IT Automation

Conductor【Practice】

※In this document, “Exastro IT Automation” is described as “ITA".

Exastro IT Automation Version 1.7.1

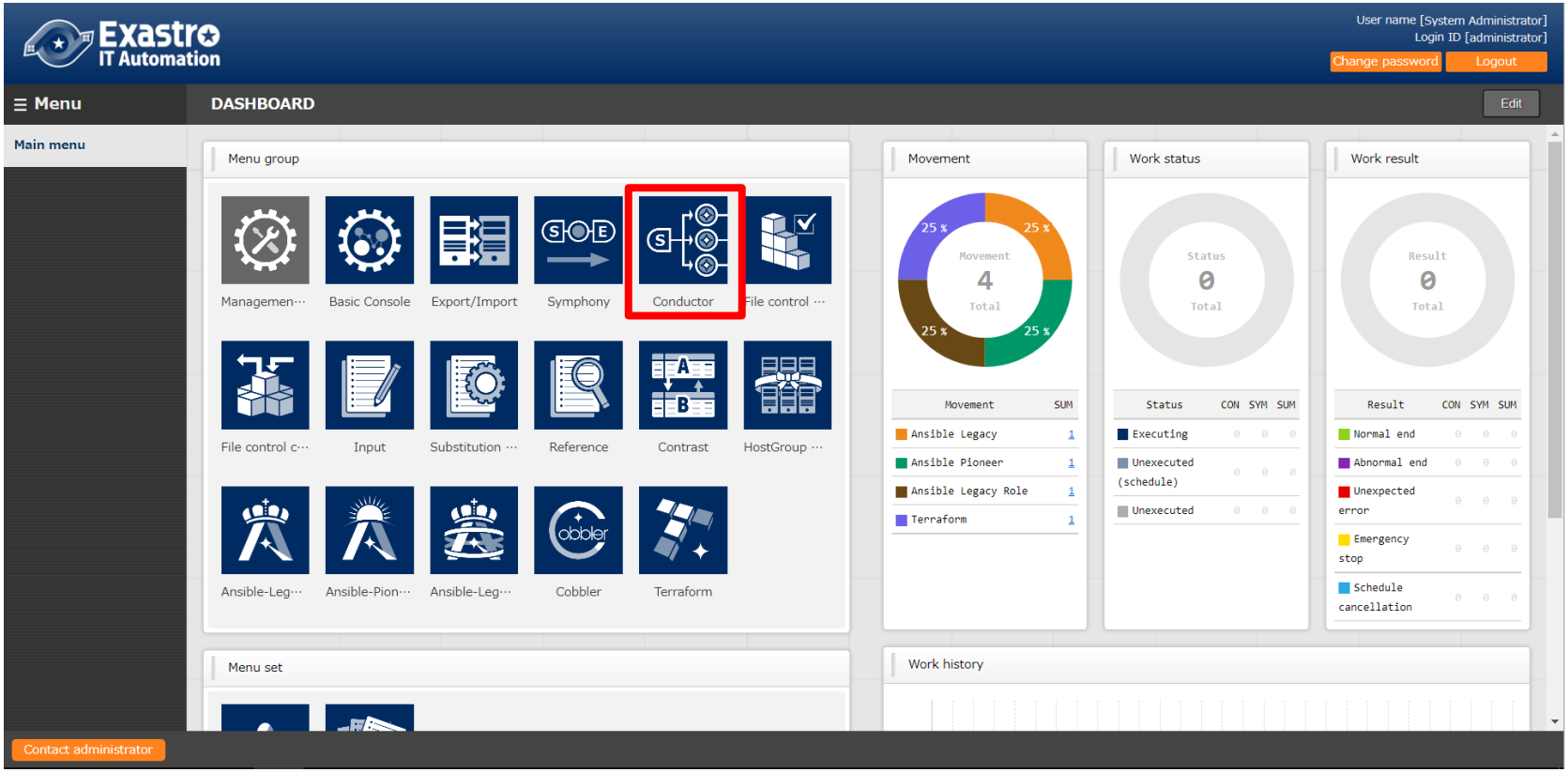
Exastro developer

Table of contents

1. Introduction

1.1 About this document

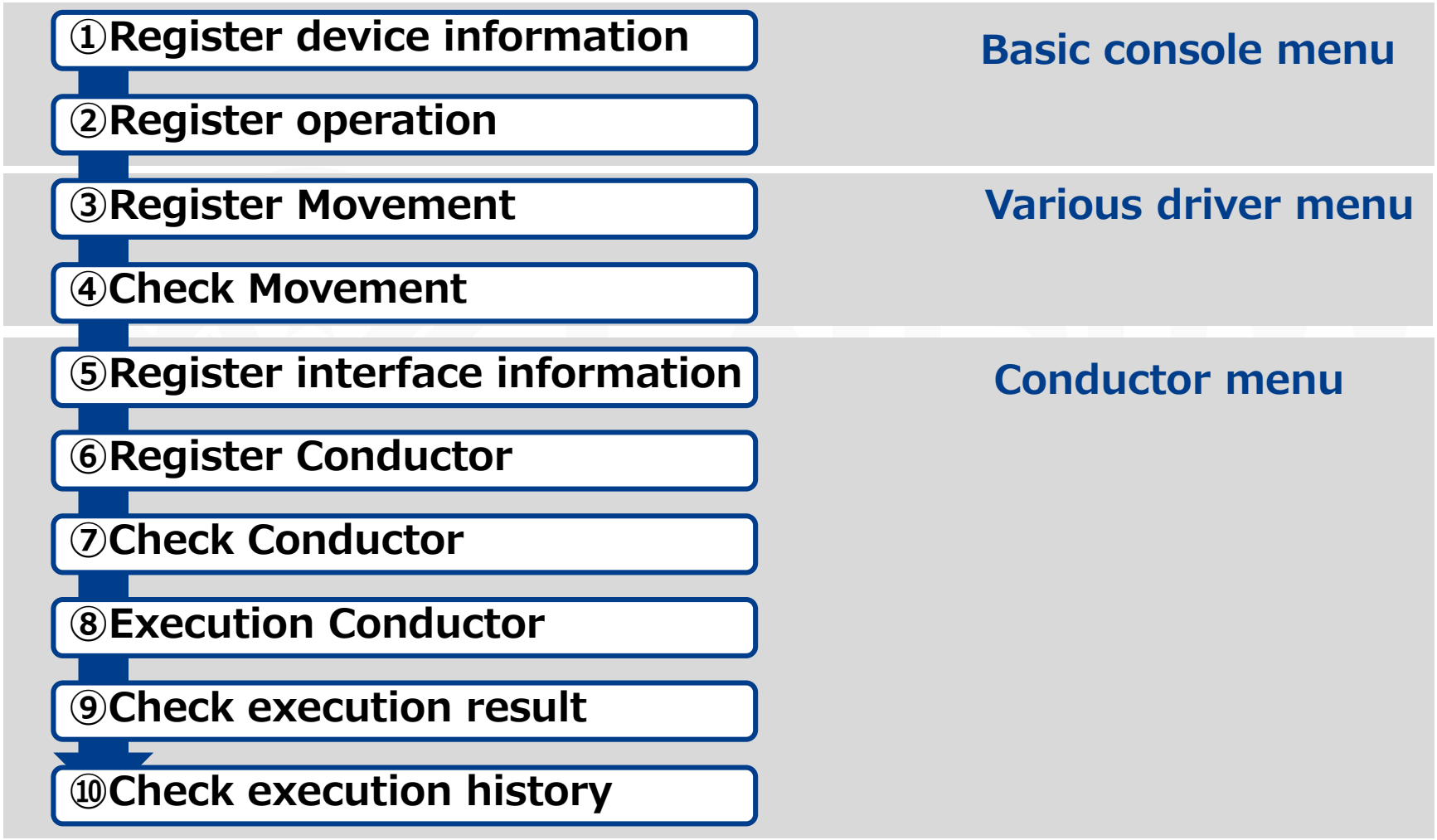
- The "**Conductor**" menu group is explained in this document.



2. About Conductor

2.1 Scenario (1/2)

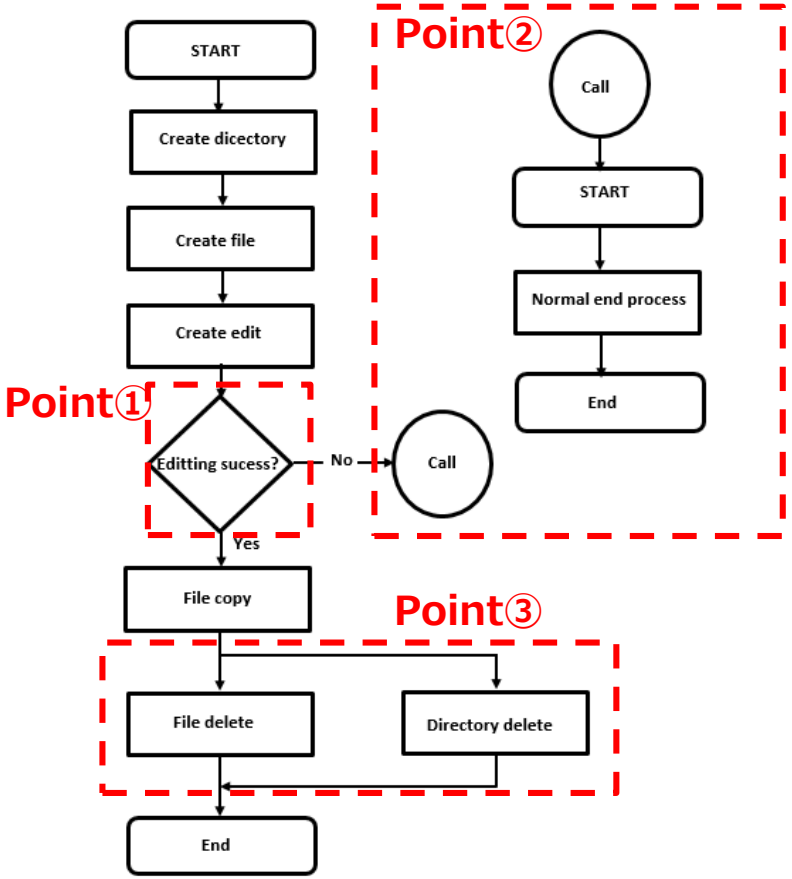
- The Scenario used in this document is as following.
- Ansible driver is required to proceed with the scenario, so in this scenario, we will explain using Ansible-Legacy.



