



# IT Automation Quick Start

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

# Table of Contents

## 1. Introduction

- 1.1 Web Console Login Screen
- 1.2 Screen Description: Main menu
- 1.3 Screen Description: Other Menus (1/2)
- 1.4 Screen Description: Other Menus (2/2)

## 2. Procedure Description

- 2.1 Overall Procedure and Work Scope

## 3. Preparation

- 3.1 Registering an IaC (1/3)
- 3.2 Registering an IaC (2/3)
- 3.3 Registering an IaC (3/3)
- 3.4 Creating the Workflow Including the IaC
- 3.5 Registering a Target Linux Machine in Device list

## 4. Execution

- 4.1 Registering a New Operation Name
- 4.2 Connecting the Target to the IaC
- 4.3 Executing the Workflow

## A Appendix

- Reference 1 Ansible-Legacy: Single Execution
- Reference 2 Ansible-Legacy: Checking the Operation Results
- Reference 3 How to Check the Symphony Execution Results
- Reference 4 Sample Collection of Playbook

# 1. Introduction

# 1.1 Web Console Login Screen

## Web console **Login** Screen

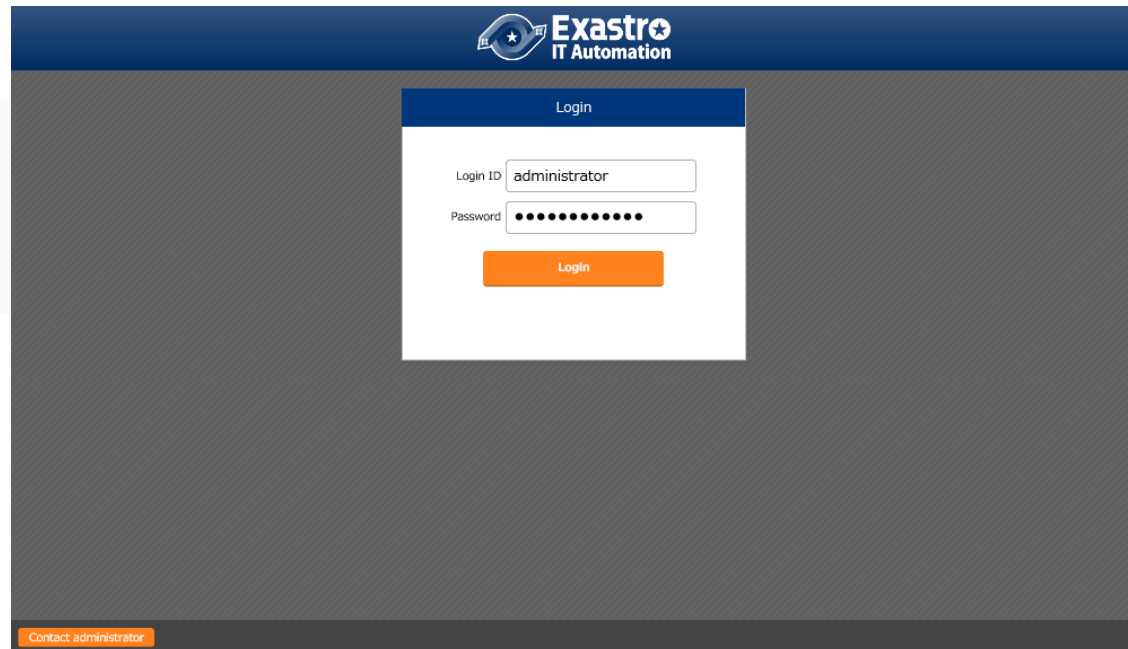
- Access the following URL to display the **Login** screen:  
<https://exastro-it-automation/>

### POINT

Immediately after the first login, you will be prompted to change your password.

### POINT

To deploy IT Automation, refer to "[ITA-online-install\\_en.pdf](#)".



