



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

- This document aims to introduce the reader to the **“Conductor”** menu group

Exastro IT Automation Management Console

User name [System Administrator]
Login ID [administrator]
Role Change password Logout

Menu DASHBOARD Edit

Main menu

System settings
Menu group list
Menu list
Role list
User list
Role - Menu link list
Role - User link list
Sequence list
Single-sign-on Basic Preference
Single-sign-on Attribute Preference
version

Menu group

Management Con... Basic Console Export/Import Symphony **Conductor** Create Menu

Input Substitution value Reference Compare HostGroup manag... Ansible Common

Ansible-Legacy Ansible-Pioneer Ansible-LegacyRole Cobbler Terraform CI/CD for IaC

Movement

Movement 7 Total 100 %

| Movement | SUM |
|---------------------|-----|
| Ansible Legacy | 2 |
| Ansible Pioneer | 0 |
| Ansible Legacy Role | 0 |
| Terraform | 0 |

Work status

Status 0 Total

| Status | CON | SYN | SUM |
|-----------------------|-----|-----|-----|
| Executing | 0 | 0 | 0 |
| Unexecuted (schedule) | 0 | 0 | 0 |
| Unexecuted | 0 | 0 | 0 |

Work result

Result 0 Total

| Result | CON | SYN | SUM |
|-----------------------|-----|-----|-----|
| Normal end | 0 | 0 | 0 |
| Abnormal end | 0 | 0 | 0 |
| Unexpected error | 0 | 0 | 0 |
| Emergency stop | 0 | 0 | 0 |
| Schedule cancellation | 0 | 0 | 0 |

