



IT Automation Conductor 【Tutorial】

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

Table of contents

1. Introduction

1. About this document

2. Conductor

1. About Conductor

2. Conductor feature

3. Conductor Function Description

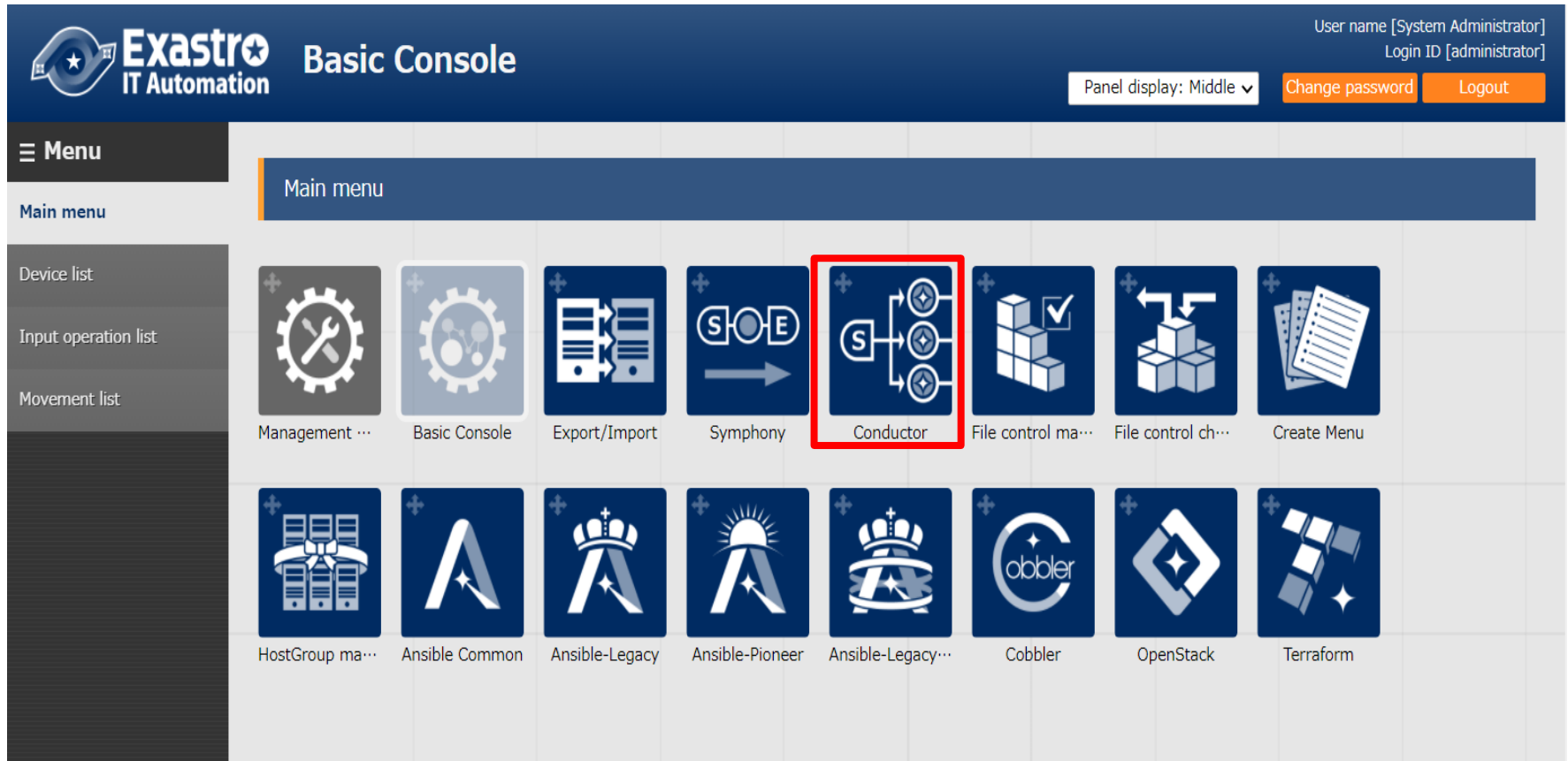
4. Conductor workflow

1. Introduction

1.1 About this document

Main menu

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

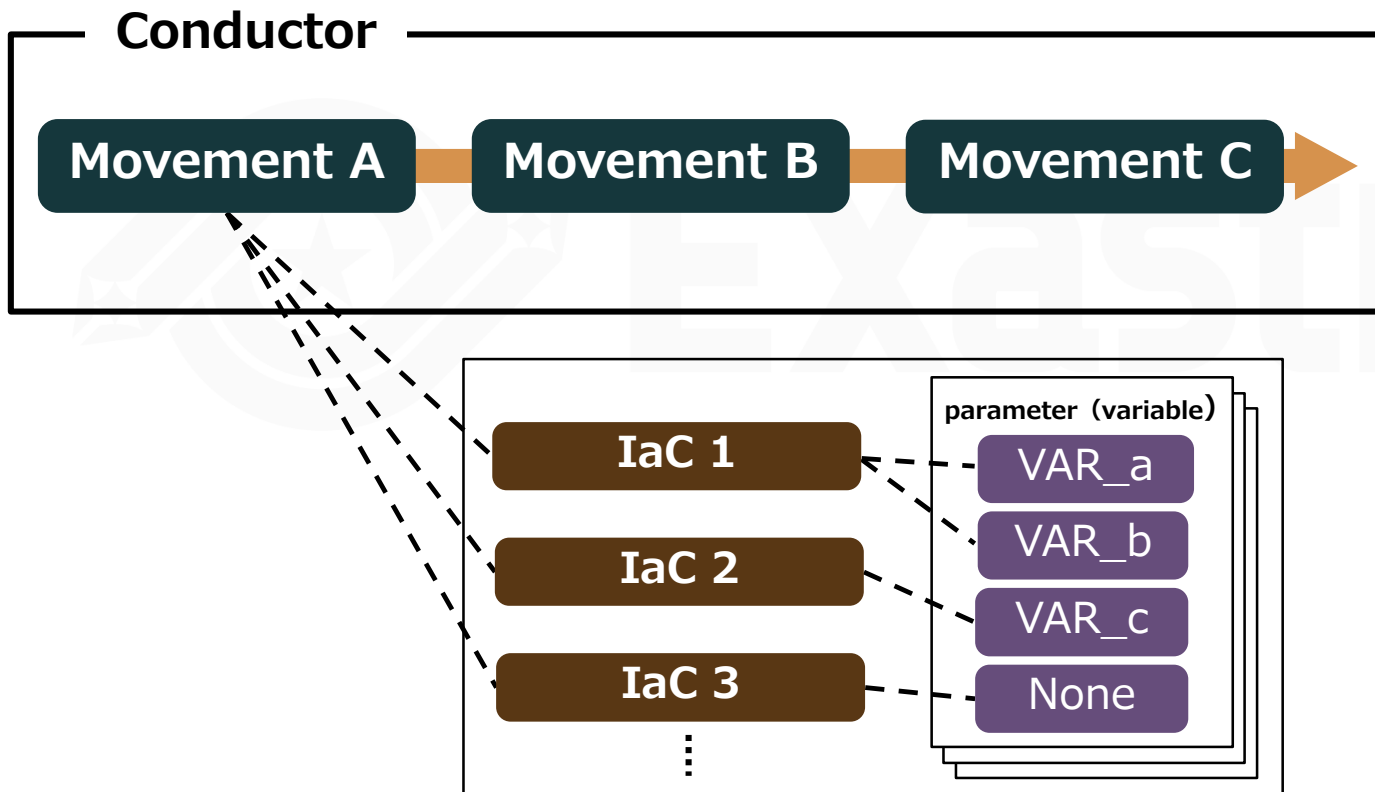


