



IT Automation

クイックスタート

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

Exastro IT Automation Version 1.10
Exastro developer

目次

1. はじめに

- 1.1 [はじめに](#)
- 1.2 [本書のシナリオと作業範囲の位置づけ](#)
- 1.3 [各種用語の説明](#)

2. 画面説明

- 2.1 [Webコンソール画面\(ログイン\)](#)
- 2.2 [画面説明\(メインメニュー\)](#)
- 2.3 [画面説明\(各メニュー\)](#)

3. 実行前準備

- 3.1 [Playbookをアップロードしてジョブ\(Movement\)に紐付け](#)
- 3.2 [ジョブ\(Movement\)をジョブフロー\(Conductor\)に組込み](#)
- 3.3 [CMDBにパラメータシートを設定](#)
- 3.4 [パラメータシートの項目とPlaybookの変数の紐付け](#)

4. 実行操作(1回目)

- 4.1 [機器一覧にターゲットとなるLinuxマシンを登録](#)
- 4.2 [作業名\(Operation\)の登録](#)
- 4.3 [パラメータシートにデータを登録](#)
- 4.4 [ジョブフロー\(Conductor\)の実行](#)
- 4.5 [実行結果の確認](#)

目次

5. 実行操作(2回目)

- 5.1 [作業名\(Operation\)の登録](#)
- 5.2 [パラメータシートにデータを登録](#)
- 5.3 [ジョブフロー\(Conductor\)の実行](#)
- 5.4 [実行結果の確認](#)

6. CMDBパラメータの履歴確認

- 6.1 [作業実行と履歴管理](#)
- 6.2 [CMDBパラメータの履歴を確認する](#)

A 付録

- 参考① [【Ansible-Legacy】単体実行](#)
- 参考② [【Ansible-Legacy】実行確認](#)

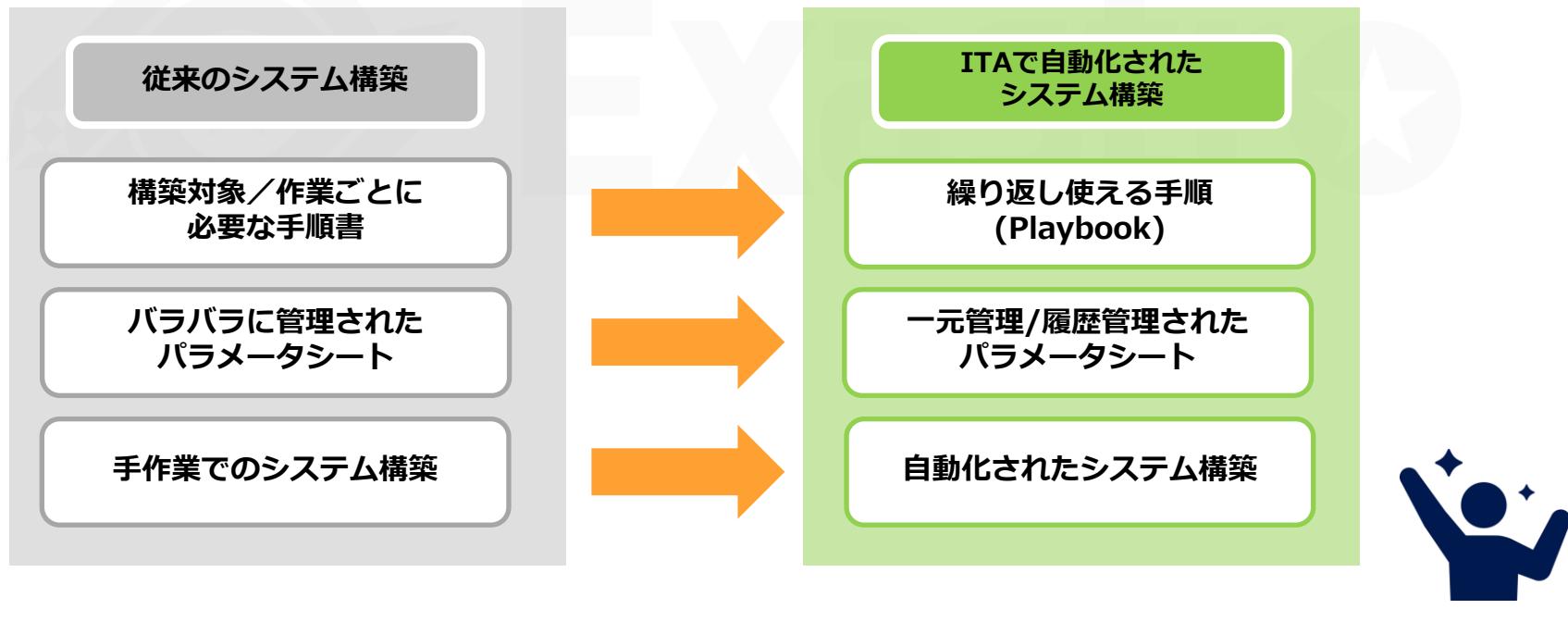
1. はじめに



1.1 はじめに (1/2)

本書は、はじめて IT Automation (以下ITAと記載) に触れるユーザが、ITAのインターフェースをスムーズに体感できるクイックスタートの手順書としてご活用できます。

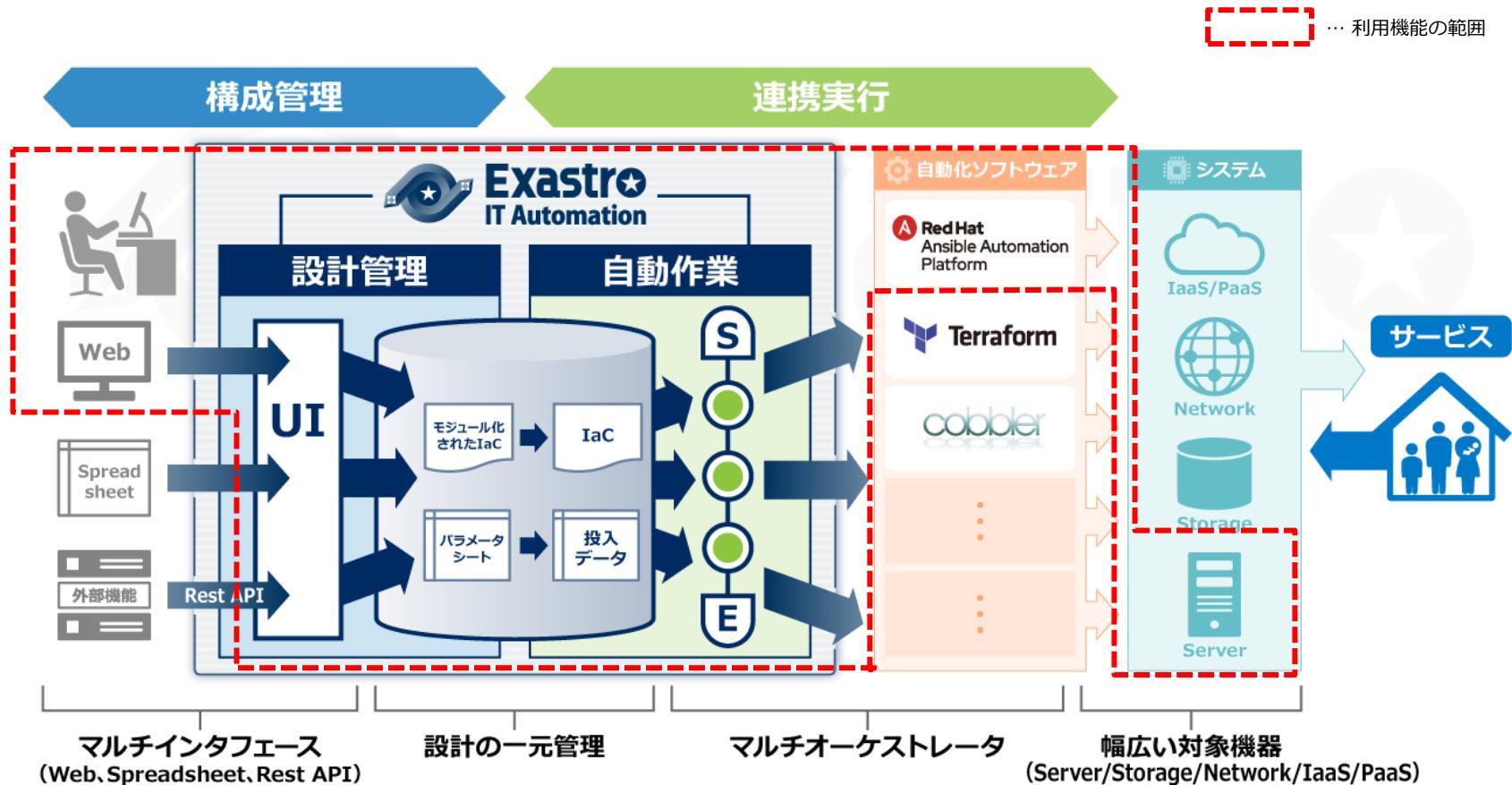
システム構築においてよくあるLinuxサーバのパッケージのインストール作業を通して、構築対象サーバごとの作業とパッケージ管理を自動化・一元管理化を行い、従来のシステム構築とは異なるITAを使用した効率的なシステム構築を体感できます。