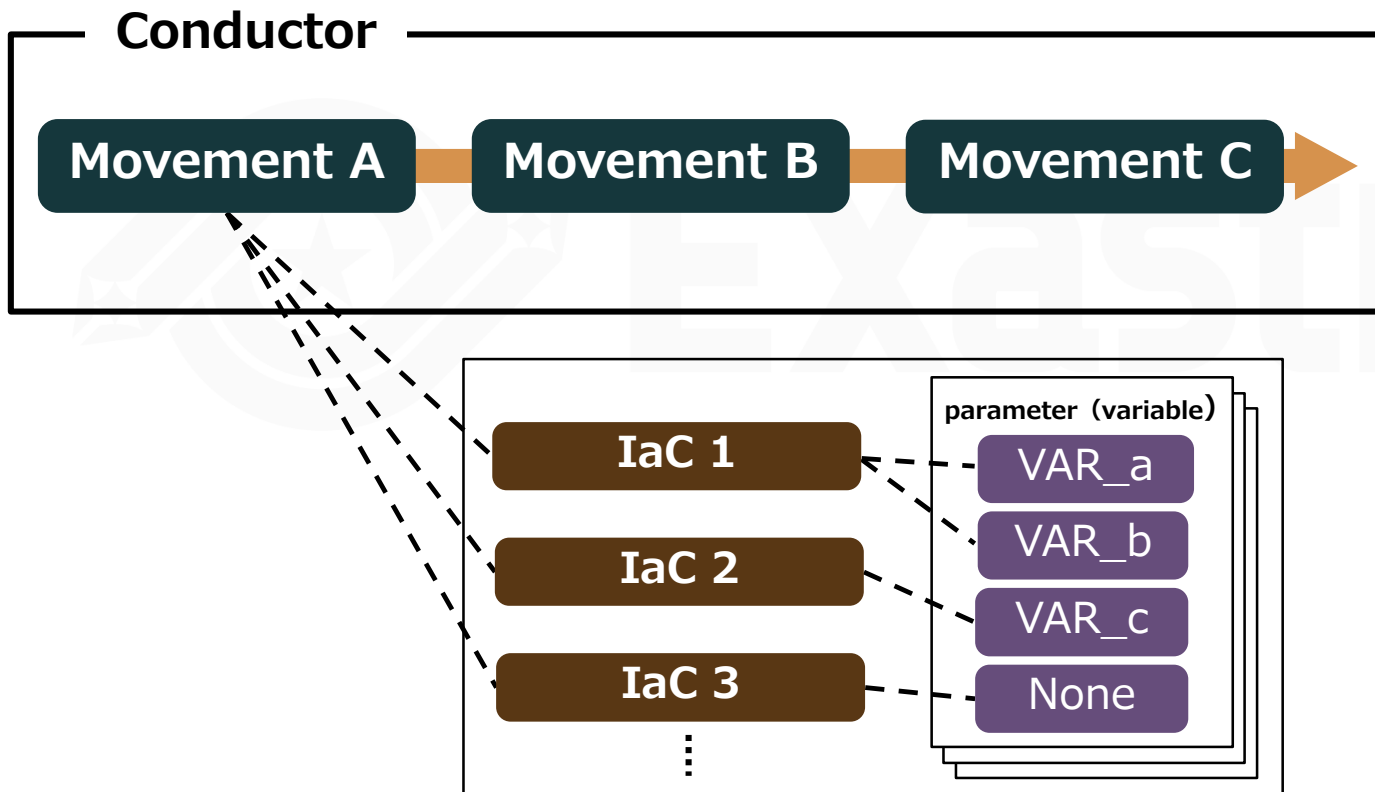
Work history

2. About Conductor



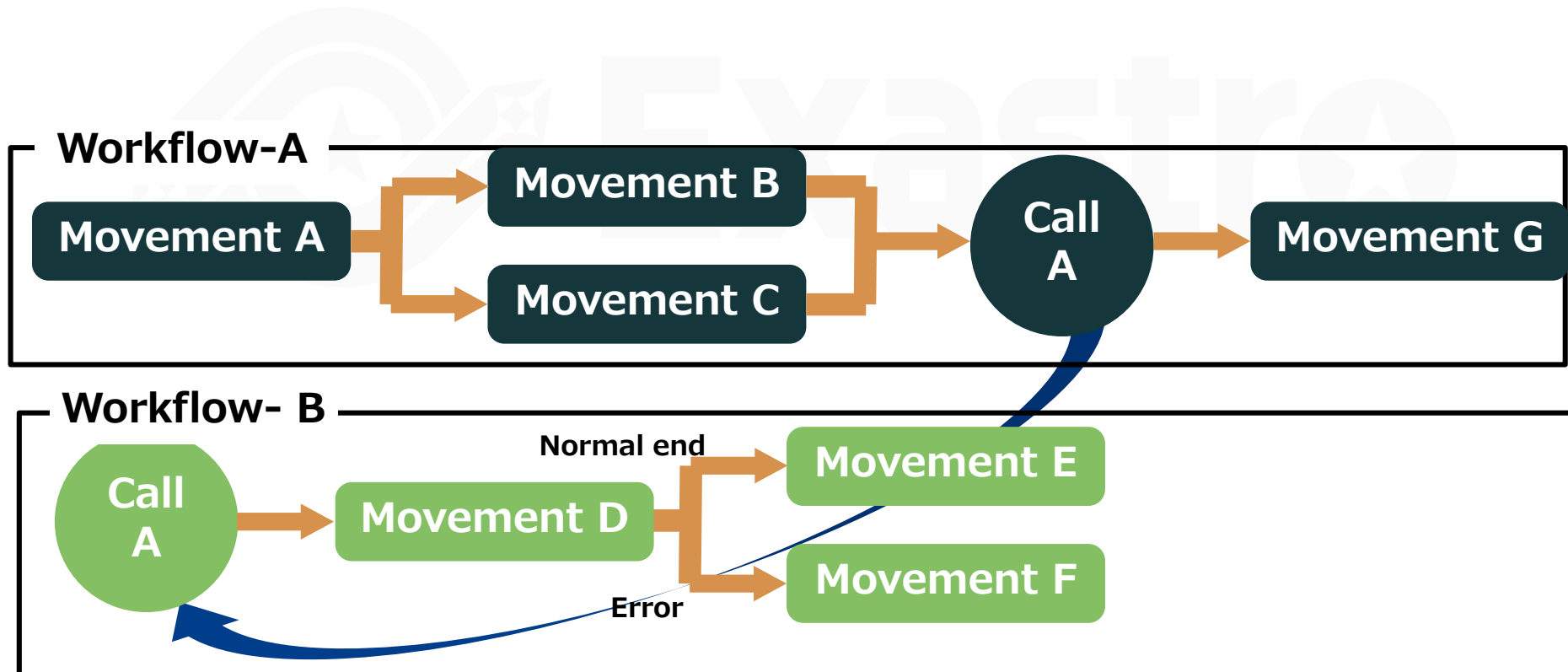
2.1 About Conductor

- Conductor is a function added in ITA Ver. 1.5.0
- Conductor allows users to specify multiple Movements into one sequence and links it to an operation before executing it.