2. About Conductor



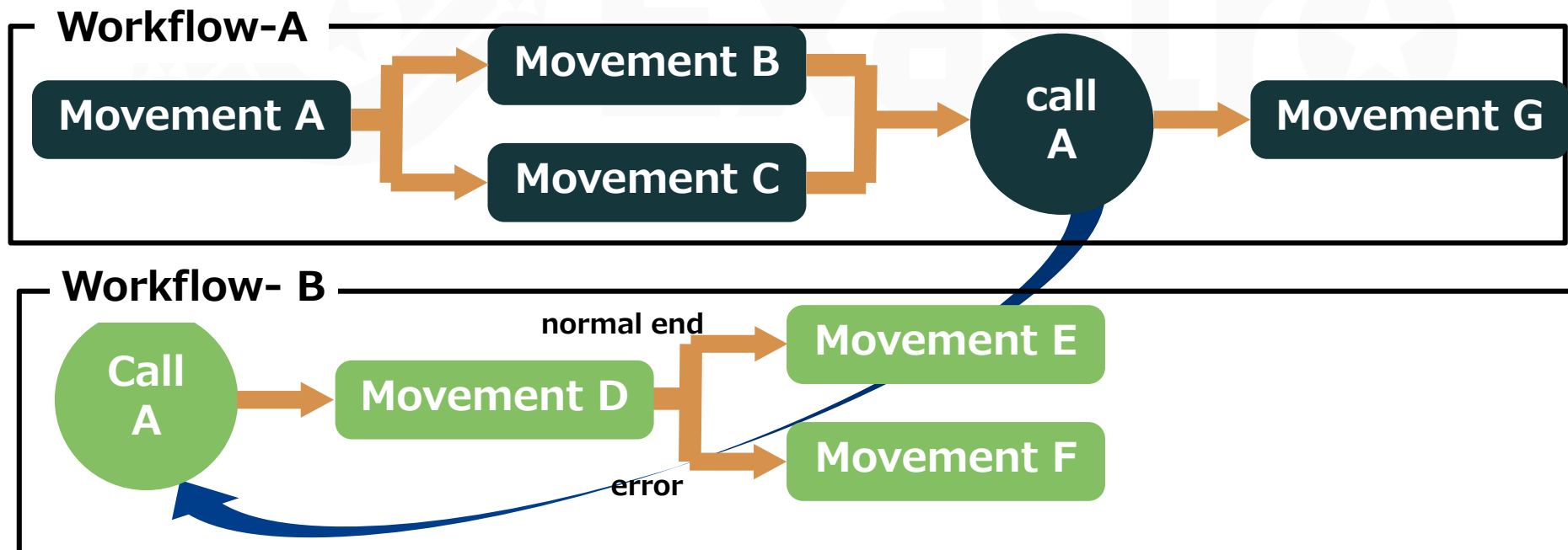
2.1 About Conductor

- Conductor is a function added to ITA from ver1.5.0.
- Conductor specifies Movements into one sequence and links it to an operation before executing it.



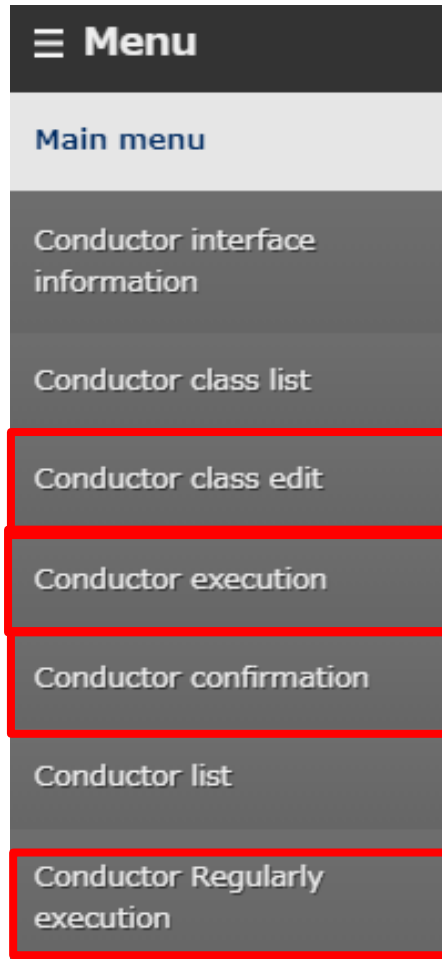
2.2 Conductor features

- Conductor is, in addition to being able to use similar functions that can be found in Symphony.
- As a result, Conductor is able to execute more complicated jobflows.
 - **Parallel movement executions**
 - **Ability to call other jobflows**
 - **Conditional branching according to the execution result of movement**



2.3 Conductor Function Description (1/9)

- The main menus and their functions in Conductor are as following.



① Conductor class edit

Create an operation using previously created movement.

② Conductor execution

Execute operations.

③ Conductor confirmation

Confirm previously created operations.

