

# IT Automation Conductor 【実習編】

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

# 目次

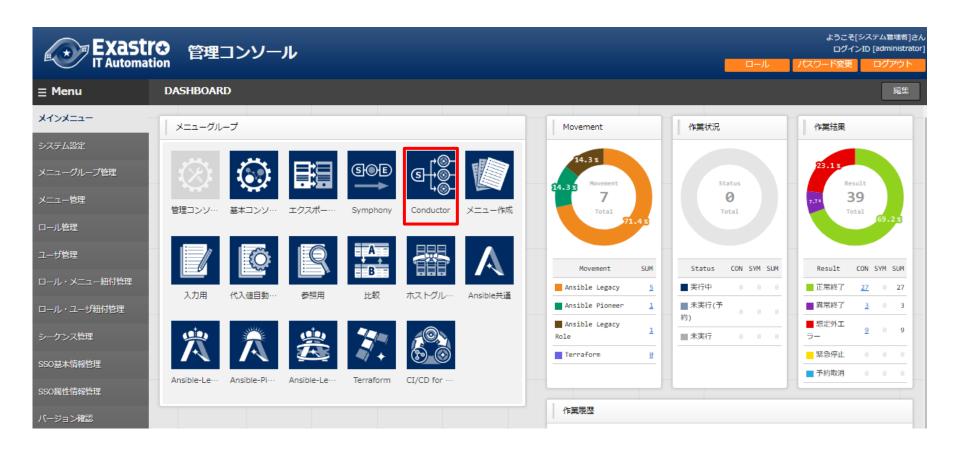
- 1. はじめに
  - 1. 本書について
- 2. Conductor
  - 1. シナリオ
  - 2. 事前準備
- 3. 実習
  - 1. 作業対象ホストの登録
  - 2. オペレーションの登録
  - 3. IaCの登録
  - 4. Movementの登録
  - 5. Movement-Playbook紐付の登録
  - 6. オペレーションに関連付くMovementとホストの登録
  - 7. 代入值管理
  - 8. Conductorの登録
  - 9. Conductorの実行
  - 10.Conductorの完了確認

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」について解説しています。



# 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。
- ●Ansible driverはAnsible-Legacyを使用しています。

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

4 Movement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

®Conductorの実行

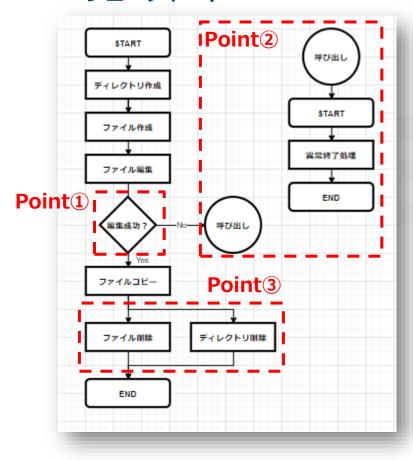
9実行結果確認

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に為に、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



## 口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

#### **□Point**②

登録済のOperation/Conductorの呼び出し機能

#### **□Point**③

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 BOMなし"、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)