1.1 はじめに (2/2)

■ 本クイックスタートで体感できる主なITA機能の範囲

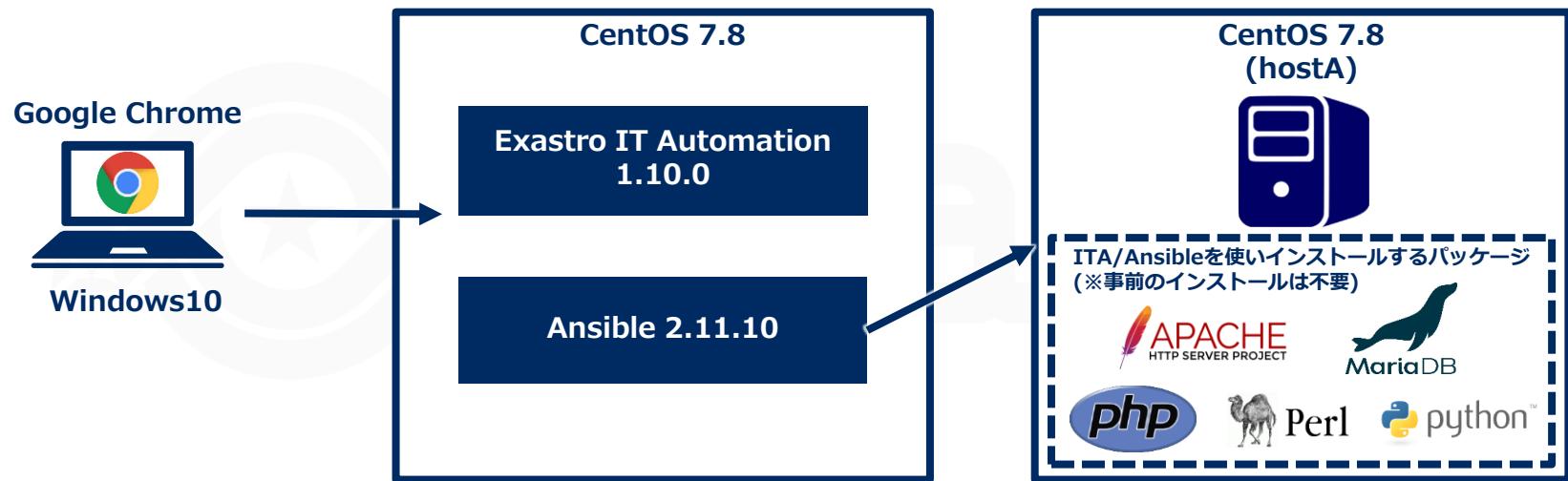
- ・自動化ソフトウェア(Ansible)との連携
- ・パラメータ管理(メニュー作成・登録・履歴管理等)
- ・変数紐付け(代入値自動登録)



1.2 本書のシナリオと作業範囲の位置づけ (1/3)

本シナリオではAnsibleドライバを使用し、Linuxサーバ構築で実施するyumパッケージのインストール作業を構築対象サーバごとにパラメータ管理し、構築作業の自動化を行う内容となっています。

■ 作業環境

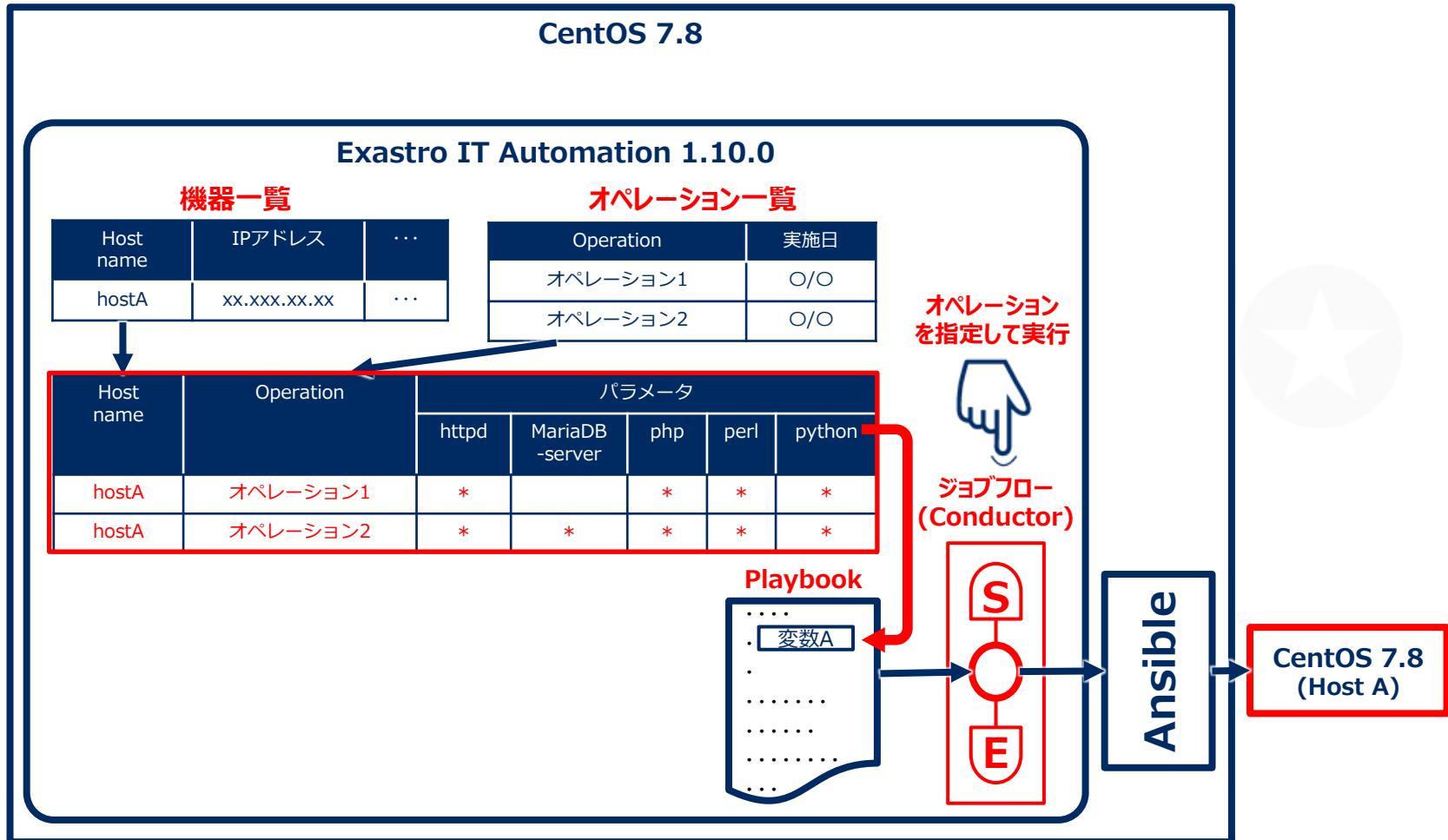


使用するシステム

- Exastro IT Automation 1.10.0
- CentOS Linux 7.8(ITAサーバ用)
- CentOS Linux 7.8(ターゲットマシン用)
- Windows 10(クライアント)
- Google Chrome (Win10側)

1.2 本書のシナリオと作業範囲の位置づけ (2/3)