2.1 Scenario (2/2)

- In order to experience the Conductor functions in this document, we will create a Conductor similar to the following flowchart.

- **Flowchart**



Features of Conductor function

Point①

Conditional branch function according to the success/end judgment of pre-processing.

Point②

Registered Operation/Conductor Call function.

Point③

Movement parallel processing function.

Create IaC(1/2)

In this scenario, Ansible-Legacy is explained with the use of an example.

Save the following IaC as an yml file for each module.

※Character code is "UTF-8", Newline code is "LF", file name extension is "yml"format.
Please be careful about indents.

```
- name: create directory
  file:
    path=/tmp/{{ VAR_dir_name_1 }}
    state=directory
    mode=0755

- name: remove directory
  file:
    path=/tmp/{{VAR_dir_name_1 }}
    state=absent

- name: create file
  file:
    path=/tmp/{{VAR_dir_name_1 }}/{{VAR_file_name }}
    state=touch
    mode=0755
```

●Create IaC(2/2)

Similarly, save the following IaC as yml files for each module.

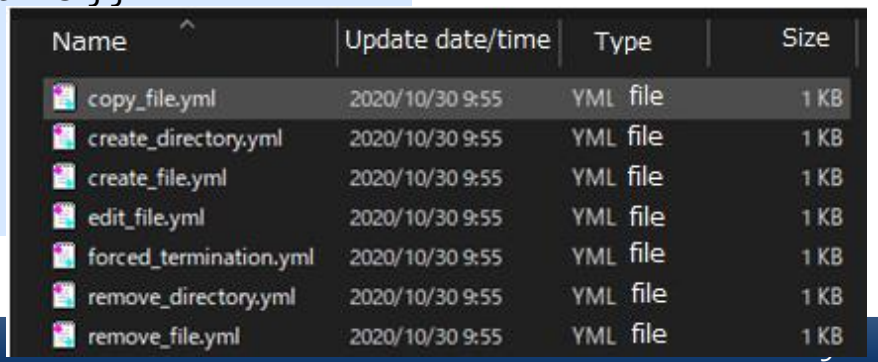
```
- name: remove file
  file:
    path=/tmp/{{VAR_dir_name_1 }}/{{VAR_file_name }}
    state=absent

- name: copy file
  copy:
    src=/tmp/{{VAR_dir_name_1 }}/{{VAR_file_name }}
    dest=/tmp/{{VAR_dir_name_2 }}/{{VAR_file_name }}
    owner=root
    group=root
    mode=0644

- name: edit file
  copy:
    dest=/tmp/{{VAR_dir_name_1 }}/{{VAR_file_name }}
    content= {{VAR_edit_param_1 }}

- name: forced termination
  fail: msg={{VAR_message_text }}
```

● Created image



Name	Update date/time	Type	Size
copy_file.yml	2020/10/30 9:55	YML file	1 KB
create_directory.yml	2020/10/30 9:55	YML file	1 KB
create_file.yml	2020/10/30 9:55	YML file	1 KB
edit_file.yml	2020/10/30 9:55	YML file	1 KB
forced_termination.yml	2020/10/30 9:55	YML file	1 KB
remove_directory.yml	2020/10/30 9:55	YML file	1 KB
remove_file.yml	2020/10/30 9:55	YML file	1 KB

3. Operation

●Register Target host

"Basic Console“ menu group >> "Device list“ menu >> "Register“ submenu >>"Start register“ button.

① Enter the following "Host name", "IP address", "Login ID", “Management", "Login password“ and "Authentication method“.

② Click "Register" button.

Enter value for the item

Item	Value
Host name	testserver
IP address	(Free value)
Login ID	(Free value)
Management	●
Login password	(Free value)
Authentication method	Password authentication

POINT

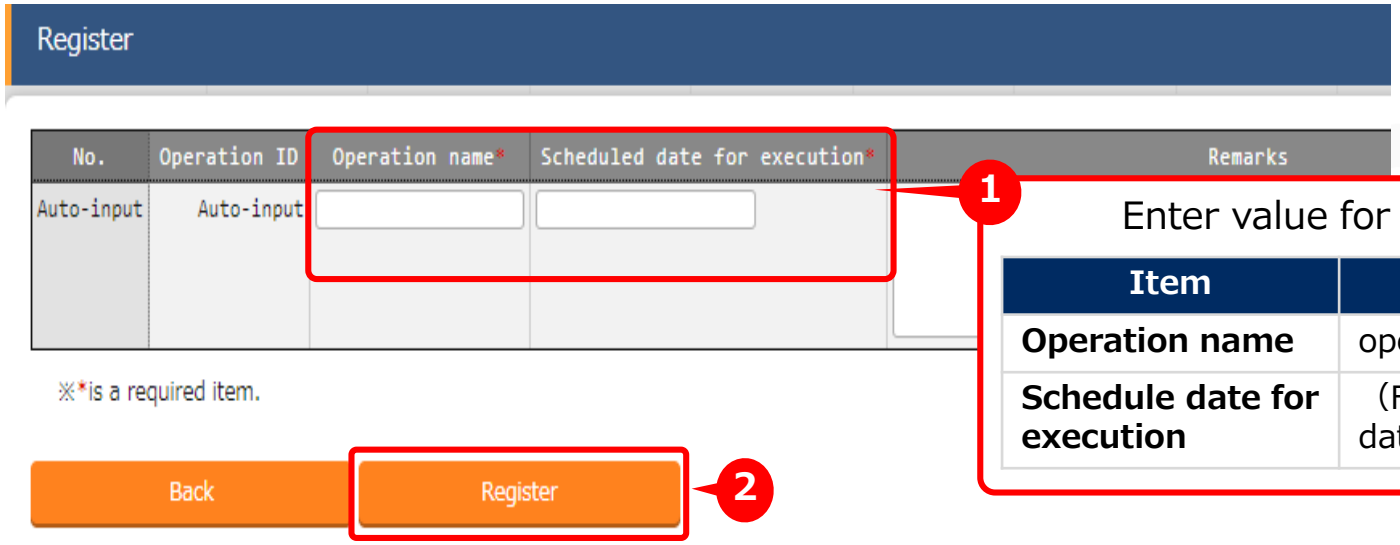
This scenario assumes that you want to have a ssh password connection to the Target host.
Please enter “IP Address”, “Login ID", "Login password”
appropriate to the settings set in the users environment.

●Register operation

"Basic Console“ menu group >> "Input list“ menu >> "Register“ submenu >>"Start register“ button.

① Enter "Operation name“, "Schedule date for execution“.

② Click “Registration” button.



Enter value for item

Item	Value
Operation name	operation1
Schedule date for execution	(Free date/time)

POINT

The process will not be executed at the date and time specified here.

3.3 Register IaC (1/2)

●Register IaC

"Ansible-Legacy" menu group >> "Playbook files" menu >> "Register" submenu >> "Start register" button.

① Input “Playbook file name“.
Click the ”Reference” button in the “Playbook files” column.
Upload all previously created yml files.
(Click "upload in Advance" button)

② Click “Registration” button.

Enter value for item

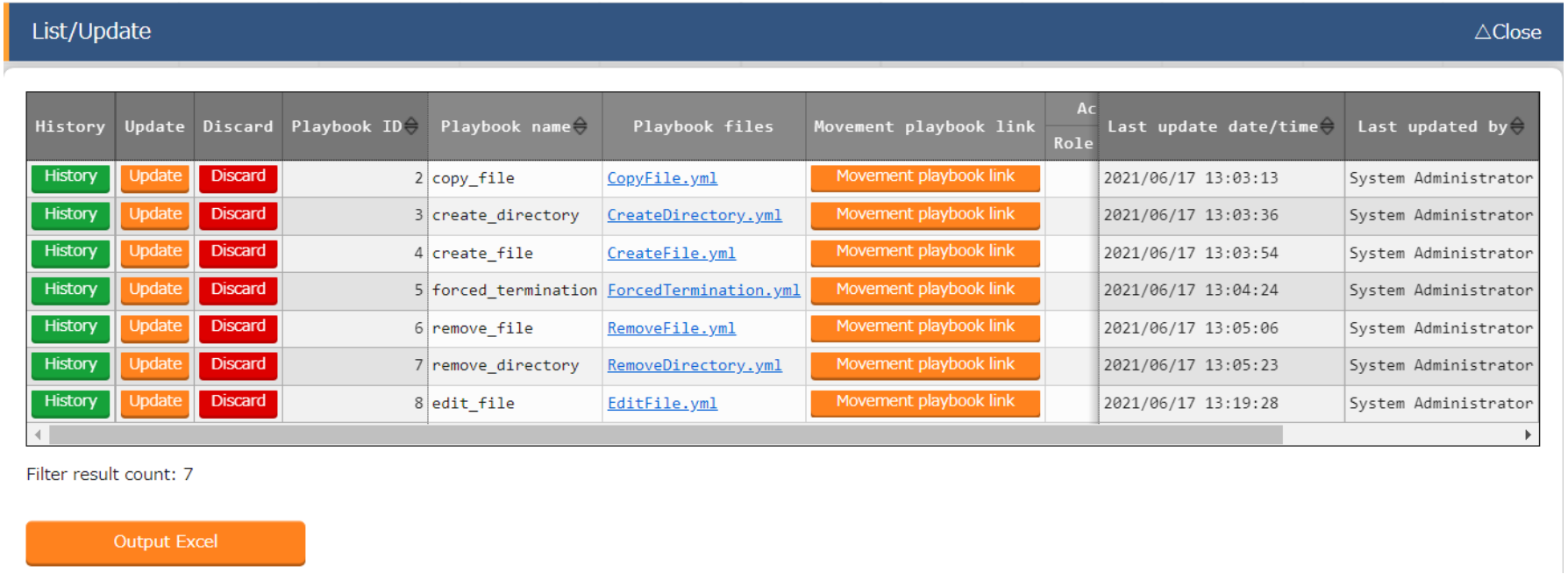
Item	Value
Playbook file name	<free>
Playbook file	<free> .yml

POINT

Please refer to "2.2 Pre-preparation“.
For more information on how to create IaC.

