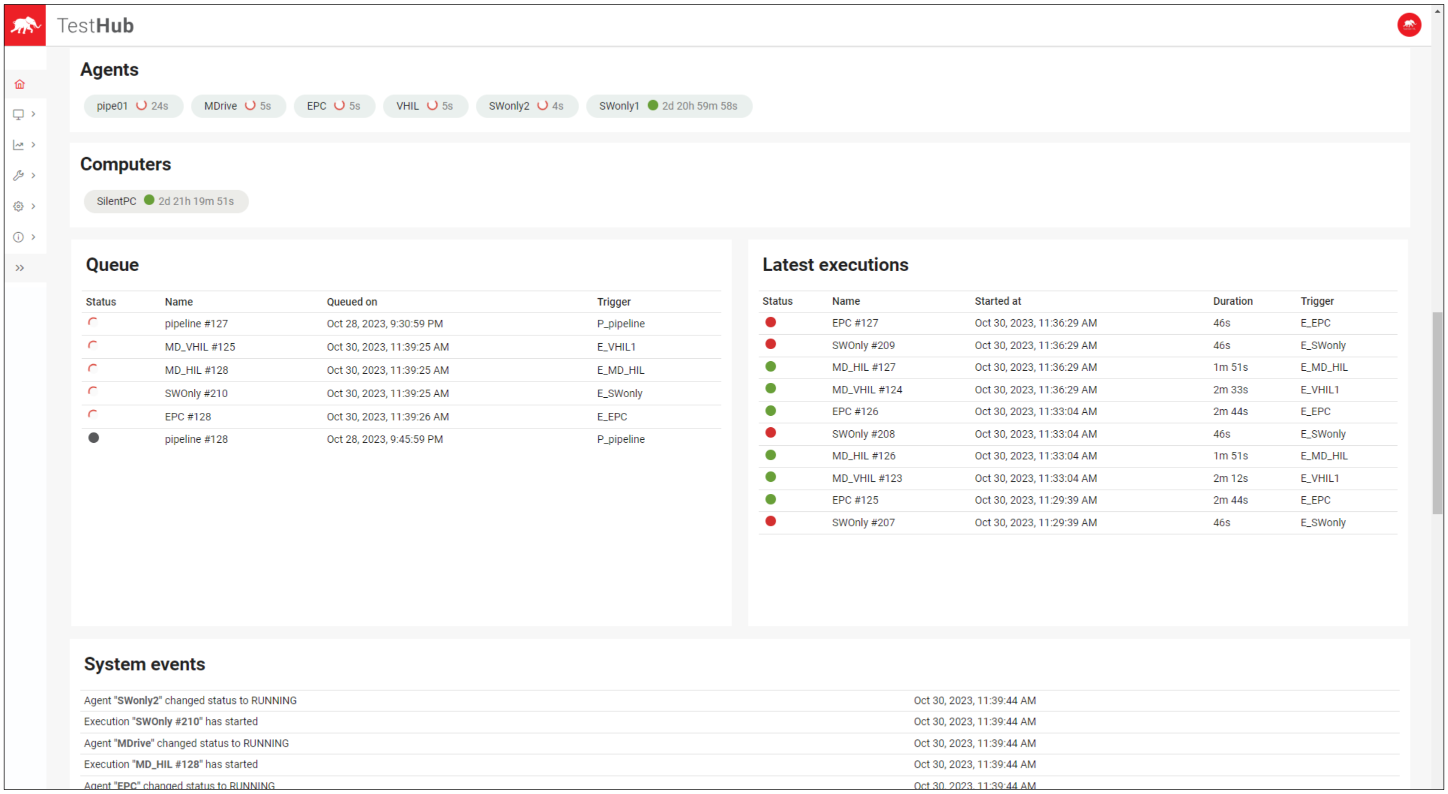
Overview

The overview page is setup for a quick overview of the whole system. Most of the items on this page are linked and will direct you to another page where more details about the item can be located. At the very top, you can see the latest results for a group of tests (3 donut graphs) and how long ago they were updated, as well as a trend (in the line or bar graph) of test results. You can easily select a different group of tests from among those displayed in each graph.





Further down, there is a list of all Agents, informing which are currently running, online, or offline. Similarly, there is a list of all the Computers currently connected to the Hub, their status, and for how long they are in that state. The Queue table shows executions which are currnetly running or queued to run. The Latest Executions table shows executions that have already completed.



TestHub

Agents

pipe01 24s | MDrive 5s | EPC 5s | VHIL 5s | SWonly2 4s | SWonly1 ● 2d 20h 59m 58s

Computers

SilentPC ● 2d 21h 19m 51s

Queue

Status	Name	Queued on	Trigger
◌	pipeline #127	Oct 28, 2023, 9:30:59 PM	P_pipeline
◌	MD_VHIL #125	Oct 30, 2023, 11:39:25 AM	E_VHIL1
◌	MD_HIL #128	Oct 30, 2023, 11:39:25 AM	E_MD_HIL
◌	SWOnly #210	Oct 30, 2023, 11:39:25 AM	E_SWonly
◌	EPC #128	Oct 30, 2023, 11:39:26 AM	E_EPC
●	pipeline #128	Oct 28, 2023, 9:45:59 PM	P_pipeline

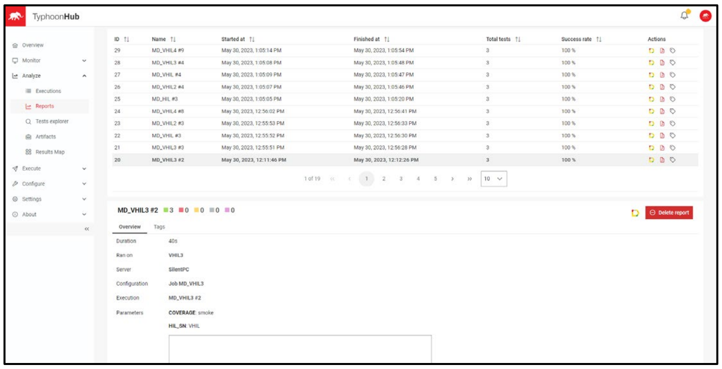
Latest executions

Status	Name	Started at	Duration	Trigger
●	EPC #127	Oct 30, 2023, 11:36:29 AM	46s	E_EPC
●	SWOnly #209	Oct 30, 2023, 11:36:29 AM	46s	E_SWonly
●	MD_HIL #127	Oct 30, 2023, 11:36:29 AM	1m 51s	E_MD_HIL
●	MD_VHIL #124	Oct 30, 2023, 11:36:29 AM	2m 33s	E_VHIL1
●	EPC #126	Oct 30, 2023, 11:33:04 AM	2m 44s	E_EPC
●	SWOnly #208	Oct 30, 2023, 11:33:04 AM	46s	E_SWonly
●	MD_HIL #126	Oct 30, 2023, 11:33:04 AM	1m 51s	E_MD_HIL
●	MD_VHIL #123	Oct 30, 2023, 11:33:04 AM	2m 12s	E_VHIL1
●	EPC #125	Oct 30, 2023, 11:29:39 AM	2m 44s	E_EPC
●	SWOnly #207	Oct 30, 2023, 11:29:39 AM	46s	E_SWonly

System events

Agent "**SWonly2**" changed status to RUNNING	Oct 30, 2023, 11:39:44 AM
Execution "**SWOnly #210**" has started	Oct 30, 2023, 11:39:44 AM
Agent "**MDrive**" changed status to RUNNING	Oct 30, 2023, 11:39:44 AM
Execution "**MD_HIL #128**" has started	Oct 30, 2023, 11:39:44 AM
Agent "**EPC**" changed status to RUNNING	Oct 30, 2023, 11:39:44 AM

As mentioned before, this is an interactive Overview. Clicking on items on the Overview page will take you to a corresponding screen where more details can be found. For example, if you click on a graph, it will take you to a report page specific to that result. From the report page, you can see a list of all reports in addition to more details per each selected report:



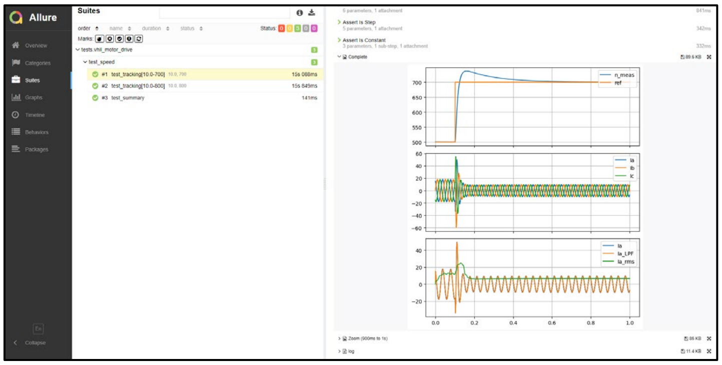
ID	Name	Started at	Finished at	Total tests	Success rate	Actions
29	MD_VHIL4 #9	May 30, 2023, 1:05:14 PM	May 30, 2023, 1:05:54 PM	3	100 %	
28	MD_VHIL3 #4	May 30, 2023, 1:05:08 PM	May 30, 2023, 1:05:48 PM	3	100 %	
27	MD_VHIL #4	May 30, 2023, 1:05:09 PM	May 30, 2023, 1:05:47 PM	3	100 %	
26	MD_VHIL2 #4	May 30, 2023, 1:05:07 PM	May 30, 2023, 1:05:46 PM	3	100 %	
25	MD_HIL #3	May 30, 2023, 1:05:05 PM	May 30, 2023, 1:05:20 PM	3	100 %	
24	MD_VHIL4 #8	May 30, 2023, 12:56:02 PM	May 30, 2023, 12:56:41 PM	3	100 %	
23	MD_VHIL2 #3	May 30, 2023, 12:55:53 PM	May 30, 2023, 12:56:33 PM	3	100 %	
22	MD_VHIL #3	May 30, 2023, 12:55:52 PM	May 30, 2023, 12:56:30 PM	3	100 %	
21	MD_VHIL3 #3	May 30, 2023, 12:55:51 PM	May 30, 2023, 12:56:28 PM	3	100 %	
20	MD_VHIL3 #2	May 30, 2023, 12:11:46 PM	May 30, 2023, 12:12:26 PM	3	100 %	

1 of 19 « ‹ 1 2 3 4 5 › » 10

MD_VHIL3 #2 🟩3 🟥0 🟨0 ⬜0 🟪0 Delete report

Overview Tags

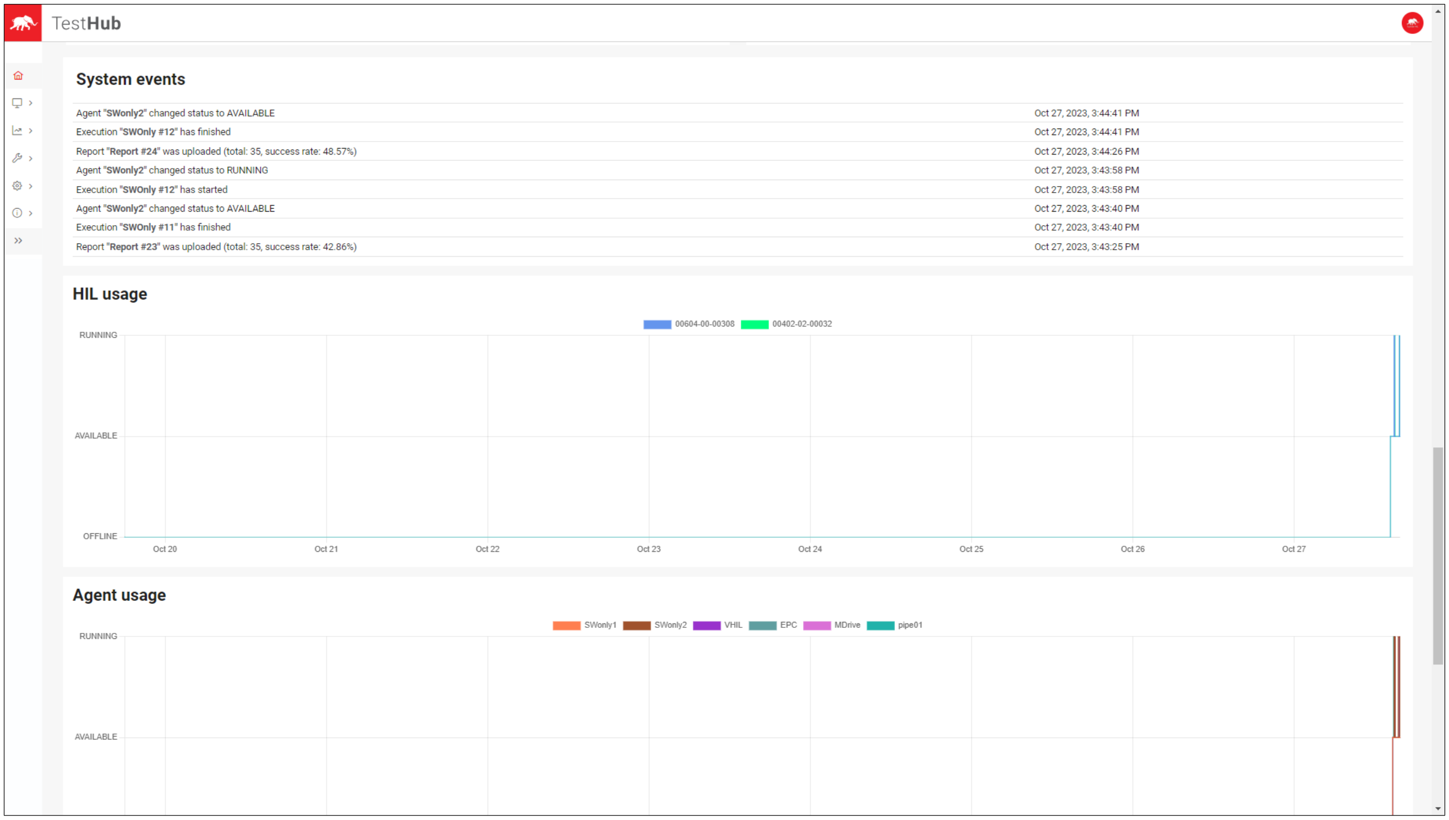
Duration 40s
Ran on VHIL3
Server SilentPC
Configuration Job MD_VHIL3
Execution MD_VHIL3 #2
Parameters **COVERAGE**: smoke
HIL_SN: VHIL



tests.vhil_motor_drive 3
test_speed 3
#1 test_tracking[10.0-700] 10.0, 700 15s 088ms
#2 test_tracking[10.0-800] 10.0, 800 15s 849ms
#3 test_summary 141ms

6 parameters, 1 attachment 841ms
Assert Is Step
5 parameters, 1 attachment 342ms
Assert Is Constant
3 parameters, 1 sub-step, 1 attachment 332ms
Complete 89.6 KB
Zoom (900ms to 1s) 86 KB
log 11.4 KB

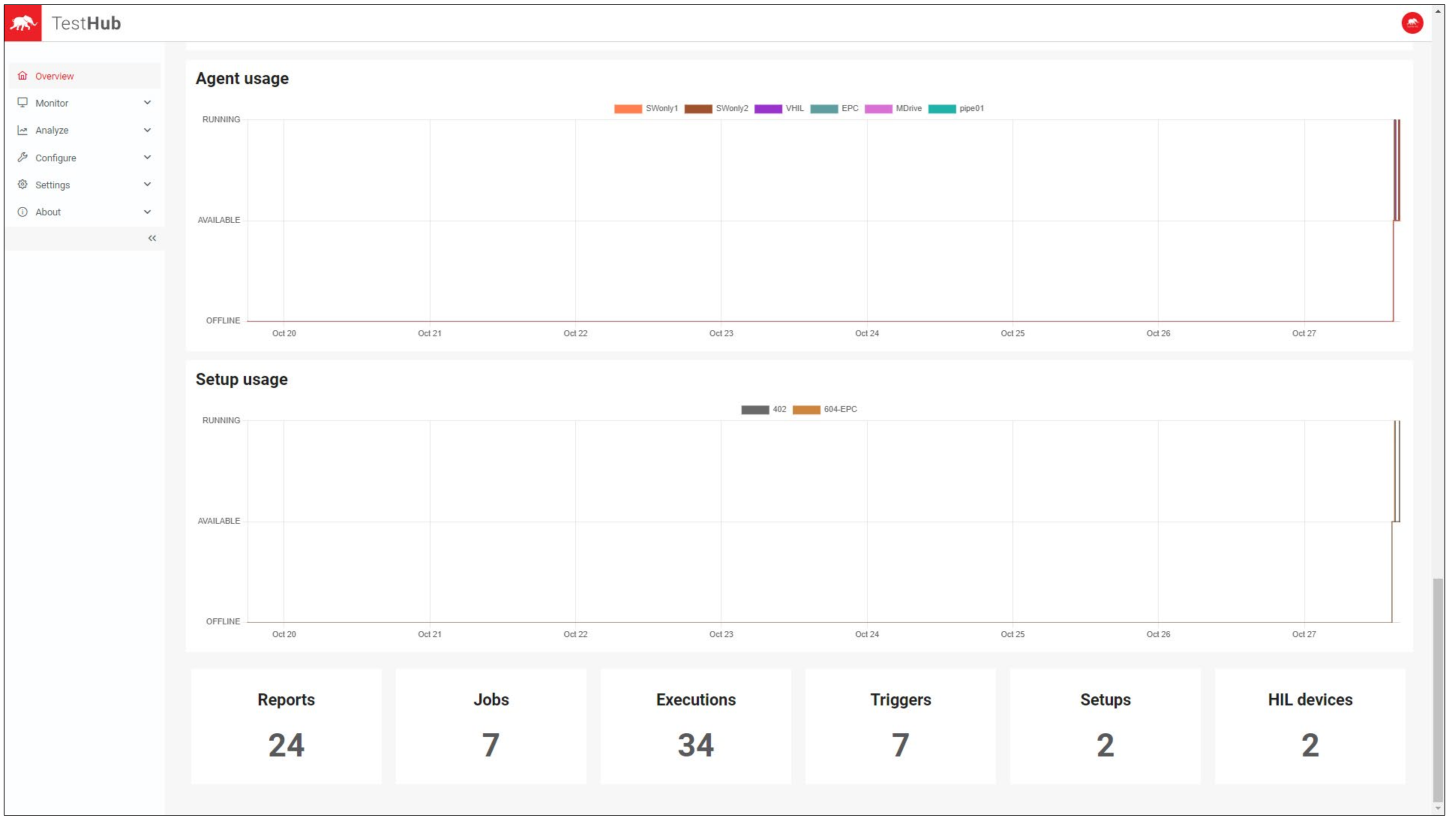
System events and configuration changes are stored so you can trace back activities and actions in your system. You can also visualize your system utilization in the time trace, which shows if HILs, Agents, and Setups are offline, available, or running.