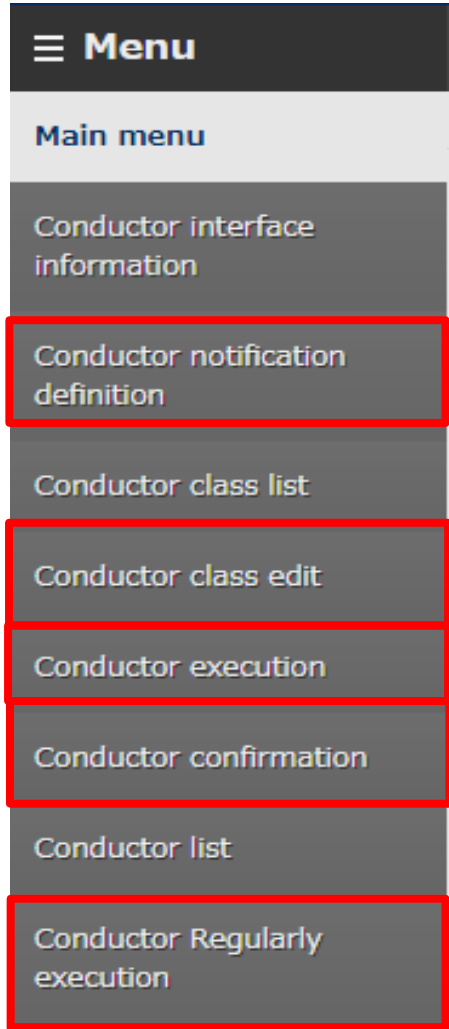
2.2 Conductor features

- While the Conductor Function has ,in similarity to the Symphony function, an execution function, it also contains the following functions.
- Hence, Conductor allows for execution of more complicated jobflows.
- **Parallel movement executions**
 - **Call other jobflows**
 - **Conditional branch depending on Movement results**



2.3 Conductor Function Description (1/14)

- The main menus the Conductor menu group and their functions 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.

⑤ Conductor notification definition

Set definition for conductor operation notifications.

2.3 Conductor Function Description (2/14)

●Conductor class edit (1/4)

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

The screenshot displays the Exastro software interface. At the top, a menu bar includes 'EDIT', 'New', 'Save', 'Read', 'Cancel', 'Redo', and 'Delete node'. The main workspace shows a diagram with three components: a 'Start' node (blue circle with 'S'), a central 'Legacy1' node (red circle with '1' and 'Ansible Legacy' text), and an 'End' node (blue circle with 'E'). Red arrows indicate connections between these nodes. A callout box points to the 'in'/'out' circles of the 'Legacy1' node, stating: 'Movements can be linked by dragging a line between the "in"/"out" circles.'

On the right side, a 'Movement' panel is visible. It has a 'Function' tab selected, a 'Filter:' field, and a table with columns '+', 'ID', 'Movement', and 'name'. The table contains the following data:

| + | ID | Movement | name |
|----------------------------------|----|-----------------|------|
| <input checked="" type="radio"/> | 1 | Legacy1 | |
| <input type="radio"/> | 3 | Legacy_movement | |
| <input type="radio"/> | 5 | Move1 | |
| <input type="radio"/> | 4 | movement | |

A callout box points to the 'Legacy1' row in the table, stating: 'Choose between various Functions.'

At the bottom left, a 'log' window and a 'Registration' button are visible. A callout box points to the 'Legacy1' node in the diagram, stating: 'Arrange Movement by dragging and dropping.'

2.3 Conductor Function Description (3/14)

●Conductor class edit (2/4)

- 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 with nodes: a 'Start' node, a 'Legacy1' node (labeled '1'), and a 'Legacy_movement' node (labeled '3'). A red arrow points from the 'Function' tab in the right-hand panel to the 'Legacy_movement' node. The 'Function' tab is highlighted with a red box, and a list of functions is visible: '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 workflow diagram, stating: 'Similar to Movements, **Operations** can also be linked by dragging a line between them.' A third blue callout box points to the function list, stating: 'Users can choose between different Functions.'

EDIT New Save Read Cancel Redo Delete node

log

Registration

Default skip : ☐

Operation select

Operation :

Select Clear

Note

Function

Conductor end

Conductor pause

Conductor call

Conditional branch

Parallel branch

Parallel merge

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 function description (4/14)

●Conductor class edit (3/4)

- By selecting multiple Nodes, users can use the buttons in the menu on the left to align the items. To select multiple Nodes, drag the mouse while holding the mouse button and drag over the Nodes you want to select or hold the shift key and press the Nodes you want selected individually.