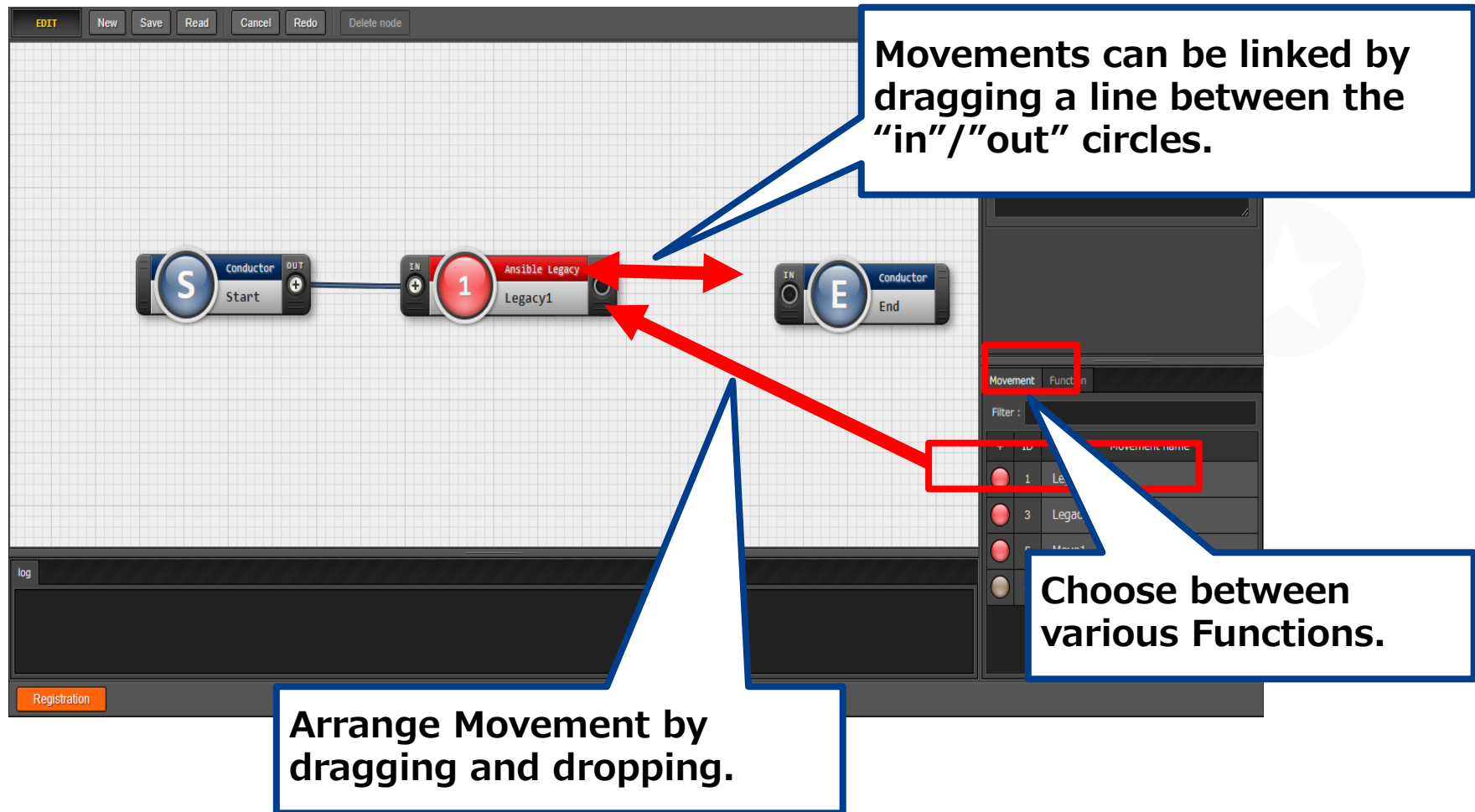
④ Conductor Regularly execution

Register operations and configure regularly executed jobflows.

2.3 Conductor Function Description (2/9)

●Conductor class edit (1/3)

- In the "Conductor class edit" menu, Movements and different functions can be added and deleted.



2.3 Conductor Function Description (3/9)

●Conductor class edit (2/3)

- Users can use the Conditional branch function by selecting it from the “Function” tab on the right side of the screen.