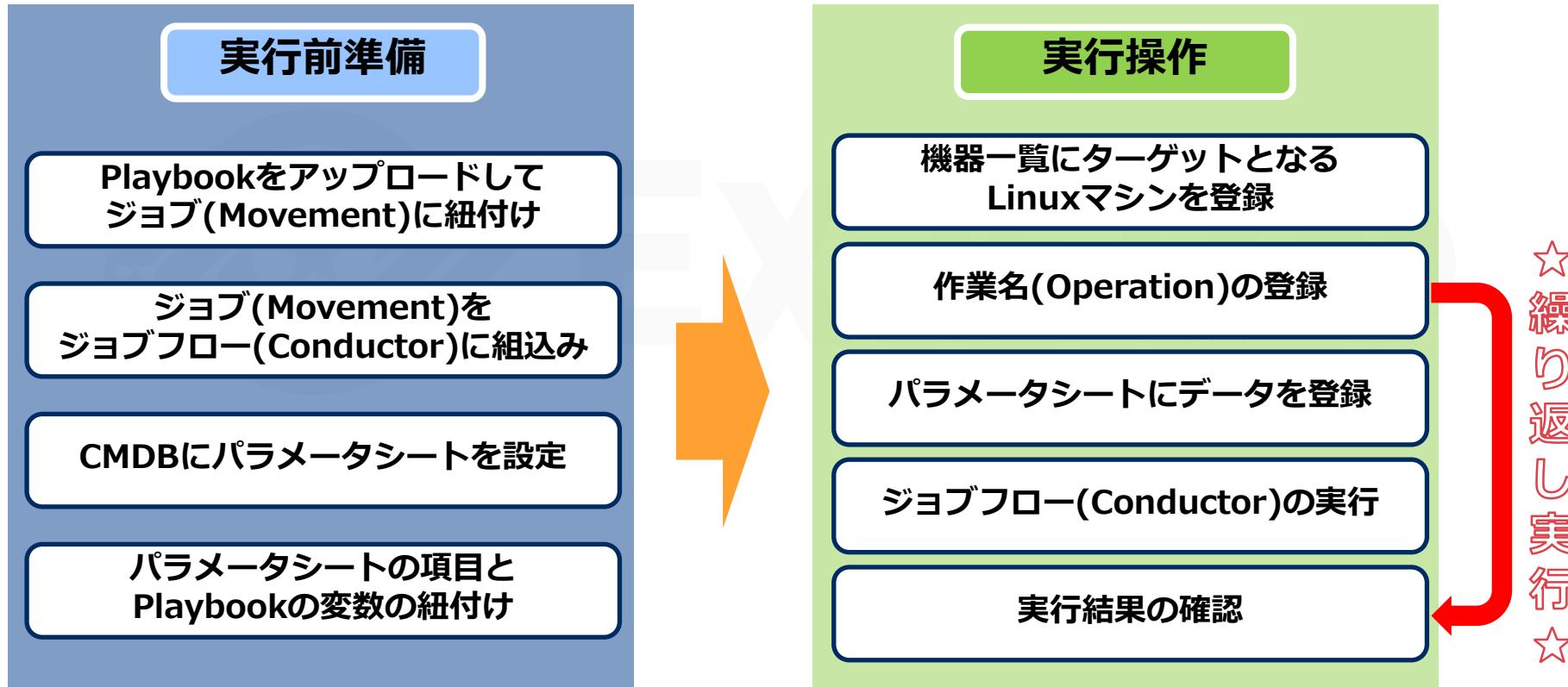
シナリオの実行イメージ



1.2 本書のシナリオと作業範囲の位置づけ (3/3)

インストール後からAnsible-Legacyを実行するまでのシナリオ

- シナリオと、開発者(実行前準備)／作業者(実行操作)の作業範囲については以下の通り。



1.3 各種用語の説明

■ 本シナリオに登場する主な用語

用語	説明
Playbook	定型業務をタスクで記述し、Ansibleに実行させるためのファイルです。YAML形式で使用します。
Ansible-Legacy	ITA から Ansible を利用する機能です。Legacy コンソールでは、構築コードとして単体のYAMLファイルを使う場合に使用します。
オペレーション名 (Oparation)	ITA での作業実行単位。作業予定、実行履歴などを管理することができます。
Conductor	ITA での一連の作業の単位。オペレーション名と関連付けて実行します。Node と呼ぶ各種パートを組み合わせて、ジョブフローを作成し、複数の機器に対して、一連の構築・設定などの作業を行います。
Movement	各機器に対する構築ツールを使った構築、設定などの作業の単位です。

Exastro ITAについてのより詳しい説明は[ドキュメントサイト](#)をご活用ください。

2. 画面説明



2.1 Webコンソール画面（ログイン）

Webコンソールログイン

- ITAのインストールが完了しURLへアクセスすると、ログイン画面が表示されます。
※インストール手順については”[IT Automation オンラインインストール](#)”をご参照ください。



Point

初回ログイン時は、ログイン直後に
パスワード変更を求められます。

2.2 画面説明(メインメニュー)

画面説明（メインメニュー）

- 基本的な名称は以下の通りです。

The screenshot shows the Exastro Management Console dashboard. A red box highlights the left sidebar under the heading "Menu". A larger red box highlights the "DASHBOARD" section, which contains several cards:

- メニュー**: A card showing icons for Management Console, Basic Console, Export, Symphony, Conductor, and Menu Creation.
- メニュー**: A card showing icons for Comparison, Post-Group, Ansible Interoperability, Ansible-Legacy, Ansible-Pioneer, and Ansible-Legacy.
- メニュー**: A card showing icons for Terraform and CI/CD for IaC.
- Movement**: A card with a donut chart showing Movement Total (0) and a table of movement status counts.
- 作業状況**: A card with a donut chart showing Status Total (0) and a table of job status counts.
- 作業結果**: A card with a donut chart showing Result Total (0) and a table of job result counts.

Annotations with red lines and labels:

- A red line points from the "バージョン確認" item in the sidebar to the "Version Confirmation" section at the bottom of the dashboard.
- A red line points from the "管理者に連絡" button at the bottom of the sidebar to the "Contact Manager" section at the bottom of the dashboard.
- A red line points from the "メニュー" label at the bottom to the "Menu" card in the dashboard.
- A red line points from the "メニュー" label at the bottom to the "Menu Group" card in the dashboard.

2.3 画面説明(各メニュー) (1/2)

画面説明 (各メニュー)

- 基本的な名称は以下の通りです。

■ サブメニューの概略

■ 説明 : 表示中メニューの説明

■ 表示フィルタ : 登録情報の検索機能

■ 一覧/更新 : 登録情報の表示

Exastro IT Automation Ansible-Legacy

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

ロール 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

廃止 MovementID Movement名 オーケストレータ 遅延タイマー

廃止含まず ~ ブルダウンド検索 ブルダウンド検索 ブルダウンド検索

▼開く △閉じる

△開く △閉じる

フィルタ フィルタクリア

オートフィルタ

一覧/更新

履歴 備考 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

ホスト指定形式 WinRM接続 virtualenv ヘッダーセクション オプション

最終更新日時 最終更新者

履歴 備考 更新 廃止 2 パッケージインストール Ansible Legacy IP 2021/12/16 11:51:37 システム管理者

ファイル結果件数: 1

Excel出力

登録

全件ダウンロードとファイルアップロード編集

管理者に連絡

2.3 画面説明(各メニュー) (2/2)

画面説明 (各メニュー)

- 基本的な名称は以下の通りです。

サブメニュー

■ サブメニューの概略

登録 : Webからのレコード登録

全件ダウンロードとファイルアップロード編集

**: Excelのダウンロード/アップロードからの編集
Excelでの変更履歴全件をダウンロード**

変更履歴 : 登録レコードの変更履歴

3. 実行前準備



3.1 Playbookをアップロードしてジョブ(Movement)に紐付け (1/5)

Playbookの準備

- 最初に今回利用するPlaybookの作成をします。

お好みのエディタを使用してymlを作成し自身のローカルフォルダに保存してください。

yum_package_install.yml

```
- name: install the latest version of packages
  yum:
    name: "{{ item }}"
    state: latest
  with_items:
    - "{{ VAR_packages }}"
```

Playbookをアップロードして
ジョブ(Movement)に紐付け

ジョブ(Movement)を
ジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目と
Playbookの変数の紐付け

機器一覧にターゲットとなる
Linuxマシンを登録

Point

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ください。

yum_package_install.yml ×

```
1   name: install the latest version of packages
2     yum:
3       name: "{{ item }}"
4       state: latest
5     with_items:
6       - "{{ VAR_packages }}"
7
```

3.1 Playbookをアップロードしてジョブ(Movement)に紐付け (2/5)

「Movement一覧」へ新規Movementを登録

- 次にMovementの登録を行っていきます。

メインメニューより、「Ansible-Legacy」メニューグループ >> 「Movement一覧」メニューをクリックします。

The screenshot shows the Ansible-Legacy dashboard with the 'Movement' section highlighted. A sidebar on the right lists five steps for Movement registration:

- Playbookをアップロードしてジョブ(Movement)に紐付け
- ジョブ(Movement)をジョブフロー(Conductor)に組込み
- CMDBにパラメータシートを設定
- パラメータシートの項目とPlaybookの変数の紐付け
- 機器一覧にターゲットとなるLinuxマシンを登録

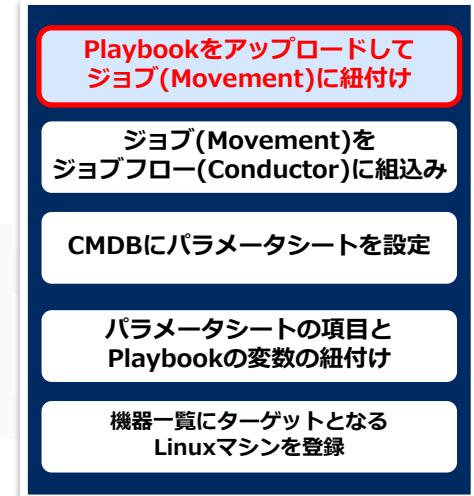
Numbered circles 1 and 2 indicate specific actions: circle 1 points to the 'Ansible Legacy' icon in the main menu, and circle 2 points to the 'Movement' link in the sidebar.

3.1 Playbookをアップロードしてジョブ(Movement)に紐付け (3/5)

「Movement一覧」へ新規Movementを登録

- 「登録開始」ボタンをクリック。
各項目へ下表のように入力し、登録をクリックしてください。

Movement名	ホスト指定形式	オプションパラメータ
パッケージインストール	IP	-vvv



Point

詳細なログ確認を行うために
オプションパラメータを-vvvにして
ログレベル上げます。

詳しくは[マニュアル](#)を参照してください。

MovementID	Movement名	遅延タイマー	ホスト指定形式*	Ansible利用情報	WinRM接続	ヘッダーセクション	オプションパラメータ
新規登録	パッケージインストール		IP				-vvv

3.1 Playbookをアップロードしてジョブ(Movement)に紐付け (4/5)

「Playbook素材集」へ新規Playbookを登録

- 次に作成したPlaybookの登録を行います。

「Ansible-Legacy」メニューグループ >> 「Playbook素材集」メニューをクリック。

登録開始をクリックし、各項目へ下表のように入力し登録を
クリックしてください。

The screenshot shows the 'Ansible-Legacy' interface with the 'Playbook素材集' menu selected. A red box labeled '1' highlights the 'Playbook素材名' field, which contains 'yum_package_install'. A red box labeled '2' highlights the 'Playbook素材' field, which contains 'yum_package_install.yml'. A red box labeled '3' highlights the '登録' (Register) button at the bottom.

