



# IT Automation Conductor 【実習編】

※本書では「Exastro IT Automation」を「ITA」として記載します。

Exastro IT Automation Version 1.8  
Exastro developer

# 目次

## 1. はじめに

### 1. 本書について

## 2. Conductor

### 1. シナリオ

### 2. 事前準備

## 3. 実習

### 1. 作業対象ホストの登録

### 2. オペレーションの登録

### 3. IaCの登録

### 4. Movementの登録

### 5. Movement詳細の登録

### 6. オペレーションに関連付く Movementとホストの登録

### 7. 代入値管理

### 8. Conductorの登録

### 9. Conductorの実行

### 10. Conductorの完了確認

# 1. はじめに

# 1.1 本書について

- 本書では、メニューグループの「**Conductor**」について、ご説明をしております。

The screenshot displays the Exastro IT Automation dashboard. The top navigation bar includes the Exastro logo, a user profile for 'ようこそ[システム管理者]さん' (Welcome [System Administrator]), and login/logout buttons. The main content area is divided into a left sidebar with a 'Menu' button and a 'DASHBOARD' header. The 'メニューグループ' (Menu Group) section is visible, showing a grid of icons. The 'Conductor' icon, which depicts a circuit board, is highlighted with a red rectangular box. Other icons in the grid include '管理コンソール', '基本コンソール', 'エクスポート/イン...', 'Symphony', 'メニュー作成', '比較', 'ホストグループ...', 'Ansible共通', 'Ansible-Legacy', 'Ansible-Pioneer', 'Ansible-Legacy...', 'Cobbler', 'Terraform', and 'CI/CD for IaC'. The right side of the dashboard features three large circular gauges labeled 'Movement', 'Status', and 'Result', each showing a 'Total' value of 0. Below these are three tables: 'Movement' with columns for 'Movement' and 'SUM'; 'Status' with columns for 'Status', 'CON', 'SYM', and 'SUM'; and 'Result' with columns for 'Result', 'CON', 'SYM', and 'SUM'. Each table contains a list of items with corresponding status indicators. At the bottom right, there is a section titled '作業履歴' (Work History) with a table of logs.

## 2. Conductorについての説明

## 2.1 シナリオ (1/2)

- 本シナリオは以下の流れとなります。
- また、シナリオを進めるにあたり、Ansible driverが必要となりますので、本シナリオでは、Ansible-Legacyを使用しご説明をいたします。

