



ITA_利用手順マニュアル

Conductor

—第1.6版—

免責事項

本書の内容はすべて日本電気株式会社が所有する著作権に保護されています。

本書の内容の一部または全部を無断で転載および複製することは禁止されています。

本書の内容は将来予告なしに変更することがあります。

日本電気株式会社は、本書の技術的もしくは編集上の間違い、欠落について、一切責任を負いません。

日本電気株式会社は、本書の内容に関し、その正確性、有用性、確実性その他いかなる保証もいたしません。

商標

- ・ LinuxはLinus Torvalds氏の米国およびその他の国における登録商標または商標です。
- ・ Red Hatは、Red Hat, Inc.の米国およびその他の国における登録商標または商標です。
- ・ Apache、Apache Tomcat、Tomcatは、Apache Software Foundationの登録商標または商標です。
- ・ Ansibleは、Red Hat, Inc.の登録商標または商標です。
- ・ AnsibleTowerは、Red Hat, Inc.の登録商標または商標です。

その他、本書に記載のシステム名、会社名、製品名は、各社の登録商標もしくは商標です。

なお、® マーク、TM マークは本書に明記していません。

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

目次

はじめに	4
1 ITA Conductor の概要	5
2 ITA Conductor のメニュー、画面構成	6
2.1 ITA Conductor メニュー 一覧	6
3 ITA Conductor 利用手順	7
3.1 作業フロー	7
4 機能・操作方法説明	8
4.1 ITA Conductor	8
4.1.1 Conductor インターフェース情報	8
4.1.2 Conductor クラス一覧	9
4.1.3 Conductor クラス編集	10
4.1.4 Conductor 作業実行	21
4.1.5 Conductor 作業確認	24
4.1.6 Conductor 作業一覧	27
4.1.7 Conductor 定期作業実行	27

はじめに

本書は、ITA の Conductor の機能および操作方法について説明します。

1 ITA Conductor の概要

本章では Conductor メニューの機能、操作方法について説明します。

Conductor では、ITA を利用して作業を行う上で共通に必要な以下の機能を提供します。

- ・ ワークフローの作成、管理、実行

ITA の操作手順における Conductor の位置づけは、「ファーストステップガイド」を参照してください。

2 ITA Conductor のメニュー、画面構成

本章では、ITA Conductor のメニュー、画面構成について説明します

2.1 ITA Conductor メニュー 一覧

ITA 共通/Conductor のメニューを以下に示します。

表 2.1-1 ITA Conductor 画面一覧

No	メニュー グループ	メニュー・画面	説明
1	Conductor	Conductor インターフェース情報	Conductor 作業実行時の Movement で共有するディレクトリパスなどをメンテナンス(閲覧/登録/更新/廃止)できます
2		Conductor クラス一覧	Conductor クラスをメンテナンス(閲覧/廃止)できます 「詳細」をクリックすると Conductor クラス編集メニューに遷移します
3		Conductor クラス編集	Conductor クラスを編集できます
4		Conductor 作業実行	Conductor 作業を実行できます
5		Conductor 作業確認	Conductor 作業の実行結果を確認できます
6		Conductor 作業一覧	Conductor 一覧(実行履歴)を閲覧できます 「詳細」をクリックすると Conductor 作業確認に遷移します
7		Conductor 定期作業実行	スケジュールに従って定期的に行う Conductor 作業を管理します。

3 ITA Conductor 利用手順

3.1 作業フロー

ITA Conductor における標準的な作業フローは以下のとおりです。
各作業の詳細は次項に記載しています。

- 「機器情報」、「オペレーション」の登録方法については、「利用手順マニュアル 基本コンソール」を参照してください。
- Movement の登録方法については、各 Driver の利用手順マニュアルを参照してください。

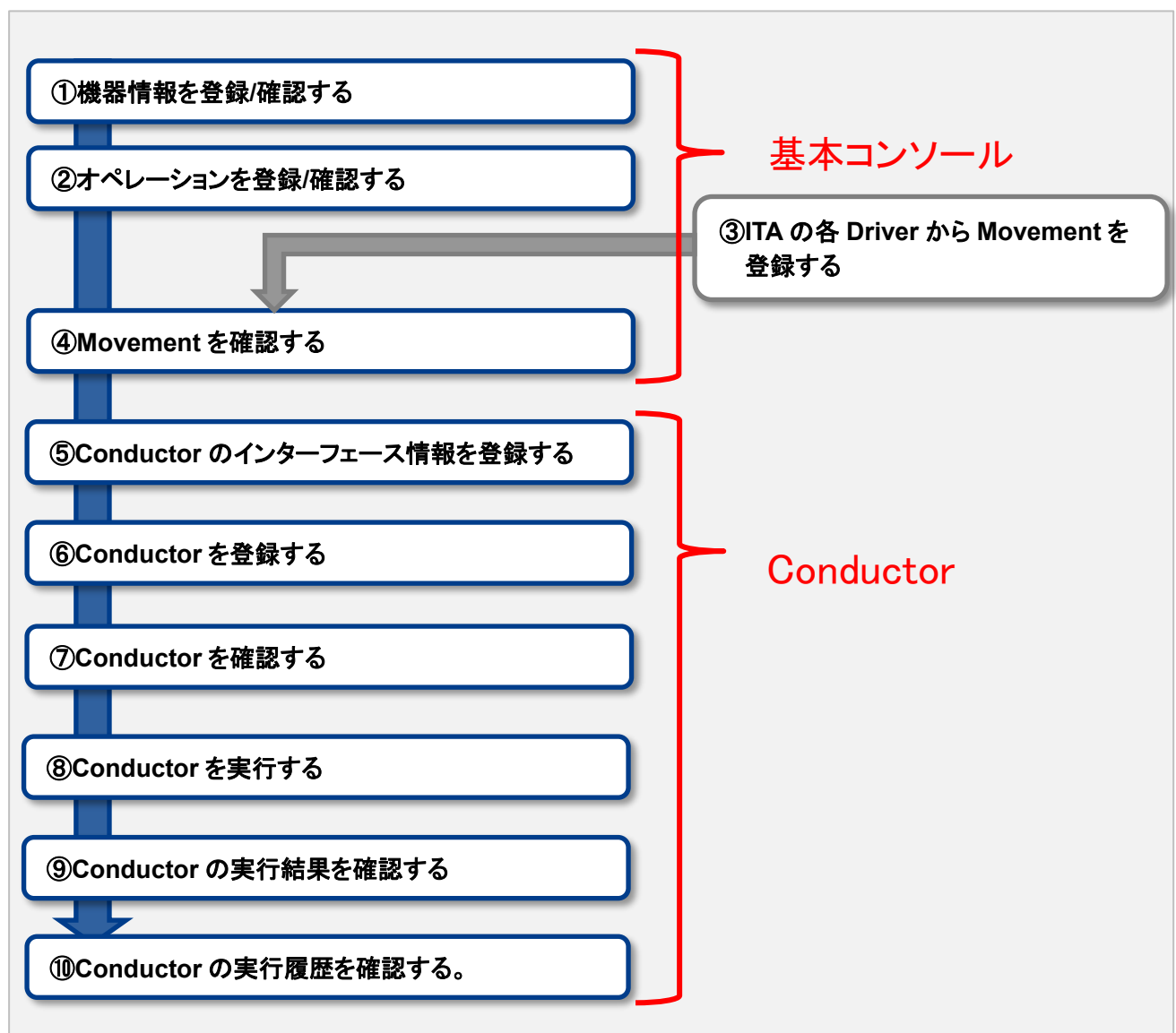


図 3.1-1 作業フロー

4 機能・操作方法説明

4.1 ITA Conductor

4.1.1 Conductor インターフェース情報

- (1) [Conductor インターフェース情報]画面では、Conductor から実行される各 Movement で共有するディレクトリのパスと[Conductor 作業確認]画面のリフレッシュ間隔を設定します。