Table settings:

素材ID	Playbook素材名	Playbook素材	アクセス権	備考	最終更新日時	最終更新者
廃止候補	yum_package_install	yum_package_install.yml	アクセス許可ロール			

File upload section:

素材ID	Playbook素材名*	Playbook素材*	アクセス権	備考	最終更新日時	最終更新者
自動入力	yum_package_install	ファイルを選択 yum_package_install.yml	設定 アクセス許可ロール		自動入力	自動入力

Point:

Playbookをアップロードする場合
ファイル指定後は必ず
「事前アップロード」ボタンをクリックしてください。

Playbookをアップロードして
ジョブ(Movement)に紐付け

ジョブ(Movement)を
ジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目と
Playbookの変数の紐付け

機器一覧にターゲットとなる
Linuxマシンを登録

3.1 Playbookをアップロードしてジョブ(Movement)に紐付け (5/5)

「Movement-Playbook紐付」への登録

- 次に登録したPlaybookをMovementに紐付けます。

「Ansible-Legacy」メニューグループ >> 「Movement-Playbook紐付」メニューをクリック。各項目へ下表のように入力、選択し登録をクリックしてください。

The screenshot shows the Ansible-Legacy interface with the 'Movement-Playbook紐付' menu item highlighted by a red box and a red circle labeled '1'. The main window displays a table with columns: 紐付項目番号, Movement, Playbook素材, インクルード順序, 最終更新日時, and 最終更新者. A red box highlights the 'Movement' column, which contains 'パッケージインストール'. The 'Playbook素材' column contains 'yum_package_install'. The 'インクルード順序' column contains '1'.

- Playbookをアップロードしてジョブ(Movement)に紐付け
- ジョブ(Movement)をジョブフロー(Conductor)に組込み
- CMDBにパラメータシートを設定
- パラメータシートの項目とPlaybookの変数の紐付け
- 機器一覧にターゲットとなるLinuxマシンを登録

The screenshot shows the '登録' dialog box. The 'Movement' field is set to '2:パッケージインストール', the 'Playbook素材' field is set to 'yum_package_install', and the 'インクルード順序' field is set to '1'. A red box highlights the 'Movement' field, and a red circle labeled '2' points to it. A red box highlights the 'Point' section below the dialog.

The screenshot shows the '登録' dialog box. The 'Point' section is highlighted with a red box and a red circle labeled '3' pointing to the '登録' (Register) button. The 'Point' text states: 'Movement 1つに対し複数のPlaybookを登録する場合、インクルード順序を指定 1 : 1 の場合は1を入力してください。' (When registering multiple Playbooks for a single Movement, specify the include order. If it's 1 : 1, enter 1). The '登録' button is highlighted with a red box and a red circle labeled '3'.

Movement 1つに対し複数のPlaybookを登録する場合、インクルード順序を指定
1 : 1 の場合は1を入力してください。

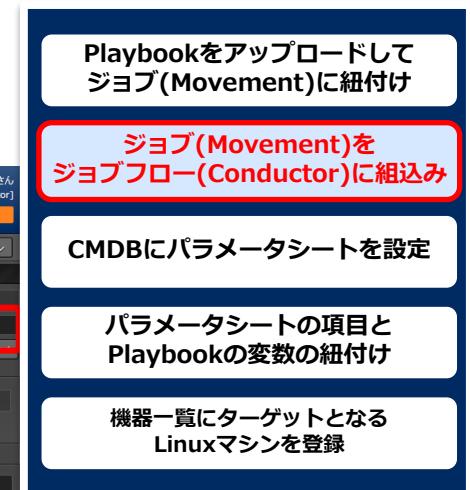
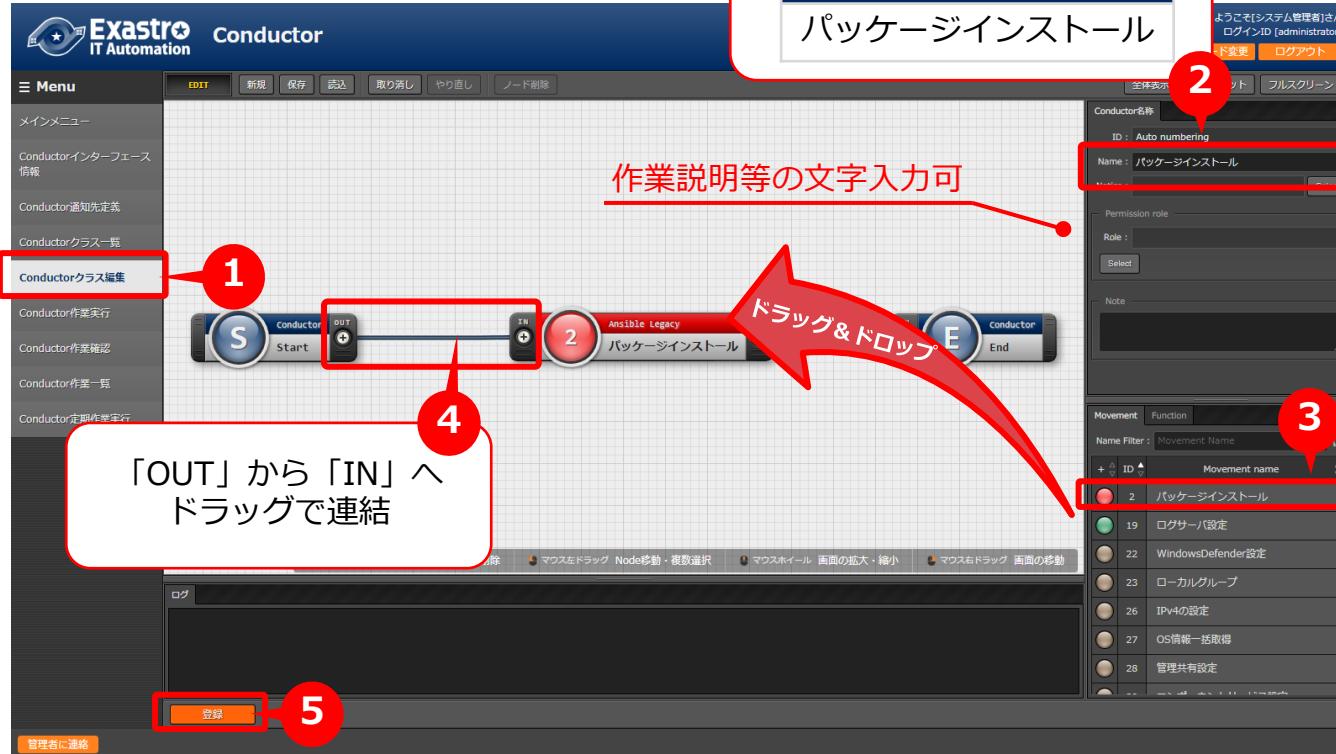
3.2 ジョブ(Movement)をジョブフロー(Conductor)に組込み

「Conductor」を作成する

- 次にMovementをConductorに組み込んでいきます。

「Conductor」メニューグループ >> 「Conductorクラス編集」メニューをクリック。

下記の順番の通り入力、移動、連結させ登録をクリックしてください。



3.3 CMDBにパラメータシートを設定(1/3)

パラメータシートを作成する

- 次にパラメータシートの作成を行います。

「メニュー作成」 メニューグループ >> 「メニュー定義・作成」 メニューをクリック。

各項目へ下表のように入力、選択して下さい。(次ページへ続く)

2 グループをクリック

3 グループ名

1 メニュー名

メニュー名	作成対象	表示順序
インストールパッケージ一覧	パラメータシート (ホスト/オペレーションあり)	1



3.3 CMDBにパラメータシートを設定(2/3)

■ パラメータシートを作成する

- 項目を追加し、各項目へ下表のように入力、選択して下さい。(次ページへ続く)

項目が5個になるようにクリック

4

5

項目名	入力方式	選択項目
httpd	プルダウン選択	メニュー作成:選択1:*(-ブランク)
MariaDB-server	プルダウン選択	メニュー作成:選択1:*(-ブランク)
php	プルダウン選択	メニュー作成:選択1:*(-ブランク)
perl	プルダウン選択	メニュー作成:選択1:*(-ブランク)
python	プルダウン選択	メニュー作成:選択1:*(-ブランク)



Point

今回はCentOS7.8を対象としています。 CentOS7系以外は『mariadb-server』と小文字で入力してください。

3.3 CMDBにパラメータシートを設定(3/3)

■ パラメータシートを作成する

- 項目の移動が完了できたら作成をクリックして下さい。

The screenshot shows the Exastro IT Automation interface for creating a menu. On the left, there's a sidebar with 'Menu' and 'メニュー定義・作成' sections. The main area has tabs for 'メニュー作成' (highlighted), 'グループ', and 'リピート'. A red box labeled '6' highlights the 'カラムグループ' (Column Group) section where items like 'httpd', 'MariaDB-server', 'php', 'perl', and 'python' are listed. Red arrows point from this section to a preview window below. Another red box labeled '7' highlights the 'すべての項目を移動すると以下のようにになります' (What it will look like after moving all items) section, which shows the same five items now in a single row. A red box labeled '8' highlights the '作成' (Create) button at the bottom of the preview window. To the right, a sidebar lists steps: 'Playbookをアップロードしてジョブ(Movement)に紐付け', 'ジョブ(Movement)をジョブフロー(Conductor)に組込み', 'CMDBにパラメータシートを設定' (highlighted in red), 'パラメータシートの項目とPlaybookの変数の紐付け', and '機器一覧にターゲットとなるLinuxマシンを登録'.