System events

Agent "**SWonly2**" changed status to AVAILABLE Oct 27, 2023, 3:44:41 PM
Execution "**SWOnly #12**" has finished Oct 27, 2023, 3:44:41 PM
Report "**Report #24**" was uploaded (total: 35, success rate: 48.57%) Oct 27, 2023, 3:44:26 PM
Agent "**SWonly2**" changed status to RUNNING Oct 27, 2023, 3:43:58 PM
Execution "**SWOnly #12**" has started Oct 27, 2023, 3:43:58 PM
Agent "**SWonly2**" changed status to AVAILABLE Oct 27, 2023, 3:43:40 PM
Execution "**SWOnly #11**" has finished Oct 27, 2023, 3:43:40 PM
Report "**Report #23**" was uploaded (total: 35, success rate: 42.86%) Oct 27, 2023, 3:43:25 PM

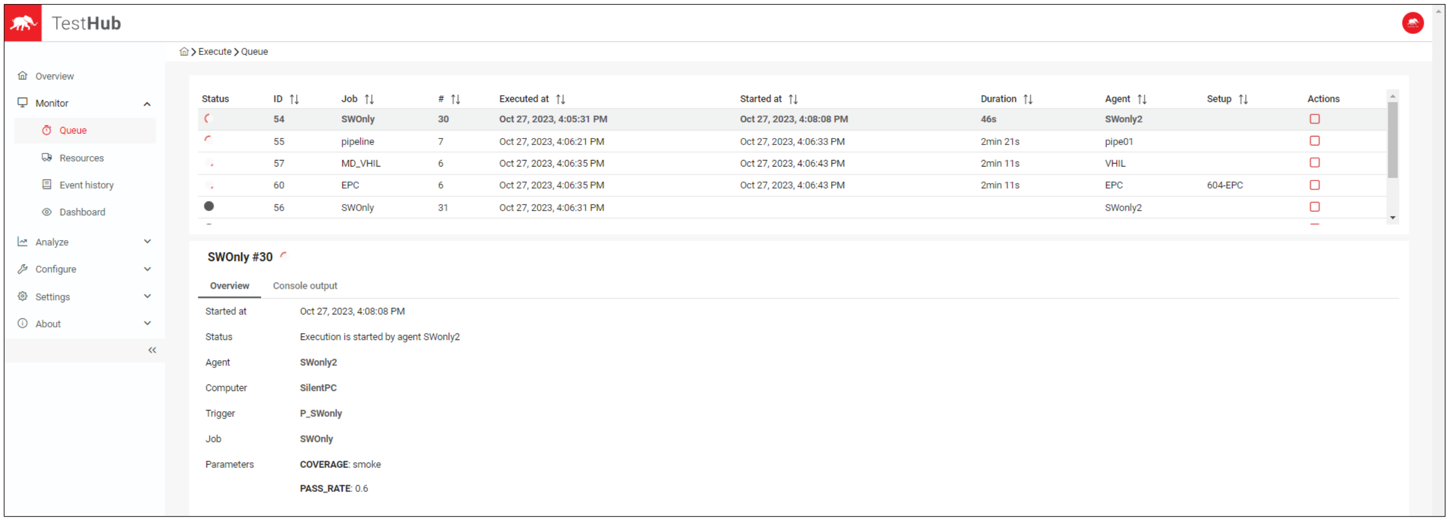
Finally, at the very bottom of the Overview, you can see some cards with summary information about your system and executions.



Reports	Jobs	Executions	Triggers	Setups	HIL devices
24	7	34	7	2	2

Queue

Queue provides details on which Executions are currently running and which ones are queued to be executed.



> Execute > Queue

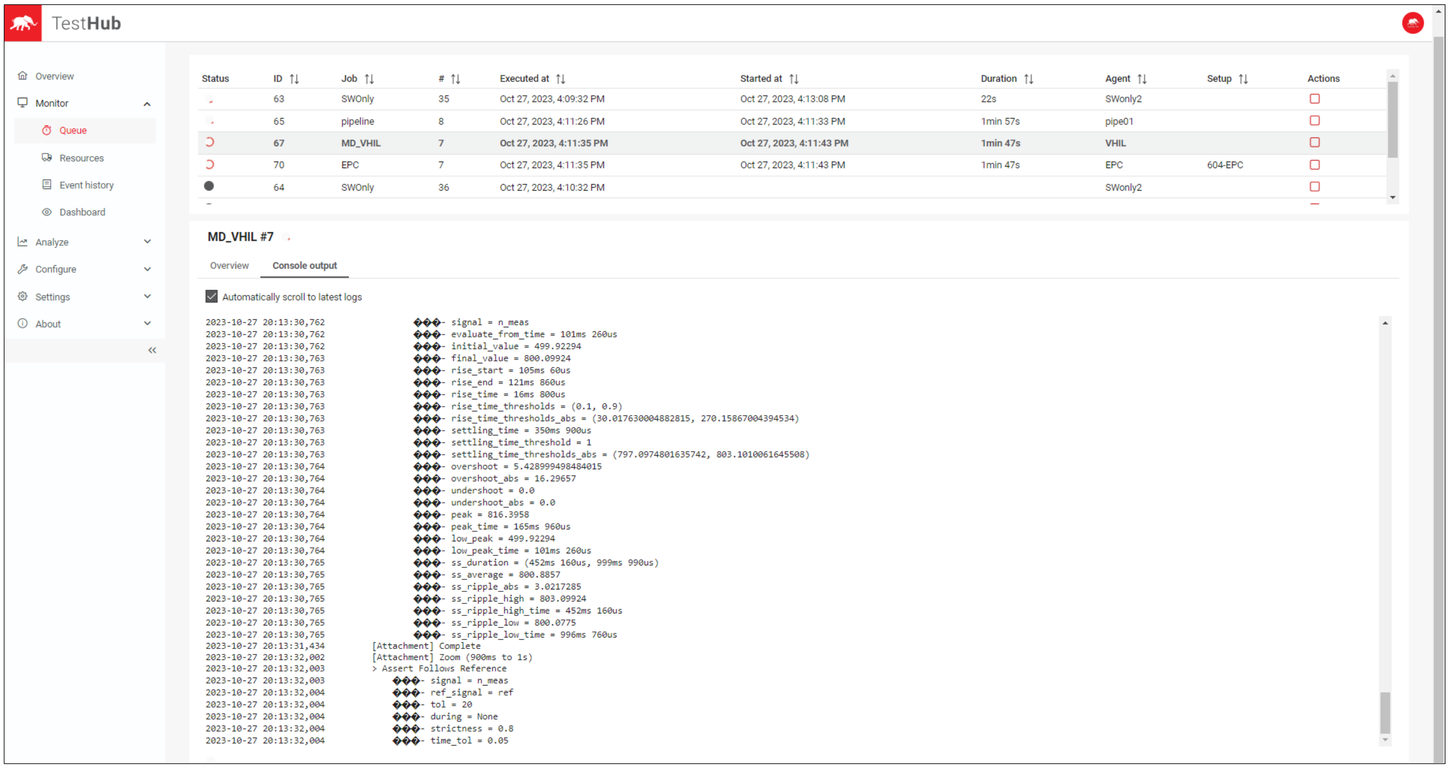
Status	ID	Job	#	Executed at	Started at	Duration	Agent	Setup	Actions
	54	SWOnly	30	Oct 27, 2023, 4:05:31 PM	Oct 27, 2023, 4:08:08 PM	46s	SWonly2		□
	55	pipeline	7	Oct 27, 2023, 4:06:21 PM	Oct 27, 2023, 4:06:33 PM	2min 21s	pipe01		□
	57	MD_VHIL	6	Oct 27, 2023, 4:06:35 PM	Oct 27, 2023, 4:06:43 PM	2min 11s	VHIL		□
	60	EPC	6	Oct 27, 2023, 4:06:35 PM	Oct 27, 2023, 4:06:43 PM	2min 11s	EPC	604-EPC	□
●	56	SWOnly	31	Oct 27, 2023, 4:06:31 PM			SWonly2		□

SWOnly #30

Overview | Console output

Started at	Oct 27, 2023, 4:08:08 PM
Status	Execution is started by agent SWonly2
Agent	SWonly2
Computer	SilentPC
Trigger	P_SWonly
Job	SWOnly
Parameters	COVERAGE: smoke PASS_RATE: 0.6

In both cases, the Overview tab provides information and links to items connected to that execution: displaying on which Agent/Computer the test is running, which Job is executing, and its parametrization. If an item is queued, it will inform which resource it is waiting for. Typhoon Test Hub is designed to optimize Setup utilization, so if two executions need different Setups, they can run in parallel. Once a test execution starts, you can track logs in real time by looking at the Console output:



Status	ID	Job	#	Executed at	Started at	Duration	Agent	Setup	Actions
	63	SWOnly	35	Oct 27, 2023, 4:09:32 PM	Oct 27, 2023, 4:13:08 PM	22s	SWonly2		□
	65	pipeline	8	Oct 27, 2023, 4:11:26 PM	Oct 27, 2023, 4:11:33 PM	1min 57s	pipe01		□
	67	MD_VHIL	7	Oct 27, 2023, 4:11:35 PM	Oct 27, 2023, 4:11:43 PM	1min 47s	VHIL		□
	70	EPC	7	Oct 27, 2023, 4:11:35 PM	Oct 27, 2023, 4:11:43 PM	1min 47s	EPC	604-EPC	□
●	64	SWOnly	36	Oct 27, 2023, 4:10:32 PM			SWonly2		□

MD_VHIL #7

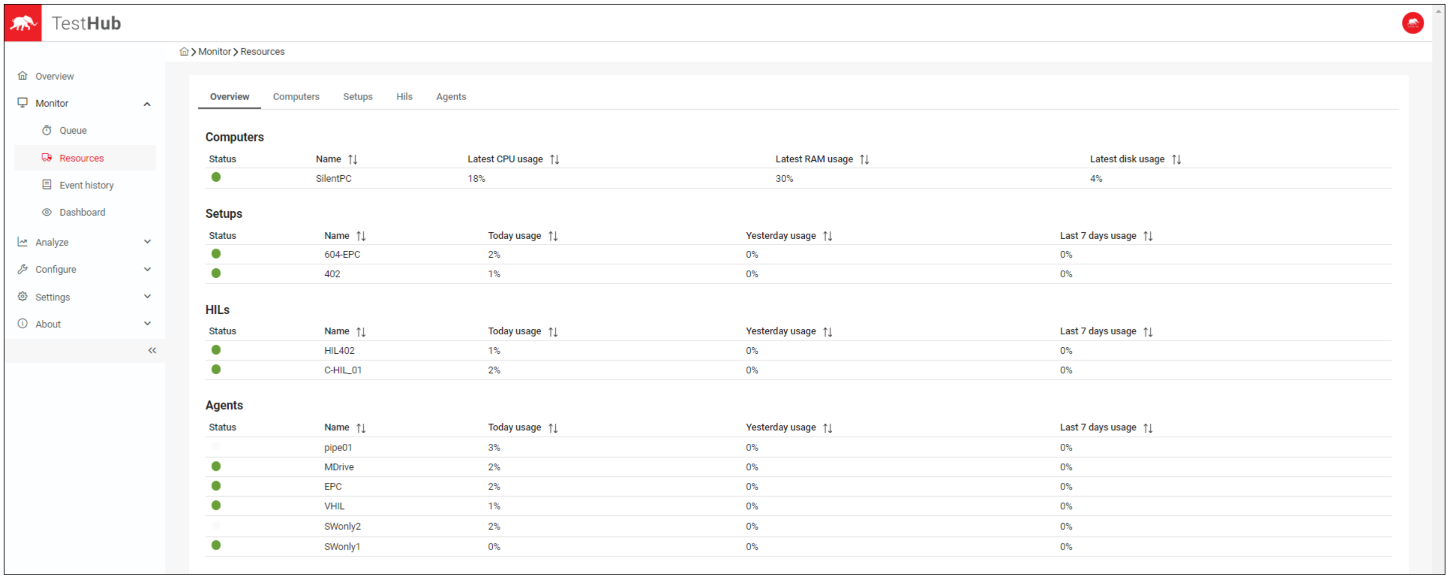
Overview | Console output

☑ Automatically scroll to latest logs

```
2023-10-27 20:13:30,762              ���- signal = n_meas
2023-10-27 20:13:30,762              ���- evaluate_from_time = 101ms 260us
2023-10-27 20:13:30,762              ���- initial_value = 499.92294
2023-10-27 20:13:30,763              ���- final_value = 800.09924
2023-10-27 20:13:30,763              ���- rise_start = 105ms 60us
2023-10-27 20:13:30,763              ���- rise_end = 121ms 860us
2023-10-27 20:13:30,763              ���- rise_time = 16ms 800us
2023-10-27 20:13:30,763              ���- rise_time_thresholds = (0.1, 0.9)
2023-10-27 20:13:30,763              ���- rise_time_thresholds_abs = (30.017630004882815, 270.15867004394534)
2023-10-27 20:13:30,763              ���- settling_time = 350ms 900us
2023-10-27 20:13:30,763              ���- settling_time_threshold = 1
2023-10-27 20:13:30,763              ���- settling_time_thresholds_abs = (797.0974801635742, 803.1010061645508)
2023-10-27 20:13:30,764              ���- overshoot = 5.428999498484015
2023-10-27 20:13:30,764              ���- overshoot_abs = 16.29657
2023-10-27 20:13:30,764              ���- undershoot = 0.0
2023-10-27 20:13:30,764              ���- undershoot_abs = 0.0
2023-10-27 20:13:30,764              ���- peak = 816.3958
2023-10-27 20:13:30,764              ���- peak_time = 165ms 960us
2023-10-27 20:13:30,764              ���- low_peak = 499.92294
2023-10-27 20:13:30,764              ���- low_peak_time = 101ms 260us
2023-10-27 20:13:30,765              ���- ss_duration = (452ms 160us, 999ms 990us)
2023-10-27 20:13:30,765              ���- ss_average = 800.8857
2023-10-27 20:13:30,765              ���- ss_ripple_abs = 3.0217285
2023-10-27 20:13:30,765              ���- ss_ripple_high = 803.09924
2023-10-27 20:13:30,765              ���- ss_ripple_high_time = 452ms 160us
2023-10-27 20:13:30,765              ���- ss_ripple_low = 800.0775
2023-10-27 20:13:30,765              ���- ss_ripple_low_time = 996ms 760us
2023-10-27 20:13:31,434          [Attachment] Complete
2023-10-27 20:13:32,002          [Attachment] Zoom (900ms to 1s)
2023-10-27 20:13:32,003          > Assert Follows Reference
2023-10-27 20:13:32,003            ���- signal = n_meas
2023-10-27 20:13:32,004            ���- ref_signal = ref
2023-10-27 20:13:32,004            ���- tol = 20
2023-10-27 20:13:32,004            ���- during = None
2023-10-27 20:13:32,004            ���- strictness = 0.8
2023-10-27 20:13:32,004            ���- time_tol = 0.05
```

Resources

Tracking your test system utilization allows you to create strategies to better utilize it. Under Monitor/Resources on the Overview tab, you can quickly see the status of all your devices and Agents.



Computers

Status	Name	Latest CPU usage	Latest RAM usage	Latest disk usage
●	SilentPC	18%	30%	4%

Setups

Status	Name	Today usage	Yesterday usage	Last 7 days usage
●	604-EPC	2%	0%	0%
●	402	1%	0%	0%

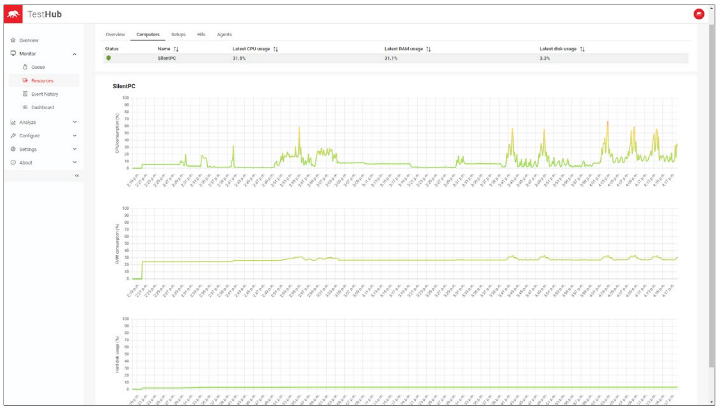
HILs

Status	Name	Today usage	Yesterday usage	Last 7 days usage
●	HIL402	1%	0%	0%
●	C-HIL_01	2%	0%	0%

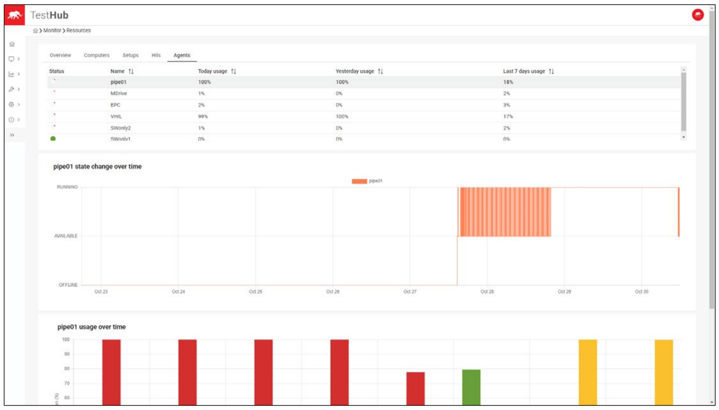
Agents

Status	Name	Today usage	Yesterday usage	Last 7 days usage
	pipe01	3%	0%	0%
●	MDrive	2%	0%	0%
●	EPC	2%	0%	0%
●	VHIL	1%	0%	0%
	SWonly2	2%	0%	0%
●	SWonly1	0%	0%	0%

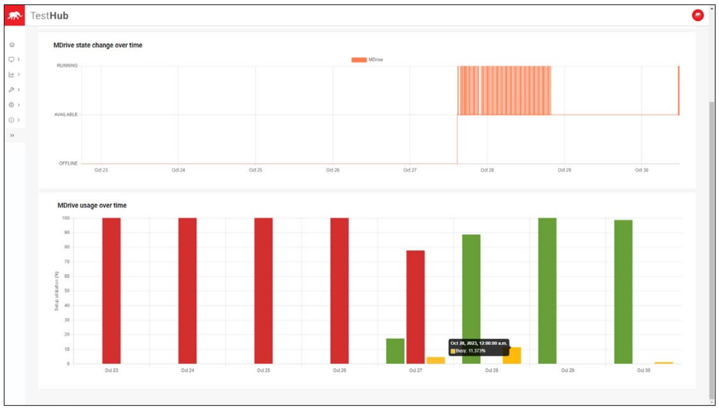
Under the Computer tab, you can visualize CPU, RAM, and Hard disk utilization in detail. This makes it easy to identify whether it is time to split test execution across multiple Computers, or if it is necessary to start deleting test Artifacts due to space limitations. Under the Setup/HILs/Agents tab, you can view detailed utilization data, both as a line plot or as a daily aggregated amount.