図 4.1-1 サブメニュー画面(Conductor インターフェース情報)

- (2) 画面の項目一覧は以下のとおりです。

表 4.1-1 登録画面項目一覧(Conductor インターフェース情報)

項目	説明	入力 必須	入力形式	制約事項
データリレイストレージパス	Conductor 実行時、各 Movement で共有するディレクトリを、ITA サーバから見たディレクトリパスで入力します。 各 Driver から見たパスは、各 Driver の利用手順マニュアルのインターフェース情報を参照して下さい。 ディレクトリを共有することが出来る Driver は以下です。 ・Ansible ・Ansible-Tower	○	手動入力	最大長 128 バイト
状態監視周期(単位ミリ秒)	「4.1.4 Conductor 作業実行」の表示をリフレッシュする間隔を入力します。通常は 3000 ミリ秒程度が推奨値です。	○	手動入力	最小値 1000 ミリ秒
備考	自由記述欄です	—	手動入力	—

4.1.2 Conductor クラス一覧

- (1) [Conductor クラス一覧]画面では、登録済みの Conductor クラスを参照/廃止できます。
一覧の「詳細」ボタンをクリックすると、その Conductor クラスの編集画面
[4.1.3 Conductor クラス編集]へ遷移できます。

廃止	ConductorクラスID	Conductor名称	説明	アクセス権 アクセス許可ロール	最終更新日時	最終更新者
廃止	1	SAVPLE1	システム管理者	システム管理者	2020/11/25 13:38:15	システム管理者
廃止	2	SAVPLE2	システム管理者, ロール1, ロール2	システム管理者	2020/11/25 13:38:05	システム管理者
廃止	3	SAVPLE3	システム管理者	システム管理者	2020/11/25 13:37:47	システム管理者
廃止	4	一時停止	データポートビリティプロシージャ	データポートビリティプロシージャ	2020/11/09 15:19:04	データポートビリティプロシージャ

図 4.1-2 サブメニュー画面(Conductor クラス一覧)

4.1.3 Conductor クラス編集

Conductor クラス編集画面、Conductor を構成するパーツ(以下、Node)、について、以下になります。

(1) 各 Node について

表 4.1-2 Node 一覧

画像	名称	動作説明
	Conductor start	Conductor を開始します
	Conductor end	Conductor 終了します。 ※複数の Conductor end がある場合、全ての Conductor end が終了を待ちます。
	Conductor pause	ワークフローを一時停止します。 一時停止を解除すると、次の処理へ進みます。
	Conductor call	別の登録済みの Conductor クラスを呼び出し実行します。
	Symphony call	登録済みの Symphony クラスを呼び出し実行します。
	Conditional branch	接続された、「Movement」、「Conductor call」、「Symphony call」の結果によって、後続の処理を分岐させます。 指定可能なステータスは、以下になります。 ・正常終了 ・異常終了 ・緊急停止 ・準備エラー ・想定外エラー ・SKIP 完了
	Parallel branch	並列して、「Movement」、「Conductor call」、「Symphony call」を実行します。 ※並列可能な実行数は、ITA の構成やサーバースペックに依存します。
	Parallel merge	接続している全 Node の処理が完了後、次の処理を実行します。
	Movement 各種	Movement を実行します。

制約事項

- ・登録/更新を行うには、全ての Node の IN/OUT が接続されている必要があります。
- ・Parallel merge を使用する場合、Parallel branch を使用している必要があります。



図 4.1-3 Node 制約事項(正常例:Parallel branch)

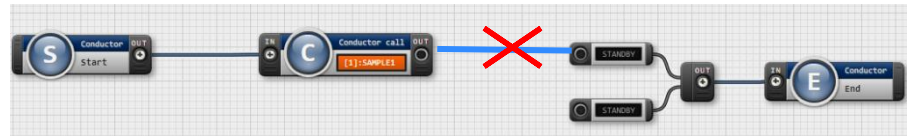


図 4.1-4 Node 制約事項(NG 例:Parallel branch)

- ・Conditional branch で分岐されたフローについて Parallel merge でマージする事はできません。

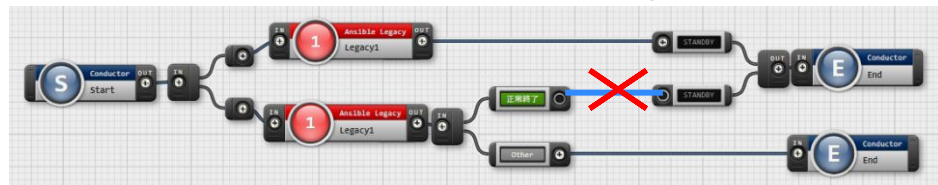


図 4.1-5 Node 制約事項(NG 例:Conditional branch)

- ・Parallel branch、Conditional branch、Parallel merge、Conductor pause について、連続して、同じ種類の Node を接続する事はできません。

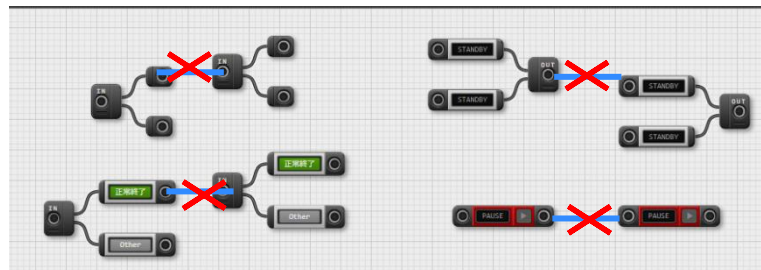


図 4.1-6 Node 制約事項(NG 例:連続使用)

- ・Conductor call について、更新中の Conductor を Conductor call で指定し、更新することはできません。

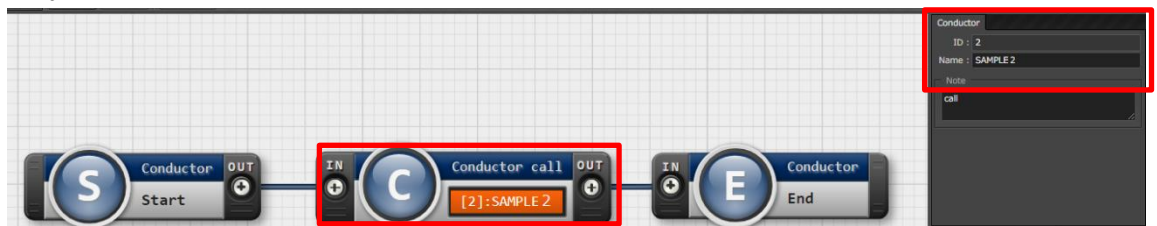


図 4.1-7 Node 制約事項(NG 例:Conductor call)

(2) [Conductor クラス編集]画面について

- Conductor クラス名称とワークフローの Node を登録します。
- 画面上のモードについて、以下のモードがあります。
各モードで可能な操作について、「表 4.1-12 Conductor クラス編集画面実行操作一覧」を参照。

・ 表 4.1-3 Conductor クラス編集画面モード一覧

モード	説明
EDIT	• Conductor クラスを編集できるモード • Conductor クラス編集画面のデフォルトのモード • EDIT モードから登録/更新ボタン押下後に、VIEW モードへ変更.
VIEW	• Conductor クラスを閲覧のみできるモード • Conductor クラス一覧の詳細から画面遷移後のモード • VIEW モードから編集ボタン押下後に、EDIT モードへ変更