6 カラムグループの領域に項目をドラッグ&ドロップ

すべての項目を移動すると以下のようにになります

7

8 作成

Playbookをアップロードしてジョブ(Movement)に紐付け

ジョブ(Movement)をジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目とPlaybookの変数の紐付け

機器一覧にターゲットとなるLinuxマシンを登録

3.4 パラメータシートの項目とPlaybookの変数の紐付け(1/3)

「代入値自動登録設定」作成

- 最後に代入値自動登録を行います。

「Ansible-Legacy」メニュー群 >> 「代入値自動登録設定」メニューをクリック。
各項目へ下表のように入力、選択して下さい。(次ページへ続く)

Ansible-Legacy

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

三 Menu

メインメニュー

Movement一覧

Ansible-Legacy

Movement-Playbook紐付

代入値自動登録設定

1

2

説明

表示フィルタ

一覧/更新

登録

△閉じる

△開く

△開く

△開く

△閉じる

パラメータシート(From)

メニュー群:メニュー

項目

最終更新日時 最終更新者

項目番号:自動入力

2100011611:代入値自動登録用:205:インストールパッケージ一覧

パラメータ/インストールパッケージ/httpd

自動入力 自動入力

パラメータシートの項目と Playbookの変数の紐付け

ジョブ(Movement)を ジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

機器一覧にターゲットとなる Linuxマシンを登録

メニュー群:メニュー	項目	登録方式	Movement	Key変数 変数名	代入順序
2100011611:代入値自動登録用:3:インストールパッケージ一覧	パラメータ/インストールパッケージ/httpd	Key型	1:パッケージインストール	1:VAR_packages	1
2100011611:代入値自動登録用:3:インストールパッケージ一覧	パラメータ/インストールパッケージ/MariaDB-server	Key型	1:パッケージインストール	1:VAR_packages	2
2100011611:代入値自動登録用:3:インストールパッケージ一覧	パラメータ/インストールパッケージ/php	Key型	1:パッケージインストール	1:VAR_packages	3
2100011611:代入値自動登録用:3:インストールパッケージ一覧	パラメータ/インストールパッケージ/perl	Key型	1:パッケージインストール	1:VAR_packages	4
2100011611:代入値自動登録用:3:インストールパッケージ一覧	パラメータ/インストールパッケージ/python	Key型	1:パッケージインストール	1:VAR_packages	5

Playbookをアップロードして ジョブ(Movement)に紐付け

ジョブ(Movement)を ジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目と Playbookの変数の紐付け

機器一覧にターゲットとなる Linuxマシンを登録

3.4 パラメータシートの項目とPlaybookの変数の紐付け(2/3)

「代入値自動登録設定」作成

- 各項目へ入力、選択が完了したら登録をクリックして下さい。

The screenshot shows the 'Ansible-Legacy' interface with the 'Movement' registration screen. On the left, there's a sidebar with various menu items like 'Menu', 'Movement一覧', 'Playbook素材集', and 'Movement-Playbook紐付'. The main area has tabs for '説明', '表示フィルタ', '一覧/更新', and '登録'. The '登録' tab is active, showing a table titled '代入値自動登録設定' (Movement Registration Settings). The table has columns for '項目' (Item), '登録方式' (Registration Method), 'Movement' (Movement), 'Key変数' (Key Variable), 'Value変数' (Value Variable), '最終更新日' (Last Update Date), and '最終更新者' (Last Updated By). A row is selected in the table, and a red box highlights the entire table area. At the bottom of the table, there are two orange buttons: '戻る' (Back) and '登録' (Register), with the '登録' button circled in red and labeled with the number '2'.

Playbookをアップロードして
ジョブ(Movement)に紐付け

ジョブ(Movement)を
ジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目と
Playbookの変数の紐付け

機器一覧にターゲットとなる
Linuxマシンを登録

Point

変数紐づけの登録方式は以下の3タイプがあります。

登録方式	今回使用	説明
Value型		基本的なタイプであり、表の中の値を変数に紐づけるものです。
Key型	●	表の項目(列名)を変数に紐づけるものです。項目の設定値が空白の場合は紐づけ対象外になります。
Key-Value型		項目の名称(Key)と設定値(Value)の両方を変数に紐づけることができます。

今回のシナリオでは、表の項目(列名)をPlaybookに具体値として代入したいので、登録方式は「Key型」を選択します。詳細はこちら[システム構築・運用の効率化ガイドブック](#)を参照してください。

3.4 パラメータシートの項目とPlaybookの変数の紐付け(3/3)

「代入値自動登録設定」作成

- 表示フィルタで5件のデータが登録できているかの確認を行って下さい。
ここまでで実行準備は終了になります。

Ansible-Legacy

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

ロール 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

廃止 項番 メニューグループ メニュー 最終更新日時 最終更新者

ID 名称 ID 名称

廃止含まず ~ ブルダウント 検索 ブルダウント 検索 ブルダウント 検索 ブルダウント 検索

△閉じる

3

4 フィルタ フィルタクリア

5 5件のデータが登録できているかを確認

履歴 備考 更新 廃止 項番 メニューグループ メニュー 項目 登録方 最終更新日時 最終更新者

ID 名称 ID 名称

9:100011611 代入値自動登録用 205 インストールパッケージ_既 パラメータ/インストールパッケージ/httpd Key型 2021/12/16 12:08:11 システム管理者

10:100011611 代入値自動登録用 205 インストールパッケージ_既 パラメータ/インストールパッケージ/MariaDB-server Key型 2021/12/16 12:56:58 システム管理者

11:100011611 代入値自動登録用 205 インストールパッケージ_既 パラメータ/インストールパッケージ/php Key型 2021/12/16 12:57:19 システム管理者

12:100011611 代入値自動登録用 205 インストールパッケージ_既 パラメータ/インストールパッケージ/perl Key型 2021/12/16 12:57:39 システム管理者

13:100011611 代入値自動登録用 205 インストールパッケージ_既 パラメータ/インストールパッケージ/python Key型 2021/12/16 12:57:54 システム管理者

△閉じる

ファイル結果件数: 5

Excel出力

登録 △閉じる

Playbookをアップロードしてジョブ(Movement)に紐付け

ジョブ(Movement)をジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目とPlaybookの変数の紐付け

機器一覧にターゲットとなるLinuxマシンを登録

4. 実行操作(1回目)



4.1 機器一覧にターゲットとなるLinuxマシンを登録(1/3)

「機器一覧」へ新規ターゲットホストの登録

- 最初に機器一覧へ今回パッケージをインストールするターゲットホストを登録します。
「基本コンソール」メニューグループ >> 「機器一覧」メニューをクリック。
各項目へ下表のように入力して下さい。(次ページへ続く)

Exastro IT Automation 基本コンソール

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

メニュー

機器一覧

1

説明

登録

2

管理システム項目番号	HW機器種別	ホスト名*	IPアドレス*	EtherWakeOnLan	MACアドレス	ネットワークデバイス名	ログインユーザーID	最終更新日時	最終更新者
自動入力	SV	hostA	192.168.1.1					自動入力	自動入力

* *は必須項目です。

戻る 登録

管理者に連絡

HW機器種別	ホスト名	IPアドレス
SV	(任意のホスト名)	(任意のIPアドレス)



4.1 機器一覧にターゲットとなるLinuxマシンを登録(2/3)

「機器一覧」へ新規ターゲットホストの登録

- スクロールバーを右にスライドし各項目へ下表のように入力して下さい。(次ページへ続く)

Exastro IT Automation 基本コンソール

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

Menu

- メインメニュー
- 機器一覧
- オペレーション一覧
- Movement一覧
- ER図表示

説明

表示フィルタ

一覧/更新

登録

ロール パスワード変更 ログアウト

Playbookをアップロードしてジョブ(Movement)に紐付け

ジョブ(Movement)をジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目とPlaybookの変数の紐付け

機器一覧にターゲットとなるLinuxマシンを登録

2

管理システム項目番号	ログインユーザID	管理	ログインパスワード	ssh鍵認証情報	パスフレーズ	最終更新日時	最終更新者
自動入力	root	●	ファイルを選択 選択されていません	Q	自動入力	自動入力
事前アップロード							
アップロード状況:							

※*は必須項目です。

戻る 登録

ログインユーザID ログインパスワード管理 ログインパスワード

(任意のログインユーザID) ● (任意のパスワード)

管理者に連絡

4.1 機器一覧にターゲットとなるLinuxマシンを登録(3/3)