3.3 Register IaC (2/2)

- **Register IaC**
- When you're done registering, the list should be something like this:



List/Update △Close

History	Update	Discard	Playbook ID	Playbook name	Playbook files	Movement playbook link	Ac / Role	Last update date/time	Last updated by
History	Update	Discard	2	copy_file	CopyFile.yml	Movement playbook link		2021/06/17 13:03:13	System Administrator
History	Update	Discard	3	create_directory	CreateDirectory.yml	Movement playbook link		2021/06/17 13:03:36	System Administrator
History	Update	Discard	4	create_file	CreateFile.yml	Movement playbook link		2021/06/17 13:03:54	System Administrator
History	Update	Discard	5	forced_termination	ForcedTermination.yml	Movement playbook link		2021/06/17 13:04:24	System Administrator
History	Update	Discard	6	remove_file	RemoveFile.yml	Movement playbook link		2021/06/17 13:05:06	System Administrator
History	Update	Discard	7	remove_directory	RemoveDirectory.yml	Movement playbook link		2021/06/17 13:05:23	System Administrator
History	Update	Discard	8	edit_file	EditFile.yml	Movement playbook link		2021/06/17 13:19:28	System Administrator

Filter result count: 7

Output Excel

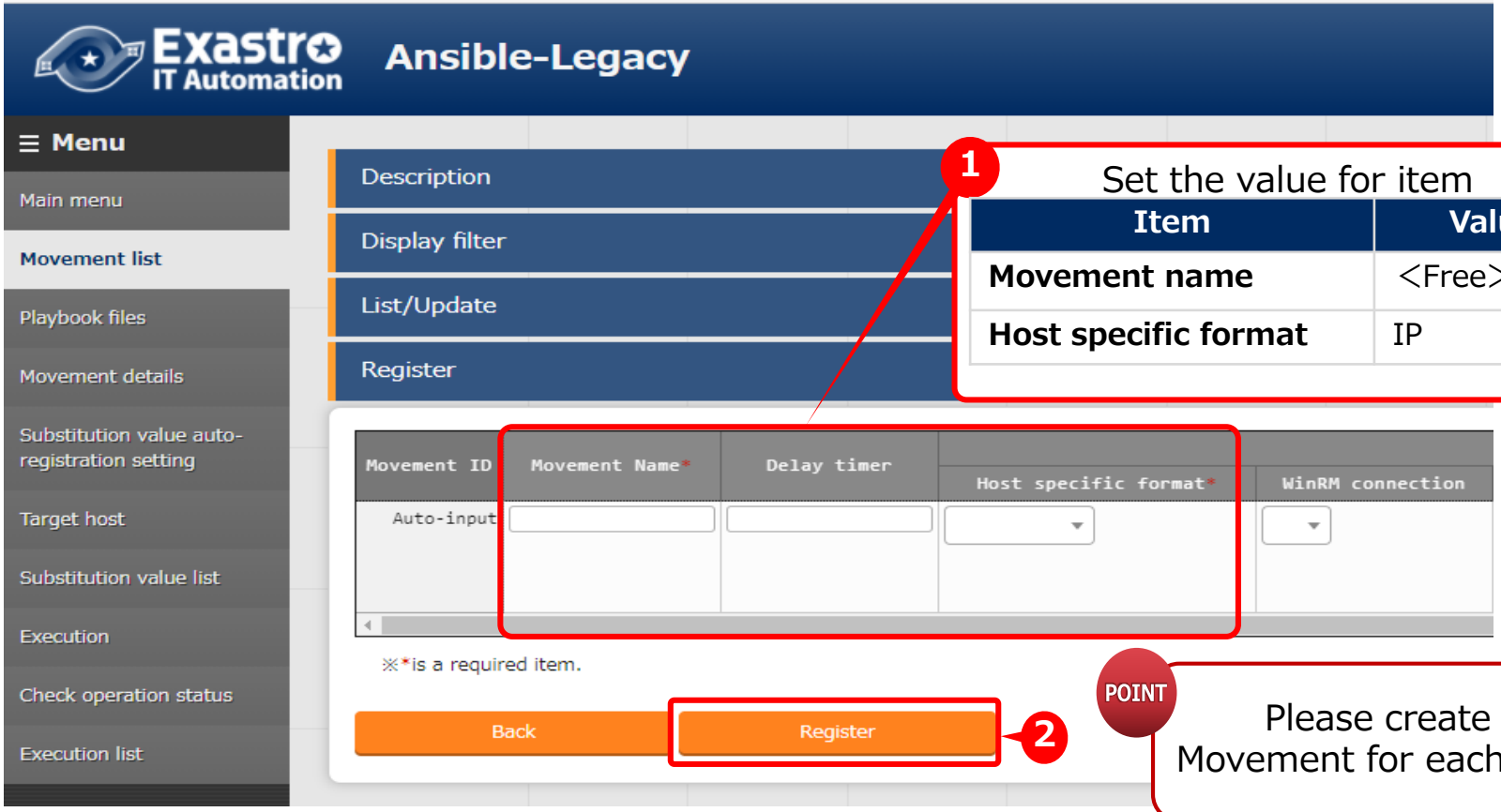
3.4 Register Movement (1/2)

●Register Movement

"Ansible-Legacy" menu group >> "Movement list" menu >> "Register" submenu >> "Start register" button.

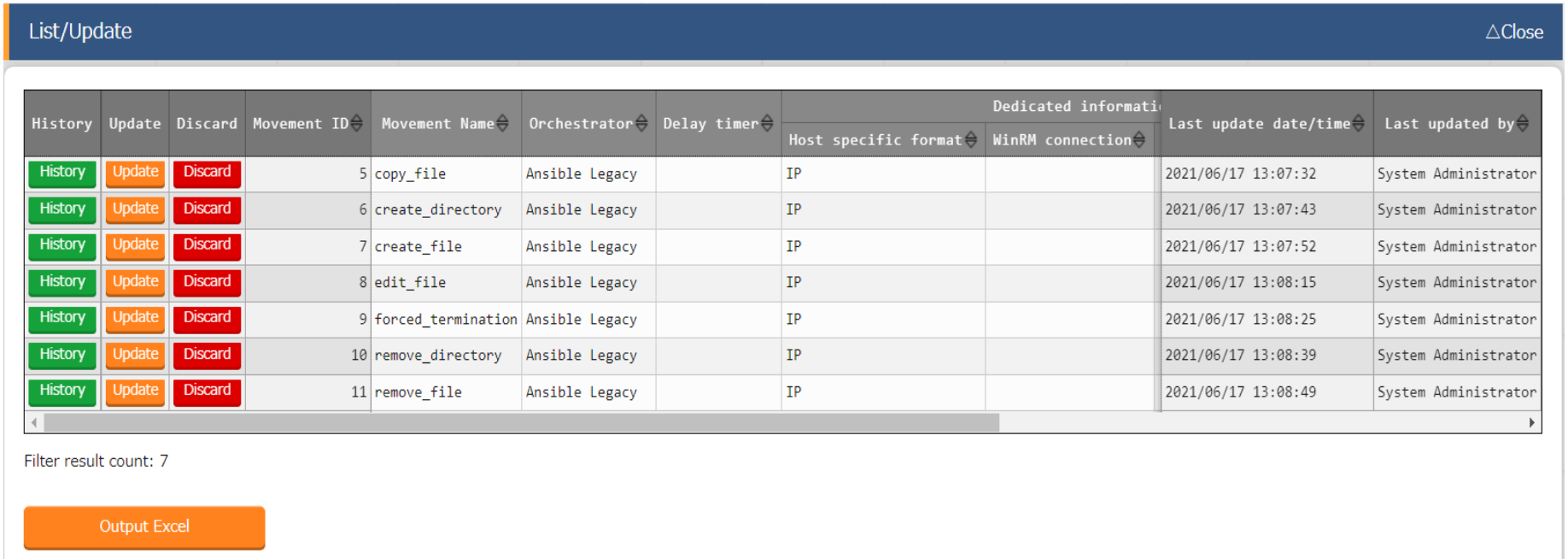
① Input "Movement name" and select ”Host specific format“.

② Click “Register” button.



3.4 Register Movement (2/2)

- **Register Movement**
- When you're done registering, the list should be something like this:

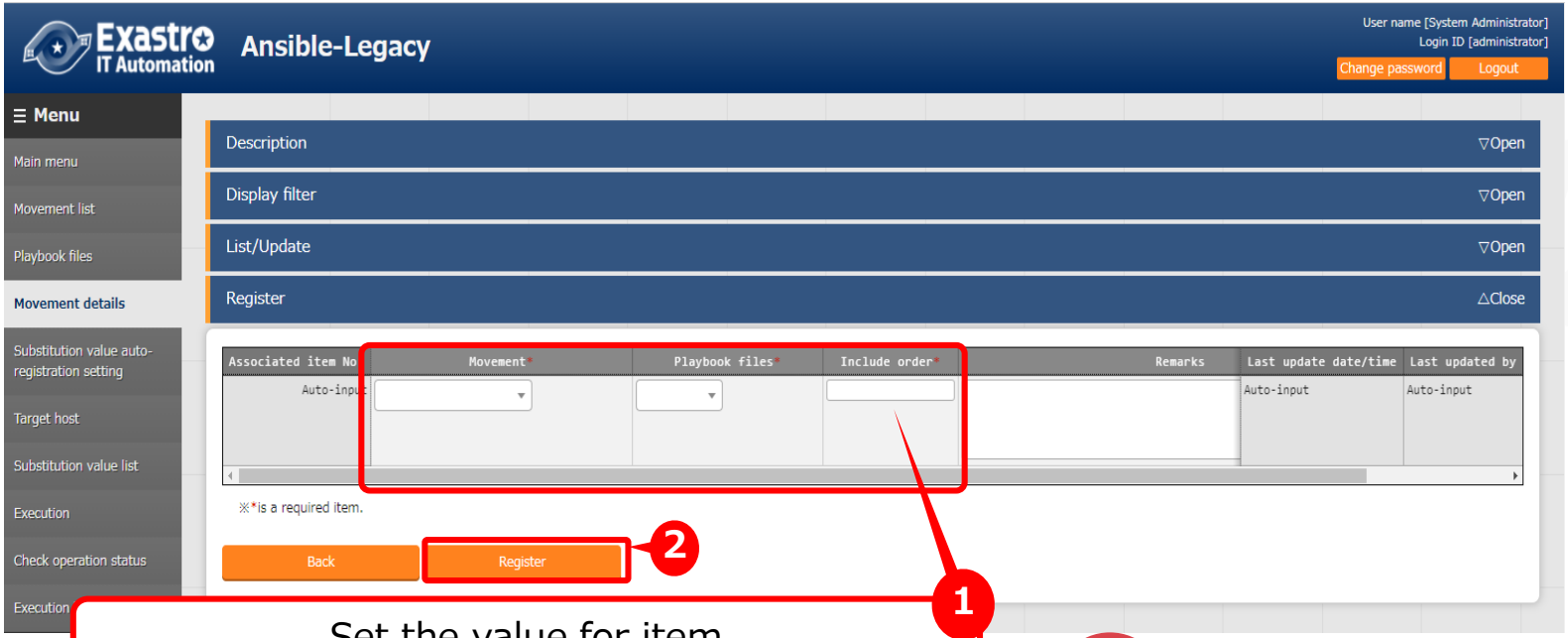


Register Movement details

"Ansible-Legacy" menu group >> “Movement playbook link" menu >> "Register" submenu >> "Start register" button.

① Input "Movement“ , "Playbook files" and "Include order"

② Click "Register" button.



Set the value for item

Item	Value
Movement	Select the create Movement
Playbook file	Select the register playbook
Include order	1

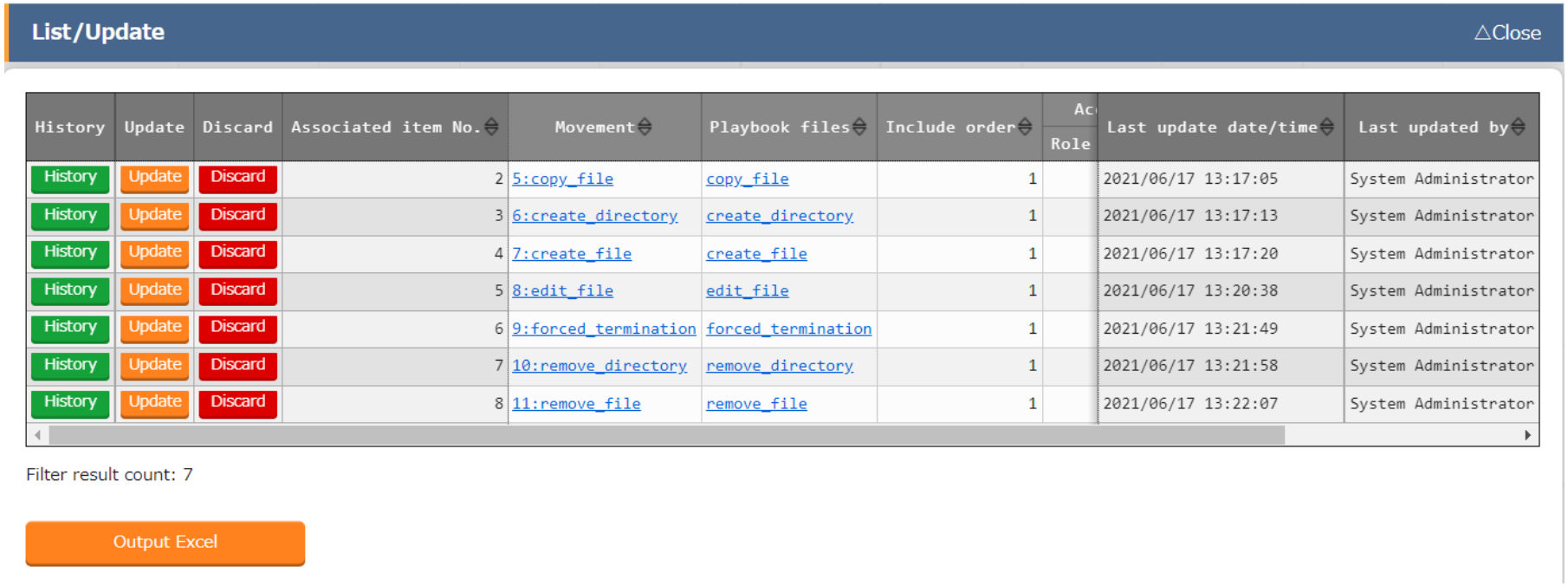
POINT

Please create the same number of yml files as registered movement information.

3.5 Register Movement details (2/2)

- **Register Movement details**

  When you're done registering, the list should be something like this:



List/Update △Close

History	Update	Discard	Associated item No.	Movement	Playbook files	Include order	Ac / Role	Last update date/time	Last updated by
History	Update	Discard	2	5:copy_file	copy_file	1		2021/06/17 13:17:05	System Administrator
History	Update	Discard	3	6:create_directory	create_directory	1		2021/06/17 13:17:13	System Administrator
History	Update	Discard	4	7:create_file	create_file	1		2021/06/17 13:17:20	System Administrator
History	Update	Discard	5	8:edit_file	edit_file	1		2021/06/17 13:20:38	System Administrator
History	Update	Discard	6	9:forced_termination	forced_termination	1		2021/06/17 13:21:49	System Administrator
History	Update	Discard	7	10:remove_directory	remove_directory	1		2021/06/17 13:21:58	System Administrator
History	Update	Discard	8	11:remove_file	remove_file	1		2021/06/17 13:22:07	System Administrator

Filter result count: 7

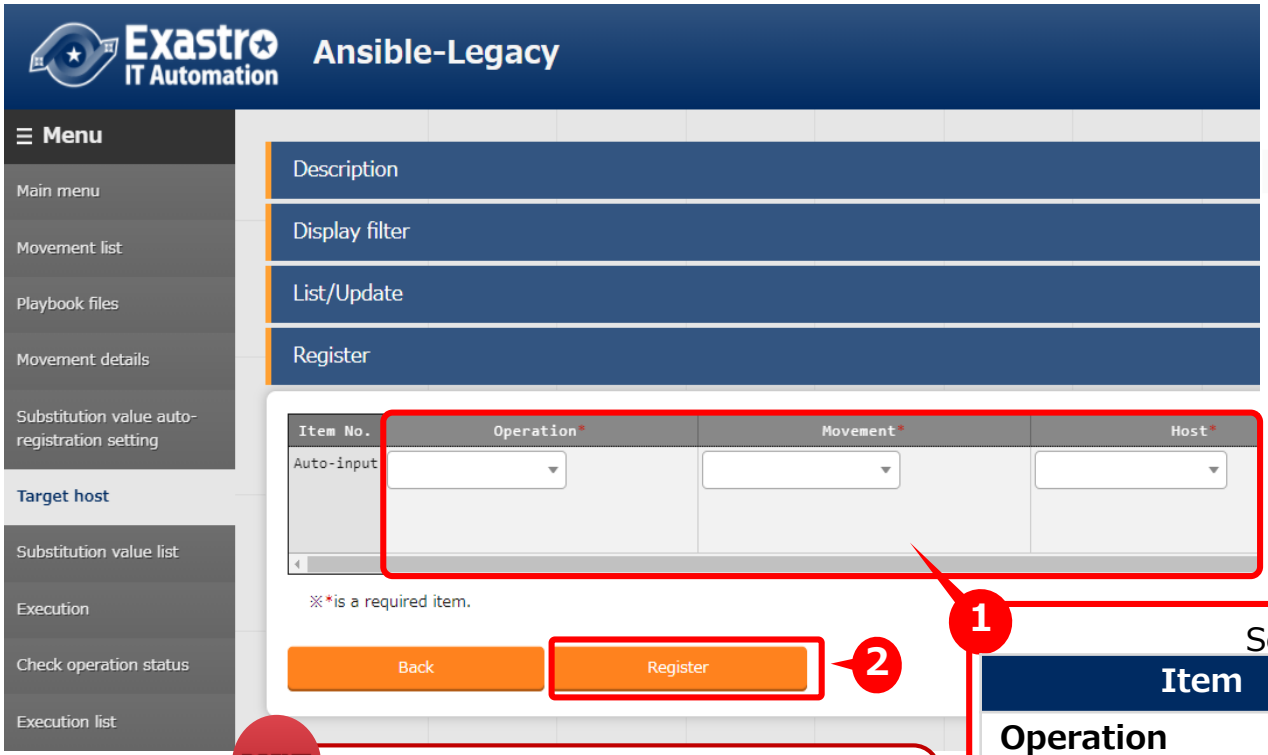
Output Excel

●Register Movement and Host connected to the operation

"Ansible-Legacy" menu group >> "Target host" menu >> "Register" submenu >> "Start register" button.

① Input "Operation“ , "Movement" and "Host".

② Click “Register” button.



Set value to item

Item	Value
Operation	operation1
Movement	Created Movement
Host	testserver

POINT

Please register all the created Movements.