For more information about the Node tab, please refer to [this manual](#).

The image consists of two screenshots of the Exastro software interface, illustrating the process of selecting and aligning multiple nodes.






Top Screenshot: The interface shows a workspace with three nodes: a 'Start' node (S), an 'Ansible Legacy Test Movement' node (2), and an 'End' node (E). A blue callout box points to the nodes with the text: "Selecting multiple Nodes will display the Object alignment function." A red arrow points from the 'Equal space' button in the 'Node' panel to the same button in the bottom screenshot. The 'Node' panel on the right has a red box around the 'Align' and 'Equal space' sections. A blue callout box points to the 'Equal space' button with the text: "Use this tab to align the selected objects to your liking."

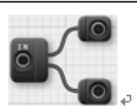
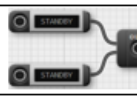
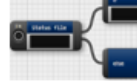

Bottom Screenshot: The same workspace is shown, but the nodes are now aligned horizontally. The 'Node' panel on the right is still visible, and the 'Equal space' button is highlighted with a blue box.

2.3 Conductor menu functions (5/14)

●Conductor class edit (4/4)

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

| Figure [↗] | Name [↗] | Description [↗] |
|--|---------------------------------|---|
|  | Conductor start [↗] | Start of Conductor [↗] |
|  | Conductor end [↗] | End of Conductor [↗] ※ If there are multiple Conductor end, the operation will end until all Conductor end are complete. [↗] |
|  | Conductor pause [↗] | Pauses the workflow temporarily. [↗] Cancel the pause to move on to next step. [↗] |
|  | Conductor call [↗] | Calls another registered Conductor class and executes it. [↗] ※ When original Conductor shuts down with error, Conductor executes process as normal call and does not have effect on original status. [↗] |
|  | Conditional branch [↗] | Branch process according to the result of "Movement" and "Conductor call" that the Node connects to. [↗] Status that can be specified is as follows. [↗] <ul style="list-style-type: none">• Normal end[↗]• Abnormal end[↗]• Emergency stop[↗]• Preparation error[↗]• Unexpected error[↗]• SKIP complete[↗]• Warning end[↗] |

| Figure [↗] | Name [↗] | Description [↗] |
|---|---------------------------------|---|
|  | Parallel branch [↗] | Execute "Movement" or "Conductor call" in parallel. [↗] ※ The maximum parallel process number depends on the configuration and server spec of ITA. [↗] |
|  | Parallel merge [↗] | Execute all process when all Nodes connected to this Node are finished. [↗] |
|  | Status file branch [↗] | Branches off process based on results of work directory of Movement file. [↗] |
|  | Movement [↗] | Execute Movement [↗] |

2.3 Conductor Function Description (6/14)

●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

Use this box to schedule when you want the Conductor to run.

Conductor [filter] ▾Open

Conductor [List] △Close

| 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

Select Conductor and Operation.