①機器情報の登録

基本コンソールメニュー

②オペレーションの登録

③Movementの登録

各種Driverメニュー

④Movementの確認

⑤インターフェース情報を登録

Conductorメニュー

⑥Conductorの登録

⑦Conductorの確認

⑧Conductorの実行

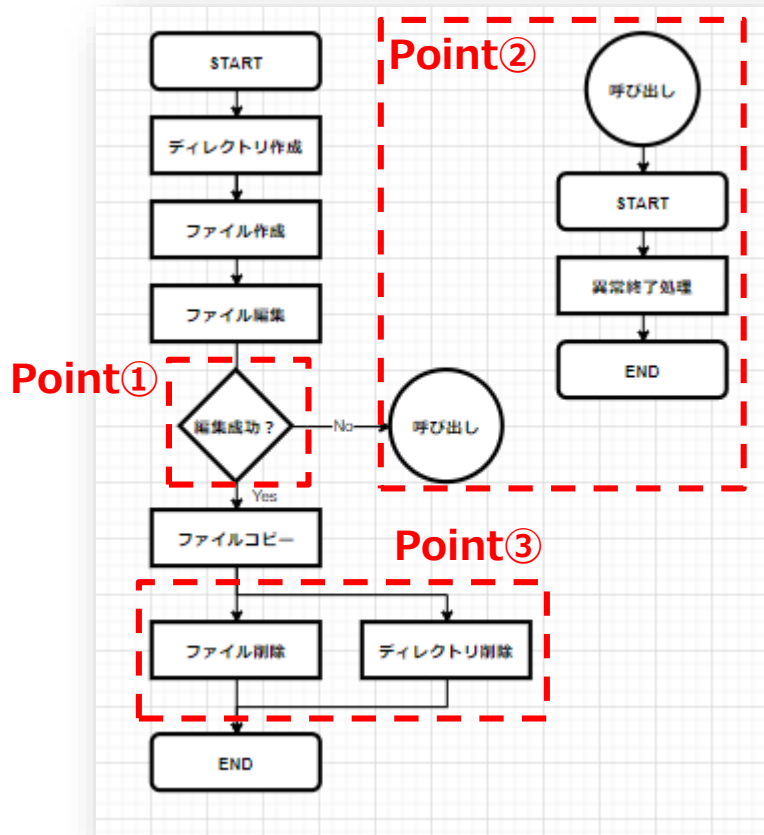
⑨実行結果確認

⑩実行履歴の確認

## 2.1 シナリオ (2/2)

- 本編ではConductor機能を体感いただくために、以下のフローチャートと同様のConductorを作成します。

- フローチャート



### □Conductor機能の特徴

#### □Point①

前処理の成功/終了判断による条件分岐機能

#### □Point②

登録済のOperation/Conductorの呼び出し機能

#### □Point③

Movementの並行処理機能

## 2.2 事前準備

### ●IaCの作成(1/2)

本シナリオでは、Ansible-Legacyを例にご説明します。

下記のIaCをモジュールごとにymlファイルとして保存してください。

※文字コードは"UTF-8"、改行コードは"LF"、拡張子は"yml"形式。  
また、インデントにご注意下さい。

```
- name: create directory
  file:
    path: /tmp/{{ item.dir }}
    state: directory
    mode: 0755
  with_items:
    - { dir: "{{ VAR_dir_name_1 }}" }
    - { dir: "{{ VAR_dir_name_2 }}" }

- 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
```



## 2.2 事前準備

### ●IaCの作成(2/2)

以下も同様にモジュールごとにymlファイルとして保存してください。

- 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  
  remote\_src: yes
- 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 }}

#### ・作成後イメージ

名前	更新日時	種類	サイズ
 copy_file.yml	2021/09/01 16:57	YML ファイル	1 KB
 create_directory.yml	2021/09/01 16:58	YML ファイル	1 KB
 create_file.yml	2021/09/01 16:58	YML ファイル	1 KB
 edit_file.yml	2021/09/01 16:58	YML ファイル	1 KB
 forced_termination.yml	2021/09/01 16:58	YML ファイル	1 KB
 remove_directory.yml	2021/09/01 16:58	YML ファイル	1 KB
 remove_file.yml	2021/09/01 16:58	YML ファイル	1 KB

### 3. 実習

## 3.1 作業対象ホストの登録

### ●作業対象ホストの登録

「基本コンソール」メニューグループ >> 「機器一覧」メニュー >> 「登録」サブメニュー >> 「登録開始」ボタン

- ① 「ホスト名」「IPアドレス」「ログインユーザID」「管理」「ログインパスワード」「認証方式」を入力
- ② 「登録」ボタンを押下

1

項目へ値を入力する

項目	値
ホスト名	testserver
IPアドレス	(任意の値)
ログインユーザID	(任意の値)
管理	●
ログインパスワード	(任意の値)
認証方式	パスワード認証

#### POINT

本シナリオでは、作業対象ホストにsshのパスワード接続を行う場合を想定しています。  
「IPアドレス」「ログインユーザID」「ログインパスワード」については  
ユーザ様のご利用環境に適した設定をご入力ください。

## 3.2 オペレーションの登録

### ●オペレーションの登録

「基本コンソール」メニューグループ >> 「オペレーション一覧」メニュー >> 「登録」サブメニュー >> 「登録開始」ボタン

- ① 「オペレーション名」「実施予定日時」を入力
- ② 「登録」ボタンを押下

登録

△開じる

No.	オペレーションID	オペレーション名*	実施予定日時*	アクセス権	設定
自動入力	自動入力			アクセス許可ロール	

※\*は必須項目です。

戻る 登録

項目へ値を入力する

項目	値
オペレーション名	operation1
実施予定日時	(任意の日時)

全件ダウンロードとファイルアップロード編集 ▽開く

変更履歴 ▽開く

#### POINT

ここで指定した日時に  
処理が実行されるわけではありません

## 3.3 IaCの登録 (1/2)

### ●IaCの登録

「Ansible-Legacy」メニューグループ >> 「Playbook素材集」メニュー >> 「登録」サブメニュー >> 「登録開始」ボタン

- ① 「Playbook素材名」を入力、  
「Playbook素材」欄の「ファイルを選択」ボタンを押下し  
**事前に作成したyamlファイルをすべてをアップロード**  
(「事前アップロード」ボタン押下)
- ② 「登録」ボタンを押下