## 1.2 Screen Description: Main menu

### Screen description: **Main menu**

- Basic names are as follows:

The screenshot displays the Exastro Basic Console Main menu. The interface features a dark blue header with the Exastro logo and 'Basic Console' text. On the right, it shows the user name 'System Administrator', login ID 'administrator', and buttons for 'Change password' and 'Logout'. A 'Panel display: Middle' dropdown is also present. A left sidebar contains a 'Menu' section with a list of options: Main menu, OS type master, Device list, Input operation list, Movement list, Symphony Interface information, Symphony class List, Symphony class editor, Symphony execution, Symphony execution checking, and Symphony execution list. The main content area, titled 'Main menu', displays a grid of 14 function icons arranged in two rows. The first row includes Management, Basic Console, File control ma, File control ch, Create master, Create param, HostGroup ma, and Ansible Common. The second row includes Ansible-Legacy, Ansible-Pioneer, Ansible-Legacy, Cobbler, DSC, and OpenStack. A red box highlights the entire grid of icons, with a red line pointing to the 'Menu' section in the sidebar, labeled 'Menu group'. Another red line points to the 'Symphony execution checking' option in the sidebar, labeled 'Menu'. A red circle with the word 'POINT' is positioned near the bottom right, with a red line pointing to a text box that reads: 'Refer to the manual for the details of each function.'

Exastro IT Automation Basic Console

User name [System Administrator]  
Login ID [administrator]  
Panel display: Middle ▼ Change password Logout

Menu

Main menu

Management... Basic Console File control ma... File control ch... Create master... Create param... HostGroup ma... Ansible Common

Ansible-Legacy Ansible-Pioneer Ansible-Legacy... Cobbler DSC OpenStack

Menu group

Menu

POINT

Refer to the manual for the details of each function.

Contact administrator

# 1.3 Screen Description: Other Menus (1/2)

## Screen description: other menus

- Basic names are as follows:

Exastro IT Automation Basic Console

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

Menu

Main menu

OS type master

Device list

Input operation list

Movement list

Symphony Interface Information

Symphony class List

Symphony class editor

Symphony execution

Symphony execution checking

Symphony execution list

Description

Display filter

Discard	OS type ID	OS type name	Device type			Remarks
			SV	NW	ST	
Exclude discarded records	~					
	Search from pulldown	Search from pulldown	Search from pulldown	Search from pulldown	Search from pulldown	

Filter Clear filter

☒ Auto-filter

List/Update

Register

OS type ID	OS type name*	Device type			Remarks
		SV	NW	ST	
Auto-input: os					

※\* is a required item.

Back Register

Download all and edit file uploads

Trace history

Contact administrator

POINT

Refer to the manual for the details of each function.

### Submenu outline

- Description** : Describes the menu being displayed.
- Display filter** : Allows you to search for registered information.
- List/Update** : Displays the registered information.

# 1.4 Screen Description: Other Menus (2/2)

## Screen description: other menus

- Basic names are as follows:

The screenshot displays the Exastro Basic Console interface. The top header includes the Exastro logo, the title 'Basic Console', and user information: 'User name [System Administrator]' and 'Login ID [administrator]'. There are buttons for 'Change password' and 'Logout'. A left sidebar menu lists various options: 'Menu', 'Main menu', 'OS type master', 'Device list', 'Input operation list', 'Movement list', 'Symphony Interface Information', 'Symphony class List', 'Symphony class editor', 'Symphony execution', 'Symphony execution checking', and 'Symphony execution list'. The 'OS type master' menu is selected, showing a list of functions: 'Description', 'Display filter', 'List/Update', 'Register', and 'Download all and edit file uploads'. Below this list are buttons for 'Download all (Excel)' and 'Download for new registration (Excel)'. There is also an 'Upload status' section with an 'Upload file' button. At the bottom, there is a 'Trace history' section with an 'OS type ID' input field and 'Display' and 'Reset' buttons. A red box highlights the 'Register' and 'Download all and edit file uploads' functions.