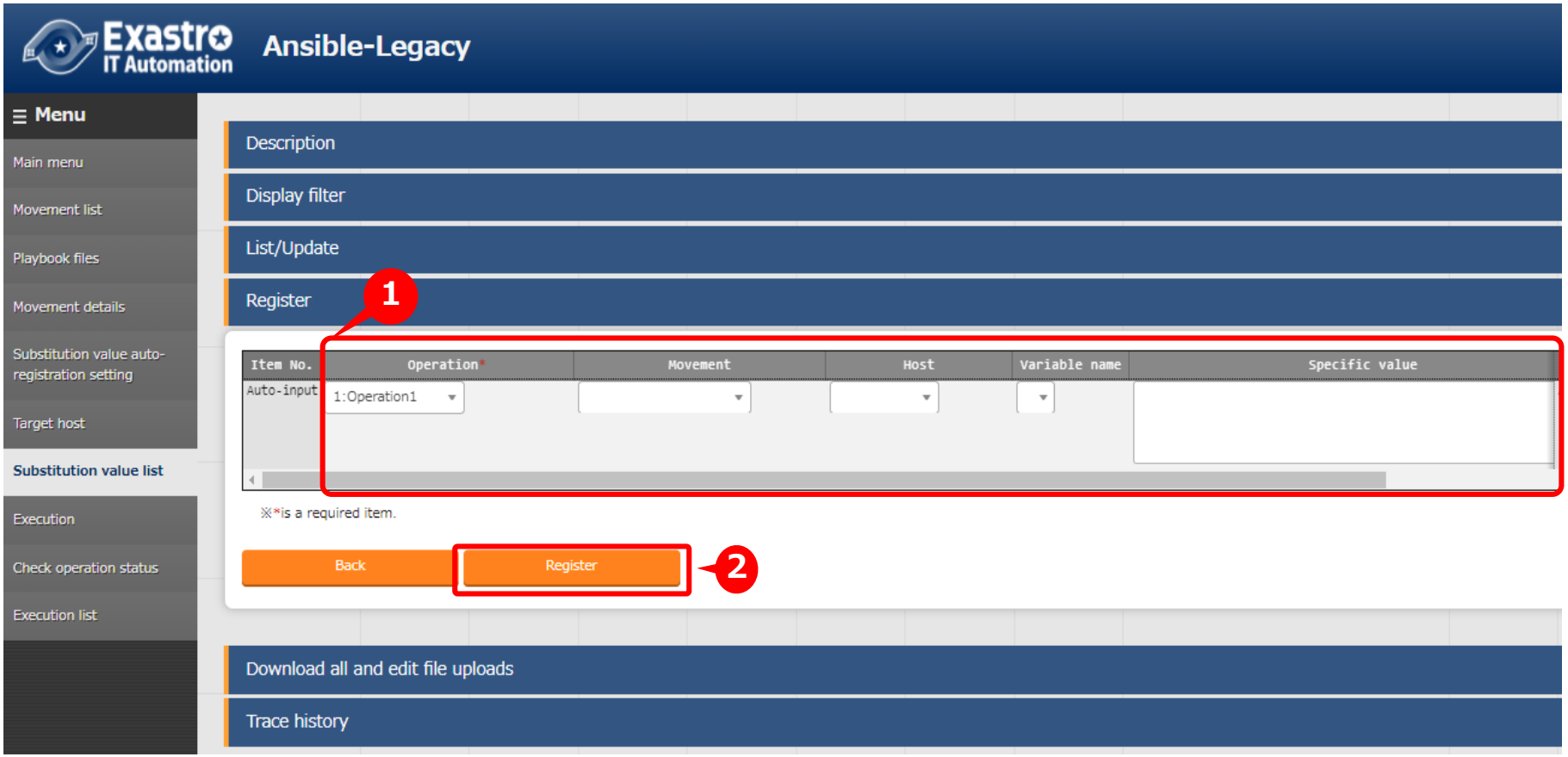
3.7 Substitution value list(1/2)

●Substitution value list

"Ansible-Legacy" menu group >> "Substitution value list" menu >> "Register" submenu >> "Start register" button.

① Input "Operation“ , "Movement“ , "Host" ,“Variable name" and “Specific value".

② Click “Register” button.



3.7 Substitution value list (2/2)

Substitution value list

Please use the list below for registering substitute values.

Operation	Host	Variable name	Specific value	Substitution order
1:operation1	1:Testserver	3:copy_file:1:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	3:copy_file:2:VAR_file_name	dir2	
1:operation1	1:Testserver	3:copy_file:3:VAR_edit_param_1	testfile	
1:operation1	1:Testserver	4:create_directory:4:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	5:create_file:5:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	5:create_file:6:VAR_file_name	testfile	
1:operation1	1:Testserver	6:edit_file:7:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	6:edit_file:8:VAR_file_name	testfile	
1:operation1	1:Testserver	6:edit_file:9:VAR_edit_param_1	param1	
1:operation1	1:Testserver	7:forced_termination:10:VAR_message_text	testmsg_fail	
1:operation1	1:Testserver	8:remove_directory:11:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	9:remove_file:12:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	9:remove_file:12:VAR_dir_name_1	testfile	

●Register Conductor

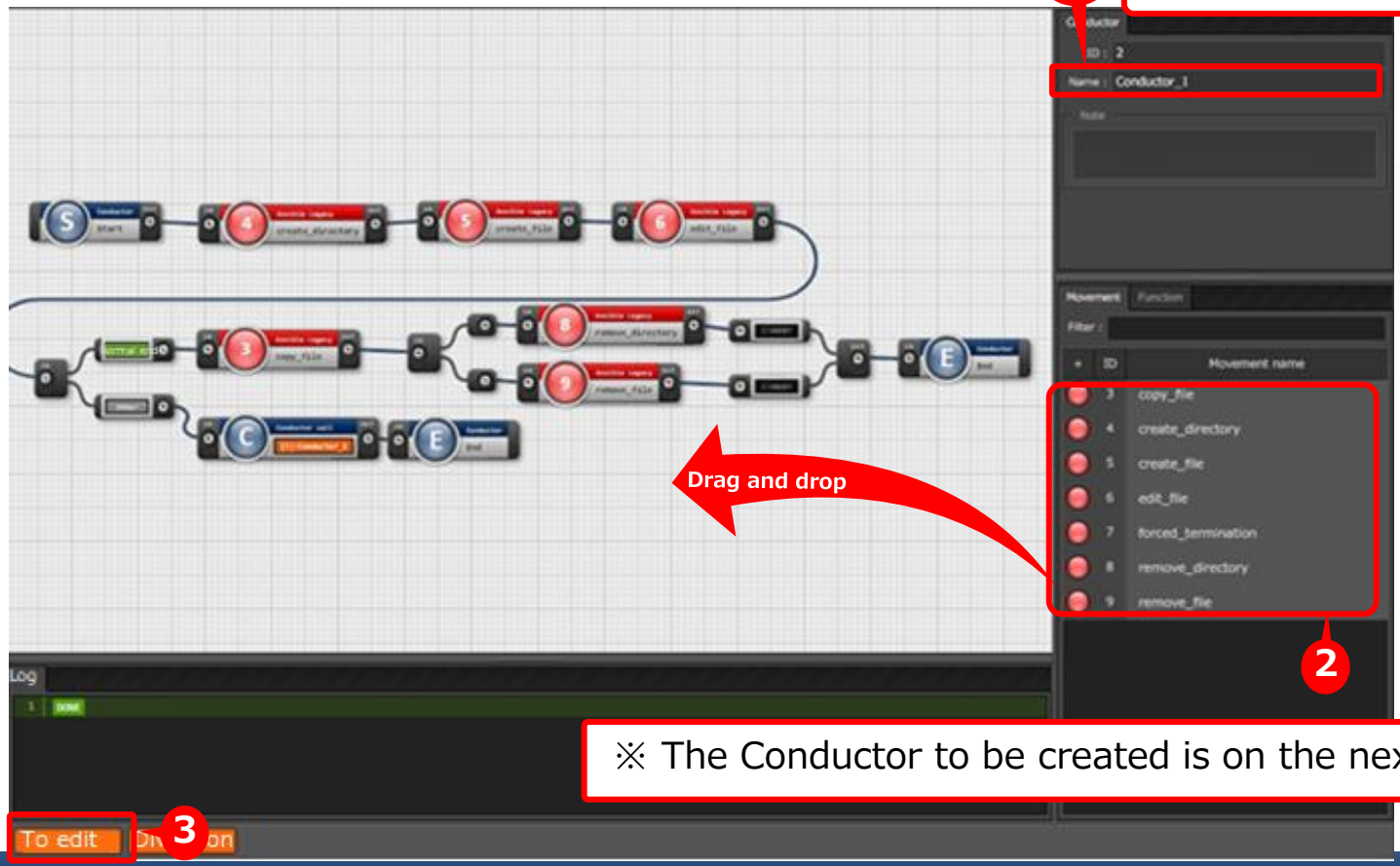
Input “Conductor name” in the Conductor menu group>> Conductor Class edit.

① Drag and drop “Movements” and “Functions” displayed from the right side of the screen to center of the screen.

② Click “Registration” button.