「機器一覧」へ新規ターゲットホストの登録

- 最後の項目へ下表のように選択し登録をクリックして下さい。

Exastro IT Automation 基本コンソール

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

メニュー

機器一覧

オペレーション一覧

Movement一覧

ER図表示

3

4

Legacy/Role利用情報

認証方式

パスワード認証

Ansible利用情報

Legacy/Role利用情報

WinRM接続情報

ポート番号 サーバー証明書

最終更新日時 最終更新者

自動入力 パスワード認証 ファイルを選択 選択されていません

事前アップロード

アップロード状況:

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

管理者に連絡

Playbookをアップロードしてジョブ(Movement)に紐付け

ジョブ(Movement)をジョブフロー(Conductor)に組込み

CMDBにパラメータシートを設定

パラメータシートの項目とPlaybookの変数の紐付け

機器一覧にターゲットとなるLinuxマシンを登録

Point

Ansible-Legacyを実行するための必須入力項目は以下の6項目です。
[ホスト名][IPアドレス][ログインユーザID]
[ログインパスワード管理][ログインパスワード]
[認証方式※]
※本書は「パスワード認証」で記載します。

4.2 作業名(Operation)の登録

「オペレーション一覧」へ新規オペレーション名を登録

- 次にオペレーション名を登録していきます。
「基本コンソール」メニューグループ >> 「オペレーション一覧」メニューをクリック。
各項目へ入力、選択が完了したら登録をクリックして下さい。

The screenshot shows the 'Basic Console' interface of Exastro IT Automation. On the left, a sidebar menu includes 'Menu', 'Main Menu', 'Machine List', and 'Operation List' (highlighted with a red box and circled with a red number 1). The main area displays the 'Operation List' screen with a search interface. A modal window titled 'Operation Registration' is open, overlaid on the list view. This modal contains fields for 'Operation Name' (set to 'Operation 1') and 'Implementation Date' (set to '(Any implementation date)'). The 'Register' button at the bottom of the modal is highlighted with a red box and circled with a red number 3. To the right of the main window, four steps are outlined in boxes:

- 作業名(Operation)の登録
- パラメータシートにデータを登録
- ジョブフロー(Conductor)の実行
- 実行結果の確認

※*は必須項目です。

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	備考	最終更新日時	最終更新者
自動入力	自動入力	オペレーション1	2021/12/17 11:20	設定	アグリゲート	2021/12/17 11:20	自動入力

4.3 パラメータシートにデータを登録(1/2)

「インストールパッケージ一覧」へ新規データを登録

- 次に実行前準備で用意したインストールパッケージ一覧(パラメータシート)にデータを入力していきます。

「入力用」メニューグループ >> 「インストールパッケージ一覧」メニューをクリック。
各項目へ入力、選択が完了したら登録をクリックして下さい。

Exastro IT Automation 入力用

説明

1 インストールパッケージ一覧

2 登録

3 登録

No	ホスト名*	オペレーション*	パッケージ	最終更新日	最終更新者
自動入力	hostA	2021/12/16 12:35_21:オペレーション1	httpd MariaDB-server php perl python	自動入力	自動入力

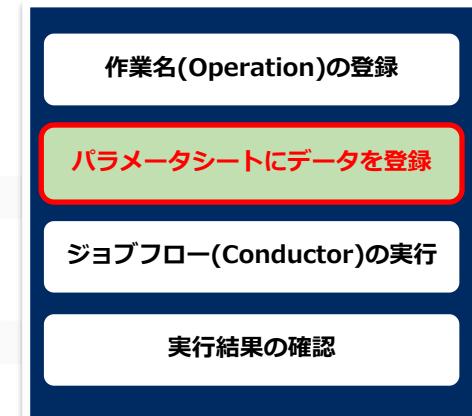
* *は必須項目です。

Table setting

戻る 登録

ホスト名	オペレーション	httpd	MariaDB-server	php	perl	python
(機器登録で登録したホスト名)	(選択した実行予定日時)_1: オペレーション1	*		*	*	*

管理者に連絡



4.3 パラメータシートにデータを登録(2/2)

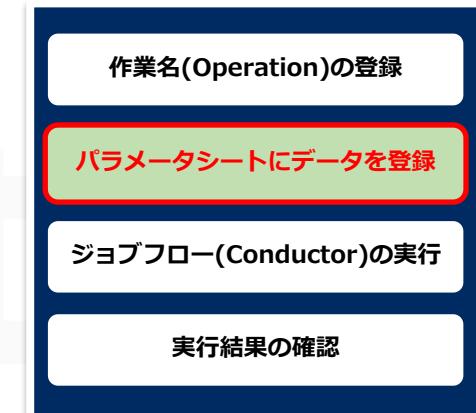
「インストールパッケージ一覧」への登録

- 実行前準備の代入値自動登録設定の時と同様、表示フィルタを開き「フィルタ」ボタンをクリックして登録したデータを確認してください。

The screenshot shows the Exastro IT Automation interface with the following steps highlighted:

- ① A red box highlights the 'Display Filter' section at the top of the main window.
- ② A red box highlights the 'Filter' button in the filter bar.
- ③ A red box highlights the 'Registration' button at the bottom of the main window.

The main window displays a table with columns: 廃止 (Delete), No., ホスト名 (Host Name), オペレーション (Operation), ID, オペレーション名 (Operation Name), 基準日時 (Standard Time), 実施予定日時 (Planned Execution Time), 最終実行日時 (Last Execution Time), インストールパッケージ (Install Package), 最終更新日時 (Last Update Time), and 最終更新者 (Last Updater). The table contains one row of data: hostA, 操作 (Operation), 21, オペレーション1 (Operation1), 2021/12/17 14:32, 2021/12/16 12:35, 2021/12/17 14:32, *, httpd, MariaDB-server, php, perl, python, 2021/12/16 13:08:52, システム管理者 (System Administrator).



4.4 ジョブフロー(Conductor)の実行(1/3)

Conductorの実行

- いよいよ実行を行っていきます。
「Conductor」メニューグループ>>「Conductor作業実行」メニューをクリック。
実行する「Conductor」と「オペレーション」を選択し実行をクリックして下さい。

The screenshot shows the Exastro IT Automation Conductor interface. The main window displays two tables: 'Conductor[一覧]' and 'オペレーション[一覧]'. Step 1 highlights the 'Conductor作業実行' menu item in the left sidebar. Step 2 highlights the first row in the 'Conductor[一覧]' table. Step 3 highlights the first row in the 'オペレーション[一覧]' table. Step 4 highlights the 'EXECUTE' button at the bottom of the screen. To the right, a vertical sidebar lists five steps: '作業名(Operation)の登録', 'パラメータシートにデータを登録', 'ジョブフロー(Conductor)の実行' (highlighted in red), and '実行結果の確認'.

1 Conductor作業実行

2

3

4 EXECUTE

作業名(Operation)の登録

パラメータシートにデータを登録

ジョブフロー(Conductor)の実行

実行結果の確認

4.4 ジョブフロー(Conductor)の実行(2/3)

実行結果確認

- 実行すると「Conductor作業確認」メニュー画面に切替わり、実行ステータスやログが表示されます。

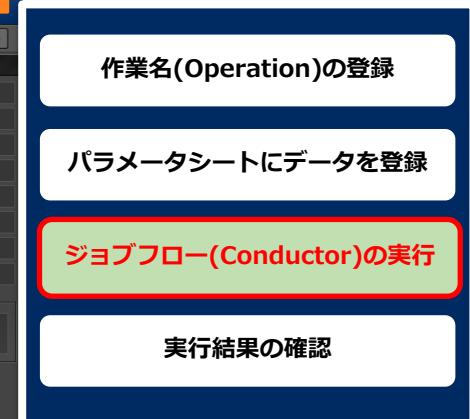
The screenshot shows the Exastro Conductor application interface. On the left is a navigation menu with options like 'Menu', 'Conductor', 'Operation', 'Log', and 'Help'. The main area displays a job flow diagram with three nodes: 'Start' (Conductor), 'Ansible Legacy' (パッケージインストール), and 'End' (Conductor). The 'Ansible Legacy' node is highlighted with a green 'DONE' status. To the right of the diagram is a detailed execution log:

Conductor name :	パッケージインストール
Status :	正常終了
Start time :	2021/12/17 11:47:15
End time :	2021/12/17 11:49:04
Execution user :	システム管理者
Reservation date :	
Emergency stop :	

Below this is another section for 'Operation' with details:

Operation ID :	21
Operation name :	オペレーション1

At the bottom left is a 'Log' panel which is currently empty. At the bottom right are several status indicators: 'マウス左クリック Node選択・作業状態確認', 'マウスホイール 画面の拡大・縮小', and 'マウス右ドラッグ 画面の移動'. A red 'Point' callout from the bottom right points to the 'Log' panel, with the text: '実行ステータスやログをリアルタイムで確認可能です。'.