POINT

**Refer to the manual  
for the details of each function.**

### Submenu outline

- Register** : Allows you to register records from the Web.
- Download all and edit file uploads** : Allows for IN/OUT processing with Excel.
- Trace history** : Allows you to display the track changes of registered records.

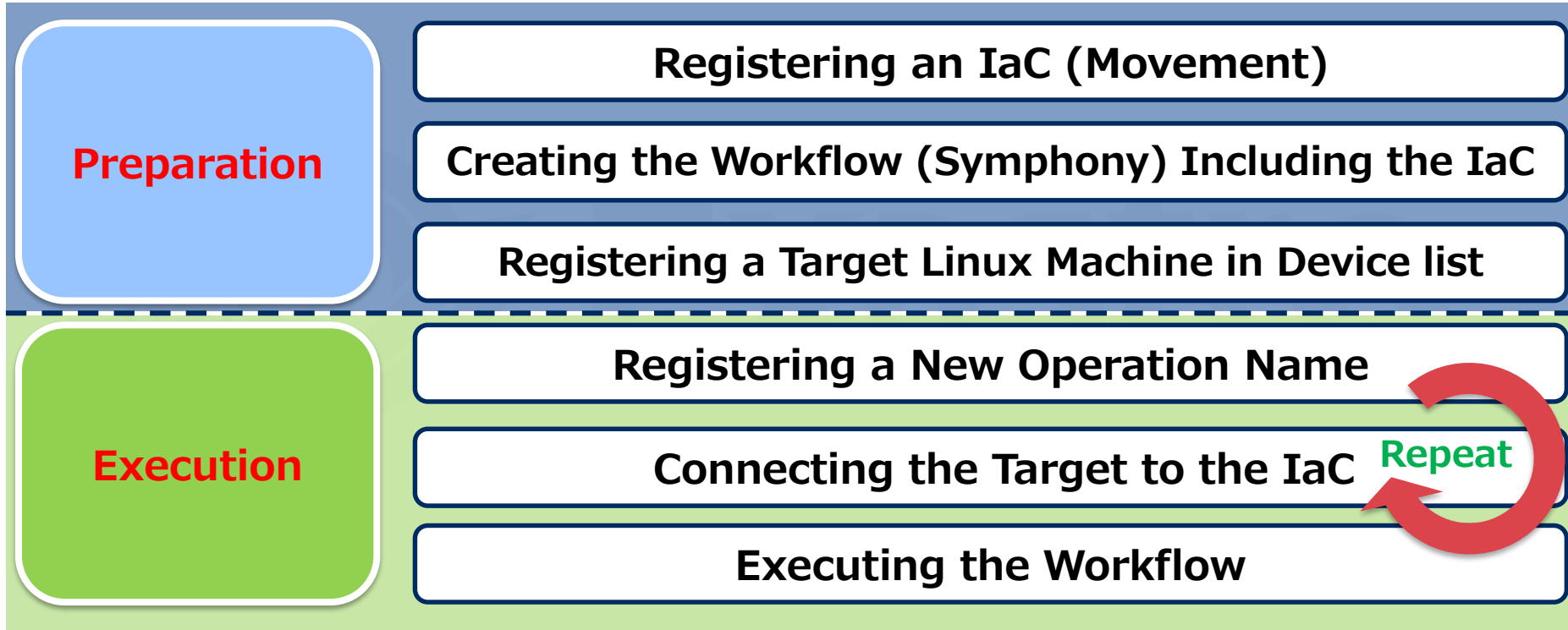
## 2. Procedure Description



## 2.1 Overall Procedure and Work Scope

Post-deployment procedure including executing Ansible-Legacy

- The following illustrates the overall procedure and work scope for developers/operators:



### POINT

The preparation contains IaC registration and workflow creation. The execution includes repeatedly performing the registered workflow.