Operation [Filter] ▾Open

Operation [List] △Close

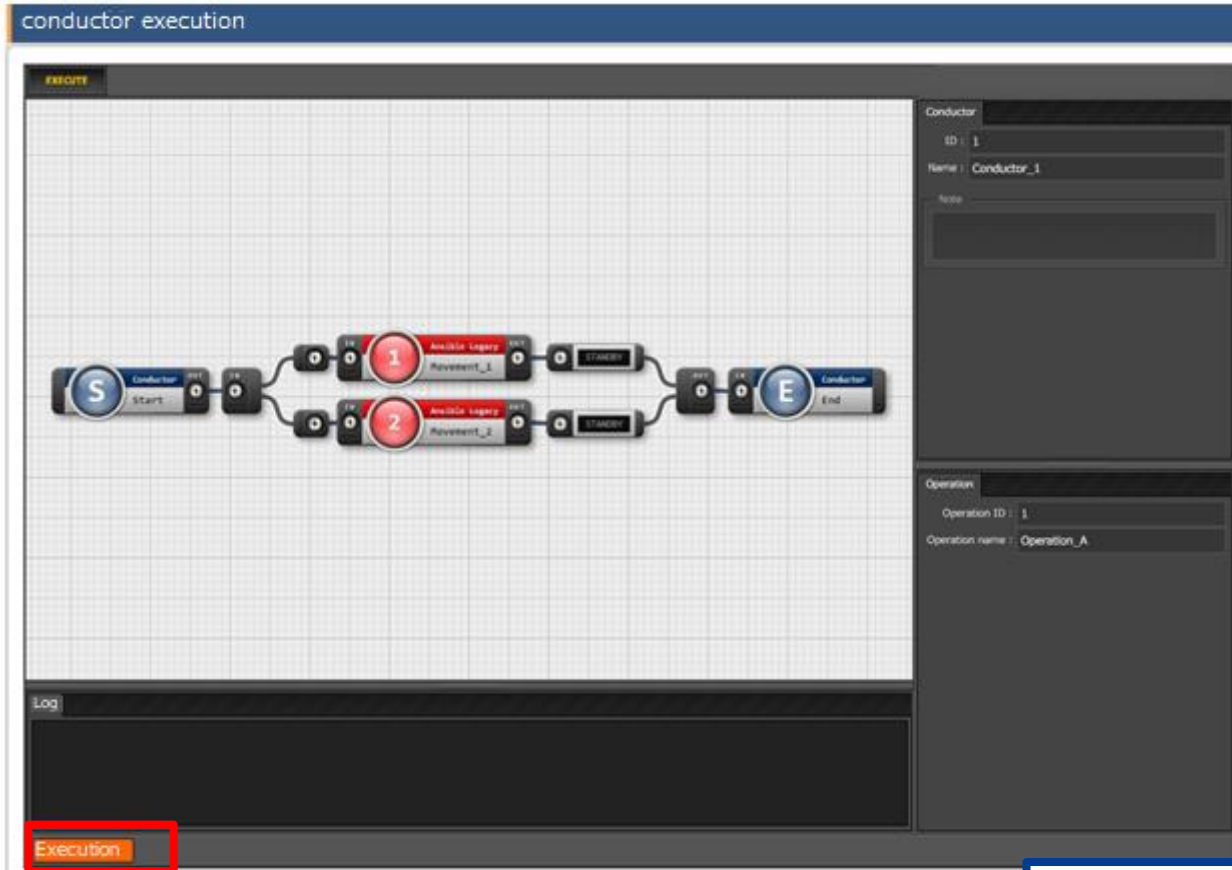
| Select | No. | Operation ID | Operation name | Scheduled date for execution | Last execution date | Remarks | Last update date/time | Last updated by |
|-----------------------|-----|--------------|----------------|------------------------------|---------------------|---------|-----------------------|----------------------------|
| <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

2.3 Conductor Function Description (7/14)

●Conductor execution (2/2)

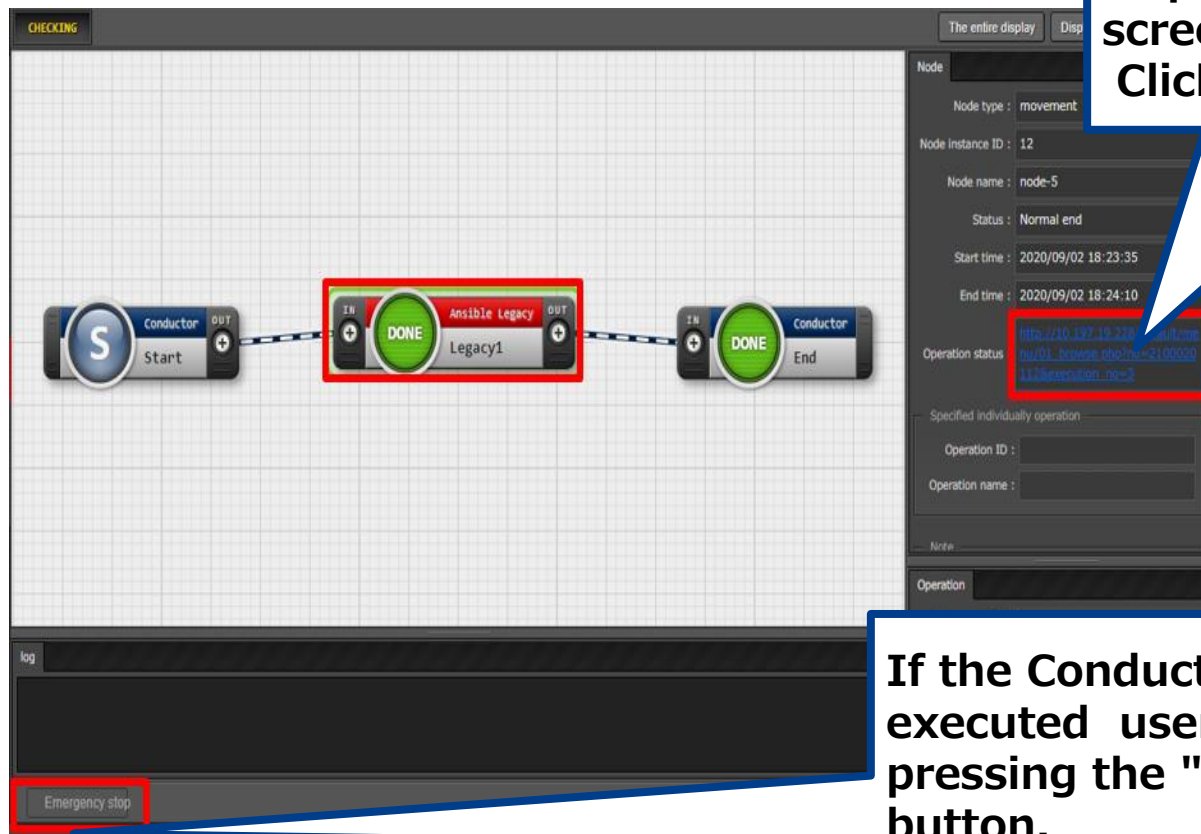
- The conductor and operation selected at the top of the page will be displayed.