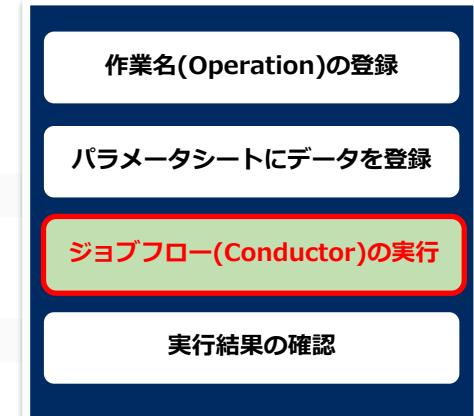


4.4 ジョブフロー(Conductor)の実行(3/3)

実行結果確認

- ジョブ(Movement)を選択し、Doneのアイコンまたは右側のOperation statusをクリックすると詳細が表示されます。

The screenshot shows the Exastro Conductor interface. On the left, there's a navigation menu with items like 'Menu', 'Main Menu', 'Conductor Interface Information', etc. The main area displays a workflow diagram with nodes: 'Start' (Conductor), 'Ansible Legacy' (Movement), and 'End' (Conductor). A red box highlights the 'Ansible Legacy' node, which has a green 'DONE' button and the text 'パッケージインストール'. A red callout bubble with the number '1' and the text '選択' points to this node. A red arrow points from this node to a detailed view window on the right. This window is titled 'Exastro Ansible-Legacy' and shows a table with various parameters for the selected movement, such as 'Movement ID: 2', 'Name: パッケージインストール', and 'Host Type: IP'.



4.5 実行結果の確認(1/3)

実行ログの確認

- 詳細画面の進行状況(実行ログ)でAnsibleの実行ログを確認していきます。

The screenshot shows the Ansible execution log details. A red box highlights the 'Execution Log (実行ログ)' window.

Execution Log (実行ログ)

File filter: [] Show only matching lines

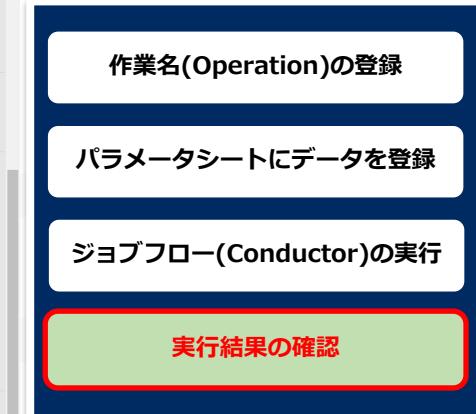
```
Upgrading : python-2.7.5-90.el7.x86_64          2/4
Cleanup   : python-2.7.5-90.el7.x86_64          3/4
Cleanup   : python-11bs-2.7.5-68.el7.x86_64      4/4
Verifying  : python-2.7.5-90.el7.x86_64          1/4
Verifying  : python-11bs-2.7.5-68.el7.x86_64      2/4
Verifying  : python-11bs-2.7.5-68.el7.x86_64      3/4
Verifying  : python-2.7.5-90.el7.x86_64          4/4

Updated:
python.x86_64 0:2.7.5-90.el7

Dependency Updated:
python-libs.x86_64 0:2.7.5-90.el7

Complete!
[]

PLAY RECAP ****
ita-test-target01 : ok=1    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



4.5 実行結果の確認(2/3)

実行ログの確認

- httpd,php,perl,pythonをインストールされているか実行ログから確認して下さい。

進行状況(実行ログ)の一部の例

```
~~~~~省略~~~~~  
Installed:  
    httpd.x86_64 0:2.4.6-97.el7.centos  
Dependency Installed:  
    httpd-tools.x86_64 0:2.4.6-97.el7.centos mailcap.noarch 0:2.1.41-2.el7  
Complete!  
"}}  
~~~~~省略~~~~~  
Installed:  
    php.x86_64 0:5.4.16-48.el7  
Dependency Installed:  
    libzip.x86_64 0:0.10.1-8.el7 php-cli.x86_64 0:5.4.16-48.el7 php-common.x86_64 0:5.4.16-  
48.el7  
Complete!  
"}}  
~~~~~省略~~~~~  
Updated:  
    perl.x86_64 4:5.16.3-299.el7_9  
Dependency Updated:  
    perl-libs.x86_64 4:5.16.3-299.el7_9  
Complete!  
"}}  
~~~~~省略~~~~~  
Updated:  
    python.x86_64 0:2.7.5-90.el7  
Dependency Updated:  
    python-libs.x86_64 0:2.7.5-90.el7  
Complete!  
"}}
```

作業名(Operation)の登録

パラメータシートにデータを登録

ジョブフロー(Conductor)の実行

実行結果の確認

4.5 実行結果の確認(3/3)

ターゲットマシンの確認

- ターゲットマシンでもパッケージがインストールできていることを確認して下さい。

hostA