- 画面右側上部の領域には、選択している Node の詳細情報が表示されます。
- 画面右側下部の領域には、利用可能な Node が表示されます。
 - i. Movemet タブ
 - ◇ 登録済みの Movement の ID と名称の一覧
 - ii. Function タブ
 - ◇ Conductor end
 - ◇ Conductor pause
 - ◇ Conductor call
 - ◇ Symphony call
 - ◇ Conditional branch
 - ◇ Parallel branch
 - ◇ Parallel merge
- 画面右下側の Node 領域から、ドラッグ & ドロップで各 Node を設定できます。
- 各 Node の Note 記入欄には、処理説明やコメントをメモしておけます。
この情報は、処理実行に影響はなく、Web 上でのみ参照できるメモ欄です。
- Node 設定後、「登録」ボタンをクリックして、Conductor クラスを登録します



図 4.1-8 サブメニュー画面(Conductor クラス編集:EDIT)

- Conductor クラス編集画面の詳細、入力項目一覧は以下のとおりです。
 - i. Conductor 詳細、入力項目

表 4.1-4 Conductor クラス編集画面項目一覧:(Conductor)

項目	説明	入力 必須	入力形式	制約事項
ID	Conductor に対応した一意の ID が自動採番されます	－	自動入力	－
Name	任意の Conductor クラス名称を入力します	○	手動入力	－
Role	この Conductor へアクセス可能なロールを選択します。 ロールが 1 つも選択されていない場合は、すべてのロールがアクセス可能となります。	－	選択	－
Note	Conductor クラスに対する説明やコメントを入力します	－	手動入力	－

Conductor名称

ID : Auto numbering

Name :

Permission role

Role : システム管理者

Permission role select

Note

図 4.1-9 詳細、入力項目(Conductor)

Select	ID	Name
<input checked="" type="checkbox"/>	1	システム管理者
<input type="checkbox"/>	2	ロール1
<input type="checkbox"/>	3	ロール2

図 4.1-10 詳細、入力項目 (Permission role select)

ii. Node (共通) 詳細、入力項目

表 4.1-5 Conductor クラス編集画面項目一覧: (Node 共通)

項目	説明	入力 必須	入力形式	制約事項
Type	選択した Node のタイプが表示されます	－	自動入力	－
Note	Node に対する説明やコメントを入力します	－	手動入力	－

図 4.1-11 詳細、入力項目 (Node 共通)

iii. Movement 詳細、入力項目

表 4.1-6 Conductor クラス編集画面項目一覧: (Movement)

項目	説明	入力 必須	入力形式	制約事項
Movement ID	選択した Movemen の ID が表示されます	－	自動入力	－
Orchestrator	選択した Movemen のオーケストレータ名が表示されます	－	自動入力	－
Name	選択した Movemen の名称が表示されます	－	自動入力	－
Default skip	対象作業をスキップする場合にチェックします Conductor 作業実行画面にて、変更可能なパラメータ	－	手動入力	
Operation	・Select ボタンを押下すると、表示されるオペレーション一覧から選択します。 ・指定したオペレーション名が表示されます。	－	選択	－

図 4.1-12 詳細、入力項目 (Movement)

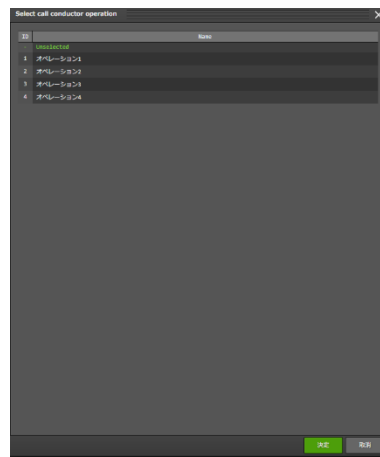


図 4.1-13 詳細、入力項目 (Operation select)

iv. Conductor call 詳細、入力項目

表 4.1-7 Conductor クラス編集画面項目一覧: (Conductor call)

項目	説明	入力 必須	入力形式	制約事項
Default skip	対象作業をスキップする場合にチェックします Conductor 作業実行画面にて、変更可能なパラメータ	－	手動入力	－
Conductor	・Select ボタンを押下すると、表示される Conductor 一覧から 選択します。 ・指定した Conductor クラス名称が表示されます。	○	選択	－
Operation	・Select ボタンを押下すると、表示されるオペレーション一覧か ら選択します。 ・指定したオペレーション名が表示されます。	－	選択	－

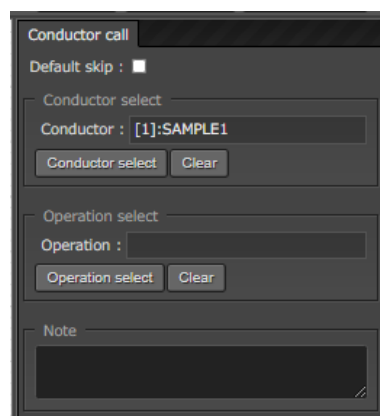


図 4.1-14 詳細、入力項目 (Conductor call)

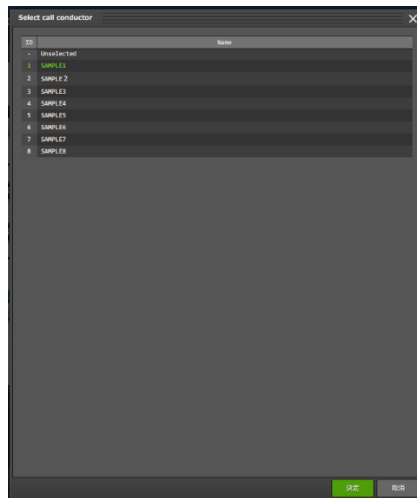


図 4.1-15 詳細、入力項目 (Conductor call select)

v. Symphony call 詳細、入力項目

・ 表 4.1-8 Conductor クラス編集画面項目一覧: (Symphony call)

項目	説明	入力 必須	入力形式	制約事項
Default skip	対象作業をスキップする場合にチェックします Conductor 作業実行画面にて、変更可能なパラメータ	－	手動入力	－
Symphony	・Select ボタンを押下すると、表示される Symphony 一覧から 選択します。 ・指定した Symphony クラス名称が表示されます。	○	選択	－
Operation	・Select ボタンを押下すると、表示されるオペレーション一覧から 選択します。 ・指定したオペレーション名が表示されます。	－	選択	－

図 4.1-16 詳細、入力項目 (Symphony call)

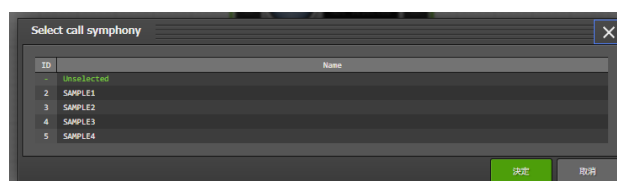


図 4.1-17 詳細、入力項目 (Symphony call select)

i. Parallel branch 詳細、入力項目

表 4.1-9 Conductor クラス編集画面項目一覧: (Parallel branch)

項目	説明	入力 必須	入力形式	制約事項
case	分岐数を設定します。 デフォルトは 2 つで以下を押下して、増減します。 ・Add ・Delete	-	選択	

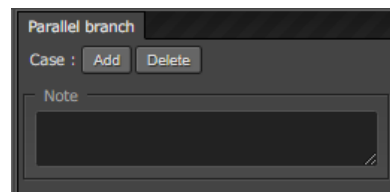


図 4.1-18 詳細、入力項目 (Parallel branch)

ii. Conditional branch 詳細、入力項目

表 4.1-10 Conductor クラス編集画面項目一覧: (Conditional branch)

項目	説明	入力 必須	入力形式	制約事項
case (1-6)	Movement、ConductorCall、SymphonyCall の実行結果による条件分岐を設定します。 ドラッグアンドドロップで設定を変更できます。 デフォルトは、 case1: 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	-	選択	