項目	値
Playbook素材名	<任意>
Playbook素材	<任意> .yaml

#### POINT

IaCの作成手順につきましては、  
「2.2事前準備」をご参照下さい

## 3.3 IaCの登録 (2/2)

### ●IaCの登録

作成後のイメージは以下ようになります。

一覧/更新

履歴	複製	更新	廃止	素材ID	Playbook素材名	Playbook素材	Movement-Playbook紐付	アクセス権	備考	最終更新日時	最終更新者
								アクセス許可ロール			
履歴	複製	更新	廃止	1	copy_file	<a href="#">copy_file.yml</a>	Movement-Playbook紐付			2021/09/09 14:37:04	システム管理者
履歴	複製	更新	廃止	2	create_directory	<a href="#">create_directory.yml</a>	Movement-Playbook紐付			2021/09/09 14:37:10	システム管理者
履歴	複製	更新	廃止	3	create_file	<a href="#">create_file.yml</a>	Movement-Playbook紐付			2021/09/09 14:37:16	システム管理者
履歴	複製	更新	廃止	4	edit_file	<a href="#">edit_file.yml</a>	Movement-Playbook紐付			2021/09/09 14:37:24	システム管理者
履歴	複製	更新	廃止	5	forced_termination	<a href="#">forced_termination.yml</a>	Movement-Playbook紐付			2021/09/09 14:37:34	システム管理者
履歴	複製	更新	廃止	6	remove_directory	<a href="#">remove_directory.yml</a>	Movement-Playbook紐付			2021/09/09 14:37:41	システム管理者
履歴	複製	更新	廃止	7	remove_file	<a href="#">remove_file.yml</a>	Movement-Playbook紐付			2021/09/09 14:36:56	システム管理者

フィルタ結果件数: 7

Excel出力

## 3.4 Movementの登録 (1/2)

### ●Movementの登録

「Ansible-Legacy」メニューグループ >> 「Movement一覧」メニュー >> 「登録」サブメニュー >> 「登録開始」ボタン

- ① 「Movement名」「ホスト指定形式」を入力
- ② 「登録」ボタンを押下

Exastro IT Automation Ansible-Legacy

ようこそ(システム管理者)さん  
ログインID [administrator]

メニュー

メインメニュー

Movement一覧

Playbook素材集

Movement-Playbook紐付

代入値自動登録設定

作業対象ホスト

代入値管理

作業実行

作業状態確認

作業管理

説明

表示フィルタ

一覧/更新

登録

項目へ値を設定する

項目	値
Movement名	<任意>
ホスト指定形式	IP

Movement ID Movement名 遅延タイマー ホスト指定形式 WinRM接続 ヘッダーセッション 最終更新日時 最終更新者

自動入力

※\*は必須項目です。

戻る 登録

全件ダウンロードとファイルアップロード編集

変更履歴

POINT

作成するMovementは  
yamlファイルと同数を作成して下さい

## 3.4 Movementの登録 (2/2)

### ●Movementの登録

登録後のイメージは以下ようになります。

一覧/更新 △閉じる

履歴	複製	更新	廃止	MovementID	Movement名	オーケストレータ	遅延タイマー	Ansible利用情報				最終更新日時	最終更新者
								ホスト指定形式	WinRM接続	ヘッダーセクション	オブジェクト		
履歴	複製	更新	廃止	1	copy_file	Ansible Legacy		IP				2021/09/09 14:47:48	システム管理者
履歴	複製	更新	廃止	2	create_directory	Ansible Legacy		IP				2021/09/09 14:48:00	システム管理者
履歴	複製	更新	廃止	3	create_file	Ansible Legacy		IP				2021/09/09 14:48:11	システム管理者
履歴	複製	更新	廃止	4	edit_file	Ansible Legacy		IP				2021/09/09 14:48:22	システム管理者
履歴	複製	更新	廃止	5	forced_termination	Ansible Legacy		IP				2021/09/09 14:48:35	システム管理者
履歴	複製	更新	廃止	6	remove_directory	Ansible Legacy		IP				2021/09/09 14:48:48	システム管理者
履歴	複製	更新	廃止	7	remove_file	Ansible Legacy		IP				2021/09/09 14:48:58	システム管理者

フィルタ結果件数: 7

Excel出力



## 3.5 Movement詳細の登録 (1/2)

### ●Movement詳細の登録

「Ansible-Legacy」メニューグループ >> 「Movement-Playbook紐付」メニュー >> 「登録」サブメニュー >> 「登録開始」ボタン

- ① 「Movement」「Playbook素材」「インクルード順序」を入力
- ② 「登録」ボタンを押下

Exastro IT Automation Ansible-Legacy

ようこそ[システム管理者]さん  
ログインID [administrator]  
パスワード変更 ログアウト

Menu

メインメニュー  
Movement一覧  
Playbook素材集  
Movement-Playbook紐付  
代入値自動登録設定  
作業対象ホスト  
代入値管理  
作業実行  
作業状態確認  
作業管理