### 3. Preparation

## 3.1 Registering an IaC (1/3)

### Registering a new Movement in **Movement list**

- From the **Main menu** screen, select **Ansible-Legacy** > **Movement list**. To start the registration, click the **Register** button.

The screenshot shows the Exastro Ansible-Legacy web interface. The top navigation bar includes the Exastro logo, the text 'Ansible-Legacy', and user information: 'User name [System Administrator]' and 'Login ID [administrator]'. There are 'Change password' and 'Logout' buttons. A left sidebar menu contains 'Menu', 'Main menu', 'Movement list' (highlighted), 'Playbook files', 'Movement details', 'Substitution value auto-registration setting', 'Target host', 'Substitution value list', 'Execution', 'Check operation status', and 'Execution list'. The main content area shows a table with columns: 'Description', 'Display filter', 'List/Update', and 'Register'. The 'Register' button is circled in red with a '1' in a speech bubble. Below the table is a form for registering a new movement. The form has a red border and contains the following fields: 'Movement ID' (Auto-input), 'Movement Name\*' (text input), 'Delay timer' (text input), 'Host specific format\*' (dropdown), and 'WinRM connection' (dropdown). A red box highlights the 'Movement Name\*' and 'Host specific format\*' fields. Below the form is a red button labeled 'Register' with a '2' in a speech bubble. At the bottom of the form, there is a note: '※ \* is a required item.' Below the form are two buttons: 'Download all and edit file uploads' and 'Trace history'. At the bottom of the page, there is a 'Contact administrator' button.

Description	Open
Display filter	Open
List/Update	Open
Register	Close

Movement ID	Movement Name*	Delay timer	Host specific format*	WinRM connection
Auto-input	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

※ \* is a required item.

Back Register

Download all and edit file uploads

Trace history

Contact administrator

#### Registering an IaC

Creating the Workflow Including the IaC

Registering a Target Linux Machine in Device list

Registering a New Operation Name

Connecting the Target to the IaC

Executing the Workflow

#### POINT

The following are mandatory fields:

- **Movement Name**
- **Host specific format**

\*Movement: A name of minimum work

## 3.2 Registering an IaC (2/3)

### Registering a new playbook in **Playbook files**

- From the **Main menu** screen, select **Ansible-Legacy** > **Playbook files**. To start the registration, click the **Register** button.

\*If no playbook is prepared, use any of the playbooks described in Appendix (**Reference 4**).

The screenshot shows the 'Ansible-Legacy' interface with a sidebar menu on the left. The 'Playbook files' option is selected. The main area displays a form for registering a new playbook. A red box labeled '1' highlights the 'Playbook name\*' and 'Playbook files\*' fields. A red box labeled '2' highlights the 'Register' button. A red callout box states: '\*When uploading a playbook, specify the file and then make sure to click the **Upload in advance** button.'

Exastro IT Automation Ansible-Legacy

User name [System Administrator]  
Login ID [administrator]

Menu

- Main menu
- Movement list
- Playbook files**
- Movement details
- Substitution value auto-registration setting
- Target host
- Substitution value list
- Execution
- Check operation status
- Execution list

Description

Display filter

Update

Register

Playbook name\* Playbook files\* Remarks Last update

Auto-input

Upload in advance

Upload status:

※\* is a required item.

Back Register

Contact administrator

#### Registering an IaC

Creating the Workflow Including the IaC

Registering a Target Linux Machine in Device list

Registering a New Operation Name

Connecting the Target to the IaC

Executing the Workflow

#### POINT

The following are mandatory fields:

- **Playbook name**
- **Playbook files**

## 3.3 Registering an IaC (3/3)

### Registering in **Movement details**

- From the **Main menu** screen, select **Ansible-Legacy** > **Movement details**. To start the registration, click the **Register** button.

Exastro IT Automation Ansible-Legacy

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

Menu

- Main menu
- Movement list
- Playbook files
- Movement details**

Substitution value auto-registration setting

Target host

Substitution value list

Execution

Check operation status

Execution list

Contact administrator

Description

Display filter

List/Update

1

In the **Include order** field, specify the order in which multiple playbooks are registered for each Movement. For 1:1, enter 1.