Enter value for item

Item	Value
Name	Conductor_1

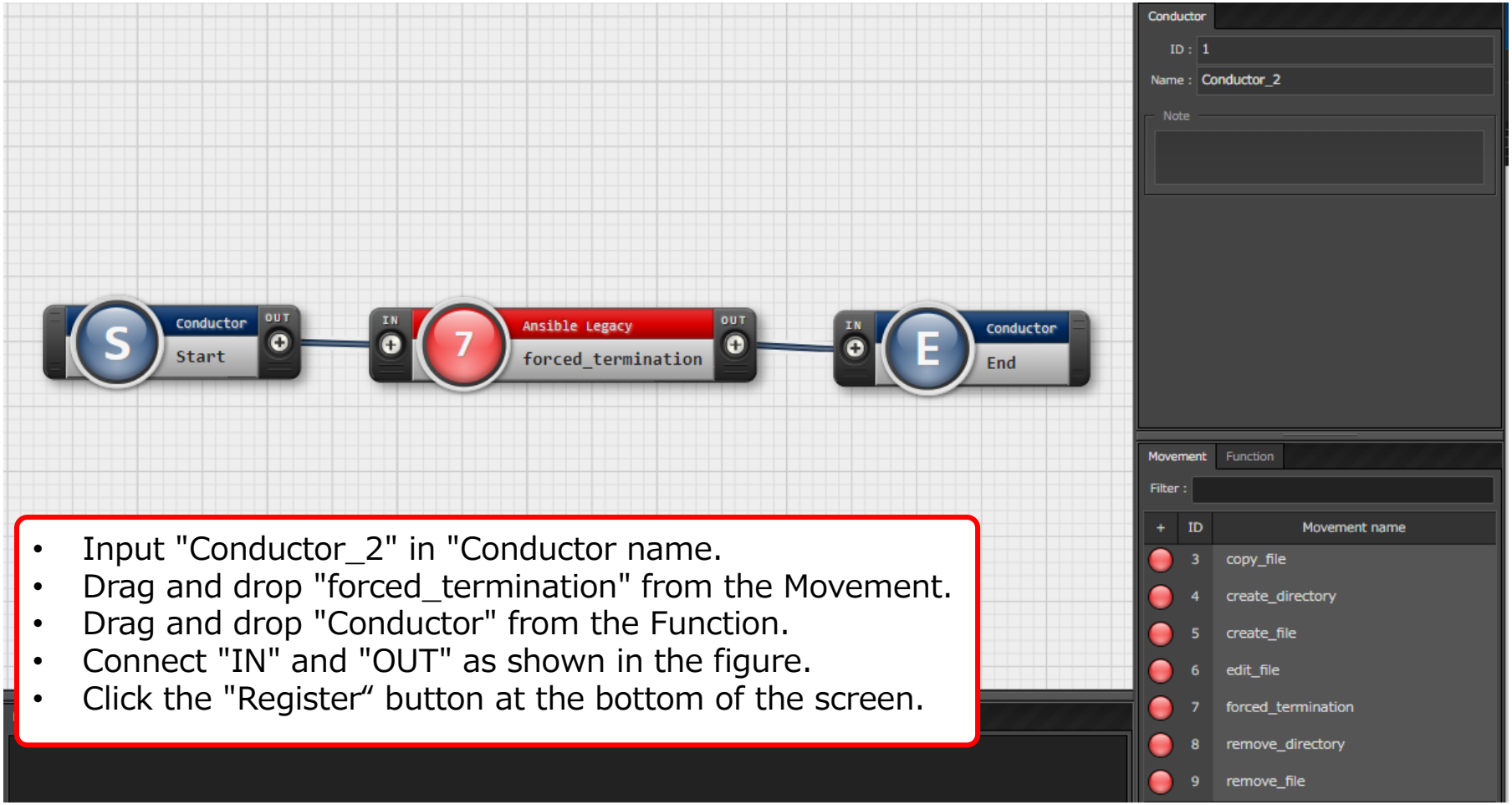


※ The Conductor to be created is on the next page.

3.8 Register Conductor(2/7)

●Register Conductor

Please create the Conductor as shown the figure below.



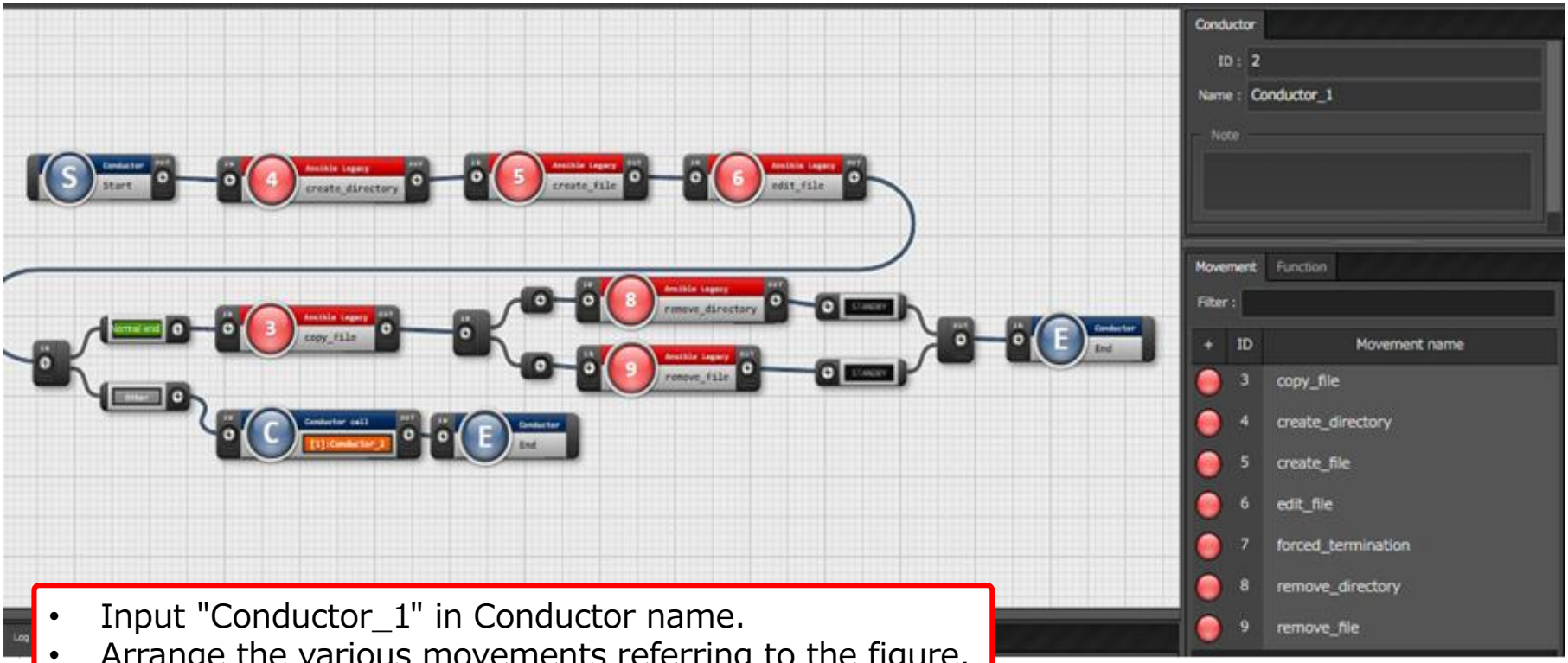
- Input "Conductor_2" in "Conductor name.
- Drag and drop "forced_termination" from the Movement.
- Drag and drop "Conductor" from the Function.
- Connect "IN" and "OUT" as shown in the figure.
- Click the "Register" button at the bottom of the screen.

3.8 Register Conductor(3/7)

●Register Conductor

The overall view of the created Conductor is as follows.

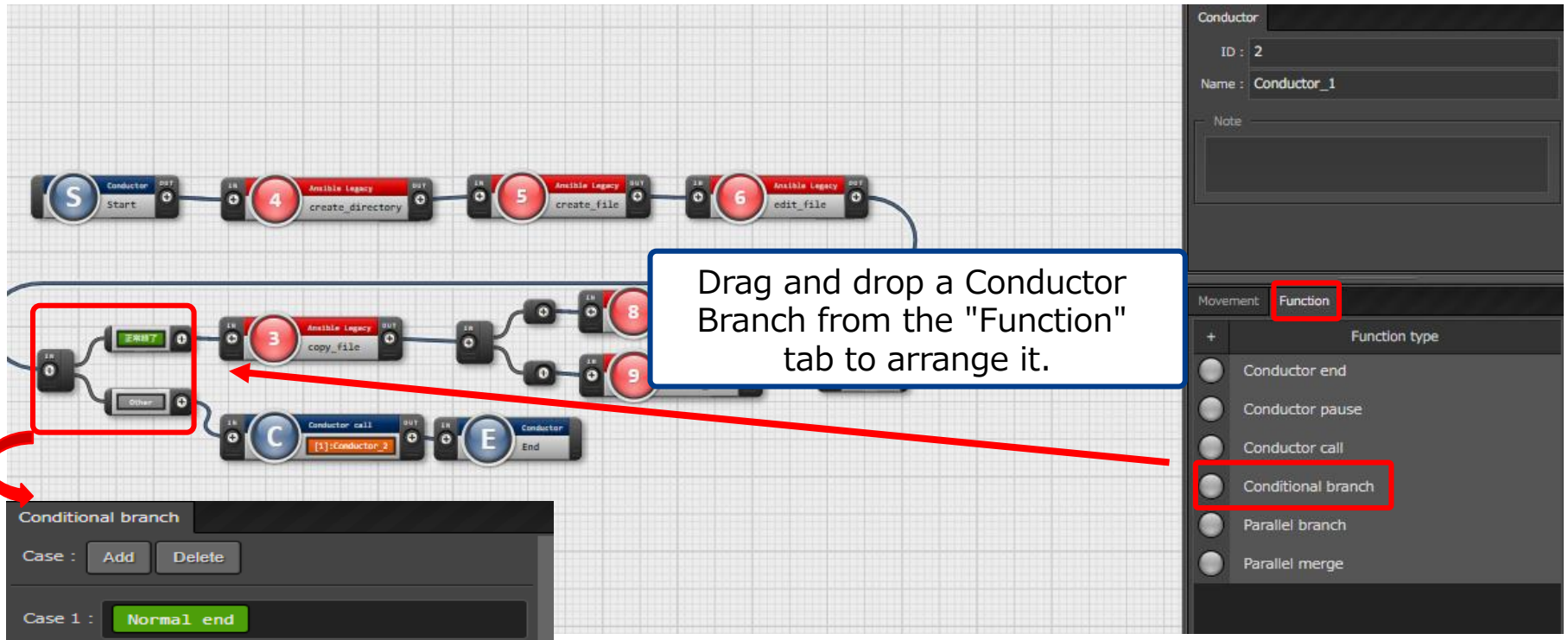
The details are explained on the next pages.

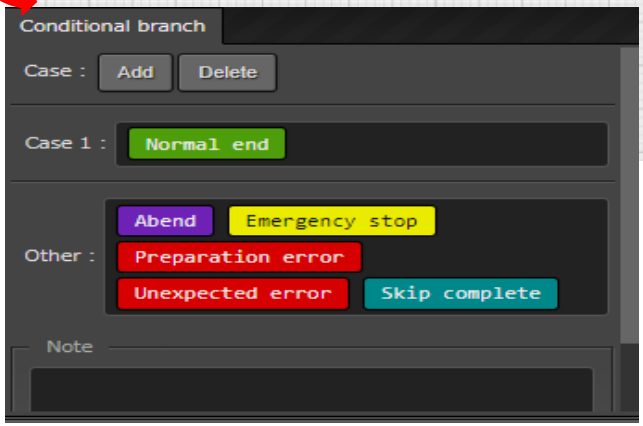


- Input "Conductor_1" in Conductor name.
- Arrange the various movements referring to the figure.