説明  
表示フィルタ  
一覧/更新  
登録

項目名 Movement Playbook素材 インクルード順序 アクセス権 備考 最終更新日時 最終更新者

自動入力 設定 自動入力 自動入力

※\*は必須項目です。

戻る 登録

項目へ値を設定する

項目	値
Movement	作成した Movementを選択
Playbook素材	登録した Playbookを選択
インクルード順序	1

POINT

登録するMovement詳細は  
yamlファイルと同数を作成して下さい

## 3.5 Movement詳細の登録 (2/2)

### ●Movement詳細の登録

登録後のイメージは以下になります。

一覧/更新

△閉じる

履歴	複製	更新	廃止	紐付項番	Movement	Playbook素材	インクルード順序	アクセス権	備考	最終更新日時	最終更新者
								アクセス許可ロール			
履歴	複製	更新	廃止	1	<a href="#">1:copy_file</a>	<a href="#">copy_file</a>	1			2021/09/08 15:49:19	システム管理者
履歴	複製	更新	廃止	2	<a href="#">2:create_directory</a>	<a href="#">create_directory</a>	1			2021/09/09 14:51:31	システム管理者
履歴	複製	更新	廃止	3	<a href="#">3:create_file</a>	<a href="#">create_file</a>	1			2021/09/09 14:51:39	システム管理者
履歴	複製	更新	廃止	4	<a href="#">4:edit_file</a>	<a href="#">edit_file</a>	1			2021/09/09 14:51:49	システム管理者
履歴	複製	更新	廃止	5	<a href="#">5:forced_termination</a>	<a href="#">forced_termination</a>	1			2021/09/09 14:51:58	システム管理者
履歴	複製	更新	廃止	6	<a href="#">6:remove_directory</a>	<a href="#">remove_directory</a>	1			2021/09/09 14:52:07	システム管理者
履歴	複製	更新	廃止	7	<a href="#">7:remove_file</a>	<a href="#">remove_file</a>	1			2021/09/09 14:52:15	システム管理者

フィルタ結果件数: 7

Excel出力

## 3.6 オペレーションに関連付くMovementとホストの登録

### ●オペレーションに関連付くMovementとホストの登録

「Ansible-Legacy」メニューグループ >> 「作業対象ホスト」メニュー >> 「登録」サブメニュー >> 「登録開始」ボタン

- ① 「オペレーション」「Movement」「ホスト」を入力
- ② 「登録」ボタンを押下

Exastro IT Automation Ansible-Legacy

ようこそ[システム管理者]さん  
ログインID [administrator]  
パスワード変更 ログアウト

Menu

- メインメニュー
- Movement一覧
- Playbook素材集
- Movement-Playbook紐付
- 代入値自動登録設定

作業対象ホスト

- 代入値管理
- 作業実行
- 作業状態確認
- 作業管理

説明

表示フィルタ

一覧/更新

登録

項目 オペレーション Movement ホスト

自動入力

アクセス権

設定

最終更新日時

最終更新者

自動入力

※\*は必須項目です。

戻る 登録

全件ダウンロードとファイルアップロード編集

変更履歴

POINT

作成したMovementは  
すべて登録を行ってください

1

項目へ値を設定する

項目	値
オペレーション	operation1
Movement	作成した Movement
ホスト	testserver

## 3.7 代入値管理 (1/2)

### ●代入値管理

「Ansible-Legacy」メニューグループ >> 「代入値管理」メニュー >> 「登録」サブメニュー >> 「登録開始」ボタン

- ① 「オペレーション」「Movement」「ホスト」「変数名」「具体値」を入力
- ② 「登録」ボタンを押下

Exastro IT Automation Ansible-Legacy

ようこそ[システム管理者]さん  
ログインID [administrator]  
パスワード変更 ログアウト

Menu

- メインメニュー
- Movement一覧
- Playbook素材集
- Movement-Playbook紐付
- 代入値自動登録設定
- 作業対象ホスト

代入値管理

- 作業実行
- 作業状態確認
- 作業管理

説明 ▽開く

表示フィルタ ▽開く

一覧/更新 ▽開く

登録 △閉じる

順番	オペレーション*	Movement*	ホスト*	変数名*	Sensitive設定	最終更新日時	最終更新者
自動入力 ▾	オペレーションを選択して下さい	Movementを選択して下さい	ホストを選択して下さい	Movementを選択して下さい	0... ▾	自動入力	自動入力