Associated item No.	Movement*	Playbook files*	Include order*	Remarks	Last
Auto-input					Auto

※ \* is a required item.

Back Register

2

Download all and edit file uploads

Trace history

#### Registering an IaC

Creating the Workflow Including the IaC

Registering a Target Linux Machine in Device list

Registering a New Operation Name

Connecting the Target to the IaC

Executing the Workflow

#### POINT

The following are mandatory fields:

- **Movement**
- **Playbook files**
- **Include order**

## 3.4 Creating the Workflow Including the IaC

### Creating a workflow in **Symphony class editor**

- From the **Main menu** screen, select **Basic Console** > **Symphony class editor**.

The screenshot displays the 'Basic Console' interface of Exastro IT Automation. The left sidebar contains a 'Menu' with options like 'Main menu', 'OS type master', 'Device list', 'Input operation list', 'Movement list', 'Symphony Interface information', 'Symphony class List', 'Symphony class editor', 'Symphony execution', 'Symphony execution checking', and 'Symphony execution list'. The main area is titled 'Edit Symphony' and contains a form for creating a new Symphony class. The form includes fields for 'Symphony class ID' and 'Auto numbering' (annotated with a red box and '1' and the text 'Register a Symphony name.'), and a 'Symphony class name' field. Below these is a 'Start' button and a large yellow area for the workflow. A 'move01' operation is shown in the workflow, with a red box and '2' and the text 'Drag and drop' pointing to it. A red box and '3' and the text 'Text such as a work description can be entered.' point to the 'move01' operation. At the bottom of the form is a 'Register' button (annotated with a red box and '3'). The top right of the console shows the user name 'System Administrator' and login ID 'administrator', with 'Logout' and 'Password' buttons. The bottom left has a 'Contact administrator' button.

Registering an IaC

Creating the Workflow Including the IaC

Registering a Target Linux Machine  
in Device list

Registering a New Operation Name

Connecting the Target  
to the IaC

Executing the Workflow

POINT

From the list of created Movements,  
drag and drop the desired Movement  
to register it.

## 3.5 Registering a Target Linux Machine in Device list

### Registering a new target host in **Device list**

- From the **Main menu** screen, select **Basic Console** > **Device list**. To start the registration, click the **Register** button.

The screenshot shows the 'Basic Console' interface of Exastro IT Automation. The left sidebar contains a 'Menu' with options: Main menu, OS type master, **Device list** (highlighted), Input operation list, Movement list, Symphony Interface information, Symphony class List, Symphony class editor, Symphony execution, Symphony execution checking, and Symphony execution list. The main content area shows the 'Device list' section with a 'Register' button highlighted by a red circle and the number 4. Above the registration form, there are links for 'Description', 'Display filter', 'List/Update', and 'Register' (highlighted by a red circle and the number 3). The registration form itself is a table with columns: Managed system item number, HW device type, Host name\*, IP address\*, MAC address, Network device name, Login user ID, and Login password. The 'Host name\*' and 'IP address\*' fields are marked as required. Below the form, there is a 'Back' button and a 'Register' button. A 'POINT' callout box is present, listing mandatory fields for Ansible-Legacy execution.

Managed system item number	HW device type	Host name*	IP address*	MAC address	Network device name	Login user ID	Login password
Auto-input							

\* is a required item.

Back Register

#### Registering an IaC

#### Creating the Workflow Including the IaC

#### Registering a Target Linux Machine in Device list

#### Registering a New Operation Name

#### Connecting the Target to the IaC

#### Executing the Workflow

#### POINT

For executing Ansible-Legacy, the following are mandatory fields:

- Host name
- IP address
- Login user ID
- Login password management
- Login password
- Authentication method\*

\*This document describes it as password authentication.