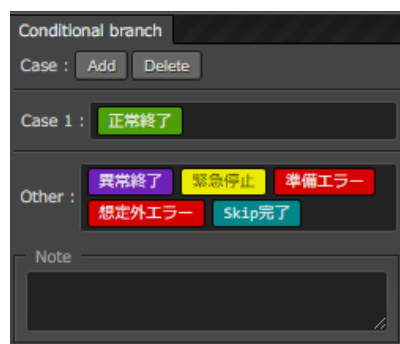


図 4.1-19 詳細、入力項目 (Conditional branch)

iii. Conditional branch 詳細、入力項目

表 4.1-11 Conductor クラス編集画面項目一覧: (Parallel merge)

項目	説明	入力 必須	入力形式	制約事項
----	----	----------	------	------

case	分岐数を設定します。 デフォルトは 2 つで以下を押下して、増減します。 ・Add ・Delete	-	選択	
------	--	---	----	--

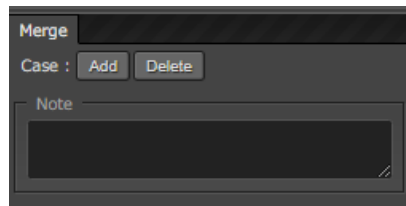


図 4.1-20 詳細、入力項目 (Parallel merge)

- ・ クラス編集画面で実行可能な操作は以下になります。

表 4.1-12 Conductor クラス編集画面実行操作一覧

項目	説明	新規 (EDIT)	更 新 (VIEW)	更 新 (EDIT)	備考
新規	初期状態へ戻します。	○	-	-	
保存	描画されている状態を、ファイル形式で保存します。	○	-	-	
読み込み	保存した、ファイルを読み込み状態を復元します。	○	-	-	
取り消し	直前の処理を取り消します。	○	-	○	
やり直し	直前の取り消しをやり直します。	○	-	○	
ノード削除	選択している Node を削除します。	○	-	○	
登録	登録を実施します。	○	-	-	
編集	EDIT モードへ変更し、Conductor の編集を行います。	-	○	○	
流用新規	登録済の Conductor を流用して、新規作成が行えます。	-	○	○	
更新	編集内容を更新します。	-	-	○	
再読込	変更をキャンセルし、変更前の状態へ戻します。	-	-	○	
キャンセル	変更をキャンセルし、VIEW モードへ変更します。	-	-	○	

- (3) [Conductor クラス一覧]画面から遷移した場合や、登録が完了した場合は、以下の画面が表示されます。

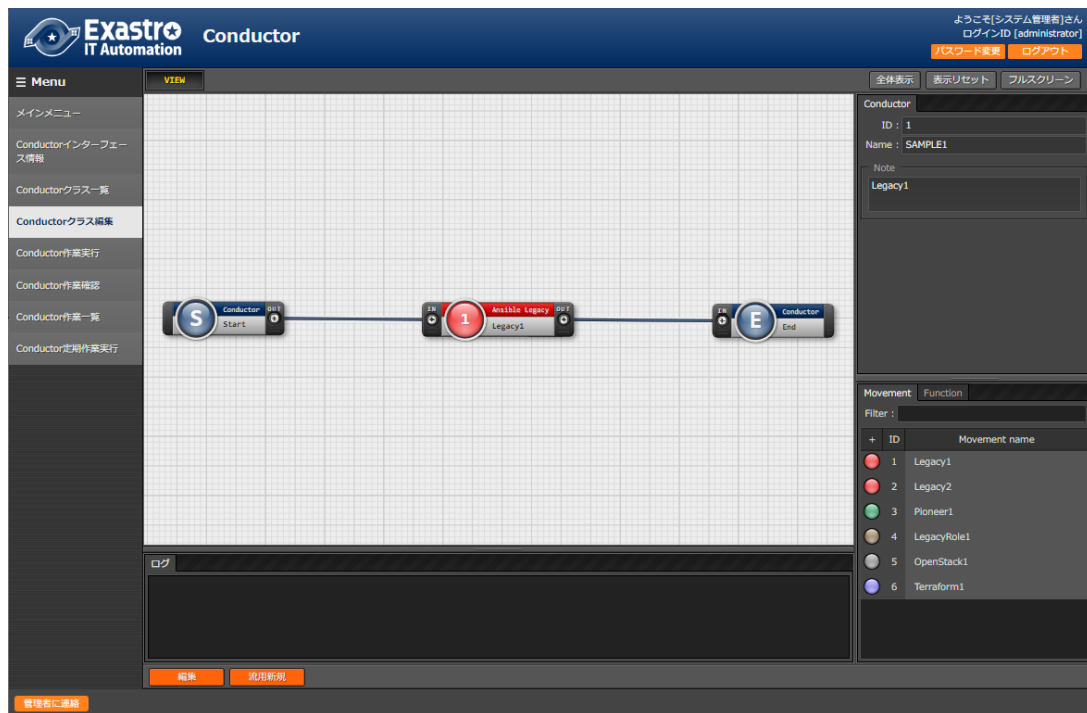


図 4.1-21 サブメニュー画面(Conductor クラス編集:VIEW)

- ・ 「編集」ボタンで、登録済みの Conductor を編集できます。
- ・ 「流用新規」ボタンで、登録済みの Conductor をコピーして新規作成が行えます。

(4) 「編集」ボタンを押下した場合は、以下の画面が表示されます。

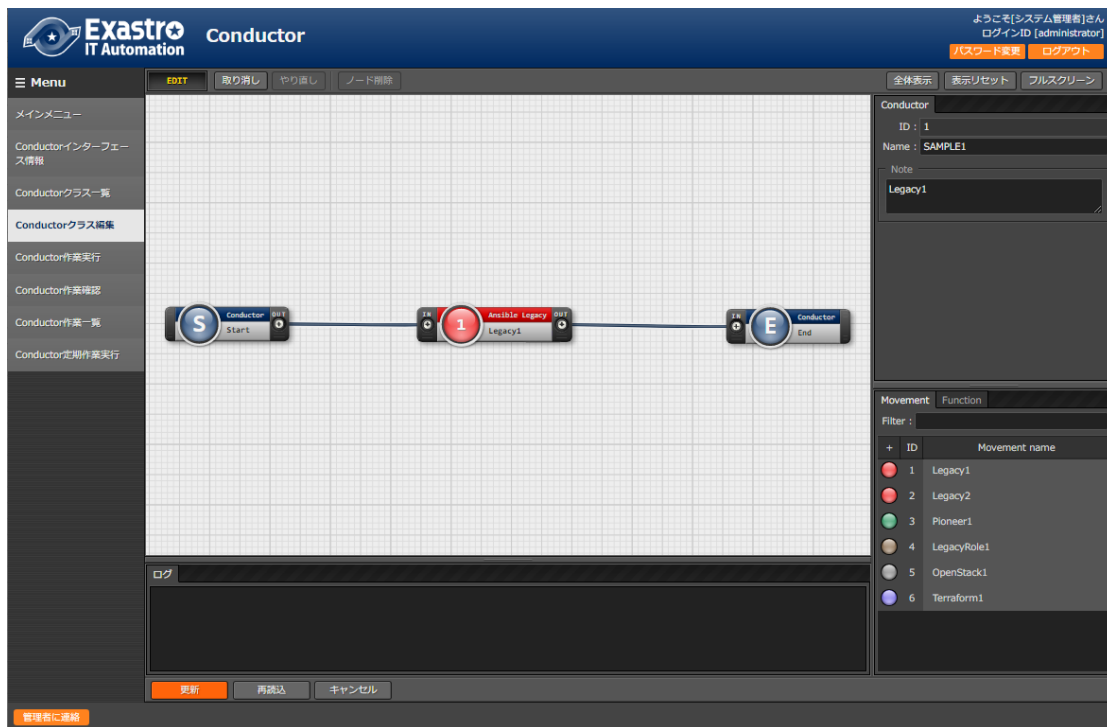


図 4.1-22 サブメニュー画面(Conductor クラス編集:EDIT)