※\*は必須項目です。

戻る 登録

全件ダウンロードとファイルアップロード編集 ▽開く

変更履歴 ▽開く

## 3.7 代入値管理 (2/2)

### ●代入値管理

代入値の登録は以下を参考に行ってください。

オペレーション	ホスト	Movement : 変数	具体値	代入順序
1:operation1	1:Testserver	1:copy_file:1:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	1:copy_file:2:VAR_file_name	testfile	
1:operation1	1:Testserver	1:copy_file:3:VAR_dir_name_2	dir2	
1:operation1	1:Testserver	2:create_directory:4:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	2:create_directory:5:VAR_dir_name_2	dir2	
1:operation1	1:Testserver	3:create_file:6:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	3:create_file:7:VAR_file_name	testfile	
1:operation1	1:Testserver	4:edit_file:8:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	4:edit_file:9:VAR_file_name	testfile	
1:operation1	1:Testserver	4:edit_file:10:VAR_edit_param_1	param1	
1:operation1	1:Testserver	5:forced_termination:11:VAR_message_text	testmsg_fail	
1:operation1	1:Testserver	6:remove_directory:12:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	7:remove_file:13:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	7:remove_file:14:VAR_file_name_1	testfile	

## 3.8 Conductorの登録 (1/7)

### ●Conductorの登録

「Conductor」メニューグループ >> 「Conductorクラス編集」 >>  
「Conductor Name」を入力

- ① 画面右側に表示されている「Movement」「Function」を画面中央にドラッグ&ドロップ
- ② 「登録」ボタンを押下

項目へ値を入力する

項目	値
Name	Conductor_1

1

2

3

ドラッグ&ドロップ

※作成するConductorは次ページを参照してください

## 3.8 Conductorの登録 (2/7)

### ●Conductorの登録

以下のようにConductorを作成してください



- ①Conductor Nameに「Conductor\_2」と入力
- ②Movementから「forced\_termination」をドラッグアンドドロップ
- ③Functionから「Conductor」をドラッグアンドドロップ
- ④図のように「OUT」と「IN」をつなぐ
- ⑤画面下の「登録」を押下

Conductor名称

ID : Auto numbering

Name : Conductor\_2

Permission role

Role :