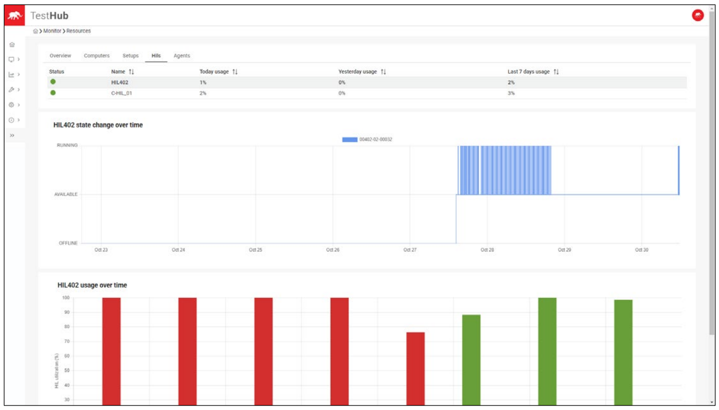


Status	Name ⇅	Latest CPU usage ⇅	Latest RAM usage ⇅	Latest disk usage ⇅
●	SilentPC	31.5%	31.1%	3.3%



Status	Name ⇅	Today usage ⇅	Yesterday usage ⇅	Last 7 days usage ⇅
	pipe01	100%	100%	18%
	MDrive	1%	0%	2%
	EPC	2%	0%	3%
	VHIL	99%	100%	17%
	SWonly2	1%	0%	2%
●	SWonly1	0%	0%	0%

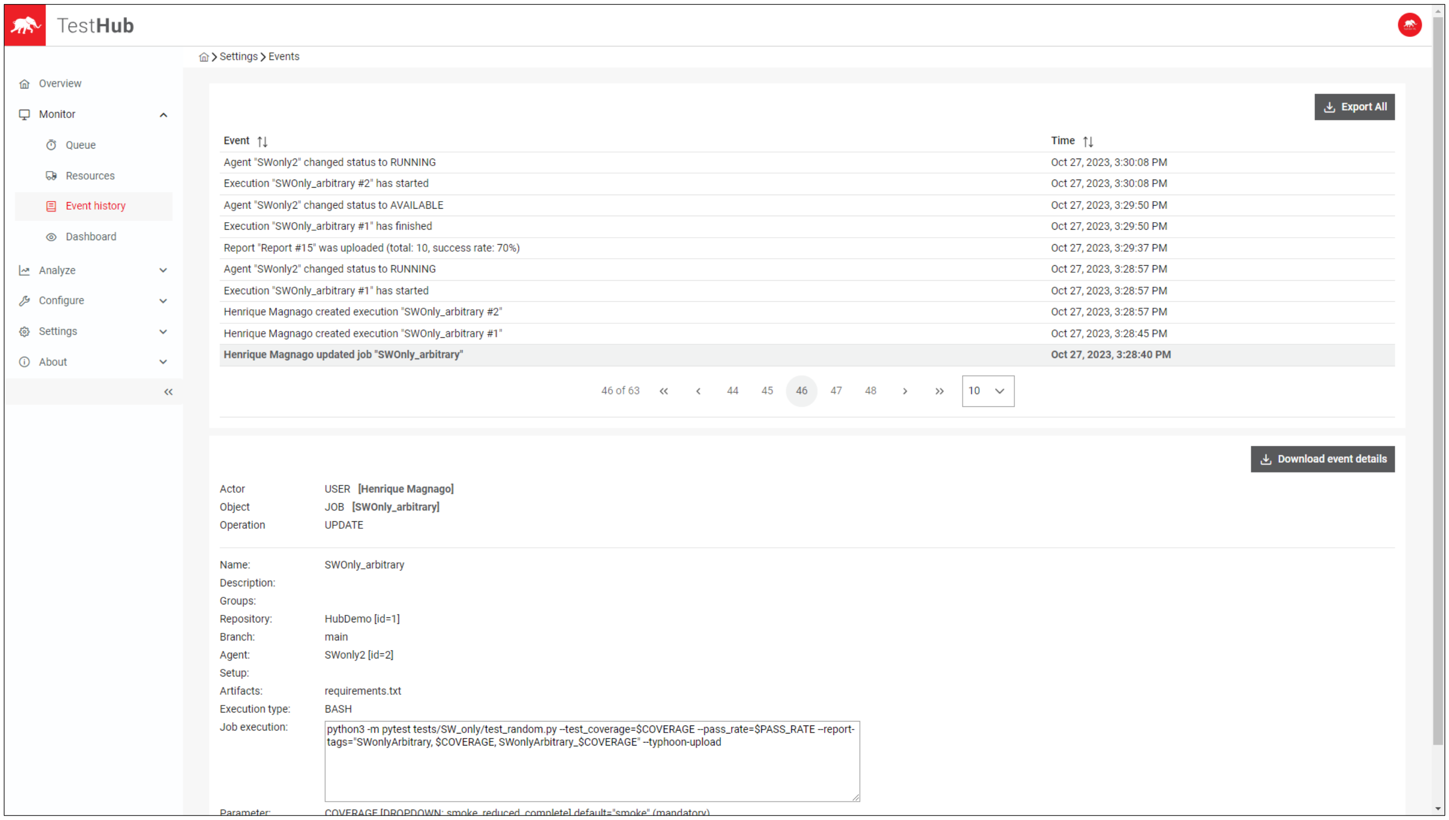




Status	Name ⇅	Today usage ⇅	Yesterday usage ⇅	Last 7 days usage ⇅
●	HIL402	1%	0%	2%
●	C-HIL_01	2%	0%	3%

Event History:

All activity in Typhoon Test Hub is logged and can be used to spot and revert accidental changes. You can quickly visualize the changes in the Event History, or export them for a more in-depth look.



Settings > Events

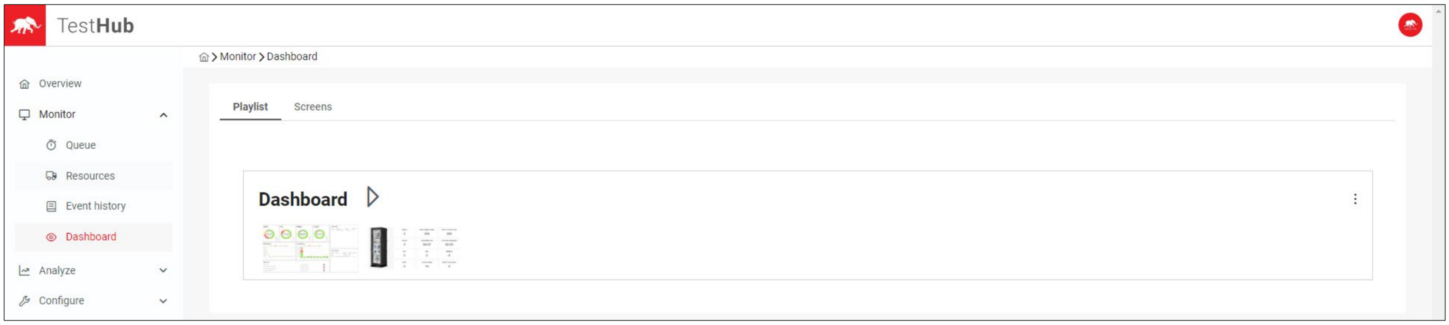
Event ⇅	Time ⇅
Agent "SWonly2" changed status to RUNNING	Oct 27, 2023, 3:30:08 PM
Execution "SWOnly_arbitrary #2" has started	Oct 27, 2023, 3:30:08 PM
Agent "SWonly2" changed status to AVAILABLE	Oct 27, 2023, 3:29:50 PM
Execution "SWOnly_arbitrary #1" has finished	Oct 27, 2023, 3:29:50 PM
Report "Report #15" was uploaded (total: 10, success rate: 70%)	Oct 27, 2023, 3:29:37 PM
Agent "SWonly2" changed status to RUNNING	Oct 27, 2023, 3:28:57 PM
Execution "SWOnly_arbitrary #1" has started	Oct 27, 2023, 3:28:57 PM
Henrique Magnago created execution "SWOnly_arbitrary #2"	Oct 27, 2023, 3:28:57 PM
Henrique Magnago created execution "SWOnly_arbitrary #1"	Oct 27, 2023, 3:28:45 PM
Henrique Magnago updated job "SWOnly_arbitrary"	**Oct 27, 2023, 3:28:40 PM**

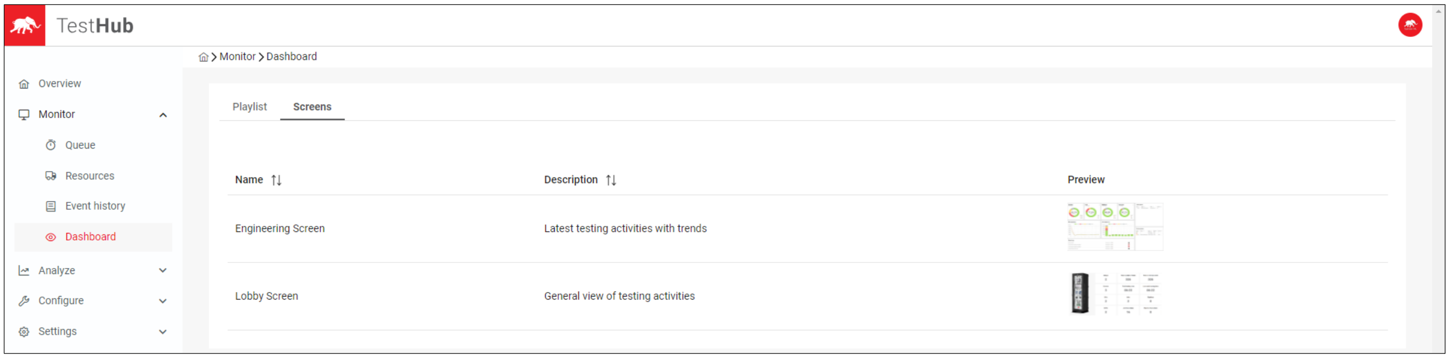
Actor	USER **[Henrique Magnago]**
Object	JOB **[SWOnly_arbitrary]**
Operation	UPDATE

Name:	SWOnly_arbitrary
Description:	
Groups:	
Repository:	HubDemo [id=1]
Branch:	main
Agent:	SWonly2 [id=2]
Setup:	
Artifacts:	requirements.txt
Execution type:	BASH
Job execution:	python3 -m pytest tests/SW_only/test_random.py --test_coverage=$COVERAGE --pass_rate=$PASS_RATE --report-tags="SWonlyArbitrary, $COVERAGE, SWonlyArbitrary_$COVERAGE" --typhoon-upload
Parameter:	COVERAGE [DROPDOWN: smoke, reduced, complete] default="smoke" (mandatory)

Dashboards

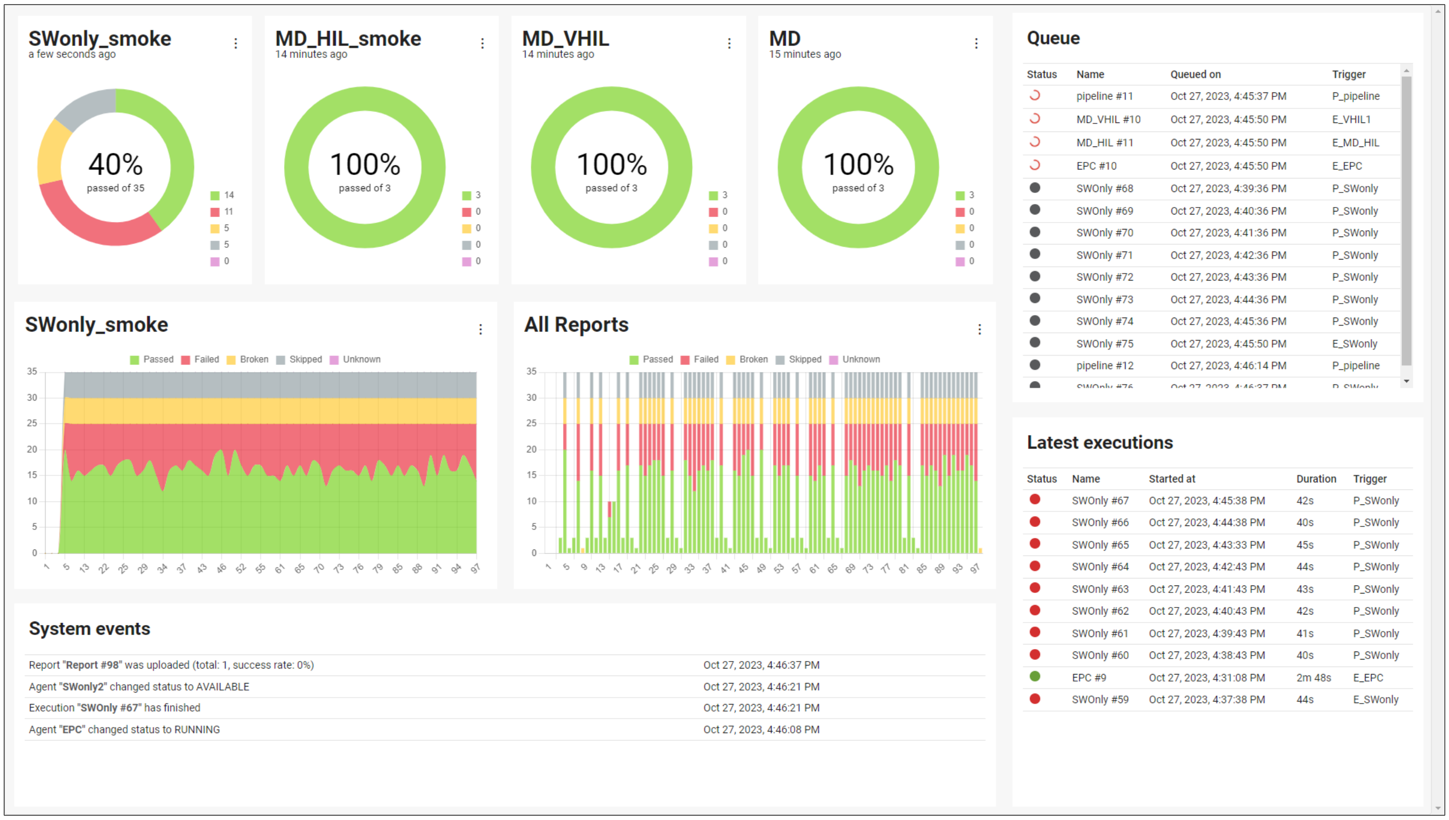
Custom Screens can also be created. Once created, screens can be grouped together into a Dashboard Playlist that will cycle through at regular intervals on the display. A single screen can be used for multiple Playlists.





Name ⇅	Description ⇅	Preview
Engineering Screen	Latest testing activities with trends	
Lobby Screen	General view of testing activities	

Dashboards can also be used for different contexts and displayed on TV screens, such as a detailed view to be displayed in the engineering room, or a more general view in the public lobby.



Queue

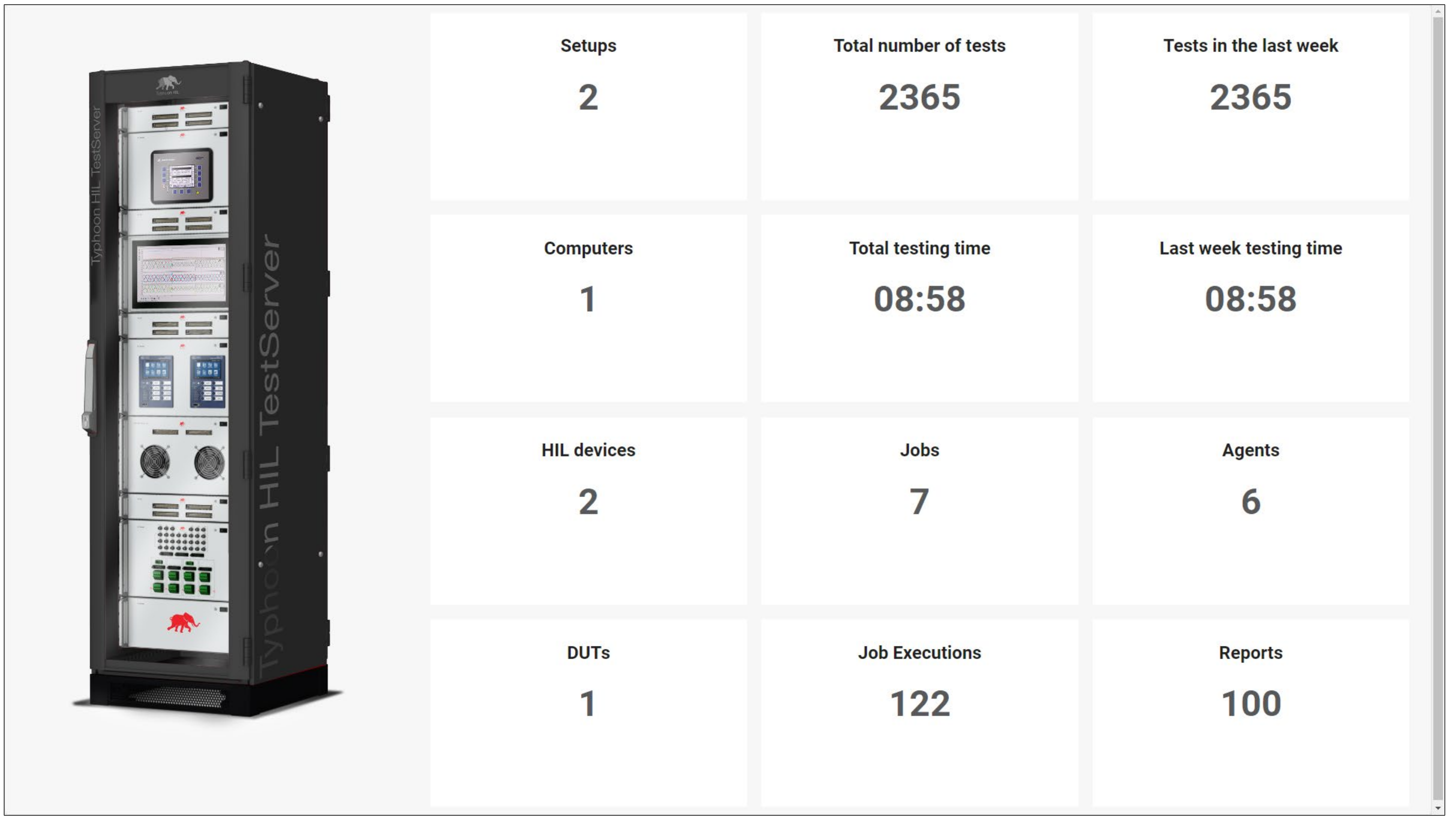
Status	Name	Queued on	Trigger
◌	pipeline #11	Oct 27, 2023, 4:45:37 PM	P_pipeline
◌	MD_VHIL #10	Oct 27, 2023, 4:45:50 PM	E_VHIL1
◌	MD_HIL #11	Oct 27, 2023, 4:45:50 PM	E_MD_HIL
◌	EPC #10	Oct 27, 2023, 4:45:50 PM	E_EPC
●	SWOnly #68	Oct 27, 2023, 4:39:36 PM	P_SWonly
●	SWOnly #69	Oct 27, 2023, 4:40:36 PM	P_SWonly
●	SWOnly #70	Oct 27, 2023, 4:41:36 PM	P_SWonly
●	SWOnly #71	Oct 27, 2023, 4:42:36 PM	P_SWonly
●	SWOnly #72	Oct 27, 2023, 4:43:36 PM	P_SWonly
●	SWOnly #73	Oct 27, 2023, 4:44:36 PM	P_SWonly
●	SWOnly #74	Oct 27, 2023, 4:45:36 PM	P_SWonly
●	SWOnly #75	Oct 27, 2023, 4:45:50 PM	E_SWonly
●	pipeline #12	Oct 27, 2023, 4:46:14 PM	P_pipeline
●	SWOnly #76	Oct 27, 2023, 4:46:37 PM	P_SWonly