## 4. Execution



# 4.1 Registering a New Operation Name

## Registering a new operation name on **Input operation list**

- From the **Main menu** screen, select **Basic Console** > **Input operation list**. To start the registration, click the **Register** button.
- \*Operation refers to the **operation name** used in the IT Automation system that indicates the whole operation.

Registering an IaC
Creating the Workflow Including the IaC
Registering a Target Linux Machine in Device list
Registering a New Operation Name
Connecting the Target to the IaC
Executing the Workflow

### POINT

- The following are mandatory fields:
- **Operation name**
  - **Scheduled date for execution**

## 4.2 Connecting the Target to the IaC

### Registering on Target host

- From the **Main menu** screen, select **Ansible-Legacy** > **Target host**. To start the registration, click the **Register** button.

Menu

- Main menu
- Movement list
- Playbook files
- Movement details
- Template list
- File list
- Substitution value auto-registration setting
- Target host**
- Substitution value list
- Execution
- Check operation status
- Execution list

Description ▾Open

Display filter ▾Open

List/Update ▾Open

Register △Close

Item No. Operation\* Movement\* Host\* /time Last updated by

Auto-input ▾ ▾ ▾ Auto-input

※\* is a required item.

Back Register

Download all and edit file uploads ▾Open

Trace history ▾Open

1

From the **Host** list, select the desired target device.

2

Registering an IaC

Creating the Workflow Including the IaC

Registering a Target Linux Machine  
in Device list

Registering a New Operation Name

Connecting the Target  
to the IaC

Executing the Workflow

POINT

The following are mandatory fields:

- **Operation**
- **Movement**
- **Host**

## 4.3 Executing the Workflow

### Executing Symphony

- From the **Main menu** screen, select **Basic Console** > **Symphony execution**.

The screenshot shows the Symphony execution interface. On the left is a 'Menu' sidebar with options: Main menu, OS type master, Device list, Input operation list, Movement list, Symphony Interface information, Symphony class List, Symphony class editor, Symphony execution, Symphony execution checking, and Symphony execution list. The main area is divided into sections for Symphony and Operation management.

**POINT**

From **Symphony** and **Operation**, select items to be executed.  
\*General procedure manual > Symphony  
\*Replacement table > Operation

**1** Select Symphony class ID, Symphony name, Description, Remarks, Last update date. Filter result count: 1.

**2** Select No., Operation ID, Operation name, Scheduled date for execution, Last execute date, Remarks, Last update date. Filter result count: 1.

**3** Execute.

**Side Menu:**

- Registering an IaC
- Creating the Workflow Including the IaC
- Registering a Target Linux Machine in Device list
- Registering a New Operation Name
- Connecting the Target to the IaC
- Executing the Workflow

# A Appendix

# Reference 1 Ansible-Legacy: Single Execution

## Execution

- In Ansible-Legacy, the **Execution** menu offers the **Execute** and **Dry run** functions for each Movement.

The screenshot shows the Ansible-Legacy web interface. The left sidebar contains a menu with options: Main menu, Movement list, Playback files, Movement details, Template list, File list, Substitution value auto-registration setting, Target host, Substitution value list, Execution, Check operation status, and Execution list. The main content area is divided into several sections. The top section is titled 'Description' and 'Scheduling'. Below this is a 'Movement [Filter]' section with a 'Movement [List]' table. A red callout with the number '1' points to the first row of the table, which is highlighted in red. The table has columns: Select, Movement ID, Movement Name, Orchestrator, Data source, Variable course, and Implement. The first row contains: 1, move1, Ansible Legacy, IP, and Implement. Below the table is a 'Filter result count: 1' label. The next section is 'Operation [Filter]' with an 'Operation [List]' table. A red callout with the number '2' points to the first row of the table, which is highlighted in red. The table has columns: Select, No., Operation ID, Operation name, Scheduled date for execution, Last execute date, Remarks, Last update date/time, and Last updated by. The first row contains: 1, 1, operation, 2020/05/01 15:00, 2019/05/10 14:18:03, System Administrator. Below the table is a 'Filter result count: 1' label. On the right side of the interface, there is a 'Movement [Filter]' section with a 'Movement [List]' table. A red callout with the number '3' points to the 'Dry run' and 'Execute' buttons at the bottom of the page. The 'Dry run' button is highlighted in orange. The 'Execute' button is highlighted in orange.