Select

Note

Movement Function

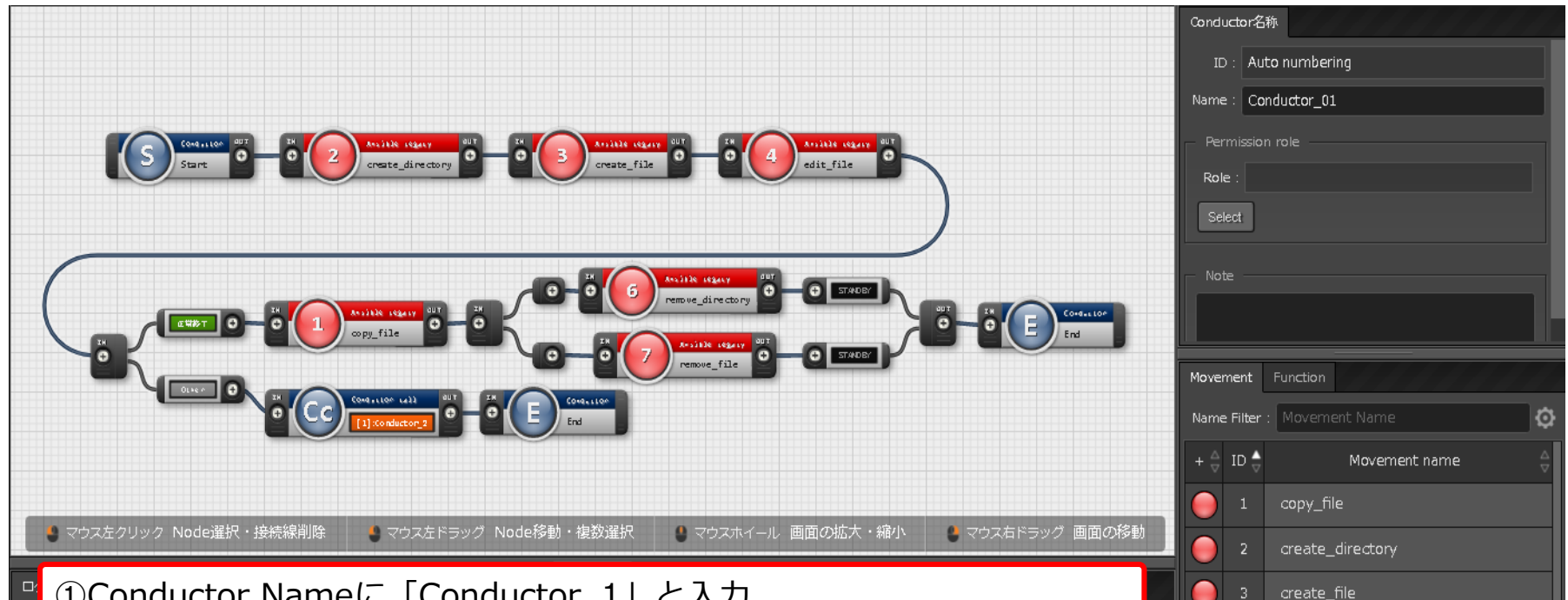
Name Filter : Movement Name

	ID	Movement name
	1	copy_file
	2	create_directory
	3	create_file
	4	edit_file
	5	forced_termination

## 3.8 Conductorの登録 (3/7)

### ●Conductorの登録

作成Conductorの全体図は以下のようになります。  
次ページ以降で細部を説明します。





## 3.8 Conductorの登録 (4/7)

### ●Conductorの登録

以下のようにConductorを作成してください

The screenshot displays the Conductor configuration interface. On the left, a workflow diagram is shown on a grid. It starts with a 'Start' node (S), followed by a sequence of nodes: 'create\_directory' (2), 'create\_file' (3), and 'edit\_file' (4). A red box highlights a 'Conditional branch' node (Cc) that is being added to the workflow. A red arrow points from the 'Conditional branch' node in the workflow to the 'Conditional branch' configuration panel on the right. The configuration panel has a 'Case' section with 'Add' and 'Delete' buttons. Below it, 'Case 1' is set to '正常終了' (Normal completion). The 'Other' section has four buttons: '異常終了' (Abnormal completion), '緊急停止' (Emergency stop), '準備エラー' (Preparation error), and '想定外エラー' (Unexpected error). A 'Skip完了' (Skip completed) button is also present. On the right side of the interface, there is a 'Function' tab selected, and a list of function types: 'Conductor pause', 'Conductor call', 'Symphony call', 'Conditional branch' (highlighted with a red box), 'Parallel branch', and 'Parallel merge'.

「Function」タブから  
Conductor branchを  
ドラッグアンドドロップして配置

Conditional branch

Case : Add Delete

Case 1 : 正常終了

Other : 異常終了 緊急停止 準備エラー 想定外エラー Skip完了

Function type

- Conductor pause
- Conductor call
- Symphony call
- Conditional branch
- Parallel branch
- Parallel merge

- ・ Conductor branchは配置された直前の Movementの終了結果に応じて次の処理を分岐します。
- ・ 今回は「正常終了」の場合のみ後続処理に続く設定にします。

## 3.8 Conductorの登録 (5/7)

### ●Conductorの登録

以下のようにConductorを作成してください

The screenshot displays the Conductor registration interface. At the top, a workflow diagram shows a sequence of steps: Start (S), create\_directory (2), create\_file (3), and edit\_file (4). Below this, a more complex diagram shows a 'Conductor call' node (Cc) connected to an 'End' node (E). A red arrow points from the 'Conductor call' node to the 'Conductor call' dialog box at the bottom left. The dialog box has fields for 'Default skip' (unchecked), 'Conductor select' (set to '[1]:Conductor\_2'), and 'Operation select' (empty). A red box highlights the 'Conductor call' node in the diagram and the 'Conductor call' dialog box. A text box in the center of the diagram reads: 「Function」タブから Conductor callを ドラッグアンドドロップして配置. On the right side, the 'Function' tab is selected in the 'Movement' section, and the 'Conductor call' option is highlighted in the 'Function type' list. A red box highlights the 'Conductor call' option in the 'Function type' list. A text box at the bottom right explains the 'Conductor call' function.