Latest executions

Status	Name	Started at	Duration	Trigger
●	SWOnly #67	Oct 27, 2023, 4:45:38 PM	42s	P_SWonly
●	SWOnly #66	Oct 27, 2023, 4:44:38 PM	40s	P_SWonly
●	SWOnly #65	Oct 27, 2023, 4:43:33 PM	45s	P_SWonly
●	SWOnly #64	Oct 27, 2023, 4:42:43 PM	44s	P_SWonly
●	SWOnly #63	Oct 27, 2023, 4:41:43 PM	43s	P_SWonly
●	SWOnly #62	Oct 27, 2023, 4:40:43 PM	42s	P_SWonly
●	SWOnly #61	Oct 27, 2023, 4:39:43 PM	41s	P_SWonly
●	SWOnly #60	Oct 27, 2023, 4:38:43 PM	40s	P_SWonly
●	EPC #9	Oct 27, 2023, 4:31:08 PM	2m 48s	E_EPC
●	SWOnly #59	Oct 27, 2023, 4:37:38 PM	44s	E_SWonly

System events

Report "**Report #98**" was uploaded (total: 1, success rate: 0%)	Oct 27, 2023, 4:46:37 PM
Agent "**SWonly2**" changed status to AVAILABLE	Oct 27, 2023, 4:46:21 PM
Execution "**SWOnly #67**" has finished	Oct 27, 2023, 4:46:21 PM
Agent "**EPC**" changed status to RUNNING	Oct 27, 2023, 4:46:08 PM



Setups

2

Total number of tests

2365

Tests in the last week

2365

Computers

1

Total testing time

08:58

Last week testing time

08:58

HIL devices

2

Jobs

7

Agents

6

DUTs

1

Job Executions

122

Reports

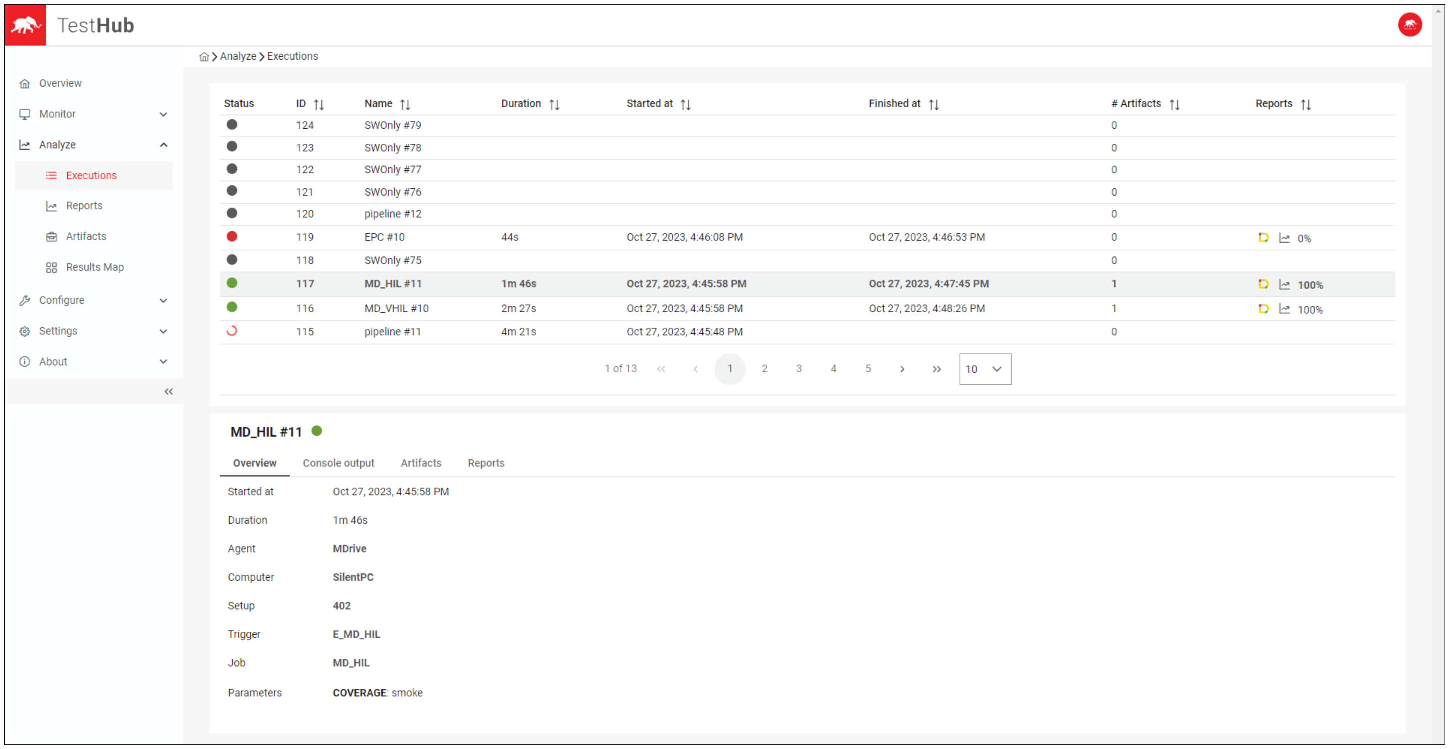
100

Analyzing Execution data

Analyzing the test results is one of the most important aspects of testing. Running a Job creates an Execution which may contain one or more Reports, each containing several Tests. Test Execution can also generate files (Artifacts) which should be stored.

Executions

Under Analyze/Executions, you can find all Executions. When selecting a specific Execution, you will be able to see more details: its configuration, which resources it used, how it was parametrized, the console output, the generated Artifacts, and all Reports. Most of the details are also links, which provide even more information about that selected item.

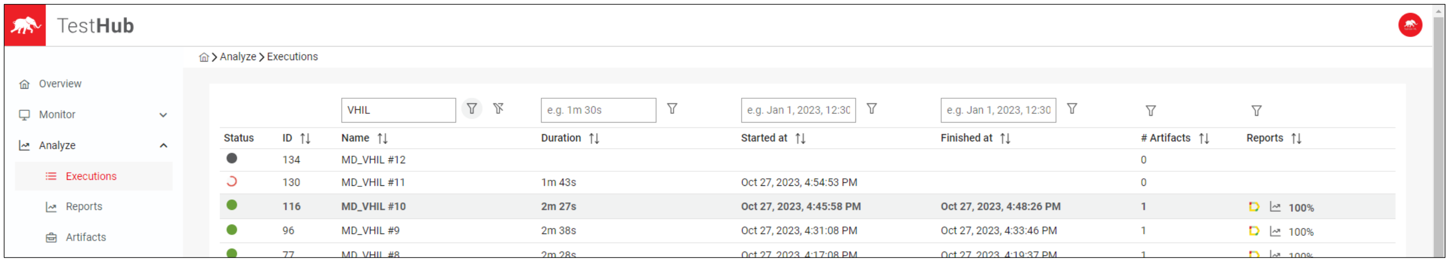


Status	ID	Name	Duration	Started at	Finished at	# Artifacts	Reports
●	124	SWOnly #79				0	
●	123	SWOnly #78				0	
●	122	SWOnly #77				0	
●	121	SWOnly #76				0	
●	120	pipeline #12				0	
●	119	EPC #10	44s	Oct 27, 2023, 4:46:08 PM	Oct 27, 2023, 4:46:53 PM	0	0%
●	118	SWOnly #75				0	
●	117	MD_HIL #11	1m 46s	Oct 27, 2023, 4:45:58 PM	Oct 27, 2023, 4:47:45 PM	1	100%
●	116	MD_VHIL #10	2m 27s	Oct 27, 2023, 4:45:58 PM	Oct 27, 2023, 4:48:26 PM	1	100%
○	115	pipeline #11	4m 21s	Oct 27, 2023, 4:45:48 PM		0	

MD_HIL #11 ●

Started at	Oct 27, 2023, 4:45:58 PM
Duration	1m 46s
Agent	MDrive
Computer	SilentPC
Setup	402
Trigger	E_MD_HIL
Job	MD_HIL
Parameters	COVERAGE: smoke

Here, it is also possible to filter Executions by different criteria.

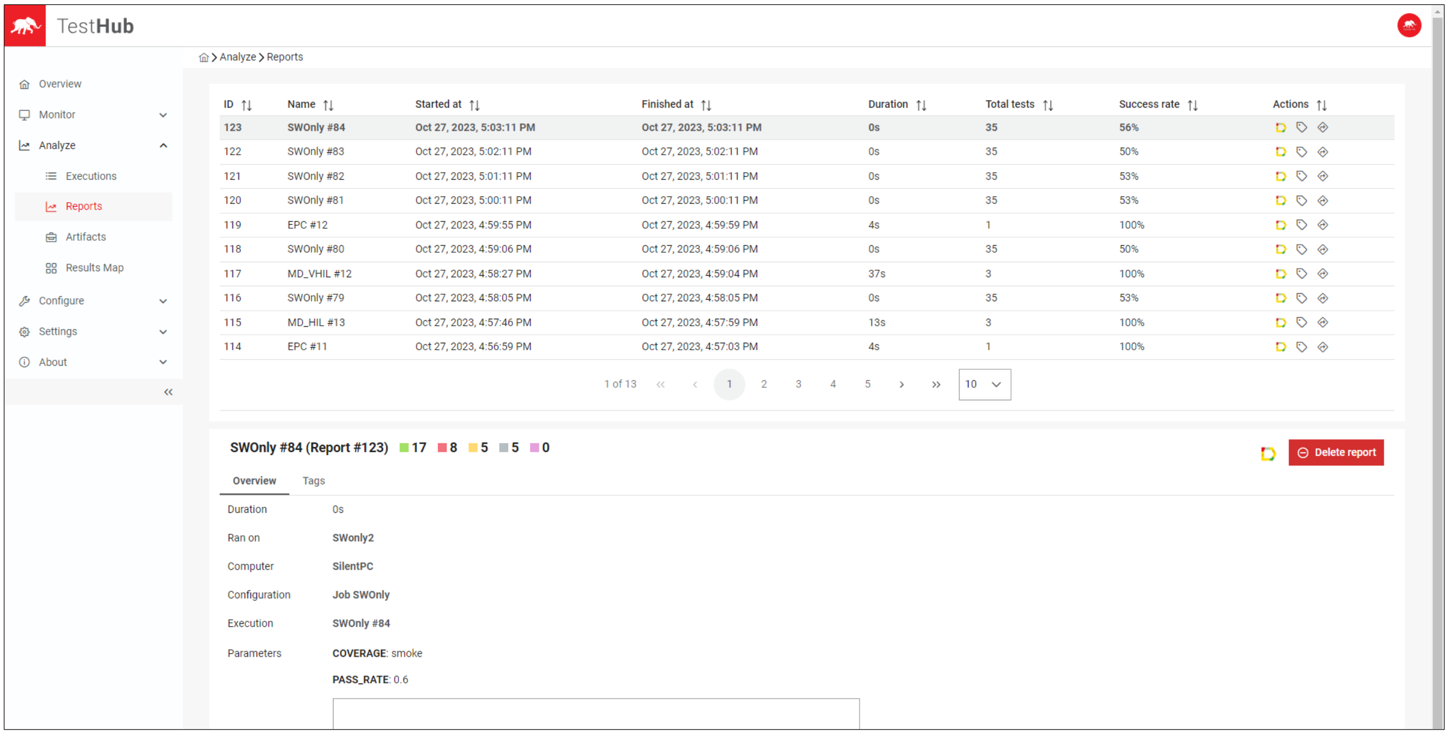


Status	ID	Name	Duration	Started at	Finished at	# Artifacts	Reports
●	134	MD_VHIL #12				0	
○	130	MD_VHIL #11	1m 43s	Oct 27, 2023, 4:54:53 PM		0	
●	116	MD_VHIL #10	2m 27s	Oct 27, 2023, 4:45:58 PM	Oct 27, 2023, 4:48:26 PM	1	100%
●	96	MD_VHIL #9	2m 38s	Oct 27, 2023, 4:31:08 PM	Oct 27, 2023, 4:33:46 PM	1	100%
●	77	MD_VHIL #8	2m 28s	Oct 27, 2023, 4:17:08 PM	Oct 27, 2023, 4:19:37 PM	1	100%

On the far right, if an Execution also generated an Allure report, it will be added there alongside the success rate. Clicking on the Allure icon will open the corresponding Allure report to that Execution, with additional details covered in the Reporting section.

Reports

The Report page is quite similar to the Execution page, with some additional granularity on the performance of the test itself.

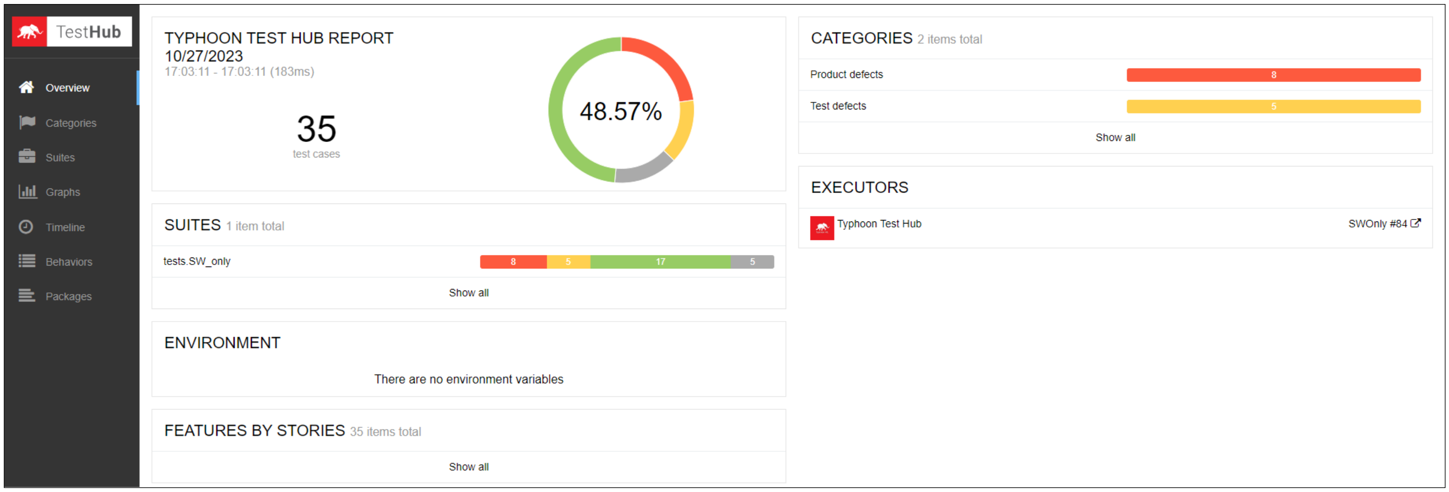


ID	Name	Started at	Finished at	Duration	Total tests	Success rate	Actions
123	SWOnly #84	Oct 27, 2023, 5:03:11 PM	Oct 27, 2023, 5:03:11 PM	0s	35	56%	
122	SWOnly #83	Oct 27, 2023, 5:02:11 PM	Oct 27, 2023, 5:02:11 PM	0s	35	50%	
121	SWOnly #82	Oct 27, 2023, 5:01:11 PM	Oct 27, 2023, 5:01:11 PM	0s	35	53%	
120	SWOnly #81	Oct 27, 2023, 5:00:11 PM	Oct 27, 2023, 5:00:11 PM	0s	35	53%	
119	EPC #12	Oct 27, 2023, 4:59:55 PM	Oct 27, 2023, 4:59:59 PM	4s	1	100%	
118	SWOnly #80	Oct 27, 2023, 4:59:06 PM	Oct 27, 2023, 4:59:06 PM	0s	35	50%	
117	MD_VHIL #12	Oct 27, 2023, 4:58:27 PM	Oct 27, 2023, 4:59:04 PM	37s	3	100%	
116	SWOnly #79	Oct 27, 2023, 4:58:05 PM	Oct 27, 2023, 4:58:05 PM	0s	35	53%	
115	MD_HIL #13	Oct 27, 2023, 4:57:46 PM	Oct 27, 2023, 4:57:59 PM	13s	3	100%	
114	EPC #11	Oct 27, 2023, 4:56:59 PM	Oct 27, 2023, 4:57:03 PM	4s	1	100%	

SWOnly #84 (Report #123) 🟩 17 🟥 8 🟨 5 ⬜ 5 🟪 0

Duration	0s
Ran on	SWonly2
Computer	SilentPC
Configuration	Job SWOnly
Execution	SWOnly #84
Parameters	**COVERAGE**: smoke **PASS_RATE**: 0.6

You can open the Allure report by clicking on the Allure report icon under the Action column.