**1** Select the created Movement

**2** Select the operation connected to the Movement

**3** Dry run: Checks the playbook connection/syntax  
Execute: Executes the playbook

# Reference 2 Ansible-Legacy: Checking the Operation Results

## Checking the operation results

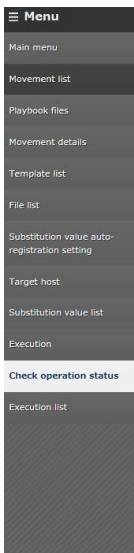
- Performing the function (**Execute** or **Dry run**) displays the execution status and logs.

Execution status:  
Allows you to check  
the execution evidence  
and the input data.

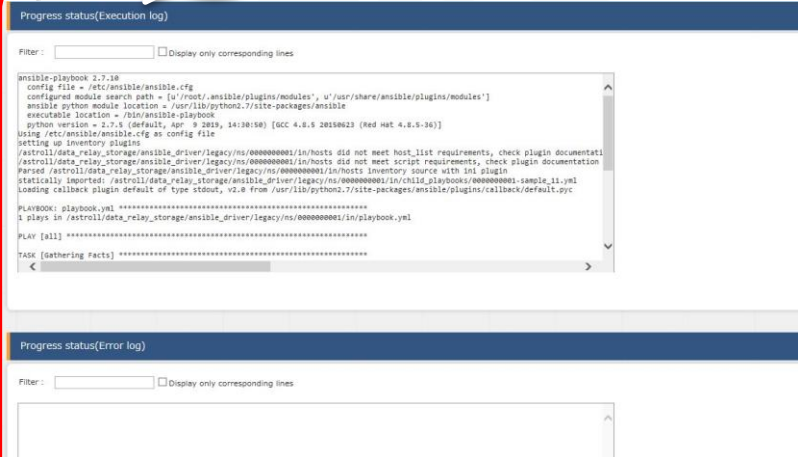
point

point

Execution log and Error log:  
Can be checked in real time.