●Register Conductor

Please create the Conductor as shown in the figure below.

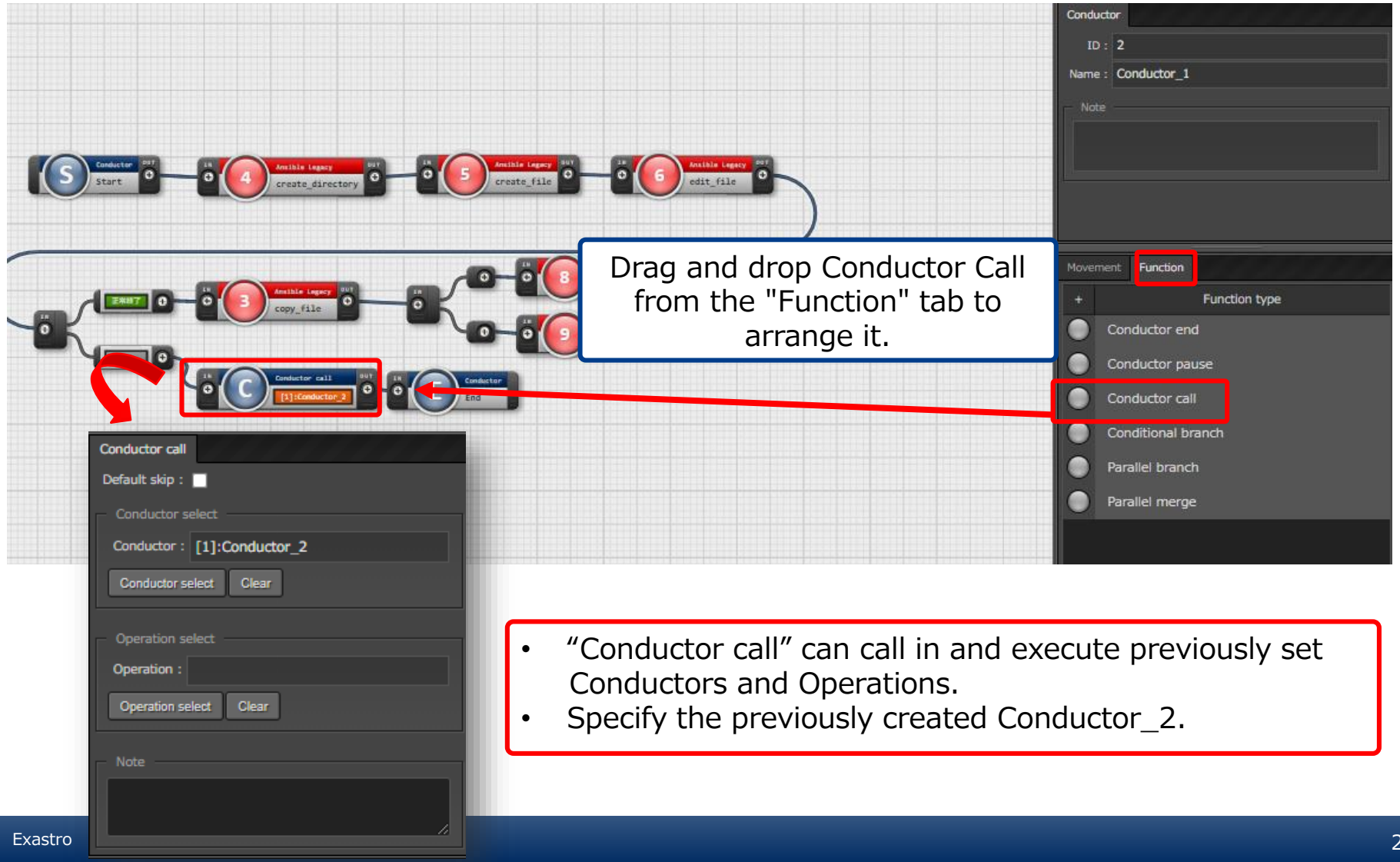




- Conductor Branches the next process according to the end result of the Movement immediately before it is arranged.
- For now, set it so subsequent processing only happens when the end status is “Normal end”.

3.8 Register Conductor (5/7)

- Register Conductor
  Please create the Conductor as shown in the figure below.

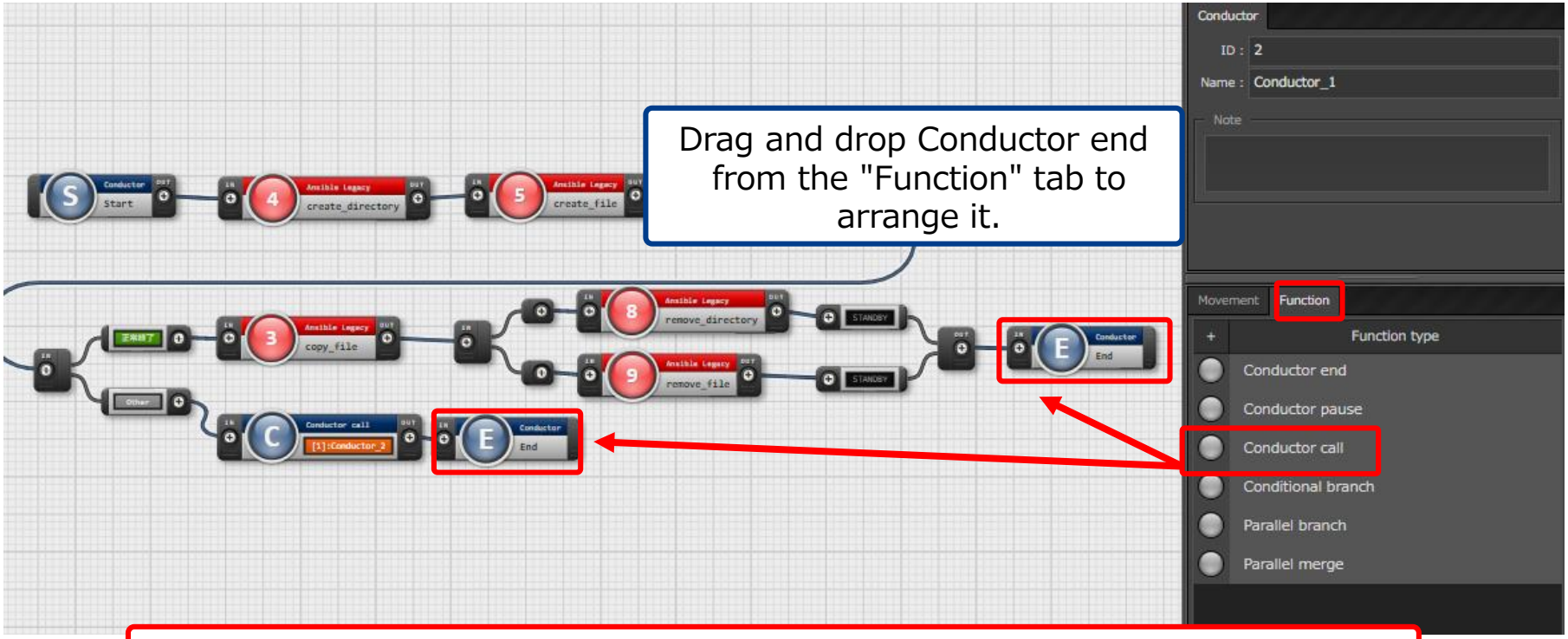


- "Conductor call" can call in and execute previously set Conductors and Operations.
- Specify the previously created Conductor_2.

3.8 Register Conductor (6/7)

●Register Conductor

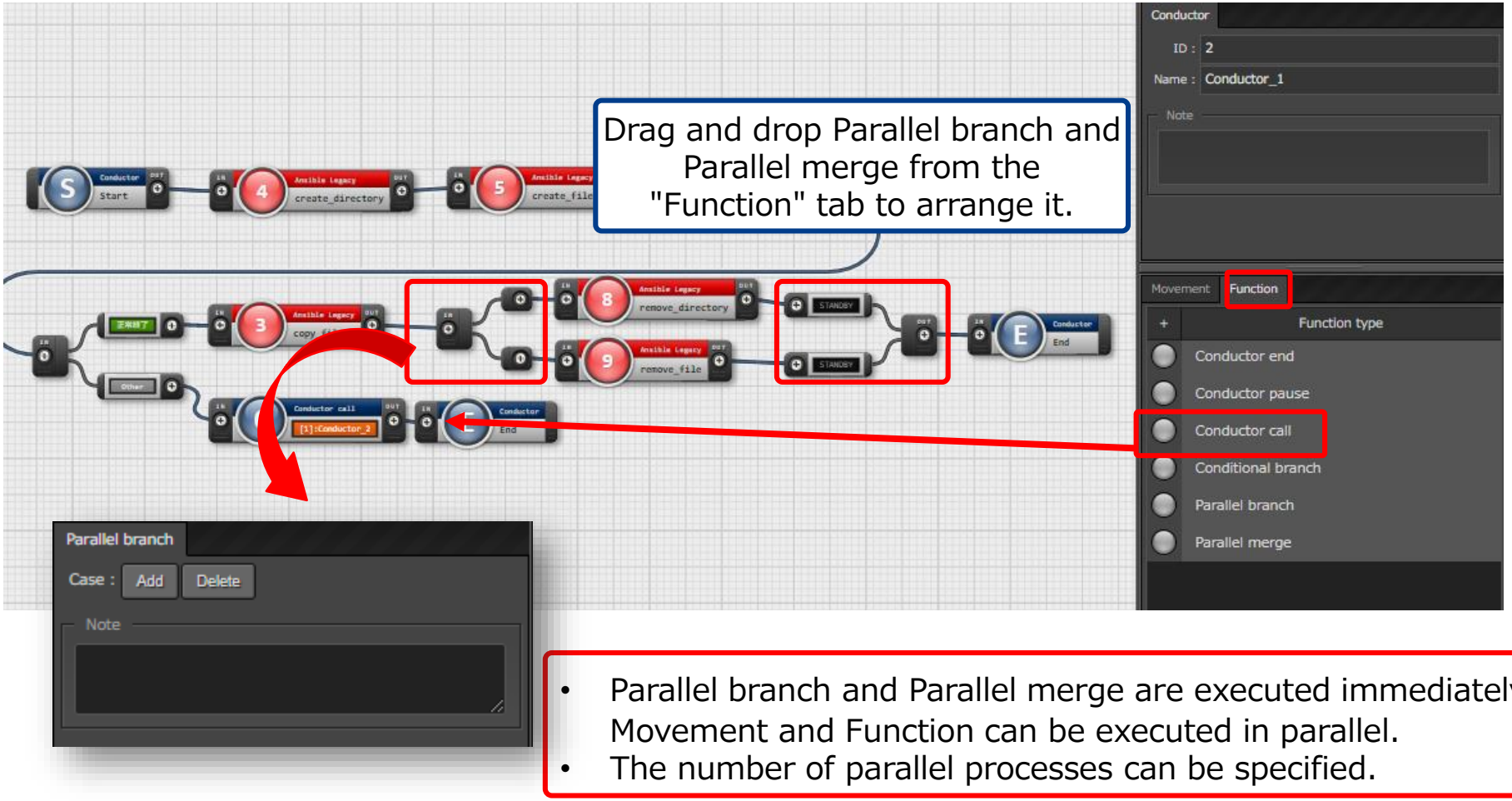
Please create the Conductor as shown in the figure below.