TYPHOON TEST HUB REPORT
10/27/2023
17:03:11 - 17:03:11 (183ms)

35
test cases

SUITES 1 item total

tests.SW_only

ENVIRONMENT

There are no environment variables

FEATURES BY STORIES 35 items total

CATEGORIES 2 items total

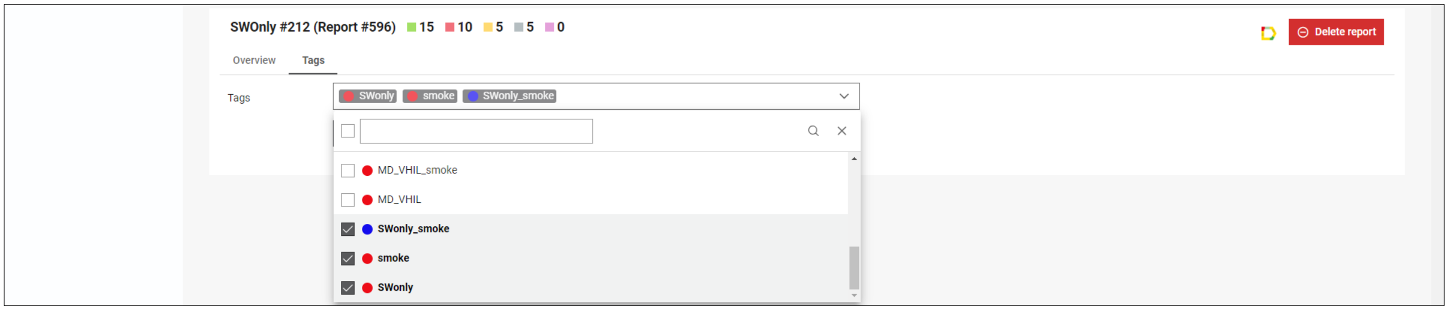
Product defects

Test defects

EXECUTORS

Typhoon Test Hub SWOnly #84

You can share a link to this report directly with your colleagues which they can open by logging into Typhoon Test Hub. If you want to check more information about the Execution which generated a specific report, simply click on the link under “Executors” to return to the details in TTH. Tags for a specific report can be added prior to Execution or after Execution. You can see them under the Tags tab.



SWOnly #212 (Report #596) 🟩 15 🟥 10 🟨 5 ⬜ 5 🟪 0

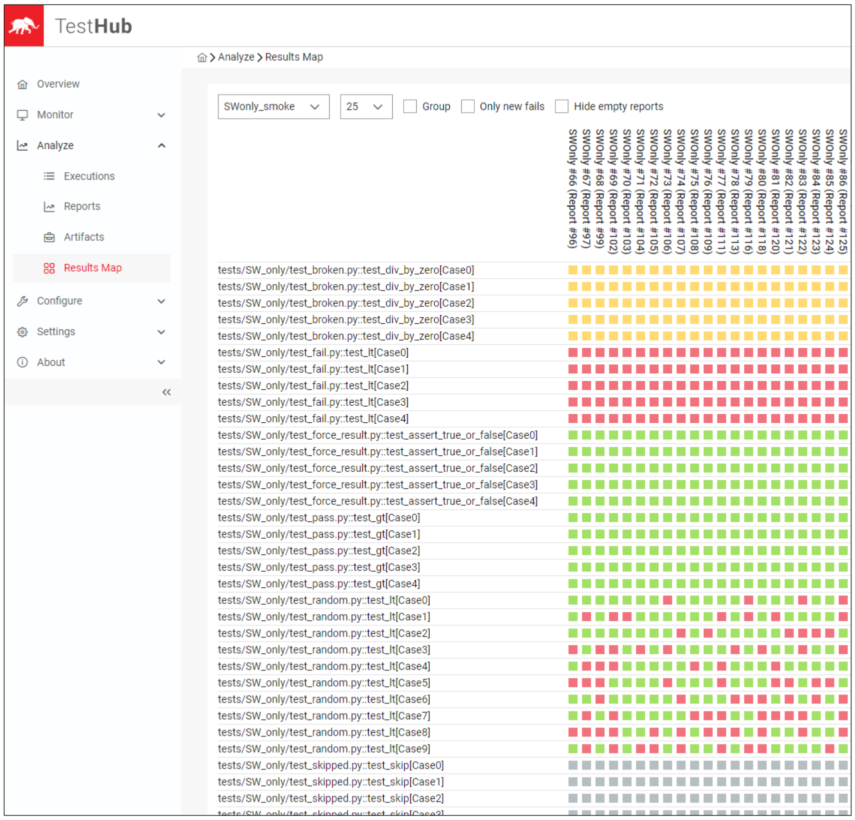
Tags: SWonly, smoke, SWonly_smoke

☐ MD_VHIL_smoke
☐ MD_VHIL
☑ SWonly_smoke
☑ smoke
☑ SWonly

Allure Reports can be easily updated to TTH at the end of a Job execution, or can be uploaded manually from anywhere by means of using our standalone *TTH report uploader* library.

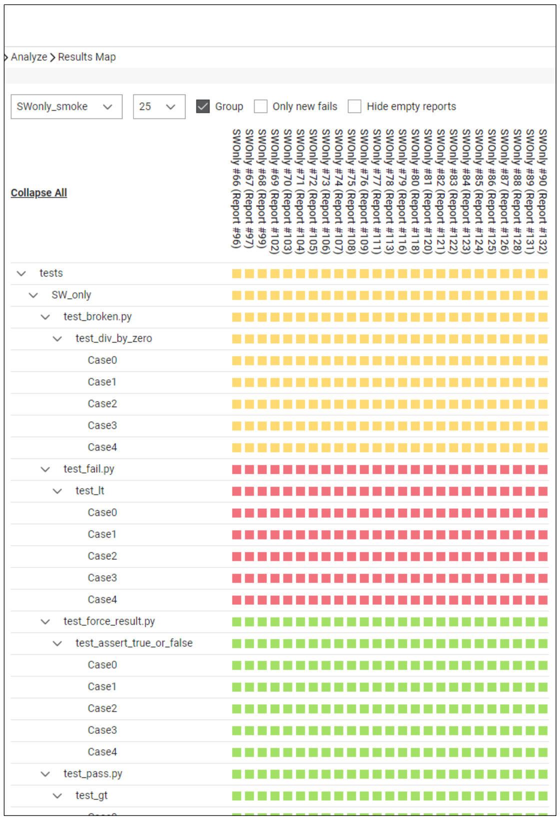
Results Map

You can quickly compare results of the same tests from different executions using the Results Map tab. You can easily switch which group of tests you want to compare and how many test executions you would like to see. On the left, you can see the test name, on the top the Execution and Report, and in the body of the table the test results. Test results are color coded for easy interpretation of the results. Clicking any square, will open the corresponding Allure report for that precise execution and test case.



SWonly_smoke ⌄ 25 ⌄ □ Group □ Only new fails □ Hide empty reports

	SWOnly #66 (Report #96)	SWOnly #67 (Report #97)	SWOnly #68 (Report #99)	SWOnly #69 (Report #102)	SWOnly #70 (Report #103)	SWOnly #71 (Report #104)	SWOnly #72 (Report #105)	SWOnly #73 (Report #106)	SWOnly #74 (Report #107)	SWOnly #75 (Report #108)	SWOnly #76 (Report #109)	SWOnly #77 (Report #111)	SWOnly #78 (Report #113)	SWOnly #79 (Report #116)	SWOnly #80 (Report #118)	SWOnly #81 (Report #120)	SWOnly #82 (Report #121)	SWOnly #83 (Report #122)	SWOnly #84 (Report #123)	SWOnly #85 (Report #124)	SWOnly #86 (Report #125)
tests/SW_only/test_broken.py::test_div_by_zero[Case0]	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
tests/SW_only/test_broken.py::test_div_by_zero[Case1]	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
tests/SW_only/test_broken.py::test_div_by_zero[Case2]	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
tests/SW_only/test_broken.py::test_div_by_zero[Case3]	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
tests/SW_only/test_broken.py::test_div_by_zero[Case4]	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
tests/SW_only/test_fail.py::test_lt[Case0]	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
tests/SW_only/test_fail.py::test_lt[Case1]	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
tests/SW_only/test_fail.py::test_lt[Case2]	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
tests/SW_only/test_fail.py::test_lt[Case3]	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
tests/SW_only/test_fail.py::test_lt[Case4]	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
tests/SW_only/test_force_result.py::test_assert_true_or_false[Case0]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_force_result.py::test_assert_true_or_false[Case1]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_force_result.py::test_assert_true_or_false[Case2]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_force_result.py::test_assert_true_or_false[Case3]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_force_result.py::test_assert_true_or_false[Case4]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_pass.py::test_gt[Case0]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_pass.py::test_gt[Case1]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_pass.py::test_gt[Case2]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_pass.py::test_gt[Case3]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_pass.py::test_gt[Case4]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_random.py::test_lt[Case0]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟥	🟩	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case1]	🟩	🟥	🟩	🟥	🟥	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟥	🟩	🟥	🟩	🟩	🟩	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case2]	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟥	🟥	🟥	🟥	🟩
tests/SW_only/test_random.py::test_lt[Case3]	🟥	🟩	🟥	🟥	🟩	🟥	🟩	🟥	🟩	🟩	🟩	🟩	🟥	🟩	🟥	🟩	🟩	🟥	🟩	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case4]	🟩	🟥	🟥	🟥	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟥	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟩
tests/SW_only/test_random.py::test_lt[Case5]	🟥	🟥	🟥	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟥	🟥	🟩	🟥	🟥	🟩
tests/SW_only/test_random.py::test_lt[Case6]	🟩	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟥	🟥	🟥	🟩	🟥	🟩	🟥	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case7]	🟩	🟥	🟩	🟥	🟩	🟩	🟩	🟩	🟩	🟥	🟥	🟥	🟩	🟩	🟥	🟥	🟥	🟥	🟩	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case8]	🟥	🟥	🟥	🟥	🟩	🟩	🟥	🟩	🟥	🟩	🟥	🟥	🟥	🟩	🟥	🟩	🟩	🟥	🟩	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case9]	🟩	🟥	🟩	🟥	🟩	🟥	🟥	🟩	🟥	🟩	🟩	🟥	🟩	🟥	🟥	🟩	🟩	🟩	🟩	🟥	🟩
tests/SW_only/test_skipped.py::test_skip[Case0]	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜
tests/SW_only/test_skipped.py::test_skip[Case1]	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜
tests/SW_only/test_skipped.py::test_skip[Case2]	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜	⬜

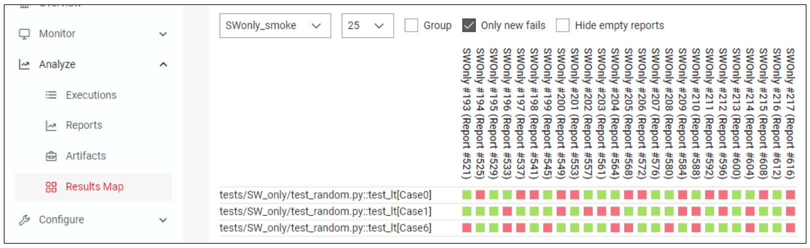


SWonly_smoke ⌄ 25 ⌄ ☑ Group □ Only new fails □ Hide empty reports

Collapse All

	SWOnly #66 (Report #96)	SWOnly #67 (Report #97)	SWOnly #68 (Report #99)	SWOnly #69 (Report #102)	SWOnly #70 (Report #103)	SWOnly #71 (Report #104)	SWOnly #72 (Report #105)	SWOnly #73 (Report #106)	SWOnly #74 (Report #107)	SWOnly #75 (Report #108)	SWOnly #76 (Report #109)	SWOnly #77 (Report #111)	SWOnly #78 (Report #113)	SWOnly #79 (Report #116)	SWOnly #80 (Report #118)	SWOnly #81 (Report #120)	SWOnly #82 (Report #121)	SWOnly #83 (Report #122)	SWOnly #84 (Report #123)	SWOnly #85 (Report #124)	SWOnly #86 (Report #125)	SWOnly #87 (Report #126)	SWOnly #88 (Report #128)	SWOnly #89 (Report #131)	SWOnly #90 (Report #132)
⌄ tests	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
⌄ SW_only	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
⌄ test_broken.py	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
⌄ test_div_by_zero	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
Case0	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
Case1	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
Case2	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
Case3	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
Case4	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨	🟨
⌄ test_fail.py	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
⌄ test_lt	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
Case0	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
Case1	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
Case2	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
Case3	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
Case4	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
⌄ test_force_result.py	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
⌄ test_assert_true_or_false	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Case0	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Case1	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Case2	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Case3	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Case4	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
⌄ test_pass.py	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
⌄ test_gt	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩

If you are looking for tests that failed in the most recent Execution, but passed before, check the “Only new Fails” box to filter the results.



SWonly_smoke ⌄ 25 ⌄ □ Group ☑ Only new fails □ Hide empty reports

	SWOnly #193 (Report #521)	SWOnly #194 (Report #525)	SWOnly #195 (Report #529)	SWOnly #196 (Report #533)	SWOnly #197 (Report #537)	SWOnly #198 (Report #541)	SWOnly #199 (Report #545)	SWOnly #200 (Report #549)	SWOnly #201 (Report #553)	SWOnly #202 (Report #557)	SWOnly #203 (Report #561)	SWOnly #204 (Report #564)	SWOnly #205 (Report #568)	SWOnly #206 (Report #572)	SWOnly #207 (Report #576)	SWOnly #208 (Report #580)	SWOnly #209 (Report #584)	SWOnly #210 (Report #588)	SWOnly #211 (Report #592)	SWOnly #212 (Report #596)	SWOnly #213 (Report #600)	SWOnly #214 (Report #604)	SWOnly #215 (Report #608)	SWOnly #216 (Report #612)	SWOnly #217 (Report #616)
tests/SW_only/test_random.py::test_lt[Case0]	🟩	🟥	🟩	🟩	🟥	🟥	🟩	🟥	🟥	🟩	🟩	🟩	🟥	🟥	🟩	🟩	🟥	🟩	🟥	🟥	🟩	🟩	🟥	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case1]	🟩	🟩	🟩	🟥	🟩	🟩	🟩	🟥	🟩	🟥	🟥	🟥	🟩	🟩	🟩	🟩	🟥	🟥	🟩	🟥	🟩	🟥	🟩	🟩	🟥
tests/SW_only/test_random.py::test_lt[Case6]	🟥	🟩	🟩	🟥	🟥	🟩	🟥	🟩	🟩	🟩	🟩	🟥	🟥	🟩	🟩	🟥	🟩	🟥	🟩	🟩	🟩	🟥	🟩	🟩	🟥

Configuring

In order to take advantage of TTH's Visualization features, Execution must be configured. This process involves five steps:

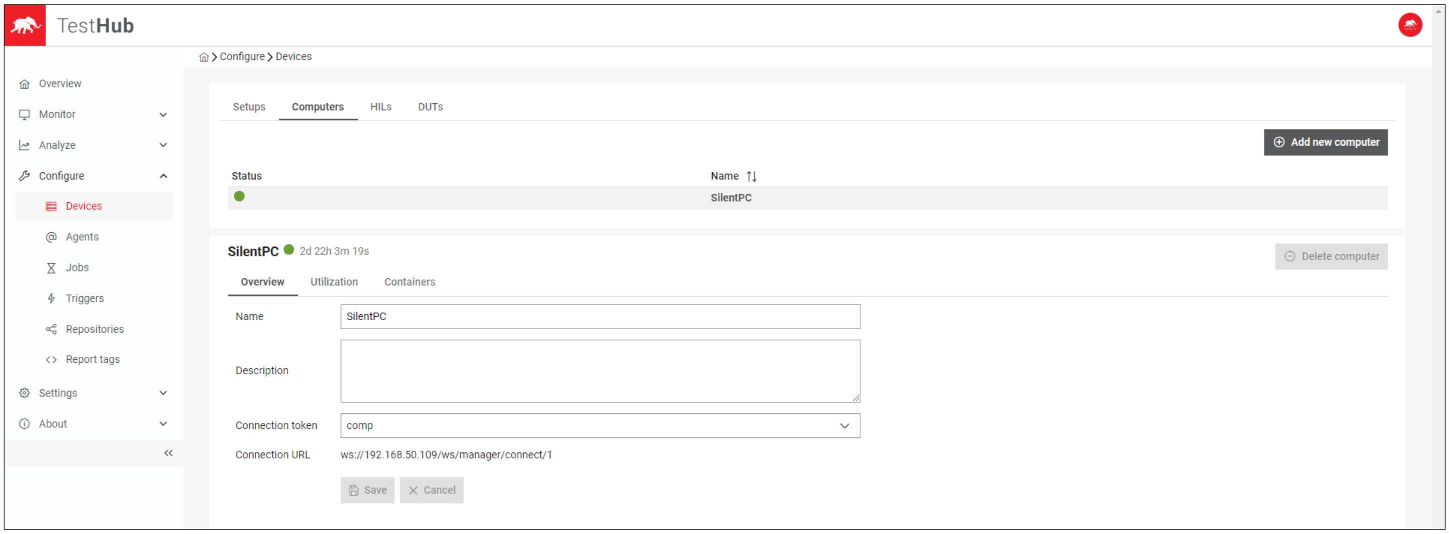
1. Adding devices and creating Setups;
2. Creating Agents;
3. Configuring Git credentials and repositories;
4. Defining the Job;
5. Starting the execution.

Devices

Here is where you can configure all Hardware devices in your testbed, so they can easily be referred to when creating new Jobs. This informs which devices should be used to run a specific test Execution.

Computers

If there are multiple Computers connected to the same Hub, they will be added and displayed here. After starting *Officer* on the Computer, it will become online on the Hub. The *Officer* application allows for creating and starting Agents on the computer; collecting resource (RAM, CPU, storage) utilization; and more.