「Function」タブから  
Conductor callを  
ドラッグアンドドロップして配置

Conductor call

Default skip : ☐

Conductor select

Conductor : [1]:Conductor\_2

Conductor select Clear

Operation select

Operation :

Operation select Clear

Note

Conductor callは設定したConductor、Operationを  
呼び出して実行することができます。  
今回は事前に作成したConductor\_2を指定します。

## 3.8 Conductorの登録 (6/7)

### ●Conductorの登録

以下のようにConductorを作成してください

The screenshot displays the Exastro Conductor configuration interface. On the left, a workflow diagram is shown on a grid background. It starts with a 'Start' node (S) connected to a 'create\_directory' node (2), which then connects to a 'create' node (3). Below this, a 'copy\_file' node (1) connects to a 'remove\_directory' node (6) and a 'remove\_file' node (7). Both (6) and (7) connect to a 'Conductor end' node (E). A callout box with a blue border points to the 'Conductor end' node, containing the text: 「Function」タブから Conductor endを ドラッグアンドドロップして配置. On the right, a panel shows the 'Conductor名称' (Conductor Name) settings, including ID (Auto numbering), Name (Conductor\_01), and a 'Function' tab. Under the 'Function' tab, a list of 'Function type' options is shown, with 'Conductor end' selected and highlighted by a red box. At the bottom, a toolbar contains icons for node selection, movement, and zooming.

「Function」タブから  
Conductor endを  
ドラッグアンドドロップして配置

Conductor名称  
ID : Auto numbering  
Name : Conductor\_01  
Permission role  
Role :  
Select  
Note

Movement Function  
+ Function type  
Conductor end  
Conductor pause  
Conductor call  
Symphony call

- Conductor endは処理の終了時に配置するfunctionです。
- (5/7)にてご紹介した分岐処理の終了時にも配置しています

## 3.8 Conductorの登録 (7/7)

### ●Conductorの登録

以下のようにConductorを作成してください

The screenshot displays the Exastro software interface. On the left, a workflow diagram is shown on a grid background. It starts with a 'Start' node (S) leading to a 'create\_directory' node (2), then a 'create\_file' node (3). Below this, a 'copy\_file' node (1) is connected to a 'Conductor call' node (C) labeled '[1]-Conductor\_2'. The 'Conductor call' node is connected to an 'End' node (E). A red box highlights the 'Conductor call' node, and a red arrow points from it to the 'Function' tab in the right-hand panel. The right-hand panel shows the 'Conductor' configuration. The 'Function' tab is selected, and the 'Conductor call' option is highlighted in the 'Function type' list. A text box at the top right explains that the 'Function' tab is used to drag and drop 'Parallel branch' and 'Parallel merge' components. A 'Parallel branch' dialog box is also visible in the bottom left corner.

「Function」タブから  
Parallel branch, Parallel mergeを  
ドラッグアンドドロップして配置

Parallel branch

Case : Add Delete

Note

Conductor名称

ID : Auto numbering

Name : Conductor\_01

Permission role

Role :

Select

Note

Movement Function

+ Function type

- Conductor end
- Conductor pause
- Conductor call
- Symphony call

- Parallel branch, Parallel mergeは直後に実行する Movement, Functionを並行して実行することができます。
- 並行する処理の数は指定することが可能です。

## 3.9 Conductorの実行

### ●Conductorの実行

「Conductor」メニューグループ >> 「Conductor作業実行」メニュー

- ① 「Conductor[一覧]」サブメニュー「予約日時」項目内のから実行日時を決定
- ② 「Conductor[一覧]」サブメニュー「Conductor名称」項目内の「Conductor\_1」を選択
- ③ 「オペレーション[一覧]」サブメニュー「オペレーション名」項目内の「operation1」を選択
- ④ 「実行」ボタンを押下

The screenshot shows the 'Conductor' menu group expanded. The 'Conductor[一覧]' (Conductor List) sub-menu is active, displaying a table with columns '予約日時' (Reservation Date/Time) and 'Conductor名称' (Conductor Name). A red box labeled '1' highlights the '予約日時' column header. Another red box labeled '2' highlights the 'Conductor名称' column header. A third red box labeled '3' highlights the 'operation1' row in the 'オペレーション[一覧]' (Operation List) sub-menu. A fourth red box labeled '4' highlights the '実行' (Execute) button at the bottom right of the interface.