The screenshot displays the Exastro Conductor class editor. The main workspace shows a workflow diagram on a grid. It starts with a 'Conductor Start' node (blue circle with 'S'), followed by a series of small black nodes with a plus sign. These lead to two 'Ansible Legacy' nodes: 'Legacy1' (top, red circle with '1') and 'Legacy_movement' (bottom, red circle with '3'). A blue line connects the 'OUT' of 'Legacy1' to the 'IN' of 'Legacy_movement'. A red arrow points from the 'Function' tab in the right-hand panel to the 'Legacy_movement' node. The right-hand panel has a 'Function' tab selected, showing a list of functions: 'Conductor end', 'Conductor pause', 'Conductor call', 'Conditional branch', 'Parallel branch', and 'Parallel merge'. A blue callout box points to the 'Function' tab, stating: 'The arrangement of the “Function” tab can be changed by dragging and dropping them'. Another blue callout box points to the function list, stating: 'Users can choose between different Functions.' A third blue callout box points to the connection line between the two legacy nodes, stating: 'Similar to Movements, Operations can also be linked by dragging a line between them.'

Similar to Movements, **Operations** can also be linked by dragging a line between them.

The arrangement of the “Function” tab can be changed by dragging and dropping them

Users can choose between different Functions.

2.3 Conductor menu functions (4/9)

●Conductor class edit (3/3)

- The following explains the different functions available.
For more details, please refer to [this manual](#).

Picture	Name	Explanation
	Conductor start	Starts Conductor.
	Conductor end	Ends Conductor ↓ (If there are multiple "Conductor end" nodes, The Conductor will only end after all nodes has ended.)
	Conductor pause	Pauses any workflows. Un-pausing will resume the workflow.
	Conductor call	Calls another Conductor class.
	Symphony call	Calls a Symphony.
	Conditional branch	Branches the workflow depending on the result of the preceding Movement, Conductor call or Symphony call. <ul style="list-style-type: none">• Normal End.• Abnormal End.• Emergency Stop.• Preparation Error.• Unexpected Error.• SKIP Complete.
	Parallel branch	Executes "Movement", "Conductor call", and "Symphony call" in parallel. The number of executions that can be done in parallel depends on the ITA configuration and server specs.
	Parallel merge	Executes the following node after all of the preceding operations have completed.
	Movement nodes	Executes Movement.

2.3 Conductor Function Description (5/9)

●Conductor execution(1/2)

- Choose and execute the created Conductor in the "Conductor execution" menu.

Description

Scheduling

Specify the scheduled date/time in (YYYY/MM/DD HH:MM). Immediately execute when blank.
Scheduled date/time

Conductor [filter]

Conductor [List]

Select	Conductor class ID	Conductor name	Explanation	Remarks	Last update date/time	Last updated by
<input type="radio"/>	1	Sample1			2020/09/02 18:28:03	System Administrator
<input type="radio"/>	2	Test			2020/10/08 09:26:00	System Administrator

Filter result count: 2

Operation [Filter]

Operation [List]

Select	No.	Operation ID	Operation name	Scheduled date for execution	Last execution date	Remarks	Last update date/time	Last updated by
<input type="radio"/>	1	1	Operation1	2020/08/27 16:15	2020/10/15 10:31		2020/10/15 10:31:18	Legacy execution procedure
<input type="radio"/>	2	2	Test Operation	2020/10/08 10:00	2020/10/16 13:45		2020/10/16 13:45:19	Legacy execution procedure