Configure > Devices

Status	Name ⇅
●	SilentPC

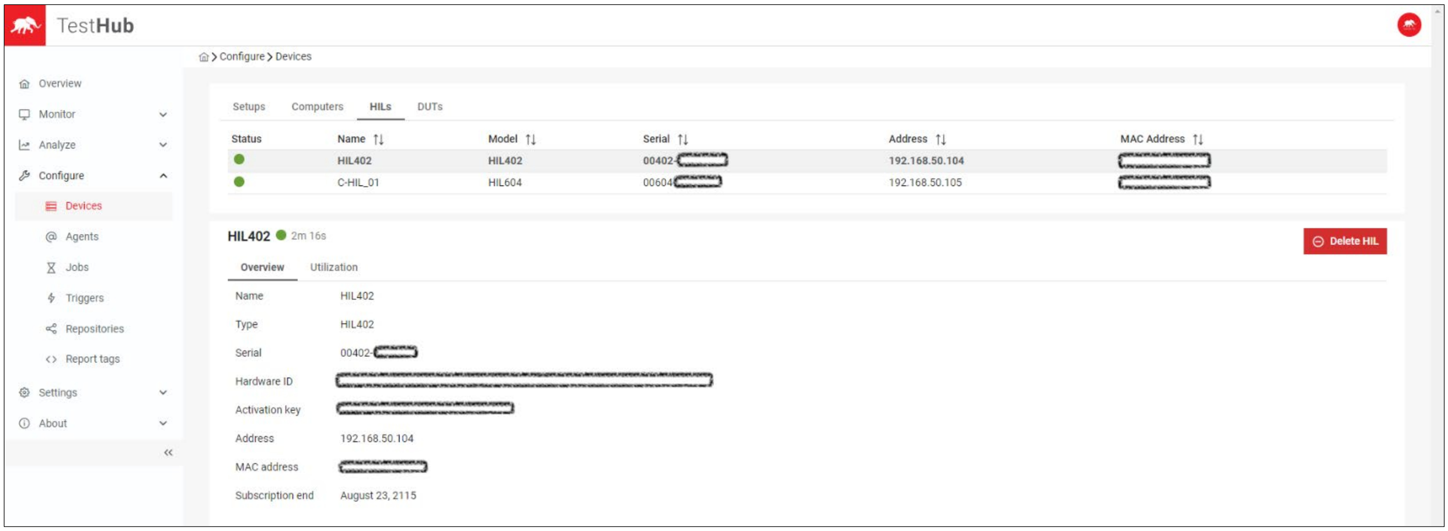
SilentPC ● 2d 22h 3m 19s

Name	SilentPC
Description	
Connection token	comp
Connection URL	ws://192.168.50.109/ws/manager/connect/1

The Typhoon Test Hub distribution package comes with the *Officer* application folder. To run *Officer* on a computer, use the interface to navigate to that folder and run the setup command, informing the Connection URL and token value.

HILs

All HIL devices in the same network as any active Computer will be automatically visible under the HIL tab. Here, you can check details for each HIL.



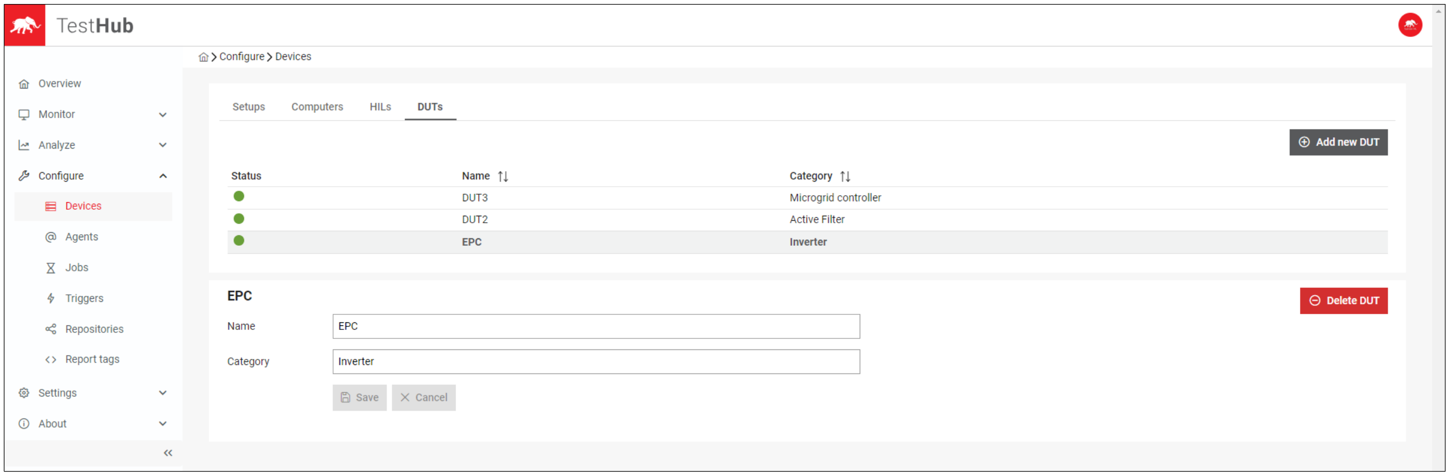
Status	Name	Model	Serial	Address	MAC Address
●	HIL402	HIL402	00402-[redacted]	192.168.50.104	[redacted]
●	C-HIL_01	HIL604	00604[redacted]	192.168.50.105	[redacted]

HIL402 ● 2m 16s

Name	HIL402
Type	HIL402
Serial	00402-[redacted]
Hardware ID	[redacted]
Activation key	[redacted]
Address	192.168.50.104
MAC address	[redacted]
Subscription end	August 23, 2115

Device Under Test (DUT)

It is also possible to add information about the DUT(s). The data displayed here, and available control options differ on a customer-to-customer basis and can be customized to your DUT(s).



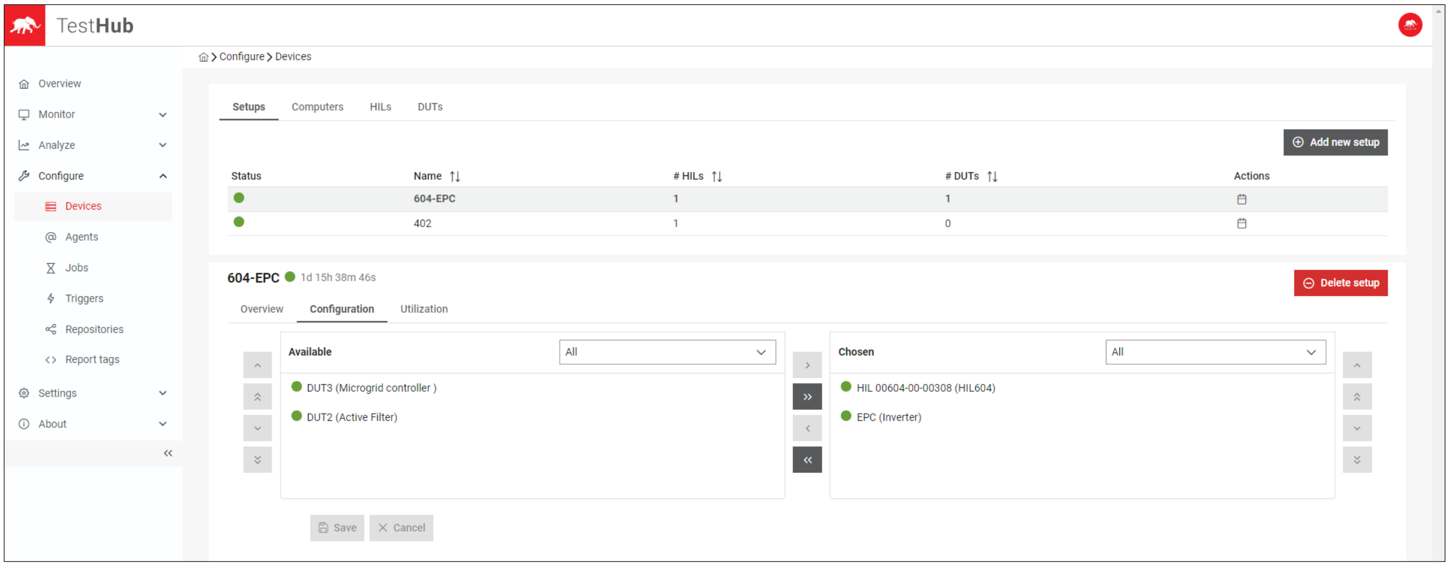
Status	Name	Category
●	DUT3	Microgrid controller
●	DUT2	Active Filter
●	EPC	Inverter

EPC

Name	EPC
Category	Inverter

Setups

Once devices are added, it is possible to build Setups. Setups define a group of Devices that are used for different tests. When configuring a Job, Hub informs the Setup, so the Execution knows which Devices are available.

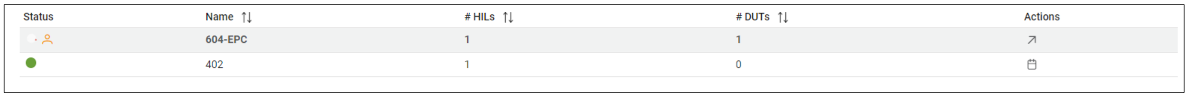


Status	Name	# HILs	# DUTs	Actions
●	604-EPC	1	1	
●	402	1	0	

604-EPC ● 1d 15h 38m 46s

Available: All	Chosen: All
● DUT3 (Microgrid controller)	● HIL 00604-00-00308 (HIL604)
● DUT2 (Active Filter)	● EPC (Inverter)

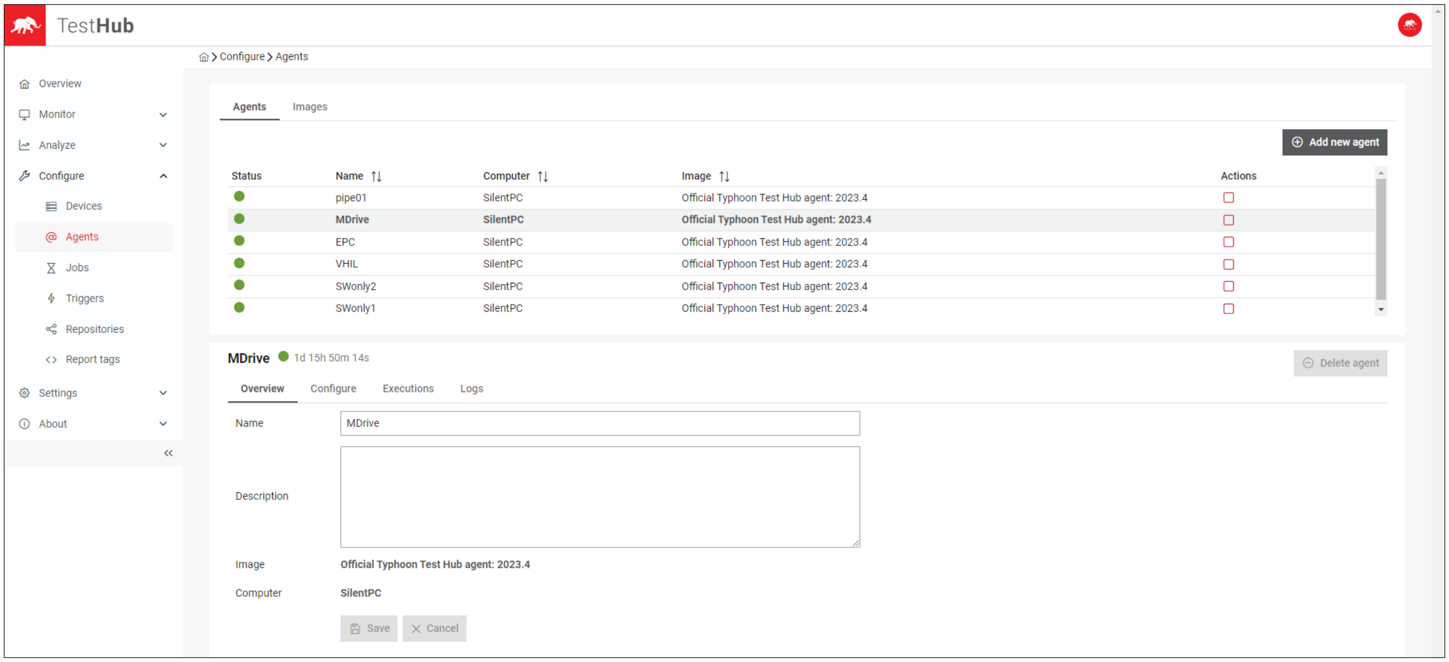
It is possible reserve Setups to be used for manual tests. The Hub avoids starting Executions with Setups marked for manual tests, but keeps them in the Queue until the Setup becomes available.



Status	Name	# HILs	# DUTs	Actions
👤	604-EPC	1	1	↗
●	402	1	0	

Agents

Agents are responsible for executing steps defined in the Job on the selected Computer, utilizing devices belonging to the Setup, and uploading traceable results to the Hub. Multiple Agents can run on the same Computer when running Linux (only a single Agent can run at a time on Windows). The recommended approach is to run Agents as docker containers in dedicated Linux Computers. The Agents use Docker Images which contain all applications and the corresponding Typhoon HIL Control Center version. Several 'instances' of an Agent can be generated from the same Image. Docker containers allow executions to start very easily with the exact same configuration, ensuring reproducibility and traceability and avoiding singular Environment, Execution, or Test dependency.



Status	Name	Computer	Image	Actions
●	pipe01	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	MDrive	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	EPC	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	VHIL	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	SWonly2	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	SWonly1	SilentPC	Official Typhoon Test Hub agent: 2023.4	□

MDrive ● 1d 15h 50m 14s

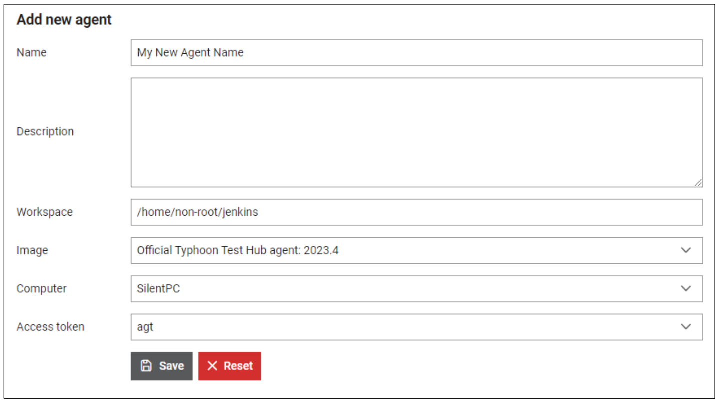
Name: MDrive

Description:

Image: Official Typhoon Test Hub agent: 2023.4

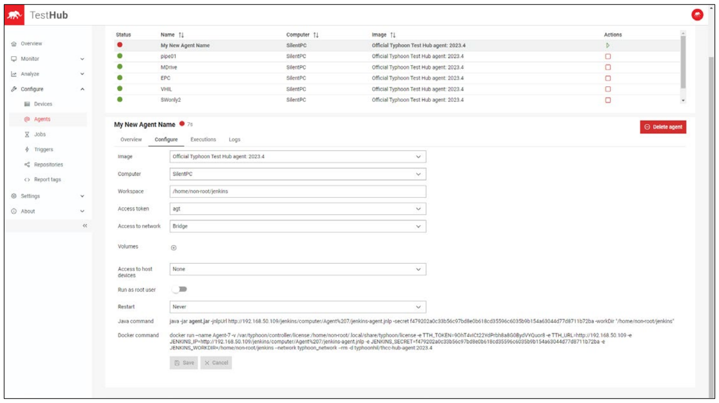
Computer: SilentPC

The process of creating new Agents is greatly simplified with TTH. When creating a new Agent, you will select on which Computer it will run, which version of THCC it will use, and its corresponding access Token. From there, you can press Play to start the Agent on the selected computer, or download the Agent file and manually start it on a Windows computer by running the provided command.



Add new agent

Name	My New Agent Name
Description	
Workspace	/home/non-root/jenkins
Image	Official Typhoon Test Hub agent: 2023.4
Computer	SilentPC
Access token	agt



Status	Name	Computer	Image	Actions
●	My New Agent Name	SilentPC	Official Typhoon Test Hub agent: 2023.4	▷
●	pipe01	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	MDrive	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	EPC	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	VHIL	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	SWonly2	SilentPC	Official Typhoon Test Hub agent: 2023.4	□

My New Agent Name ● 7s

Image	Official Typhoon Test Hub agent: 2023.4
Computer	SilentPC
Workspace	/home/non-root/jenkins
Access token	agt
Access to network	Bridge
Volumes	⊕
Access to host devices	None
Run as root user	
Restart	Never

Java command

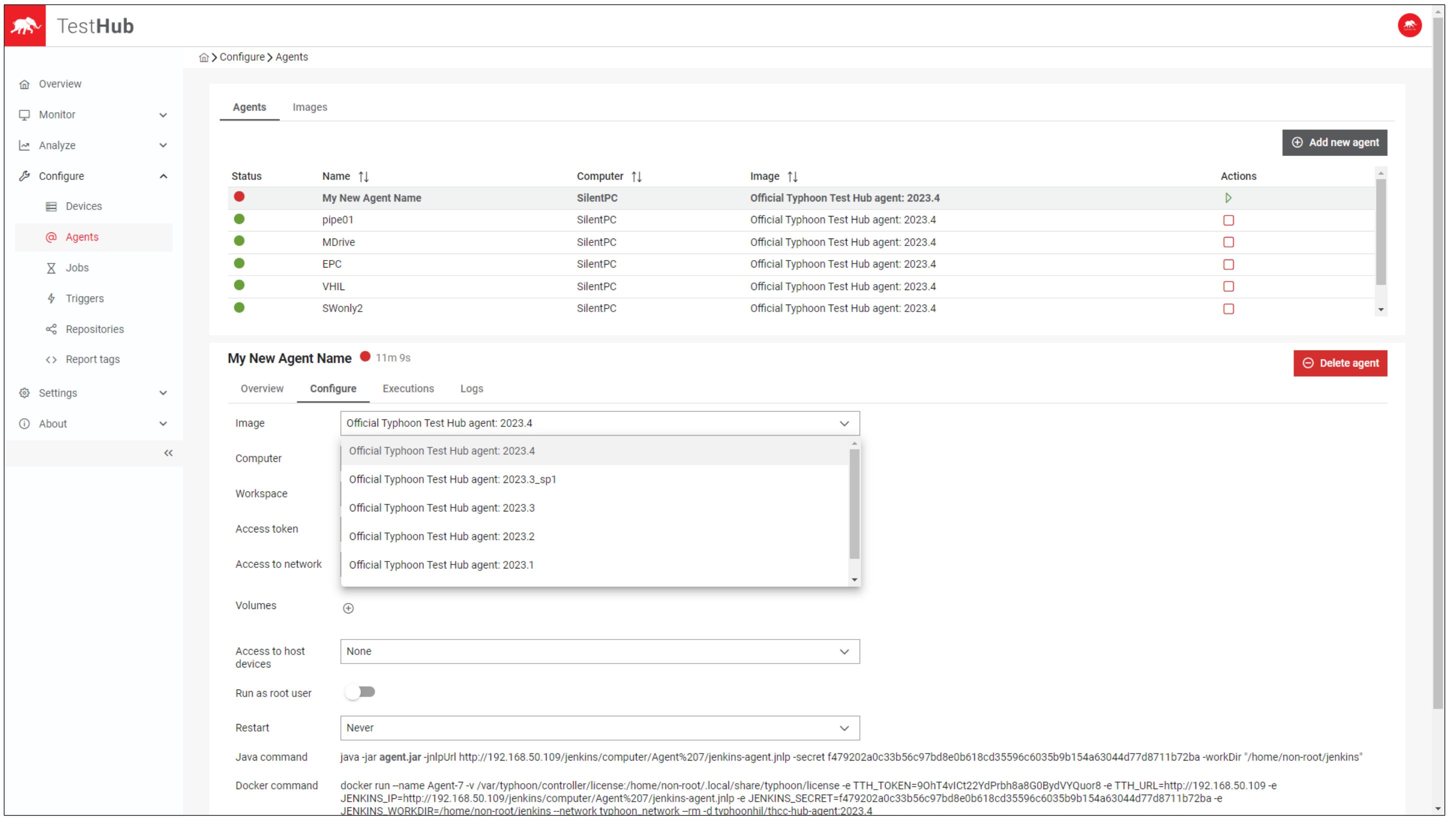
```
java -jar agent.jar -jnlpUrl http://192.168.50.109/jenkins/computer/Agent%207/jenkins-agent.jnlp -secret f479202a0c33b56c97bd8e0b618cd35596c6035b9b154a63044d77d8711b72ba -workDir "/home/non-root/jenkins"
```

Docker command

```
docker run --name Agent-7 -v /var/typhoon/controller/license:/home/non-root/.local/share/typhoon/license -e TTH_TOKEN=9OhT4vICt22YdPrbh8a8G0BydVYQuor8 -e TTH_URL=http://192.168.50.109 -e
JENKINS_IP=http://192.168.50.109/jenkins/computer/Agent%207/jenkins-agent.jnlp -e JENKINS_SECRET=f479202a0c33b56c97bd8e0b618cd35596c6035b9b154a63044d77d8711b72ba -e
JENKINS_WORKDIR=/home/non-root/jenkins --network typhoon_network --rm -d typhoonhil/thcc-hub-agent:2023.4
```

Under the Execution tab, you will also see all Executions which were performed with that specific Agent. Under the Logs tab, you can view the Logs corresponding to that Agent.