- name: forced termination

fail: msg={{ VAR message text }}

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 }}"
```

#### ・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



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



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yml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



POINT

IaCの作成手順つきましては、 「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にようになります。

#### 一覧/更新

履歷	複製	更新	廃止	素材エロ⇔	Playbook素材名令	Playbook素材	Movement -Playbook紐付	アクセス権 アクセス許可ロール <b>令</b>	備考令	最終更新日時令	最終更新者令
履歴	複製	更新	廃止	1	copy_file	copy file.yml	Movement-Playbook組付			2021/09/09 14:37:04	システム管理者
履歴	複製	更新	廃止	2	create_directory	create directory.yml	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	create file.yml	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	edit file.yml	Movement-Playbook組付			2021/09/09 14:37:24	システム管理者
履歴	複製	更新	廃止	5	forced_termination	forced termination.yml	Movement-Playbook組付			2021/09/09 14:37:34	システム管理者
履歴	複製	更新	廃止	6	remove_directory	remove directory.yml	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	remove file.yml	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



# 3.4 Movementの登録 (2/2)

## ●Movementの登録

登録後のイメージは以下にようになります。



# 3.5 Movement-Playbook紐付の登録 (1/2)

## ●Movement-Playbook紐付の登録

「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



受録するMovement詳細は ymlファイルと同数を作成して下さい

# 3.5 Movement-Playbook紐付の登録 (2/2)

# ●Movement-Playbook紐付の登録

登録後のイメージは以下にようになります。

△閉じる

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

フィルタ結果件数: 7

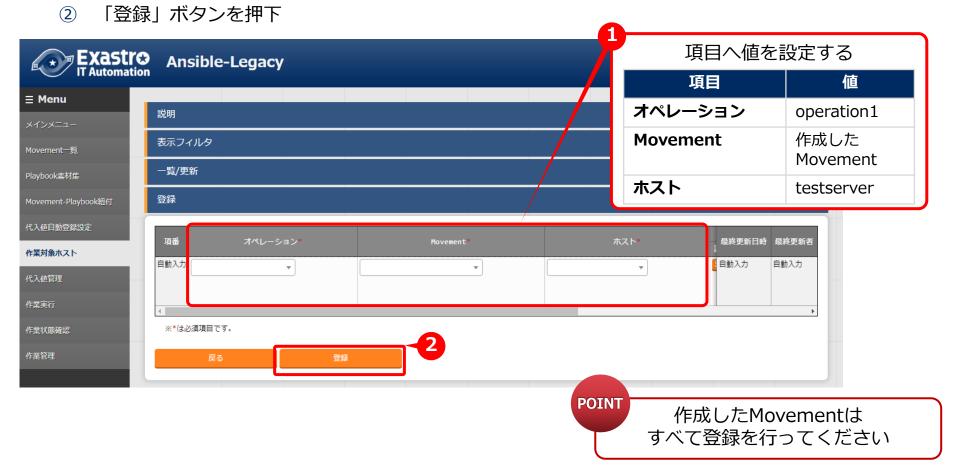
Excel出力

# <u>3.6 オペレーションに関連付くMovementとホストの登録</u>

#### ●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



# 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



# 3.7 代入值管理 (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。

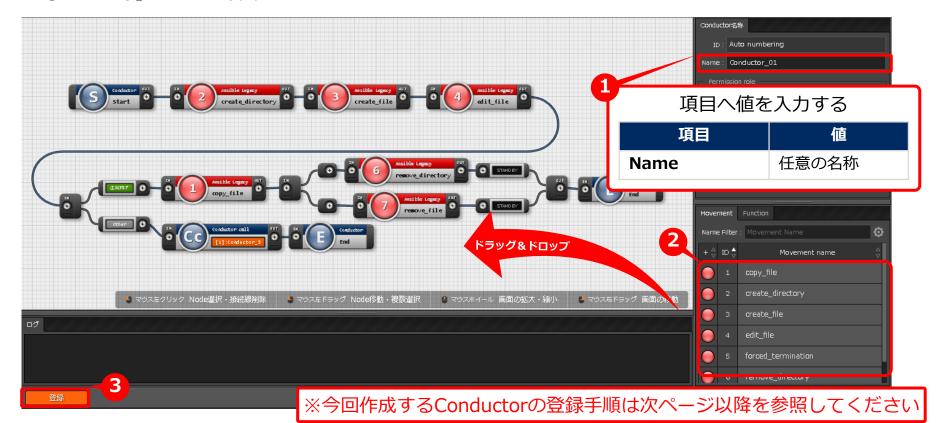
オペレーション	ホスト	Movement:変数	具体値	代入順序
1:operation1	1:Testserver	1:copy_file:1:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	1:copy_file:2:VAR_file_name	testfile	
1:operation1	1:Testserver	1:copy_file:3:VAR_dir_name_2	dir2	
1:operation1	1:Testserver	2:create_directory:4:VAR_dir_name_1	dir1	
1:operation1	1:Testserver	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」