Filter result count: 2

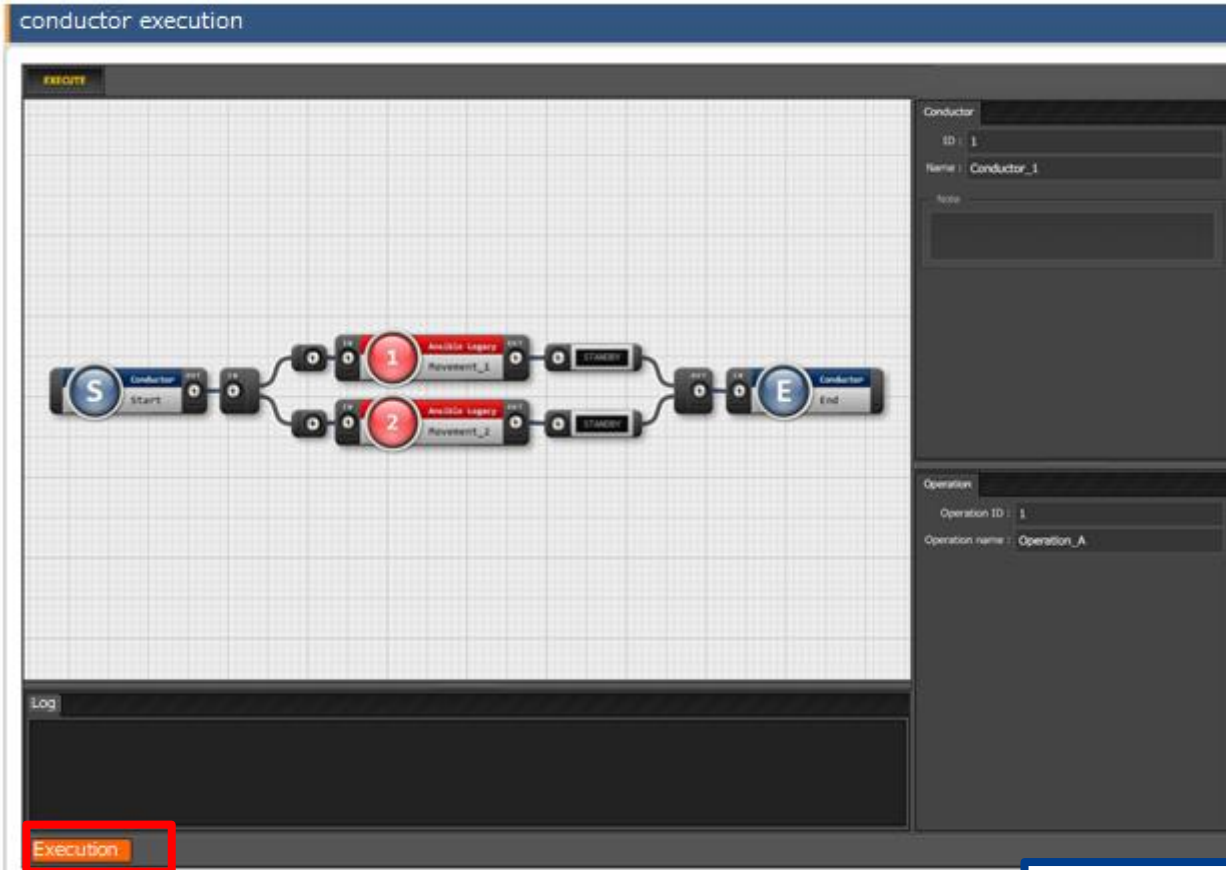
It is possible to schedule the timing of execution

Select Conductor and Operation.

2.3 Conductor Function Description (6/9)

●Conductor execution (2/2)

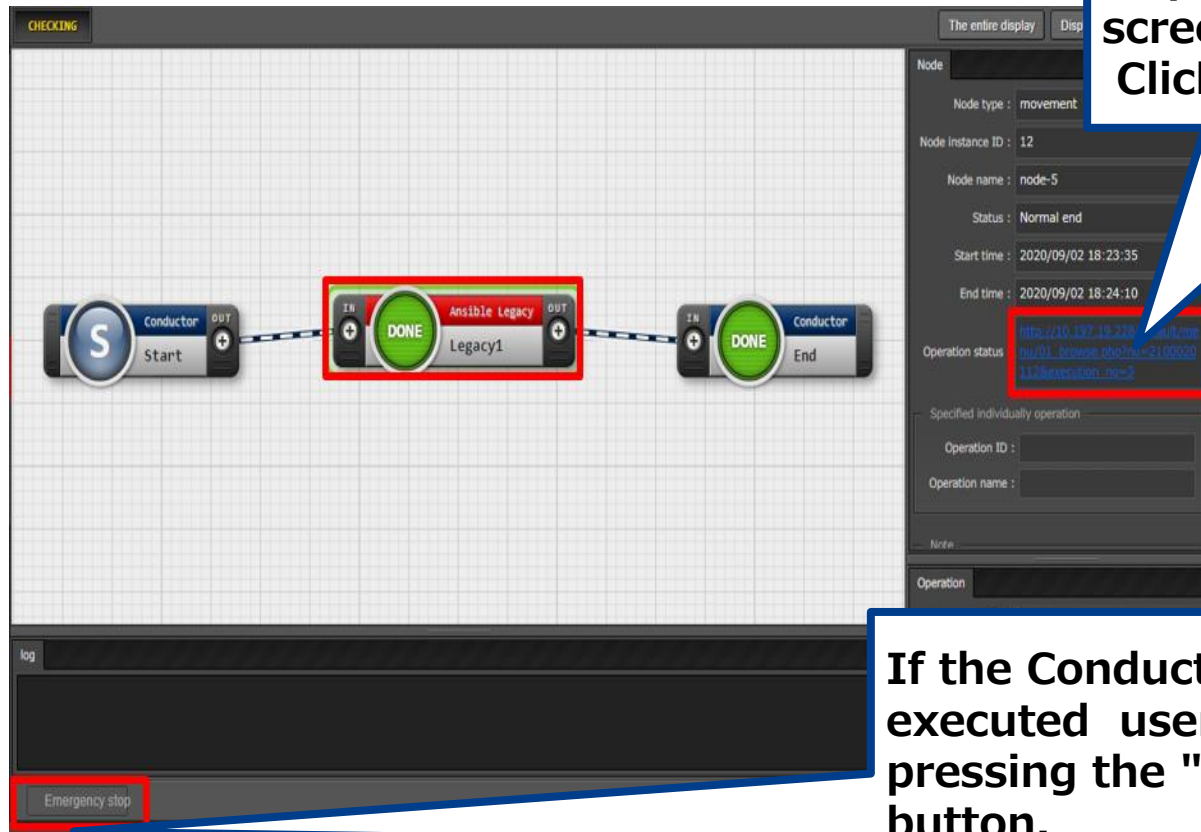
- The conductor and operation selected at the top of the page will be displayed.