Every new THCC release will be presented as a new Image. Upgrading your tests to a new version of THCC is as easy as selecting a different version under the drop-down menu.



Status	Name	Computer	Image	Actions
●	My New Agent Name	SilentPC	Official Typhoon Test Hub agent: 2023.4	▷
●	pipe01	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	MDrive	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	EPC	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	VHIL	SilentPC	Official Typhoon Test Hub agent: 2023.4	□
●	SWonly2	SilentPC	Official Typhoon Test Hub agent: 2023.4	□

My New Agent Name ● 11m 9s

Image	Official Typhoon Test Hub agent: 2023.4
Computer	
Workspace	
Access token	
Access to network	
Volumes	⊕
Access to host devices	None
Run as root user	
Restart	Never

Official Typhoon Test Hub agent: 2023.4
Official Typhoon Test Hub agent: 2023.3_sp1
Official Typhoon Test Hub agent: 2023.3
Official Typhoon Test Hub agent: 2023.2
Official Typhoon Test Hub agent: 2023.1

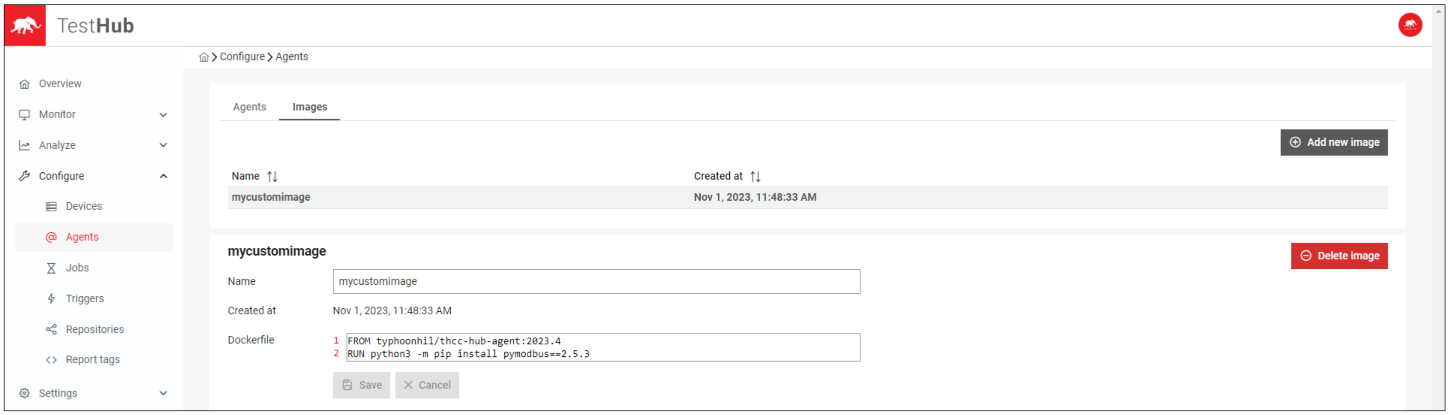
Java command

```
java -jar agent.jar -jnlpUrl http://192.168.50.109/jenkins/computer/Agent%207/jenkins-agent.jnlp -secret f479202a0c33b56c97bd8e0b618cd35596c6035b9b154a63044d77d8711b72ba -workDir "/home/non-root/jenkins"
```

Docker command

```
docker run --name Agent-7 -v /var/typhoon/controller/license:/home/non-root/.local/share/typhoon/license -e TTH_TOKEN=9OhT4vICt22YdPrbh8a8G0BydVYQuor8 -e TTH_URL=http://192.168.50.109 -e
JENKINS_IP=http://192.168.50.109/jenkins/computer/Agent%207/jenkins-agent.jnlp -e JENKINS_SECRET=f479202a0c33b56c97bd8e0b618cd35596c6035b9b154a63044d77d8711b72ba -e
JENKINS_WORKDIR=/home/non-root/jenkins --network typhoon_network --rm -d typhoonhil/thcc-hub-agent:2023.4
```

Use the Images tab in case it is necessary to make changes to the base THCC image or add other custom images. After any changes are made, the new image will appear as an available option under Agents with the given name.



Name	Created at
mycustomimage	Nov 1, 2023, 11:48:33 AM

mycustomimage

Name: mycustomimage

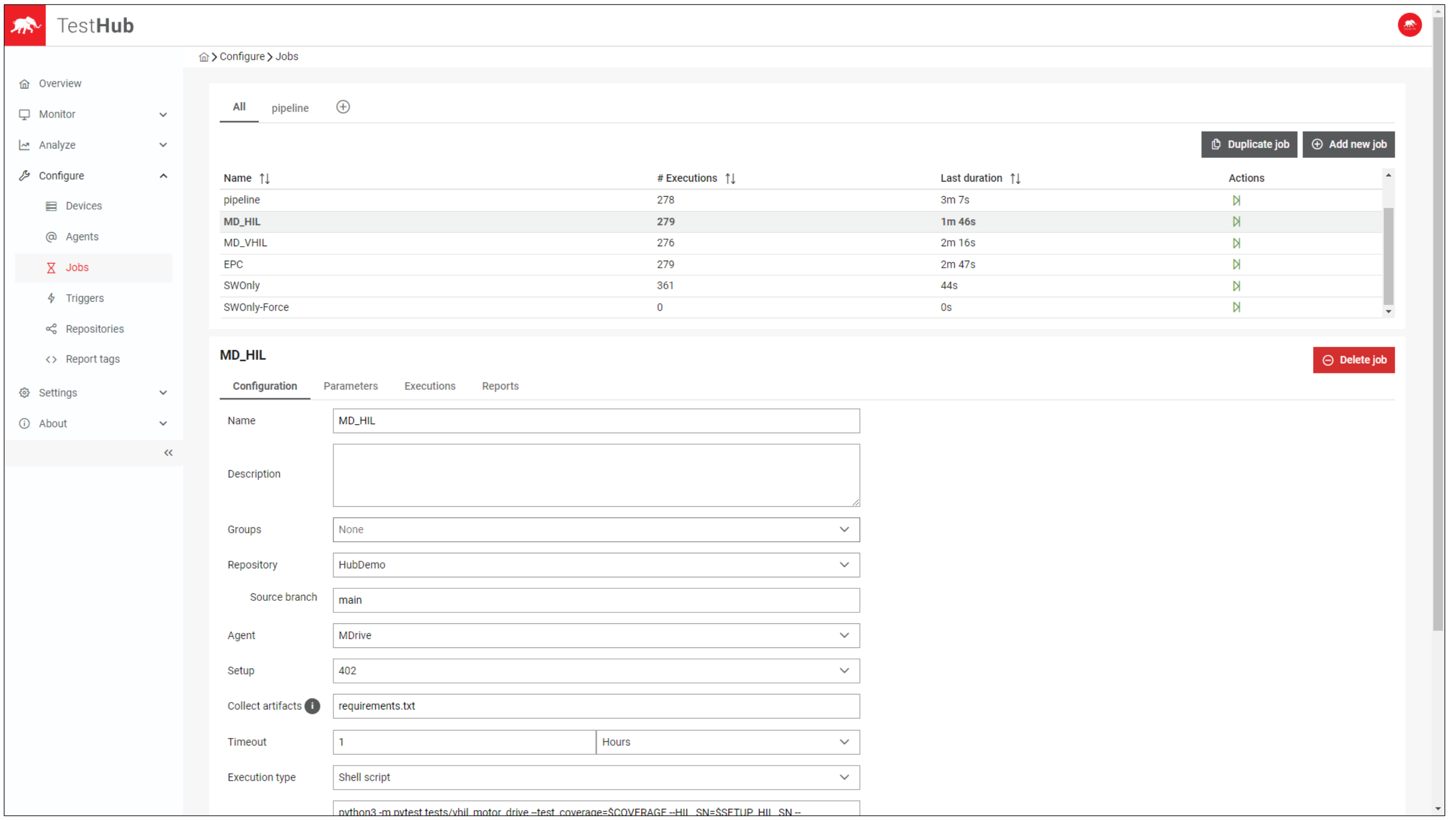
Created at: Nov 1, 2023, 11:48:33 AM

Dockerfile:

```
FROM typhoonhil/thcc-hub-agent:2023.4
RUN python3 -m pip install pymodbus==2.5.3
```

Jobs

This is where you can configure what should be executed, where, and which resources are available. Jobs can be grouped under custom categories. When clicking on a Job, you can configure it, create parameters, and see previous executions and reports.



Name	# Executions	Last duration	Actions
pipeline	278	3m 7s	▷
MD_HIL	279	1m 46s	▷
MD_VHIL	276	2m 16s	▷
EPC	279	2m 47s	▷
SWOnly	361	44s	▷
SWOnly-Force	0	0s	▷

MD_HIL

Name: MD_HIL

Description:

Groups: None

Repository: HubDemo

Source branch: main

Agent: MDrive

Setup: 402

Collect artifacts: requirements.txt

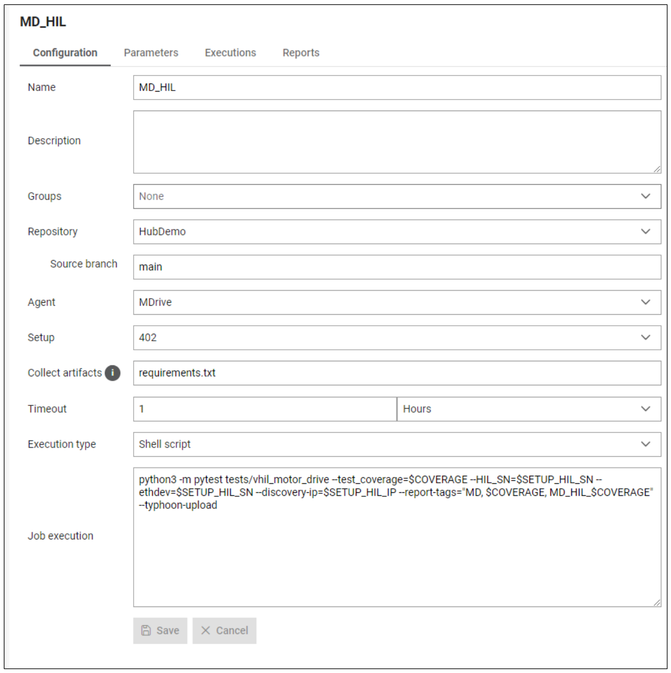
Timeout: 1 Hours

Execution type: Shell script

```
python3 -m pytest tests/vhil_motor_drive --test_coverage=$COVERAGE --HIL_SN=$SETUP_HIL_SN --
```

Under the Configuration tab, you can define what a Job should do. To define a job, complete the following steps:

- (optional) Specify the Repository and the branch - the branch can also be parametrized, allowing for easy selection;
- Select the Agent that will be used to execute the Job;
- (optional) Select the Setup that contains the necessary Devices;
- (optional) Define the Artifacts that should be collected once Execution completes;
- (optional) Specify the Timeout period for halting Execution if it takes longer than expected;
- Define the command line which should be executed.



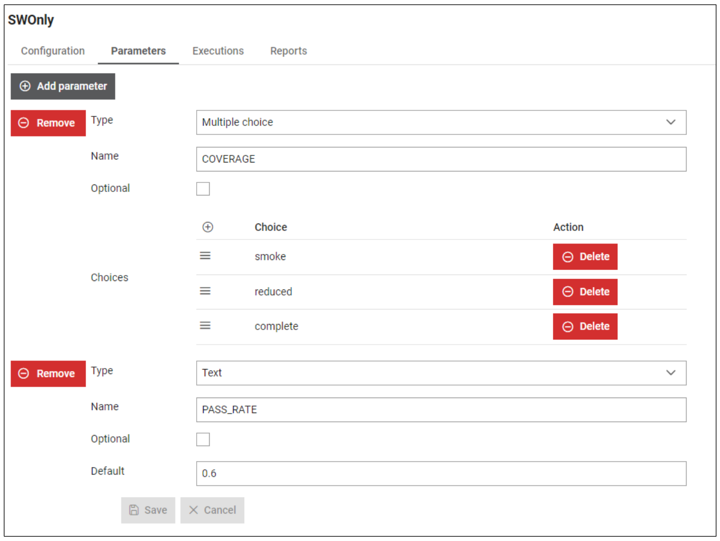
MD_HIL

Name	MD_HIL
Description	
Groups	None
Repository	HubDemo
Source branch	main
Agent	MDrive
Setup	402
Collect artifacts	requirements.txt
Timeout	1 Hours
Execution type	Shell script

Job execution

```
python3 -m pytest tests/vhil_motor_drive --test_coverage=$COVERAGE --HIL_SN=$SETUP_HIL_SN --
ethdev=$SETUP_HIL_SN --discovery-ip=$SETUP_HIL_IP --report-tags="MD, $COVERAGE, MD_HIL_$COVERAGE"
--typhoon-upload
```

You can change how a Job behaves by parametrizing it. Under the Parameters tab, it is possible to add parameters and their values, which will be used during the Execution. You can also start the Execution directly from this page, by pressing the Play button next to the Job. If the Job is parametrized, it is possible to inform the values of the parameters before the Execution goes into the Queue.



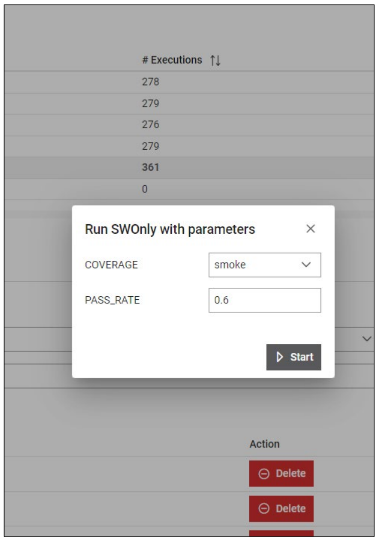
SWOnly

Type	Multiple choice
Name	COVERAGE
Optional	□

Choices

Choice	Action
smoke	Delete
reduced	Delete
complete	Delete

Type	Text
Name	PASS_RATE
Optional	□
Default	0.6

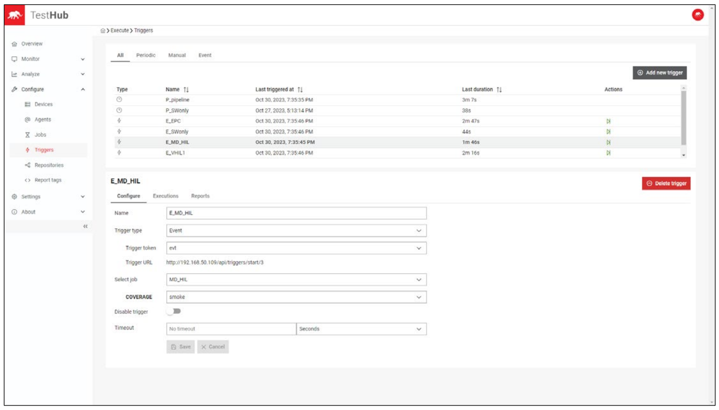


Run SWOnly with parameters

COVERAGE	smoke
PASS_RATE	0.6

Trigger

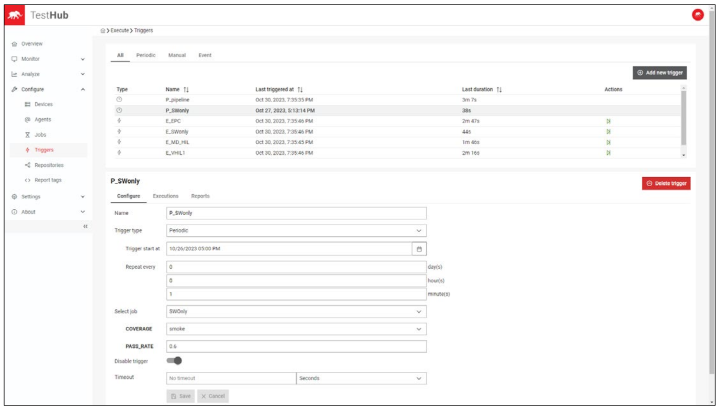
Defining when a Job should be executed is at the core of Test Automation, as this maintains a consistent order of Executions in the Queue. In Typhoon Test Hub, it is possible to create Manual, Periodic, or Event Triggers. They all have the same behavior but differ in the way they are started. With Triggers, you can select which Job should be executed and its parametrization.