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

2.3 Conductor (8/14)

- **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 (9/14)

●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 web application. The top navigation bar includes the Exastro logo, the title 'Conductor', and user information: 'User name [System Administrator]', 'Login ID [administrator]', and buttons for 'Role', 'Change password', and 'Logout'. A left sidebar menu lists various options, with 'Conductor list' highlighted by a red rectangle. The main content area is divided into two sections. The top section, titled 'Description', contains a 'Display filter' area with a table of search criteria: 'Discard' (with a dropdown 'Exclude discarded records'), 'Conductor instance ID' (with a search range), 'Conductor name', 'Operation Name', 'Status', 'Executing user', 'Emergency stop flag', 'Last update date/time', and 'Last updated by'. Each criterion has a 'Search from pulldown' button. Below the table are 'Filter' and 'Clear filter' buttons, an 'Auto-filter' checkbox, and a 'Table setting' button. The bottom section, titled 'List', contains a table with columns: 'History', 'Conductor instance ID', 'Detailed display', 'Conductor name', 'Operation Name', 'Status', 'Executing user', 'Emergency stop flag', 'Input data set (zip)', 'Result data s', 'Last update date/time', and 'Last updated by'. The first row of data shows instance ID '1' with a 'Details' button. The 'Input data set (zip)' and 'Result data s' columns for this row contain 'download(.zip)' and 'download' buttons, respectively, both highlighted by a red rectangle.

| Discard | Conductor instance ID | Conductor name | Operation Name | Status | Executing user | Emergency stop flag | Last update date/time | Last updated by |
|---------------------------|-----------------------|----------------------|----------------------|----------------------|----------------------|----------------------|-----------------------|----------------------|
| Exclude discarded records | ~ | | | | | | ~ | |
| | Search from pulldown | Search from pulldown | Search from pulldown | Search from pulldown | Search from pulldown | Search from pulldown | | Search from pulldown |

| History | Conductor instance ID | Detailed display | Conductor name | Operation Name | Status | Executing user | Emergency stop flag | Input data set (zip) | Result data s | Last update date/time | Last updated by |
|---------|-----------------------|------------------|----------------------|--------------------|--------|----------------------|---------------------|----------------------|---------------|-----------------------|--------------------------------|
| History | 1 | Details | Server basic setting | Basic settings all | Abend | System Administrator | Not issued | download(.zip) | download | 22/01/28 15:42:09 | Conductor management procedure |

2.3 Conductor Function Description (10/14)

●Conductor Routine Executions (1/3)