If there are no problems with the contents, press the "Execute" button to execute.

2.3 Conductor (7/9)

- **Conductor confirmation**
- Check the execution status from the "Conductor Confirmation Menu".



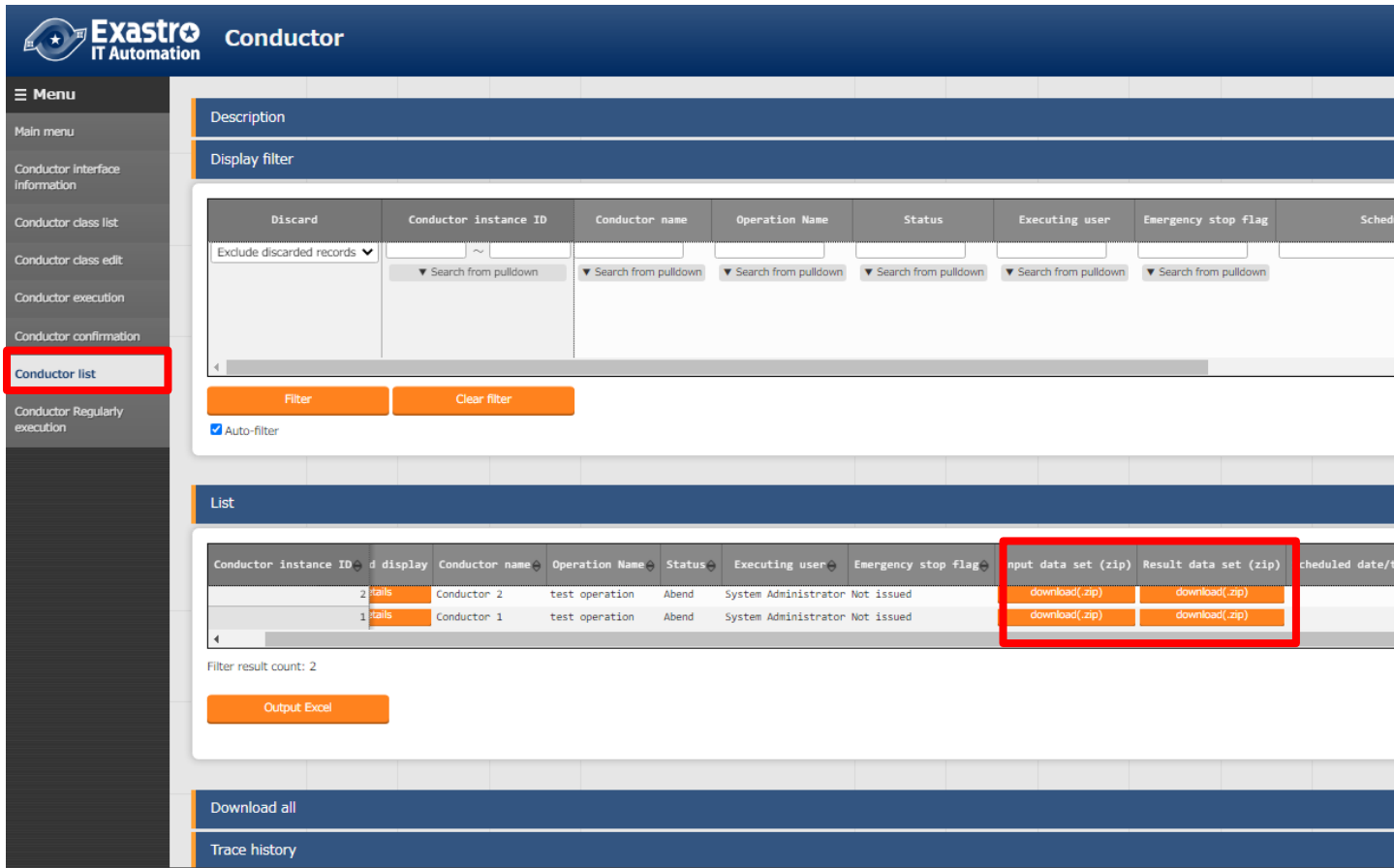
Clicking the movement will display a detailed results screen.
Click [here](#) for more details.