説明

スケジュールリング

予約日時を指定する場合は、日時フォーマットで入力してください。

予約日時

項目	値
予約日時	任意

Conductor[フィルタ]

Conductor[一覧]

選択	ConductorクラスID	Conductor名称
<input type="radio"/>	1	Conductor_1
<input type="radio"/>	2	Conductor_2

フィルタ結果件数: 2

項目	値
Conductor名称	Conductor_1

オペレーション[フィルタ]

オペレーション[一覧]

選択	No.	オペレーションID	オペレーション名
<input type="radio"/>	1	1	operation1

フィルタ結果件数: 1

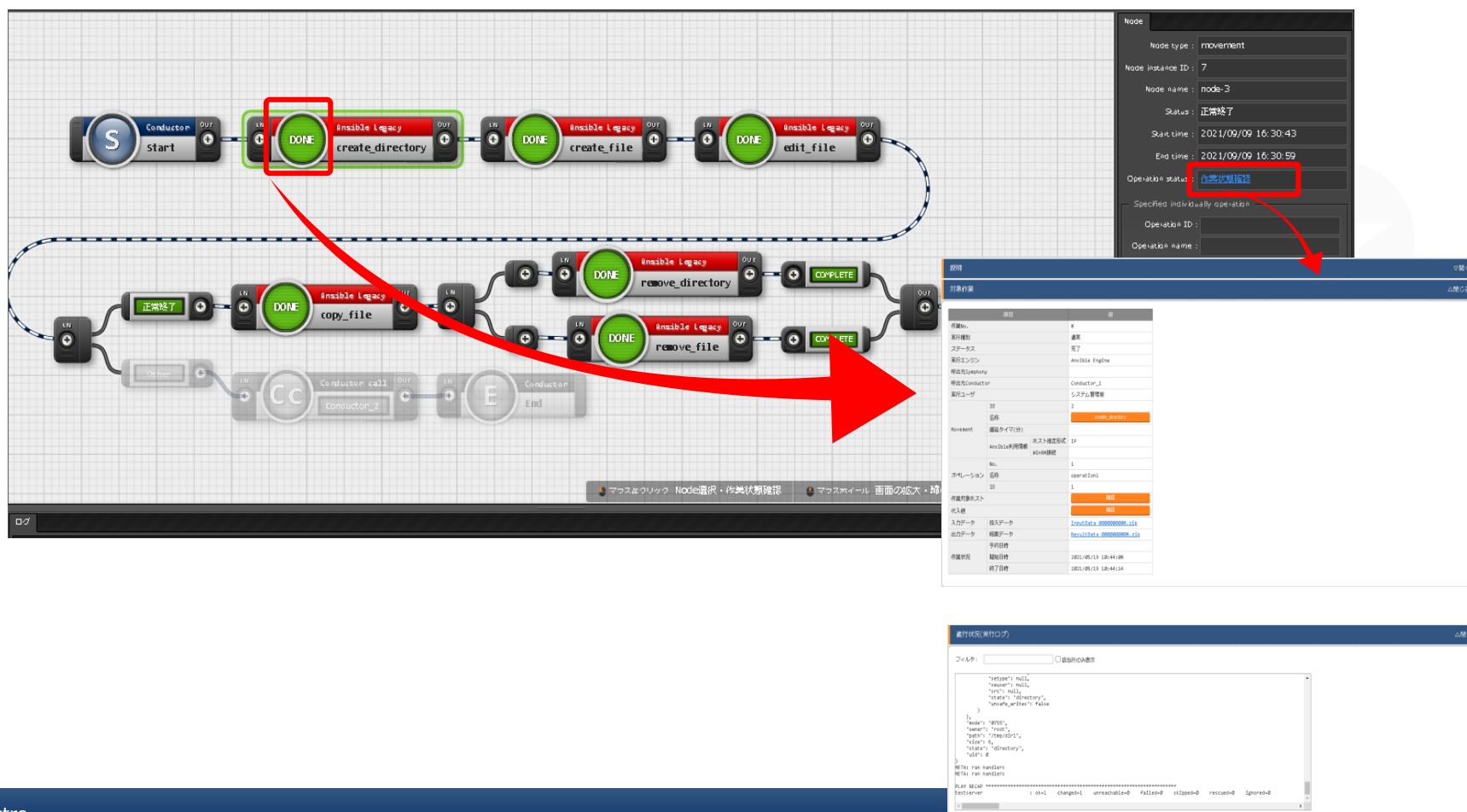
項目	値
オペレーション名	operation1

実行

### 3.10 Conductor完了確認

## ●Conductor完了確認

実行中または実行完了したMovementを選択し、  
Doneのアイコンまたは右側のOperation statusをクリックすると  
対象作業ステータスや、ログを確認できる画面に遷移します。





**Exastro**