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

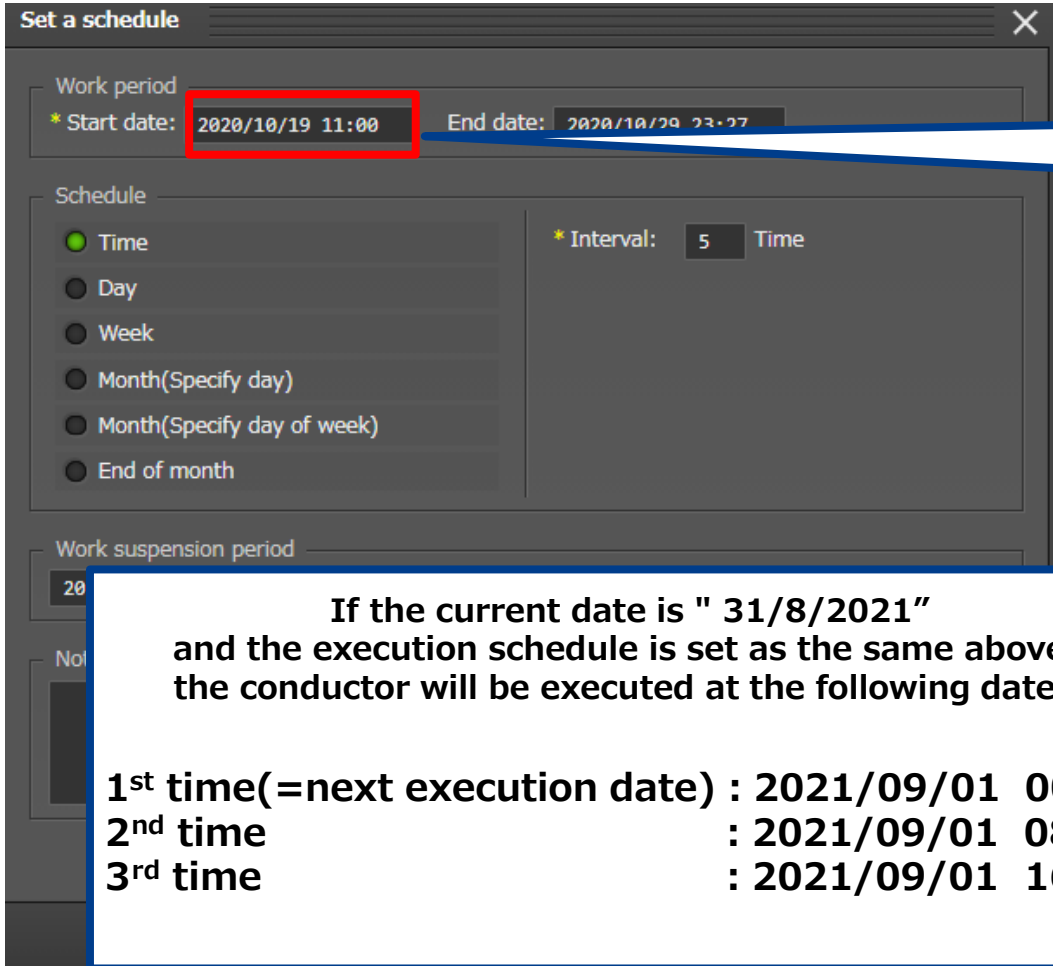
The screenshot displays the 'Conductor Regularly execution' interface. At the top, there is a 'Display filter' section with a 'Close' button. Below this is a table with columns: 'Discard', 'Periodic work execution ID', 'Conductor class name', 'Operation name', 'status', 'Next execution date', 'Last update 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 (11/14)

●Conductor Routine Executions (2/3)

- "Schedule settings" 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

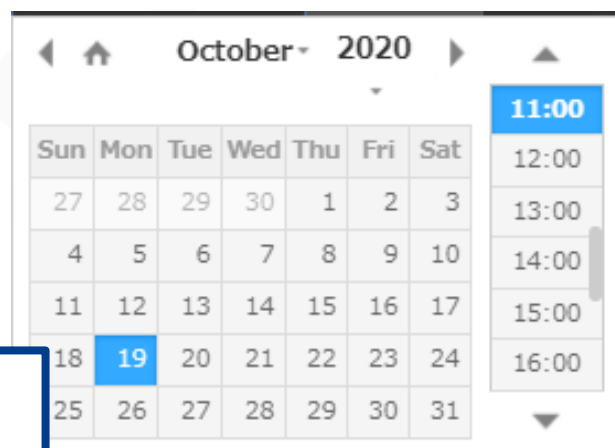
* Interval: 5 Time

Work suspension period

20

Not

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

If the current date is "31/8/2021"
and the execution schedule is set as the same above,
the conductor will be executed at the following dates:

1st time(=next execution date) : 2021/09/01 00:00
2nd time : 2021/09/01 08:00
3rd time : 2021/09/01 16:00

2.3 Conductor Function Description (12/14)

●Conductor Routine Executions (3/3)

- Any Conductor operations registered with a scheduled routine will display "Not executed (Reserved)" when the first interval time before the first scheduled execution date passes. (The default interval (time) value is 3 minutes). The Interval time can be changed by the user in the "Management console" > "System settings" menu. For more information, please see the Management console user manual.
- Conductor/Symphonies with the status "Unexecuted (Schedule)" can be seen in the "Work status" field in the dashboard and in the "Conductor list" menu.