If the Conductor is scheduled to be executed users can cancel it pressing the "schedule cancelling" button.

2.3 Conductor Function Description (8/10)

●Conductor check (1)

In the “Conductor” Menu group -> “Conductor list” menu -> “List” Sub-menu, users can download the input/results data for each Conductor.



The screenshot displays the Exastro IT Automation Conductor interface. On the left, a sidebar menu has 'Conductor list' highlighted with a red box. The main area shows the 'List' sub-menu, also with a red box. It features a table of conductor instances with columns for ID, name, operation, status, user, and flags. The 'Input data set (zip)' and 'Result data set (zip)' columns contain 'download(.zip)' buttons, which are also highlighted with a red box. Below the table, there is a 'Filter result count: 2' indicator and an 'Output Excel' button. At the bottom, there are links for 'Download all' and 'Trace history'.

Conductor instance ID	Conductor name	Operation Name	Status	Executing user	Emergency stop flag	Input data set (zip)	Result data set (zip)	Scheduled date/t
2	Conductor 2	test operation	Abend	System Administrator	Not issued	download(.zip)	download(.zip)	
1	Conductor 1	test operation	Abend	System Administrator	Not issued	download(.zip)	download(.zip)	

2.3 Conductor Function Description (8/9)

●Conductor Routine Executions (1/2)

- In the "Conductor Regularly execution" menu, users can manage regularly executed operations.

The screenshot displays the 'Conductor Regularly execution' interface. At the top, there is a 'Display filter' section with a 'Close' button. Below this is a table with columns: Discard, Periodic work execution ID, Conductor class name, Operation name, status, Next execution date, Last update date/time, and Last updated by. Each column has a search input field with a 'Search from pulldown' button. Below the table are 'Filter' and 'Clear filter' buttons, and a checked 'Auto-filter' checkbox. Below the filter section are two tabs: 'List/Update' (with an 'Open' button) and 'Register' (with a 'Close' button). The 'Register' tab is active, showing a table with columns: Periodic work execution ID, Conductor class name, Operation name, status, Schedule settings, Next execution date, Start date, End date, pe, Last update date/time, and Last updated by. The 'Schedule settings' column has a red button labeled 'Schedule settings'. Below the table is a note '※*is a required item.' and 'Back' and 'Register' buttons.

A detailed schedule can be set from the "Schedule Settings" button.

2.3 Conductor Function Description (9/9)

●Conductor Routine Executions (2/2)

- "Schedule setting" allows user to set detailed settings such as the regular execution period and the period for stopping work.

Set a schedule

Work period

* Start date: 2020/10/19 11:00 End date: 2020/10/29 23:27

Schedule

☒ Time ☐ Day ☐ Week ☐ Month(Specify day) ☐ Month(Specify day of week) ☐ End of month

* Interval: 5 Time

Work suspension period

2020/10/06 23:34 ~ 2020/10/14 23:34

Note

* is a required field

OK Close

Clicking here will display a calendar.

October 2020

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

11:00
12:00
13:00
14:00
15:00
16:00

2.4 Conductor workflow

- The Conductor workflow is as follows.
Details can be found in the Practice document.

① Register device information

Basic console menu

② Register operation

③ Register Movement

Various driver menus

④ Check Movement

⑤ Register interface information

Conductor menu

⑥ Register Conductor

⑦ Check Conductor

⑧ Execution Conductor

⑨ Check execution result

⑩ Check execution history



Exastro