- Conductor end is a function is deployed at the end of a process..
- It is also deployed at the end of the branch process introduced in (5/7).

3.8 Register Conductor (7/7)

- Register Conductor
  Please create the Conductor as shown the figure below.



- Parallel branch and Parallel merge are executed immediately. Movement and Function can be executed in parallel.
- The number of parallel processes can be specified.

●Conductor execution

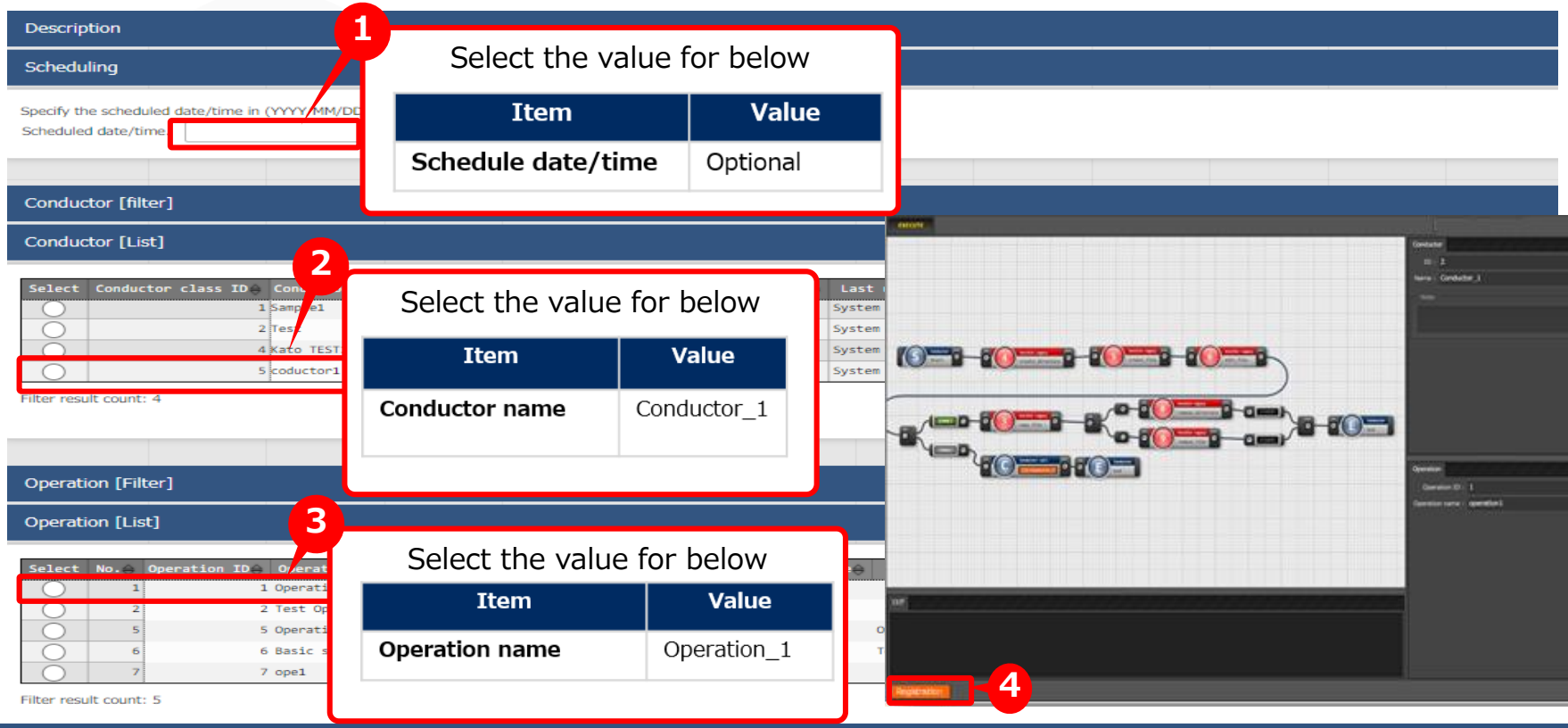
"Conductor“ menu group >> "Conductor execution" menu.

① Determine the execution date and time from the "Schedule date/time" item in the "Conductor [list]" submenu.

② Select "Conductor_1" in the "Conductor name" , "Conductor [List]" submenu items.

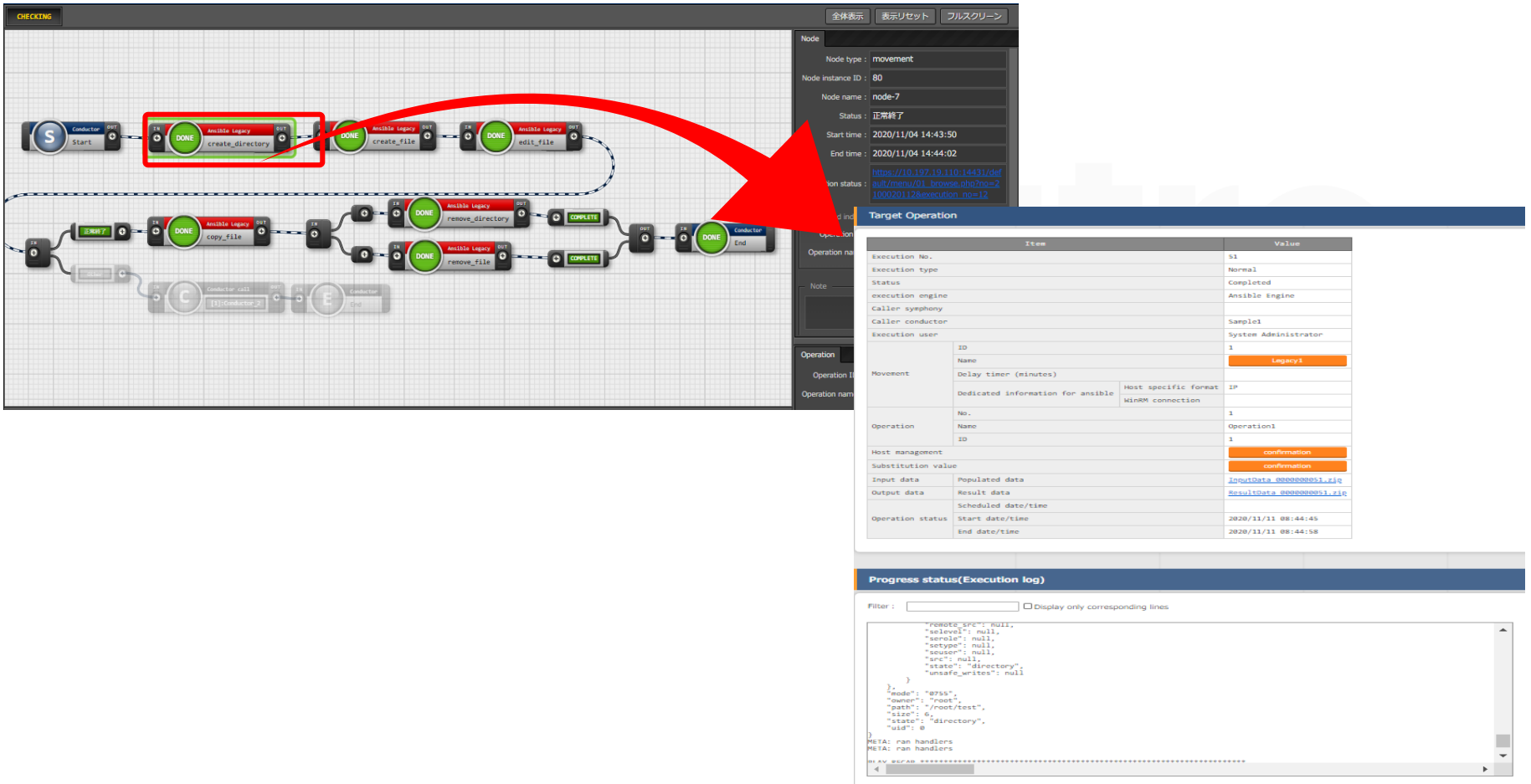
③ Select "operation" in the "operation name“ , "operation [list]" submenu items.

④ Click the "Execute" button.



Conductor confirmation

- Selecting a running or an already executed movement will move you to a screen where you can check the logs and the operation status.




Exastro