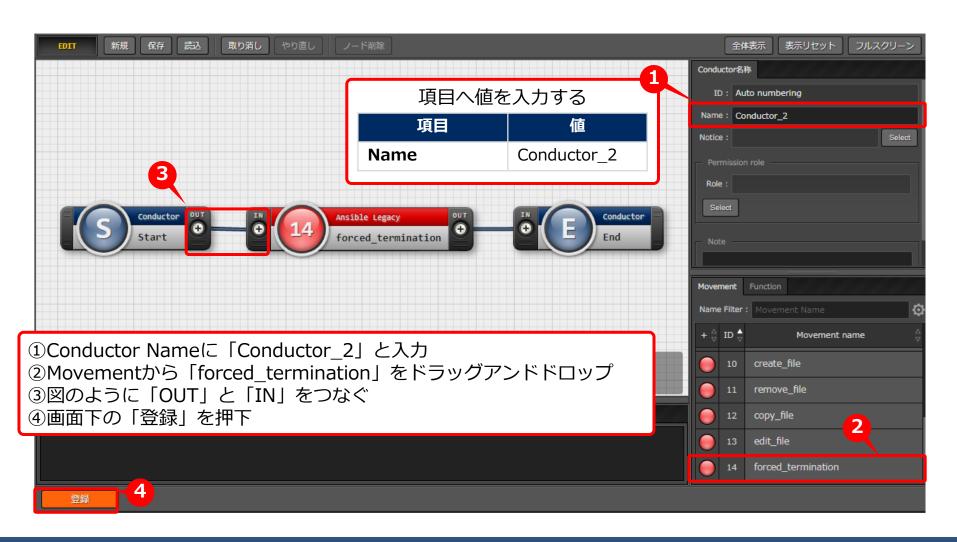
- ① 「Conductor名称」を入力
- ② 画面右側に表示されている「Movement」「Function」を 画面中央にドラッグ&ドロップ
- ③ 「登録」ボタンを押下



# 3.8 Conductorの登録 (2/7)

#### ●Conductorの登録

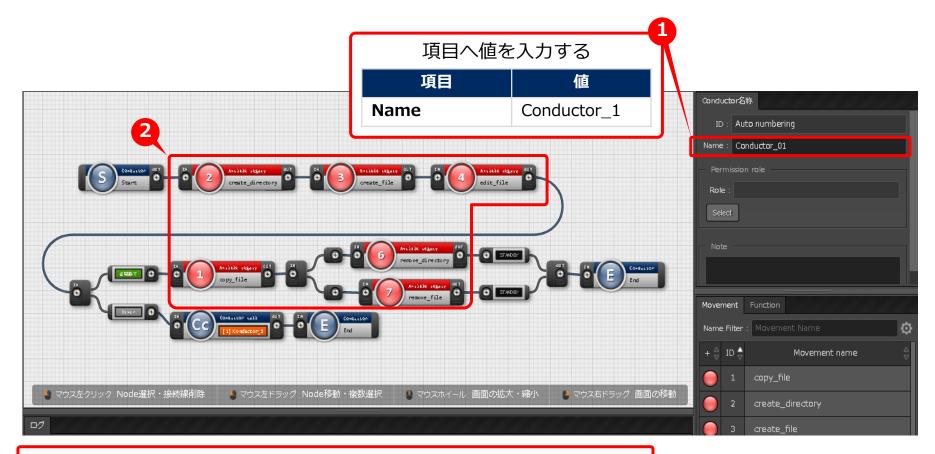
以下のようにConductorを作成してください



# 3.8 Conductorの登録 (3/7)

#### ●Conductorの登録

「Conductor\_1」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。

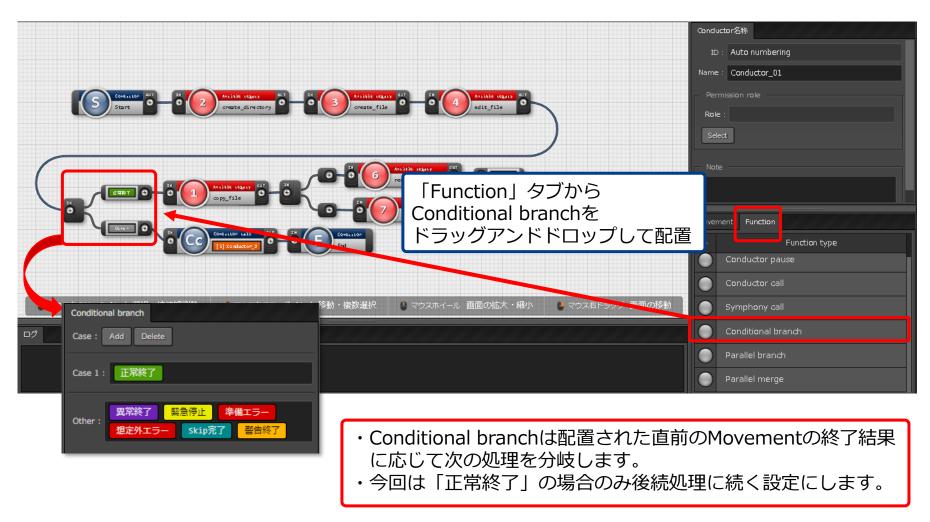


- ①Conductor Nameに「Conductor\_1」と入力
- ②各種Movementは図を参考に配置してください。

# 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

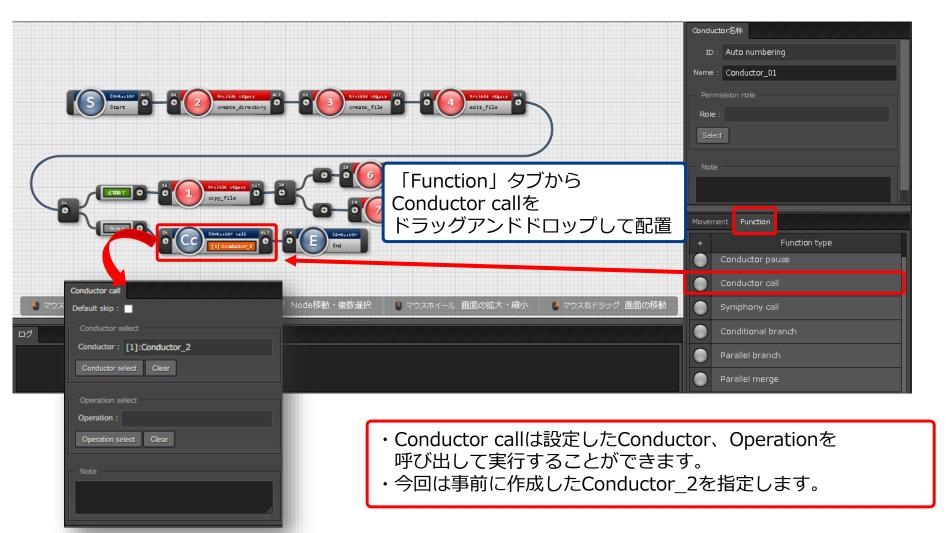


25

# 3.8 Conductorの登録 (5/7)

#### ●Conductorの登録

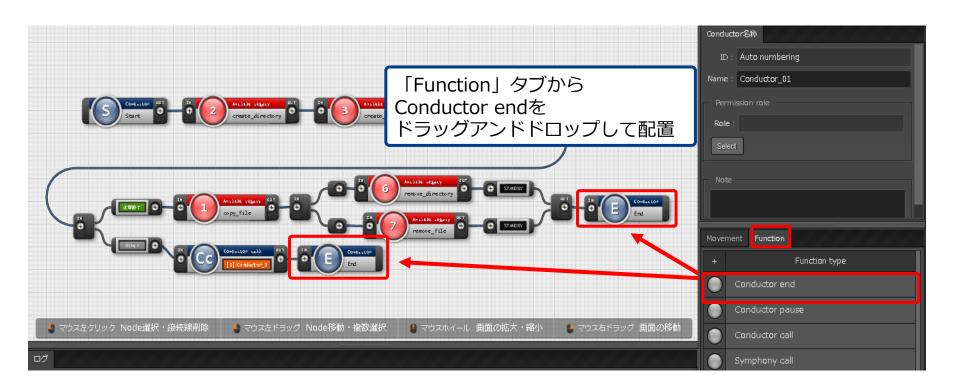
以下のようにConductorを作成してください



# 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

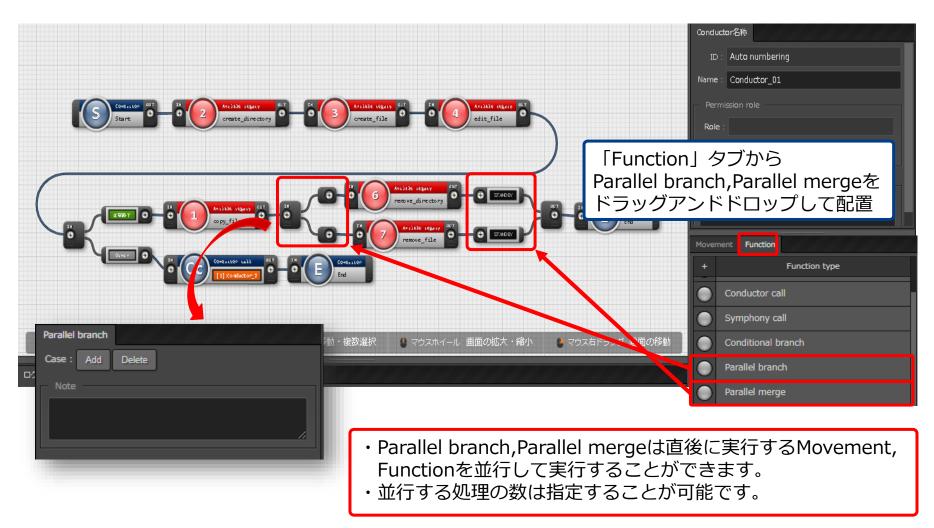


- ・Conductor endは処理の終了時に配置するfunctionです。
- ・(5/7)にてご紹介した分岐処理の終了時にも配置しています

# 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



## 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 「スケジューリング」サブメニュー「予約日時」項目内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



# 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。