- ・ 「再読み込み」ボタンで、編集内容が破棄されて登録内容の状態に戻ります。

- ・ 「更新」ボタンで、編集内容が保存されます。
- ・ 「キャンセル」ボタンで、「編集」ボタン押下前の状態に戻ります。

4.1.4 Conductor 作業実行

- (1) [Conductor 作業実行]画面では、Conductor 実行の指示を行います。
- ・ 「Conductor[一覧]」には「[4.1.2 Conductor クラス一覧](#)」で登録した Conductor が表示されます。
 - ・ 「オペレーション[一覧]」には「基本コンソール」で登録したオペレーションが表示されます。
 - i. 別紙、「利用手順マニュアル 基本コンソール」を参照。
 - ・ 「Conductor[一覧]」、「オペレーション[一覧]」のラジオボタンからそれぞれ選択し、「実行ボタン」を押すと「[4.1.5 Conductor 作業確認](#)」に遷移し、作業のトレースが始まります。
 - ・ 予約日時に入力して「実行ボタン」を押すと、作業予約が作られます。これは、「[4.1.6 Conductor 作業一覧](#)」で確認できます。※現在時刻より過去の日時は入力できません
 - ・ Movement、Conductor Call、Symphony Call の Operation, skip のみ、設定値を変更可能です。
 - i. Conductor 編集で登録したデータへ変更は反映されません。作業実行にのみ反映されます。

Exastro
IT Automation

Conductor

ようこそ[システム管理者]さん
ログインID [administrator]
パスワード入力 ロックアウト

Menu

メインメニュー

Conductorインターフェース情報

Conductorクラス一覧

Conductorクラス編集

Conductor作業実行

Conductor作業確認

Conductor作業一覧

Conductor定期作業実行

説明

スケジュール

予約日時

(YYYY/MM/DD HH:MM)で入力して下さい。ブランクの場合は即時実行となります

Conductor[フィルタ]

Conductor[一覧]

選択	ConductorクラスID	Conductor名称	説明	備考	最終更新日時	最終更新者
<input checked="" type="radio"/>	1 SAMPLE1	Legacy1			2020/07/21 16:14:31	システム管理者
<input type="radio"/>	2 SAMPLE2	call			2020/07/21 16:14:11	システム管理者
<input type="radio"/>	3 SAMPLE3	pause			2020/07/21 16:15:51	システム管理者
<input type="radio"/>	4 SAMPLE4	conditional			2020/07/21 16:17:27	システム管理者
<input type="radio"/>	5 SAMPLE5	parallel			2020/07/21 16:18:38	システム管理者
<input type="radio"/>	6 SAMPLE6	call			2020/07/29 09:52:41	システム管理者
<input type="radio"/>	7 SAMPLE7	call			2020/07/29 09:53:12	システム管理者
<input type="radio"/>	8 SAMPLE8	継続			2020/07/22 17:13:13	システム管理者

フィルタ結果件数: 8

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	実施予定日時	最終実行日時	備考	最終更新日時	最終更新者
<input checked="" type="radio"/>	1	1	オペレーション1	2020/09/01 15:54	2020/07/28 10:11		2020/07/28 10:11:23	legacy作業実行プロセス
<input type="radio"/>	2	2	オペレーション2	2020/09/02 15:54	2020/07/22 13:55		2020/07/22 13:55:34	legacy作業実行プロセス
<input type="radio"/>	3	3	オペレーション3	2020/09/03 15:55	2020/07/22 13:59		2020/07/22 13:59:28	legacy作業実行プロセス
<input type="radio"/>	4	4	オペレーション4	2021/03/23 11:02	2020/07/27 11:01		2020/07/28 18:39:14	システム管理者

フィルタ結果件数: 4

Conductor実行

EXECUTE

Conductor

ID: 1

Name: SAMPLE1

Note

Legacy1

Operation

Operation ID: 1

Operation name: オペレーション1

実行

図 4.1-23 サブメニュー画面(Conductor 作業実行)

Conductor 実行画面の共通項目一覧は以下のとおりです。

表 4.1-13 登録画面項目一覧(Conductor 実行)

項目	説明	入力 必須	入力形式	制約事項
予約日時	Conductor の実行予定日時を指定します	-	手動入力	-現在時刻 より過去の 日時は入力 不可
Conductor[一覧]	「4.1.7Conductor クラス一覧」で登録した Conductor が表示されます	○	ラジオ ボタン	
オペレーション[一覧]	「4.1.4 投入オペレーション一覧」で登録したオペレーションが表示されます	○	ラジオ ボタン	
Skip	対象作業をスキップする場合にチェックします ※以下「オペレーションの指定について」参照	-	チェック ボックス	
Operation	※以下「Operation の指定について」参照	-	手動入力	
実行	登録した Conductor を実行します	○	ボタン	

※オペレーションの指定について

「Operation 欄の select」をクリックすると、Operation のリストが表示されます。

画面のラジオボタンで指定したオペレーションのオペレーション ID とは別のオペレーションを指定することができます。

これにより、その Movement の属するオーケストレータの「代入値管理」メニュー（例：ITAAnsible-Legacy コンソールの「代入値管理」メニュー）で、ほかのオペレーション ID のものとして登録した「具体値」を代入して実行することができます。

Conductor クラス編集画面で個別指定したオペレーション ID は Conductor 登録/更新ボタンにより設定が保存されます。

また、Conductor 実行画面でも実行前に個別指定ができ、既に Conductor クラス編集メニューで個別指定登録をして保存されているオペレーション ID についても更に変更を行い Conductor 実行することができます。

ただし、Conductor 実行画面で個別指定したオペレーション ID は実行時のみの反映となり、設定は保存されません。

同じ Movement を流用し、別なサーバを操作したい時などにご活用ください。

※Skip について

Skip のチェックを変更することができます。

Conductor クラス編集画面で Default Skip の設定は登録/更新ボタンにより設定が保存されます。

また、Conductor 実行画面でも実行前に個別指定ができ、既に Conductor クラス編集メニューで保存されている Skip について変更を行い Conductor 実行することができます。

ただし、Conductor 実行画面では実行時のみの反映となり、設定は保存されません。

一時的に、処理を飛ばして、又は実施して、作業実行したい時などにご活用ください。

※実行する作業のアクセス権限について

Conductor 実行画面に表示された「Movement」および「Conductor Call」「Symphony Call」内で呼び出されるすべて作業および個別指定されたオペレーションについて、アクセス権限が無い対象が一つでも含まれている場合は「実行」押下した際にバリデーションエラーになります。

4.1.5 Conductor 作業確認

- (1) [Conductor 作業確認]画面では、Conductor の実行状態を表示します。
- 「4.1.6Conductor 作業一覧」の「詳細」表示ボタンをクリックすると、選択した Conductor 作業の処理状況をモニター表示します。状況に応じて「予約取消」、「停止解除」や「緊急停止」の投入が可能です。
- 各ノードを選択すると、実行状況を表示します。
- 作業実行状況の詳細は、「Movement」、「Conductor Call」、「Symphony Call」の「Operation status」から作業状態確認画面に遷移できます。

※「Conductor 作業実行」で実行した、作業実行済みの Conductor を「Conductor クラス編集」で編集を行うと、作業実行時の Conductor と異なる状態となるため、「詳細」表示ボタンをクリックしても処理状況が表示されない場合があります。作業実行済みの Conductor を編集して、再度実行する場合は、「Conductor クラス編集」にて、流用新規にて別の Conductor を作成してご利用いただくことを推奨します。