Type	Name	Last triggered at	Last duration	Actions
🕓	P_pipeline	Oct 30, 2023, 7:35:35 PM	3m 7s	
🕓	P_SWonly	Oct 27, 2023, 5:13:14 PM	38s	
⚡	E_EPC	Oct 30, 2023, 7:35:46 PM	2m 47s	▷
⚡	E_SWonly	Oct 30, 2023, 7:35:46 PM	44s	▷
⚡	E_MD_HIL	Oct 30, 2023, 7:35:45 PM	1m 46s	▷
⚡	E_VHIL1	Oct 30, 2023, 7:35:46 PM	2m 16s	▷

E_MD_HIL

Name: E_MD_HIL
Trigger type: Event
Trigger token: evt
Trigger URL: http://192.168.50.109/api/triggers/start/3
Select job: MD_HIL
COVERAGE: smoke
Disable trigger
Timeout: No timeout | Seconds



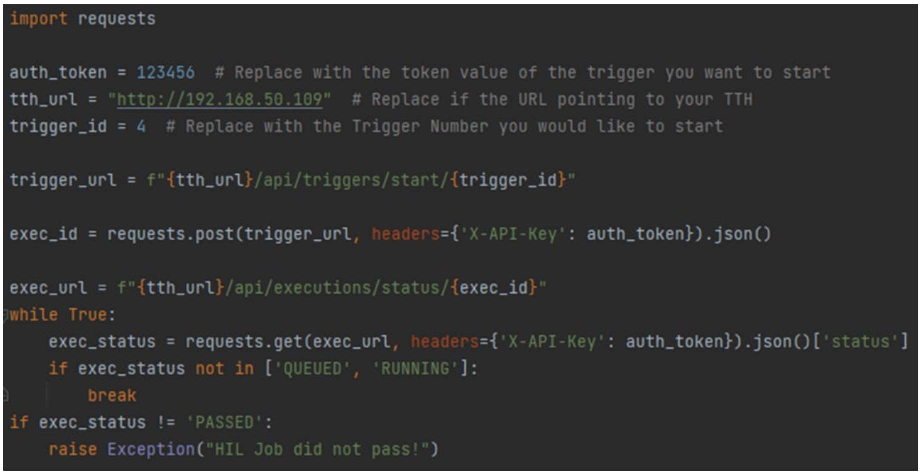
Type	Name	Last triggered at	Last duration	Actions
🕓	P_pipeline	Oct 30, 2023, 7:35:35 PM	3m 7s	
🕓	P_SWonly	Oct 27, 2023, 5:13:14 PM	38s	
⚡	E_EPC	Oct 30, 2023, 7:35:46 PM	2m 47s	▷
⚡	E_SWonly	Oct 30, 2023, 7:35:46 PM	44s	▷
⚡	E_MD_HIL	Oct 30, 2023, 7:35:45 PM	1m 46s	▷
⚡	E_VHIL1	Oct 30, 2023, 7:35:46 PM	2m 16s	▷

P_SWonly

Name: P_SWonly
Trigger type: Periodic
Trigger start at: 10/26/2023 05:00 PM
Repeat every: 0 day(s), 0 hour(s), 1 minute(s)
Select job: SWOnly
COVERAGE: smoke
PASS_RATE: 0.6
Disable trigger
Timeout: No timeout | Seconds

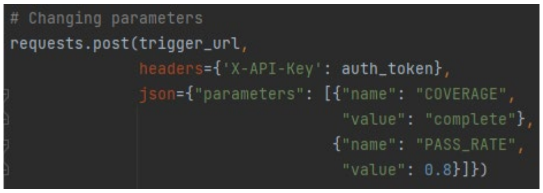
Manual Triggers are mostly used for quick and customized test executions. Periodic Triggers run at a pre-defined interval – every day at 10pm, for example. Event Triggers create a trigger URL, which is used to start the execution externally.

Event Triggers are the most flexible and allow integration with several other applications. When an Event Trigger starts, it returns its Execution ID, which can be used to track if the execution is Queued, Running, Passed, or Failed. This information can be used to approve merge requests, for example. Here is a simple Python code snippet showing how you can trigger an execution, wait for it to be done, and raise an exception in case it fails:



```
import requests

auth_token = 123456  # Replace with the token value of the trigger you want to start
tth_url = "http://192.168.50.109"  # Replace if the URL pointing to your TTH
trigger_id = 4  # Replace with the Trigger Number you would like to start

trigger_url = f"{tth_url}/api/triggers/start/{trigger_id}"

exec_id = requests.post(trigger_url, headers={'X-API-Key': auth_token}).json()

exec_url = f"{tth_url}/api/executions/status/{exec_id}"
while True:
    exec_status = requests.get(exec_url, headers={'X-API-Key': auth_token}).json()['status']
    if exec_status not in ['QUEUED', 'RUNNING']:
        break
if exec_status != 'PASSED':
    raise Exception("HIL Job did not pass!")
```

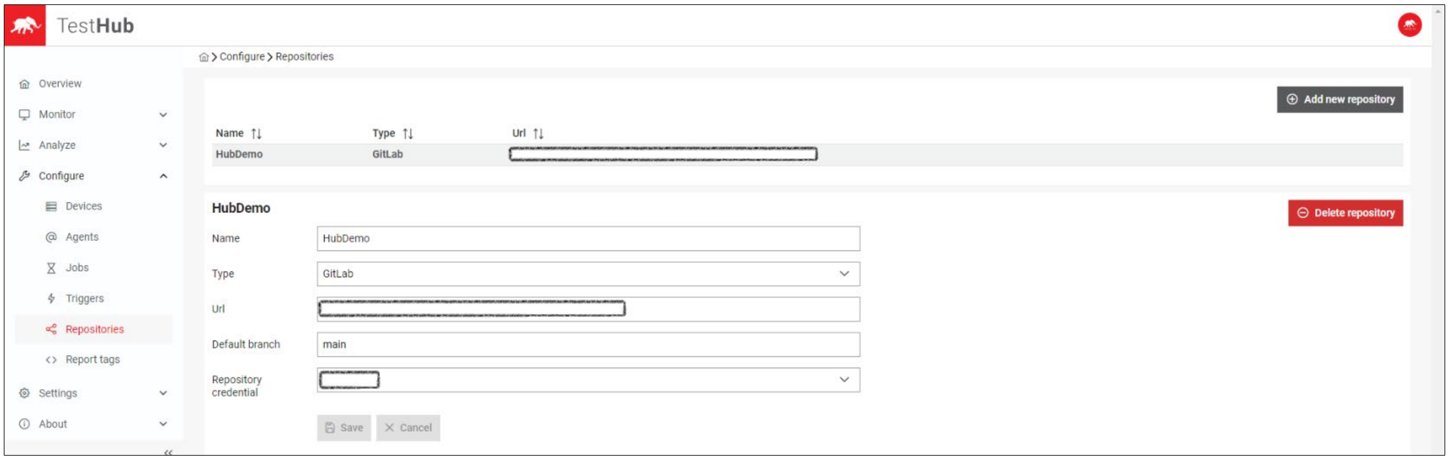
It is also possible to change the Trigger Job Parametrization when starting it, by making this change to the command:



```
# Changing parameters
requests.post(trigger_url,
              headers={'X-API-Key': auth_token},
              json={"parameters": [{"name": "COVERAGE",
                                    "value": "complete"},
                                   {"name": "PASS_RATE",
                                    "value": 0.8}]})
```

Repositories

You can configure multiple repositories to be easily accessible by your Jobs. Just provide the repository URL and pick which Credential will be used to connect to it. Once completed, it is available to be used by a Job.



Configure > Repositories

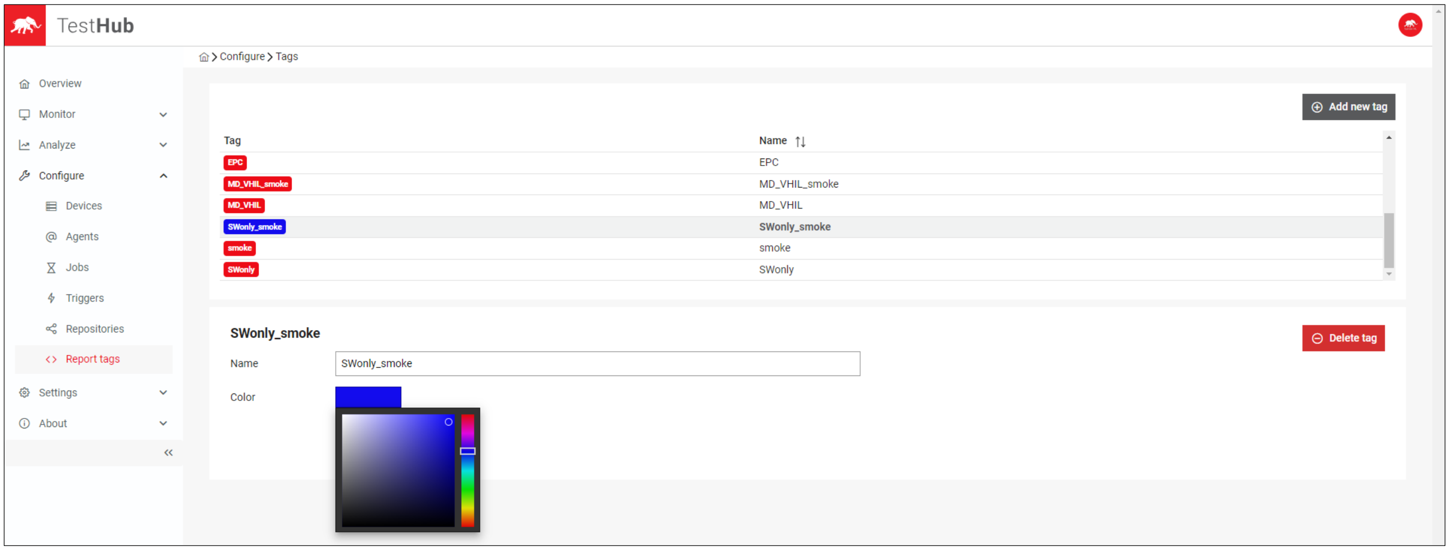
Name	Type	Url
HubDemo	GitLab	[redacted]

HubDemo

Name	HubDemo
Type	GitLab
Url	[redacted]
Default branch	main
Repository credential	[redacted]

Report Tags

Report Tags can be defined before Execution or can be added after Execution completes. Report Tags are used for filtering out results so they can be displayed in the Dashboard and Overview page. You can change their name or color under Configure/Report Tags.



Configure > Tags

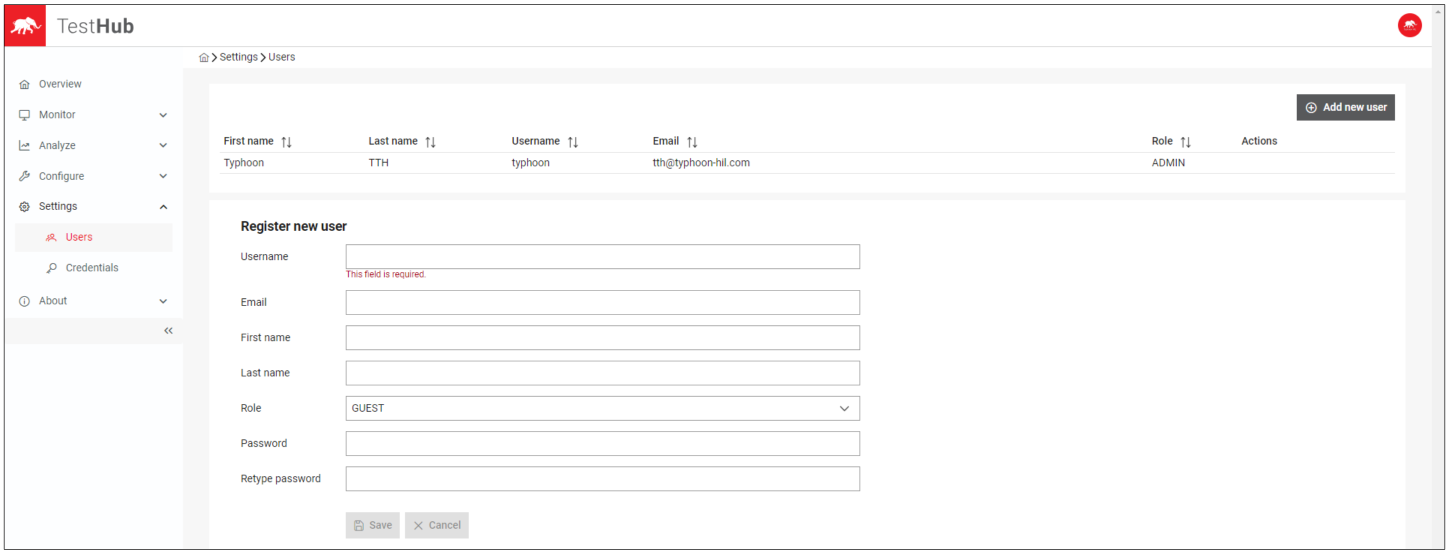
Tag	Name
EPC	EPC
MD_VHIL_smoke	MD_VHIL_smoke
MD_VHIL	MD_VHIL
SWonly_smoke	SWonly_smoke
smoke	smoke
SWonly	SWonly

SWonly_smoke

Name	SWonly_smoke
Color	🟦

Users

You can add unlimited users accounts to access TTH, free of charge. You can define their role, limiting their access and the actions they can perform in TTH. This is done with the goal of making results easily accessible company wide.



First name	Last name	Username	Email	Role	Actions
Typhoon	TTH	typhoon	tth@typhoon-hil.com	ADMIN	

Register new user

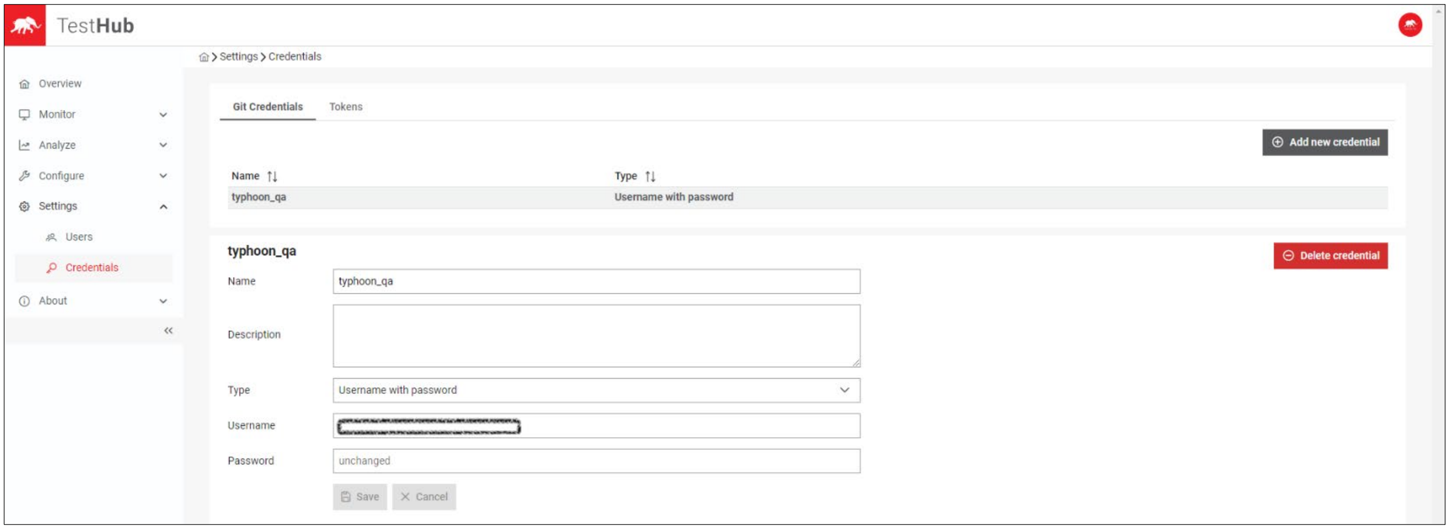
Username	This field is required.
Email	
First name	
Last name	
Role	GUEST
Password	
Retype password	

Credentials

You can add sensitive information here to refer to it within the Hub, while keeping it encrypted and safe.

Git Credentials

In order to interact with Git repositories, it is necessary to provide Git credentials. In Configure/Repository you can store all Git credentials and pick which credential to use when creating a new repository.



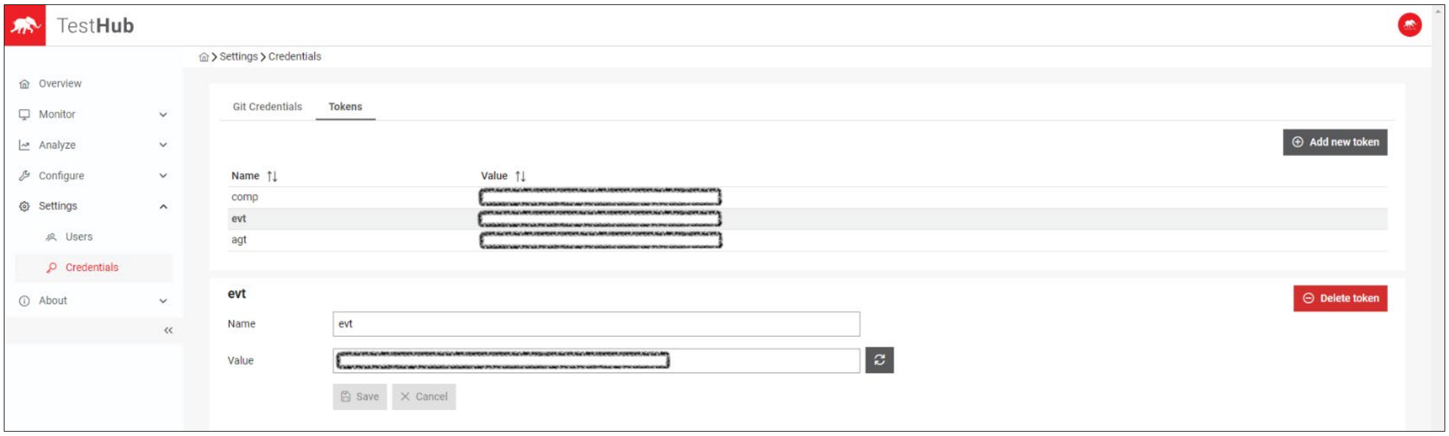
Name	Type
typhoon_qa	Username with password

typhoon_qa

Name	typhoon_qa
Description	
Type	Username with password
Username	[redacted]
Password	unchanged

Tokens

Tokens are used to validate communication internally within TTH. Here you can create new tokens, as well as manage existing ones.



Name	Value
comp	[redacted]
evt	[redacted]
agt	[redacted]

evt

Name: evt

Value: [redacted]

Final remarks

Typhoon Test Hub is a tool designed to simplify integration of automated HIL tests and increase the value of the generated results. The goal is to have you spending more time adding new features to your product and less time maintaining the necessary infrastructure to make sure the product is performing as it should. This means shorter development cycles, a product with better quality, and easier product life cycle maintenance.

For additional information, or to request a demonstration, please contact henrique.magnago@typhoon-hil.com.

Disclaimer: The contents of this document were created for the sole purpose of informing prospects, partners, and potential customers of Typhoon HIL Inc. As such, all the contents contained herein are confidential property of Typhoon HIL, Inc. and cannot be shared with any person or organization in any shape or form without the prior consent of Typhoon HIL, Inc.