If the next execution date is set to "2021/9/1 18:00" and the interval is set to 3 minutes, the status will change to "Unexecuted (Schedule)" 17:57 2021/9/1.

The screenshot shows the 'Conductor list' menu on the left. The main area displays a table of conductor instances. The table has columns: Discard, Conductor instance ID, Conductor name, Operation Name, Status, Executing user, Emergency stop flag, Last update date/time, and Last updated by. The 'Status' column for the selected item shows 'Unexecuted (Schedule)'. A red box highlights the 'Unexecuted (Schedule)' status in the table.

The screenshot shows the dashboard with three donut charts: Movement (1 Total, 100%), Status (1 Total, 100%), and Result (0 Total). The 'Status' chart is highlighted with a red box. Below the charts, there are three tables: Movement, Status, and Result. The 'Status' table shows a list of conductor instances with their status, and a red box highlights the 'Unexecuted (Schedule)' status in the table.

Will display when the interval time before the next execution date passes.

●Conductor notification definition (1/2)

- Send conductor operation results to selected message tool (Teams, Slack) by http request. Set definition for destination and content in “Conductor notification definition” menu. For detail see [this manual](#).
- Applicable for checking results of long or reserved operations.

Exastro IT Automation Conductor

User name [System Administrator]
Login ID [administrator]
Role Change password Logout

Menu

- Main menu
- Conductor interface information
- Conductor notification definition**
- Conductor class list
- Conductor class edit
- Conductor execution
- Conductor confirmation
- Conductor list
- Conductor Regularly execution

Description ▾Open

Display filter △Close

| Discard | No | Notification name | HTTP Request Options | | | Last update date/time | Last updated by |
|---|-------------------------------|-----------------------------|--|-----------------------------|------------------------------|-----------------------|-----------------------------|
| | | | Notification destination (CURLOPT_URL) | Header (CURLOPT_HTTPHEADER) | Message (CURLOPT_POSTFIELDS) | | |
| Exclude discarded records ▾ | ~ ▾ ▼ Search from pulldown | ▾ ▼ Search from pulldown | ▾ ▼ Search from pulldown | ▾ ▼ Search from pulldown | ▾ ▼ Search from pulldown | ~ ▾ | ▾ ▼ Search from pulldown |
| Filter Clear filter | | | | | | | |
| <input checked="" type="checkbox"/> Auto-filter | | | | | | | |

List/Update ▾Open

Register △Close

| No | Notification name* | HTTP Request Options | | | Last update date/time | Last updated by |
|------------|----------------------|---|------------------------------|-------------------------------|-----------------------|-----------------|
| | | Notification destination (CURLOPT_URL)* | Header (CURLOPT_HTTPHEADER)* | Message (CURLOPT_POSTFIELDS)* | | |
| Auto-input | <input type="text"/> | <input type="text"/> | <input type="text"/> | <input type="text"/> | Auto-input | Auto-input |

※* is a required item.

Back **Register**

Define notification destination · content

2.3 Conductor Function Description (14/14)

●Conductor notification definition (2/2)

- Users can press the “Notice” button in the “Conductor class edit” to select what statuses will send a notification.
The notifications must be defined in the Conductor notification definition menu.

The screenshot displays the 'Conductor class edit' interface. On the left is a 'Menu' sidebar with options: Main menu, Conductor interface information, Conductor notification definition, Conductor class list, **Conductor class edit**, Conductor execution, Conductor confirmation, Conductor list, and Conductor Regularly execution. The main workspace shows a workflow diagram with nodes: 'Start' (blue circle with 'S'), 'Install_Packages' (red circle with '1'), 'Open_Ports' (red circle with '2'), 'Start_Service' (red circle with '3'), and 'End' (blue circle with 'E'). Each node is connected by a line with 'IN' and 'OUT' labels. A callout box points to a 'Select' button in the top right corner of the workspace, with the text 'Click to display the "Notice list"'. Below the workspace, a 'Notice list' dialog is open, showing a table of notification statuses and their corresponding icons.

Notice list

| Notification name | Executing | Executing (delay) | Normal end | Warning end | Emergency stop | Abend | Unexpected error |
|---------------------|-----------|-------------------|------------|-------------|----------------|-------|------------------|
| Notification sample | | | | | | | |

Buttons: Decision, Cancel

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