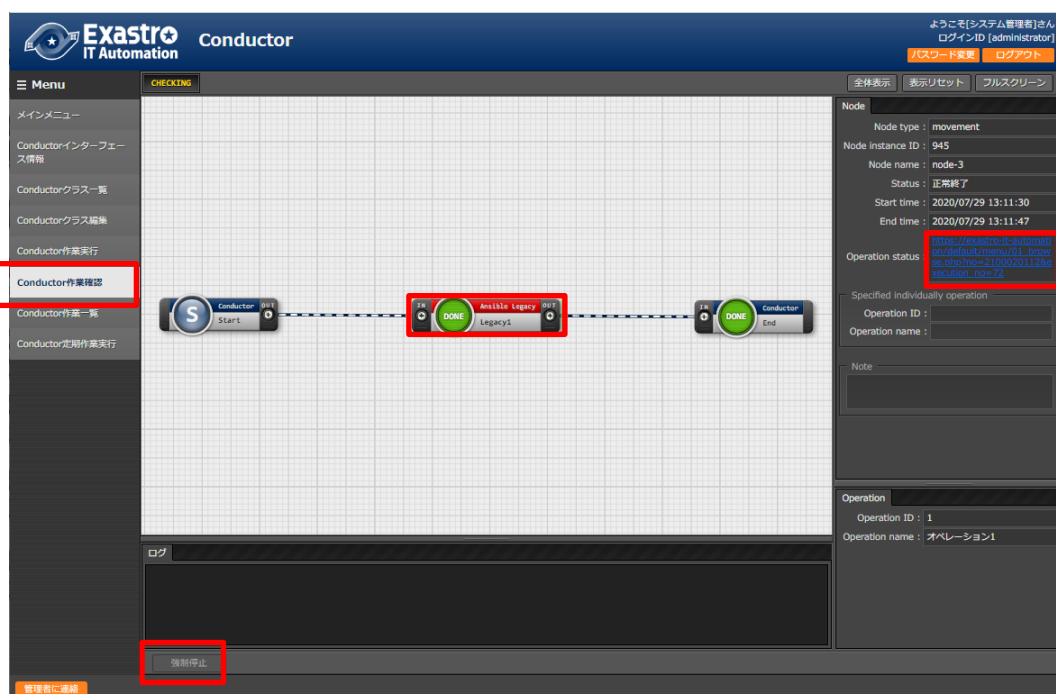


図 4.1-24 サブメニュー画面(Conductor 作業確認)

また、選択した Conductor 作業に予約日時が設定されていて、かつ未実行の場合は、予約取り消しボタンが表示されます。予約取り消しボタンを押すと、「4.1.6Conductor 作業一覧」で確認できるステータスが「予約取り消し」となり、実行されなくなります。

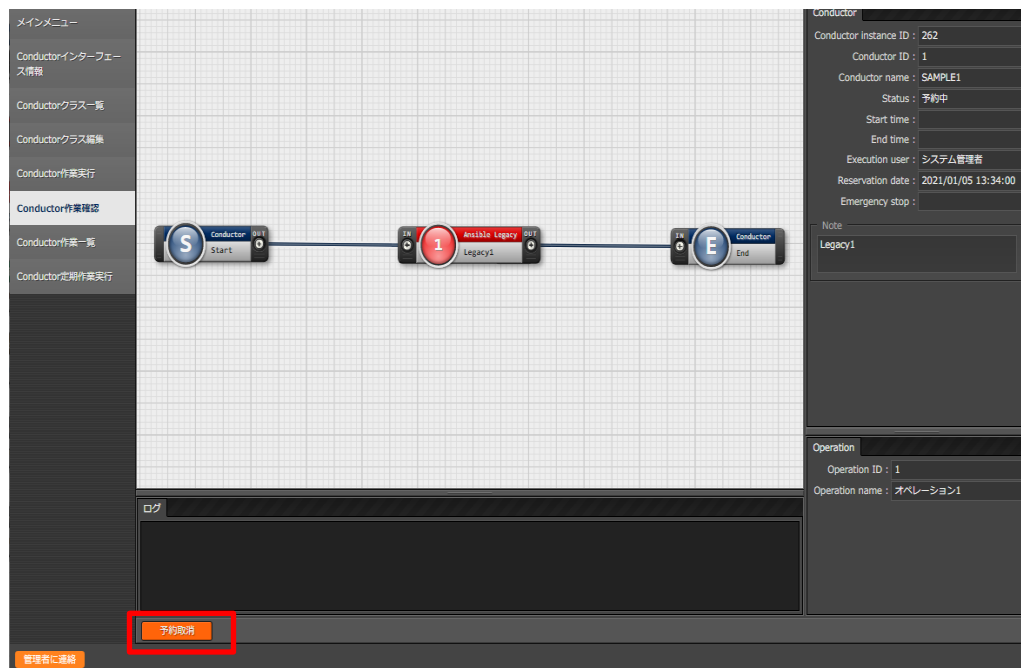


図 4.1-25 サブメニュー画面(Conductor 作業確認 - 予約取消)

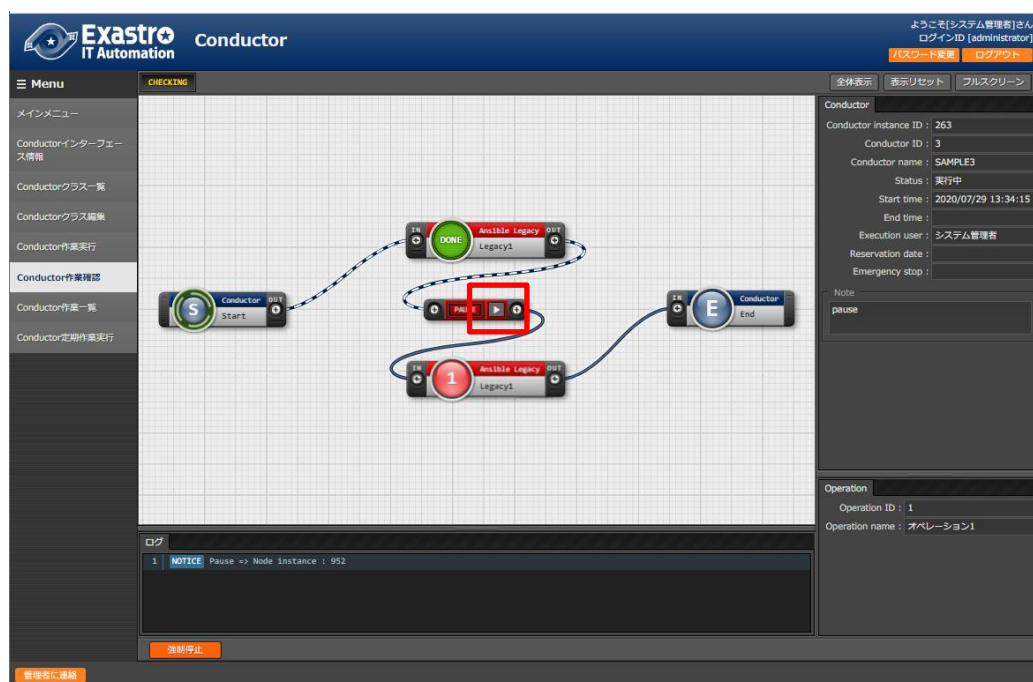


図 4.1-26 サブメニュー画面(Conductor 作業確認 - 停止解除)