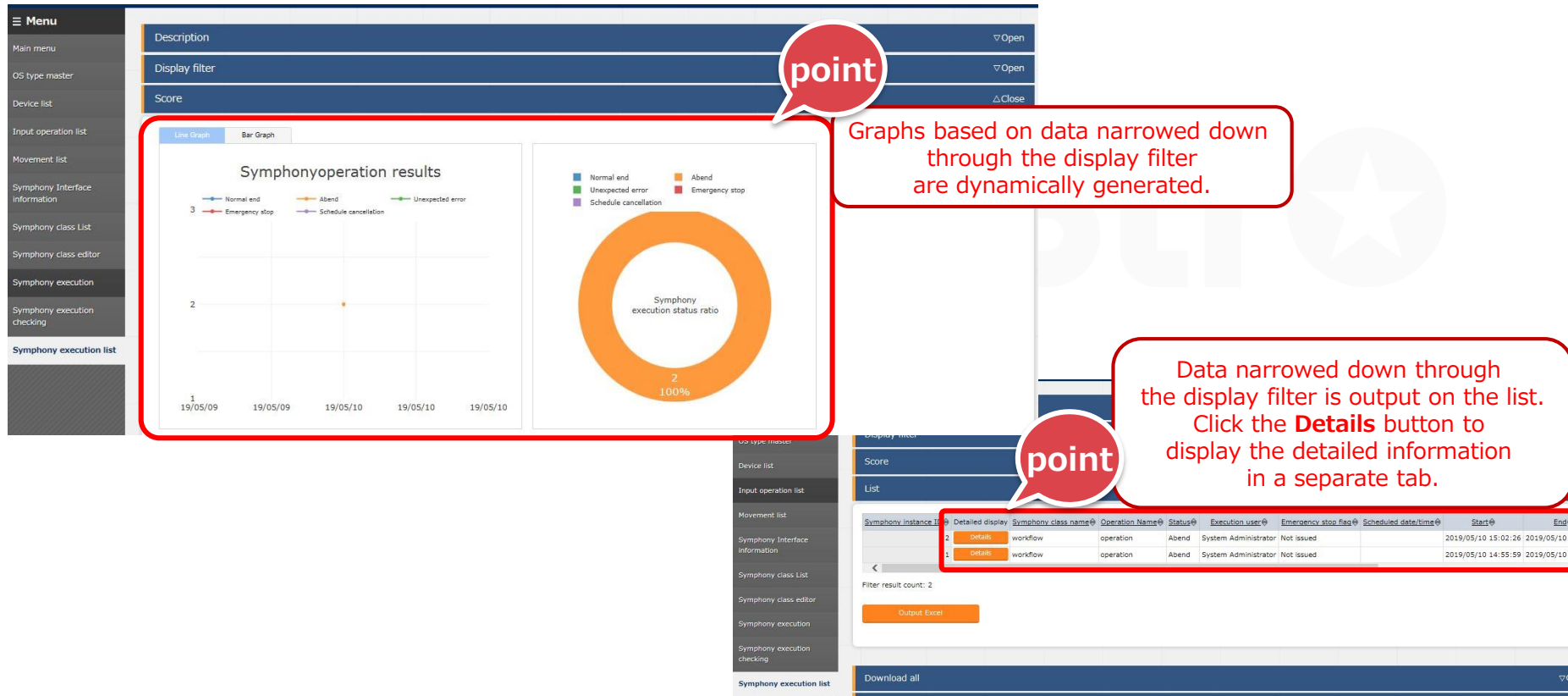
Description				▽Open
Target Operation				△Close
Item		Value		
Execution type		Normal		
Status		Completed (error)		
Execution user		System Administrator		
Movement	ID	1		
	Name	move1		
	Delay timer (minutes)			
Dedicated information for ansible	Host specific format	IP		
	Number of parallel execution			
	WinRM connection	gather_facts		
Implementation		Implement		
Operation	No.	1		
	Name	operation		
Host management	ID	1		
		confirmation		
Substitution value		confirmation		
Input data		Populated data		
Output data		Result data		
Scheduled date/time				
Operation status		Start date/time		
		End date/time		



# Reference 3 How to Check the Symphony Execution Results

## Checking the execution results on **Symphony execution list**

- From the **Main menu** screen, select **Basic Console** > **Symphony execution list**.



# Reference 4 Sample Collection of Playbook

## Sample playbooks (for Linux servers)

- The following playbooks are samples.

They can be used as is, but you can freely change the parts in red.

\*The character code is "UTF-8", the line feed code is "LF", and the extension is "yml" format.

Keep the indents in mind.

- name: Make Work Directory demonstration file:

path: `/tmp/demodirectory`  
state: `directory`  
mode: `0755`

point

A directory called "demodirectory" is created under the /tmp directory.

- name: Sample User add user:

name: `ITA`  
createhome: `no`  
uid: `4401`  
group: `users`

point

An ITA user is created.  
Delete the user after checking the operation.

- name: Collect Files fetch:

src=`/etc/hosts`  
dest={{ \_\_workflowdir\_\_ }}/{{ inventory\_hostname }}  
flat=yes

point

The following definition is reserved variables prepared in advance that are used when a file is brought back to the IT Automation server.

{{ \_\_workflowdir\_\_ }}/{{ inventory\_hostname }}

point

/etc/hosts files are collected.  
The collected files are gathered in the zip file of the result data.





**Exastro**