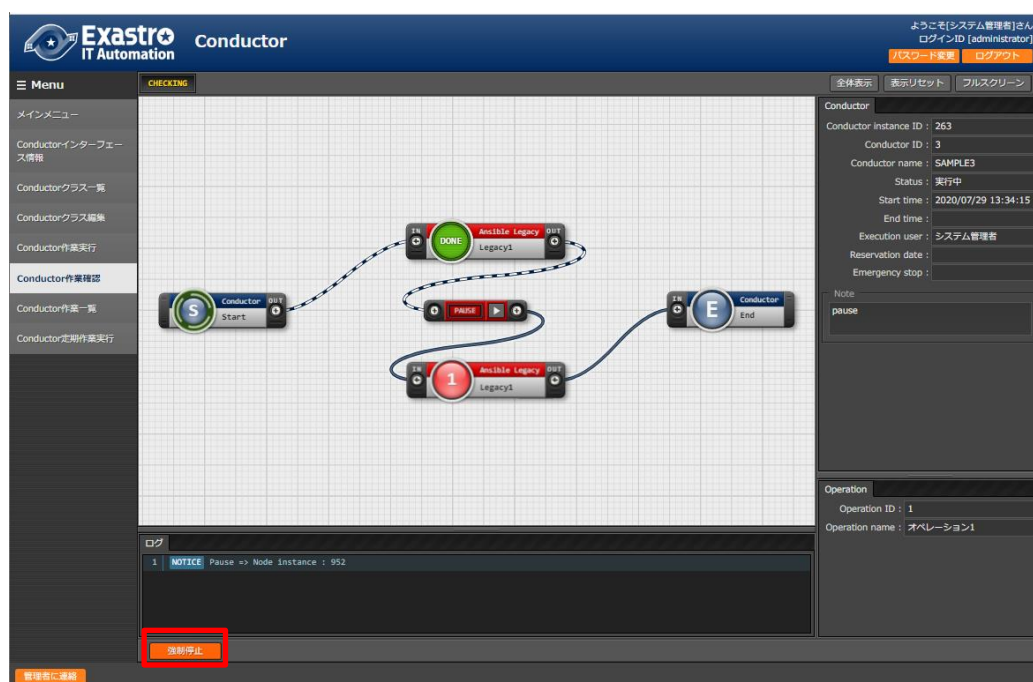


図 4.1-27 サブメニュー画面(Conductor 作業確認 – 緊急停止)

Conductor 作業確認の共通項目一覧は以下のとおりです。

表 4.1-14 登録画面項目一覧(Conductor 作業確認)

項目	説明	入力必須	入力形式	制約事項
停止解除	一時停止を解除します	－	ボタン	－
緊急停止	Conductor の実行を中止します	－	ボタン	－
予約取消	Conductor の実行予約を取り消します	－	ボタン	予約日時が設定されていて、かつ未実行の場合に表示される

4.1.6 Conductor 作業一覧

- (1) [Conductor 作業一覧]画面では、実行済みの Conductor の作業を管理します。
条件を指定し「フィルタ」ボタンをクリックすると、作業一覧テーブルを表示します。

作業表示欄の「詳細」ボタンをクリックすると、「4.1.5 Conductor 作業確認」画面に遷移します。

「投入データ式(zip)」の「download(.zip)」をクリックすると、実行された全ての Conductor 配下の Movement(*)の実行ファイルなどをまとめてダウンロードすることができます。

「結果データ式(zip)」の「download(.zip)」をクリックすると、実行された全ての Conductor 配下の Movement(*)の実行ログ、エラーログなどをまとめてダウンロードすることができます。

(*)Conductor が階層構造になっている場合は、末端の Movement も対象になります。

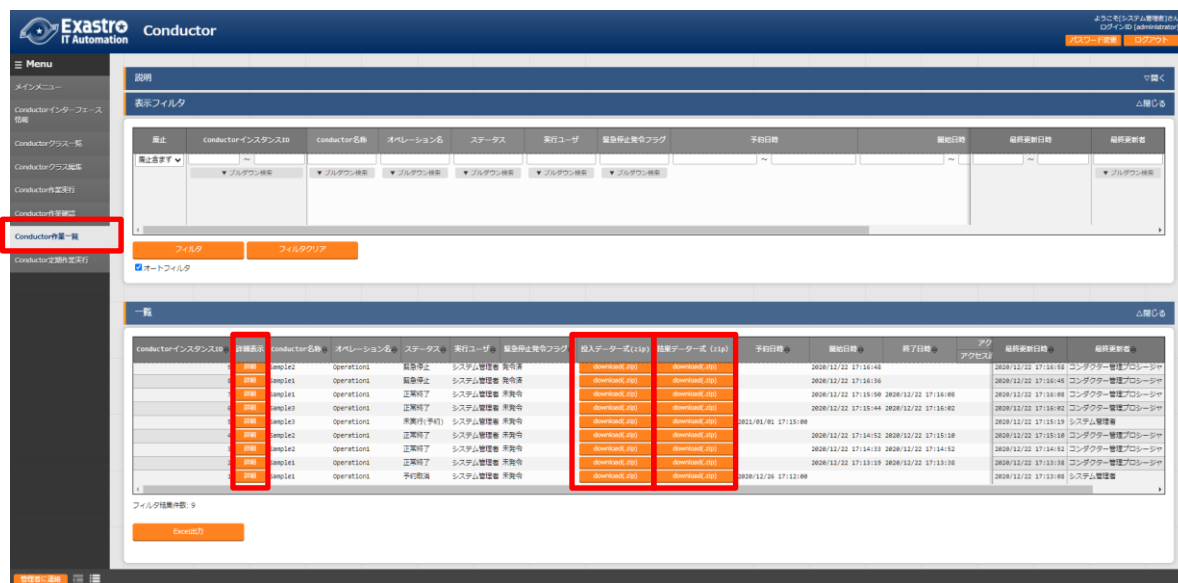


図 4.1-28 サブメニュー画面(Conductor 作業一覧)

4.1.7 Conductor 定期作業実行

- (1) [定期作業実行]画面では、スケジュールに従って定期的に実行する Conductor 作業を管理します。
一覧の「作業一覧確認」ボタンをクリックすると、その定期作業で実行した対象をフィルタした状態で「4.1.6 Conductor 作業一覧」へ遷移できます。

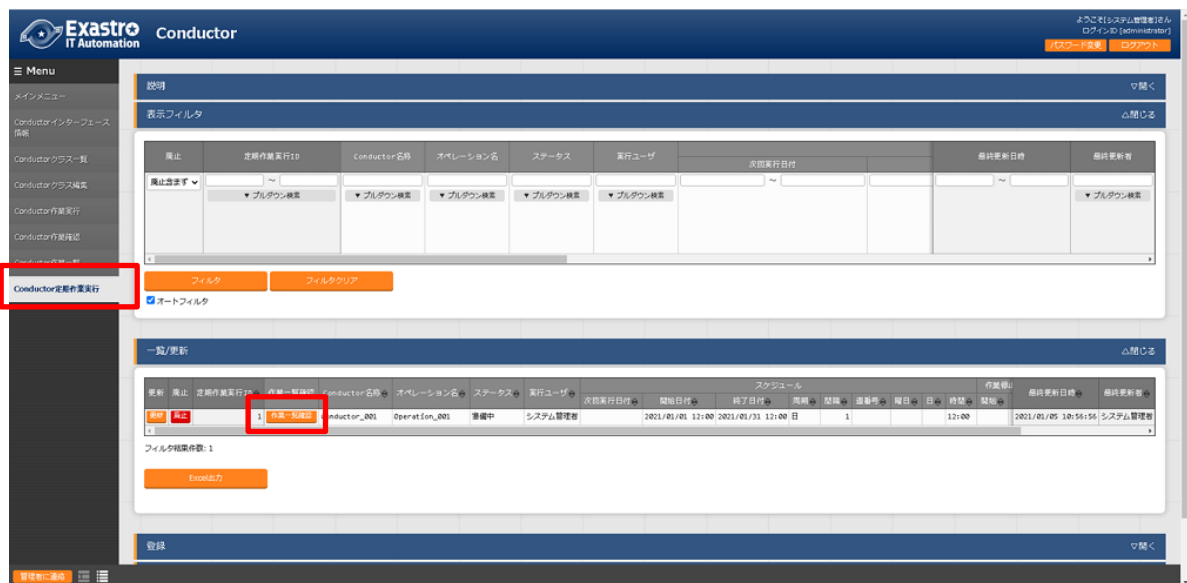


図 4.1-29 サブメニュー画面(Conductor 定期作業実行)

- (2) 「登録」-「登録開始」ボタンより、定期作業実行の登録を行います。
スケジュールの詳細は「スケジュール設定」ボタンをクリックすることで表示されるウインドウからのみ設定できます。

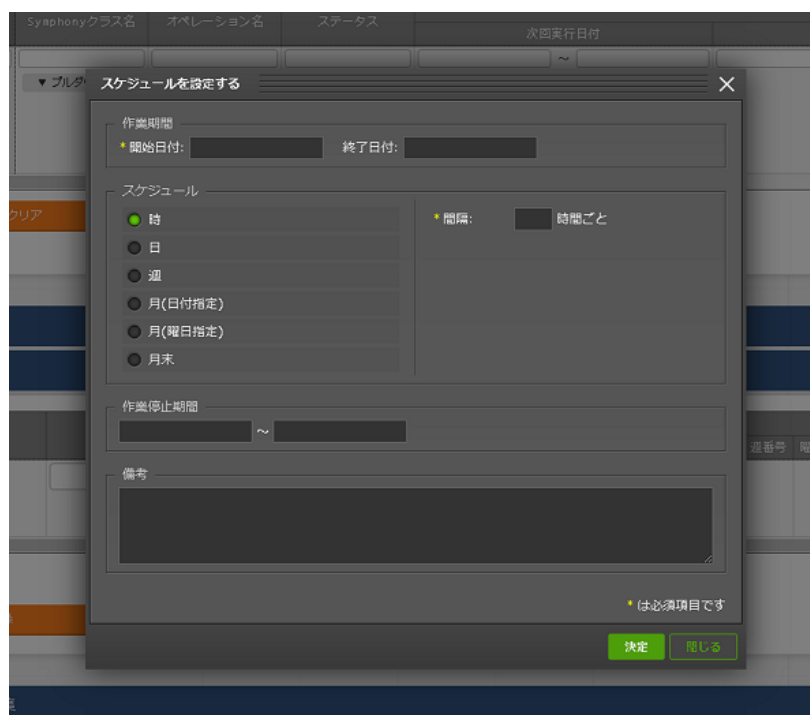


図 4.1-30 スケジュール設定画面(定期作業実行)

- (3) 登録画面の項目一覧は以下のとおりです

表 4.1-15 登録画面項目一覧(定期作業実行)

項目	説明	入力 必須	入力形式	制約事項
Conductor クラス名	「4.1.2Conductor クラス一覧」で登録した Conductor がリストに表示されます。	○	リスト選択	-

オペレーション名		「基本コンソール-投入オペレーション一覧」で登録したオペレーションがリストに表示されます。	○	リスト選択	-
ステータス		下記「表 4.2 11 ステータス一覧(定期作業実行)」参照	-	自動入力	-
実行ユーザ		「登録」「更新」を実行したユーザが Conductor を実行するユーザとして登録されます。 定期作業実行が「4.1.6Conductor 作業一覧」へ作業登録をする際、「実行ユーザ」が引き継がれて登録されます。 「実行ユーザ」が指定した「Conductor クラス名」を実行できない場合 (Conductor で実行される Movement のアクセス権限を持っていない等)、ステータス「紐付けエラー」となります。	-	自動入力	
スケジュール設定		詳細なスケジュールを設定するウインドウをオープンするボタンです。	-	-	-
スケジュール	次回実行日付	登録完了後にスケジュール設定に基づいて、次に作業を実行する日付が自動的に更新されます。	-	自動入力	-
	開始日付	定期作業実行を開始する日付を入力します。 「次回実行日付」は必ず「開始日付」以降の日付で更新されます。	○	手動入力	スケジュール設定からのみ入力可能
	終了日付	定期作業実行を終了する日付を入力します。 「次回実行日付」が「終了日付」を超えた場合にステータスが「完了」となります。	-	手動入力	スケジュール設定からのみ入力可能
	周期	定期的に行う周期を選択します。 「時」「日」「週」「月(日付指定)」「月(曜日指定)」「月末」があります。	○	ラジオボタン	スケジュール設定からのみ入力可能
	間隔	設定した周期に基づいた、定期的に行う間隔を入力します。	○	手動入力	スケジュール設定からのみ入力可能
	週番号	周期で「月(曜日指定)」を選択した場合に利用する、定期的に行う週番号を選択します。	※1	リスト選択	スケジュール設定からのみ入力可能
	曜日	周期で「曜日」「月(曜日指定)」を選択した場合に利用する、定期的に行う曜日を選択します。	※2	リスト選択	スケジュール設定からのみ入力可能
	日	周期で「月(日付指定)」を選択した場合に利用する、定期的に行う日にちを入力します。	※3	手動入力	スケジュール設定からのみ入力可能
	時間	定期的に行う時間を入力します。	※4	手動入力	スケジュール設定からのみ入力可能
作業停止期間	開始	作業停止期間の開始日付を入力します。 開始日付以降から終了日付以前までの間、Conductor 作業登録が実行されないようにします。	※5	手動入力	スケジュール設定からのみ入力可能
	終了	作業停止期間の終了日付を入力します。 開始日付以降から終了日付以前までの間、Conductor 作業登録が実行されないようにします。	※5	手動入力	スケジュール設定からのみ入力可能
備考		自由記述欄です	-	手動入力	-

※1 週番号は周期に「月(曜日指定)」を選択した場合は必須です。

※2 曜日は周期に「月(曜日指定)」を選択した場合は必須です。

※3 日は周期に「月(日付指定)」を選択した場合は必須です。

※4 時間は周期に「日」「週」「月(日付指定)」「月(曜日指定)」「月末」を選択した場合は必須です。

※5 作業停止期間を設定する場合は「開始」と「終了」両方が入力されていることが必須です。

表 4.1-16 ステータス一覧(定期作業実行)

ステータス名	説明
準備中	登録した直後のステータスです。 backyard が「次回実行日付」を自動更新するタイミングで「稼働中」になります。
稼働中	正常稼働中のステータスです。

	「次回実行日付」の3分前に「4.1.6Conductor 作業一覧」への作業登録を実行し、再びスケジュール設定に基づいて「次回実行日付」が更新されます。
完了	「次回実行日付」が「終了日付」を超えた場合になるステータスです。以降は Conductor 作業登録を行いません。
不整合エラー	スケジュールの設定値に不正がある場合になるステータスです。
紐付けエラー	「4.1.6Conductor 作業一覧」への作業登録が失敗した場合になるステータスです。ステータス「稼働中」同様、「4.1.6Conductor 作業一覧」への作業登録を実行し、再びスケジュール設定に基づいて「次回実行日付」を更新するという動作をします。その際再び作業登録が失敗した場合、ステータス「紐付けエラー」が継続されます。
想定外エラー	ステータス「不整合エラー」「紐付けエラー」以外の不具合が発生した場合になるステータスです。
Conductor 廃止	登録した Conductor が廃止された場合になるステータスです。廃止された Conductor を復活させた場合はステータス「準備中」へと更新されます。
operation 廃止	登録した operation が廃止された場合になるステータスです。廃止された operation を復活させた場合はステータス「準備中」へと更新されます。

- (4) 定期作業実行に登録した直後はステータスが「準備中」となり、その後 backyard がスケジュール設定に基づいて「次回実行日付」を更新し、そのタイミングでステータスは「稼働中」となります。ステータスが「稼働中」か「紐付けエラー」の作業は「次回実行日付」の3分前に「4.1.6Conductor 作業一覧」への作業登録を実行し、再びスケジュール設定に基づいて「次回実行日付」が更新されます。

※一時停止を設定している Conductor を定期作業実行に登録した場合、作業登録後に「4.1.5 Conductor 作業確認」より「保留解除」の投入を行わないと「4.1.6Conductor 作業一覧」にステータス「実行中」で残り続けます。