```
$ yum list installed httpd
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
* base: ftp-srv2.kddilabs.jp
* extras: ftp-srv2.kddilabs.jp
* updates: ftp-srv2.kddilabs.jp
Installed Packages
httpd.x86_64          2.4.6-97.el7.centos      @updates
```

作業名(Operation)の登録

パラメータシートにデータを登録

ジョブフロー(Conductor)の実行

実行結果の確認

5. 実行操作(2回目)



5.1 作業名(Operation)の登録

「オペレーション一覧」へ新規オペレーション名を登録

- ここからは1回目のオペレーション名登録以降の作業と同様になります。
「基本コンソール」メニューグループ >> 「オペレーション一覧」メニューをクリック。
各項目へ入力、選択が完了したら登録をクリックして下さい。

The screenshot shows the Exastro IT Automation basic console interface. On the left, there is a sidebar with various menu items. The 'Operation List' item is highlighted with a red box and a red number '1'. The main area displays two windows: a search window at the top and a registration window below it. In the registration window, several fields are highlighted with red boxes and numbered circles: 'Operation ID' (2), 'Operation Name' (3), and 'Implementation Date' (3). A modal dialog box is overlaid on the registration window, showing the input values: 'Operation Name' is 'Operation2' and 'Implementation Date' is '(任意の日時)' (Any Date). The overall title bar of the application is '作業名(Operation)の登録'.

Exastro IT Automation 基本コンソール

操作者に連絡

「作業名(Operation)の登録」

「パラメータシートにデータを登録」

「ジョブフロー(Conductor)の実行」

「実行結果の確認」

1
2
3
3
3

操作名
No.
オペレーションID
オペレーション名*
実施予定日時
登録

最終更新日時 最終更新者

※*は必須項目です。

オペレーション名
オペレーション2
実施予定日時
(任意の日時)

5.2 パラメータシートにデータを登録

「インストールパッケージ一覧」新規データを登録

- 「入力用」メニューグループ >> 「インストールパッケージ一覧」メニューをクリック。
各項目へ入力、選択が完了したら登録をクリックして下さい。
1回目とインストールするパッケージが異なっているので注意して下さい。

Exastro IT Automation 入力用

ようこそ[システム管理者]さん
ログインID [administrator] ログアウト

Menu

メインメニュー

インストールパッケージ一覧 1

説明

表示フィルタ

登録

※*は必須項目です。

戻る 登録 3

全件ダウンロードとファイルアップロード編集

作業名(Operation)の登録

パラメータシートにデータを登録

ジョブフロー(Conductor)の実行

実行結果の確認

ホスト名	オペレーション	httpd	MariaDB-server	php	perl	python
(機器登録で登録したホスト名)	(選択した実行予定日時)_2: オペレーション2	*	*	*	*	*

5.3 ジョブフロー(Conductor)の実行(1/3)

Conductorの実行

- 2回目の実行も行っていきます。

「Conductor」メニューグループ>>「Conductor作業実行」メニューをクリック。
実行する「Conductor」と「オペレーション」を選択し実行をクリックして下さい。

The screenshot shows the Exastro IT Automation Conductor interface. On the left, a sidebar menu lists various options, with 'Conductor作業実行' (1) highlighted by a red box. The main content area has two tabs: 'Conductor[一覧]' (2) and 'オペレーション[一覧]' (3). Both tabs have their own filter and list components. A large red box highlights the 'Conductor[一覧]' tab's list, which contains one item: '1 パッケージインストール'. Below it, another list for 'オペレーション[一覧]' shows two items: '21 オペレーション1' and '22 オペレーション2'. At the bottom right of the main area, a large orange button labeled 'Conductor実行' (4) is also highlighted with a red box. To the right of the main interface, there is a vertical sidebar with five colored boxes: '作業名(Operation)の登録', 'パラメータシートにデータを登録', 'ジョブフロー(Conductor)の実行' (highlighted with a red border), and '実行結果の確認'.

Exastro IT Automation Conductor

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

Menu

- メインメニュー
- Conductorインターフェース情報
- Conductor通知先定義
- Conductorクラスタ一覧
- Conductorクラス編集
- Conductor作業実行 (1)
- Conductor作業確認
- Conductor作業一覧
- Conductor定期作業実行

説明

△開く

スケジューリング

△開く

Conductor[フィルタ]

△開く

Conductor[一覧] (2)

△閉じる

選択 ConductorクラスID □ Conductor名称 □ 説明 □ アクセス権 備考 □ 最終更新日時 □ 最終更新者 □

選択	ConductorクラスID	Conductor名称	説明	アクセス権	備考	最終更新日時	最終更新者
<input checked="" type="radio"/>	1	パッケージインストール		アクセス許可ロール		2021/12/17 11:34:47	システム管理者

フィルタ結果件数: 1

オペレーション[フィルタ]

オペレーション[一覧] (3)

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 □ アクセス権

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	アクセス権
<input type="radio"/>	21	21	オペレーション1	2021/12/16 12:35	2021/12/17 14:32	アクセス許可ロール
<input checked="" type="radio"/>	22	22	オペレーション2	2021/12/16 13:25	2021/12/17 14:44	アクセス許可ロール

フィルタ結果件数: 2

Conductor実行

作業名(Operation)の登録

パラメータシートにデータを登録

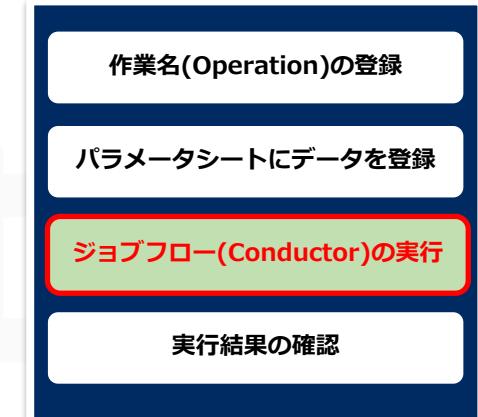
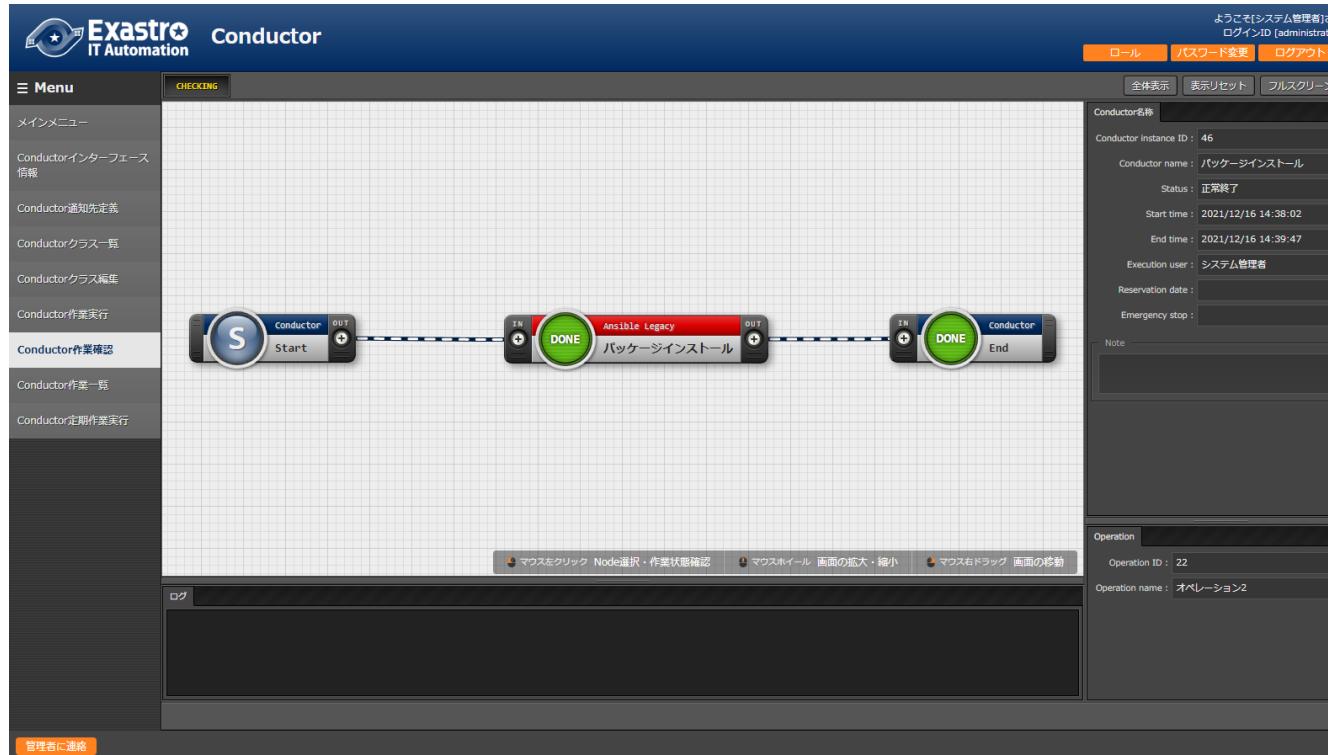
ジョブフロー(Conductor)の実行

実行結果の確認

5.3 ジョブフロー(Conductor)の実行(2/3)

作業結果確認

- 実行すると「Conductor作業確認」メニュー画面に切替わり、実行ステータスやログが表示されます。



Point

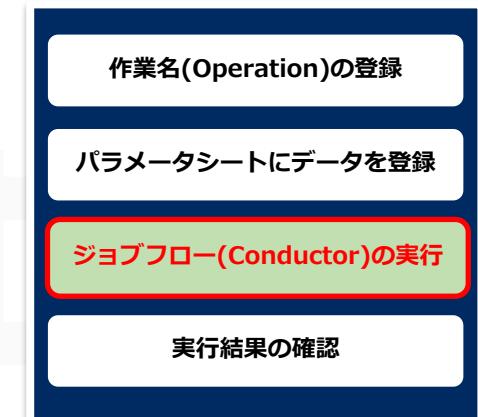
実行ステータスやログを
リアルタイムで
確認可能です。

5.3 ジョブフロー(Conductor)の実行(3/3)

作業結果確認

- ジョブ(Movement)を選択し、Doneのアイコンまたは右側のOperation statusをクリックすると詳細が表示されます。

The screenshot shows the Exastro Conductor interface. On the left, there's a navigation menu with items like 'Menu', 'Checking', 'Role', 'Password Change', and 'Logout'. The main area displays a workflow diagram with nodes: 'Start' (S), 'Ansible Legacy' (labeled 'パッケージインストール'), and 'End'. A red callout bubble labeled '1 選択' points to the 'Ansible Legacy' node. To the right, a detailed view of the 'Ansible Legacy' node is shown, including its Node type (movement), Node instance ID (192), Node name (node-3), Status (正常終了), Start time (2021/12/16 14:38:02), End time (2021/12/16 14:39:44), and Operation status (手動実行確認). Below this, there's a 'Specified individual operation' section with fields for Operation ID and Operation name. At the bottom of the interface, there's a 'Logs' section and a '管理者に連絡' button.



5.4 実行結果の確認(1/2)

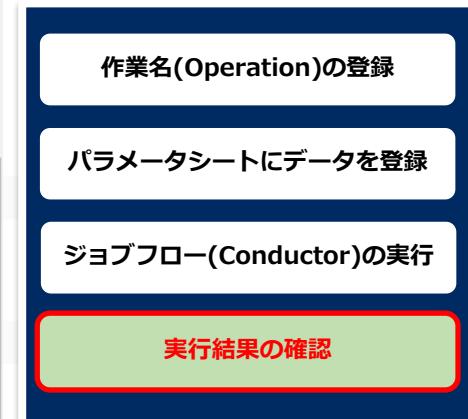
作業結果確認

- 詳細画面の進行状況(実行ログ)でAnsibleの実行ログを確認します。

The screenshot shows the Ansible execution log details. On the left, there's a sidebar with various menu items. In the center, a table displays input and output data, and a log viewer window is open. The log viewer has two tabs: '進行状況(実行ログ)' and '進行状況(エラーログ)'. The '進行状況(実行ログ)' tab is selected and contains the following log output:

```
        "update_cache": false,
        "update_only": false,
        "use_backed": "auto",
        "validate_certs": true
    },
    "item": "python",
    "msg": "",
    "rc": 0,
    "results": [
        "results": [
            "All packages providing python are up to date",
            ""
        ]
    ]
}
META: ran handlers
META: ran handlers
PLAY RECAP ****hostA : ok=1    changed=0      unreachable=0      failed=0      skipped=0      rescued=0      ignored=0
```

The '進行状況(実行ログ)' tab is highlighted with a red box.



5.4 実行結果の確認(2/2)

実行ログの確認

- 新たにMariaDBのインストールと他のパッケージとの依存関係の解決、他の4つのパッケージ(httpd,php,perl,python)のバージョンアップが行われていることを確認して下さい。

進行状況(実行ログ)の一部の例

```
~~~~~省略~~~~~  
"results": ["All packages providing httpd are up to date",  
    ""]}  
~~~~~省略~~~~~  
Installed:  
    MariaDB-server.x86_64 0:10.8.4-1.el7.centos  
Dependency Installed:  
    perl-Compress-Raw-Bzip2.x86_64 0:2.061-3.el7  
    perl-Compress-Raw-Zlib.x86_64 1:2.061-4.el7  
    perl-DBD-MySQL.x86_64 0:4.023-6.el7  
    perl-DBI.x86_64 0:1.627-4.el7  
    perl-IO-Compress.noarch 0:2.061-2.el7  
    perl-Net-Daemon.noarch 0:0.48-5.el7  
    perl-PIRPC.noarch 0:0.2020-14.el7  
Complete!  
"]}  
~~~~~省略~~~~~  
"results": ["All packages providing php are up to date",  
    ""]}  
~~~~~省略~~~~~  
"results": ["All packages providing perl are up to date",  
    ""]}  
~~~~~省略~~~~~  
"results": ["All packages providing python are up to date",  
    ""]}
```

作業名(Operation)の登録

パラメータシートにデータを登録

ジョブフロー(Conductor)の実行

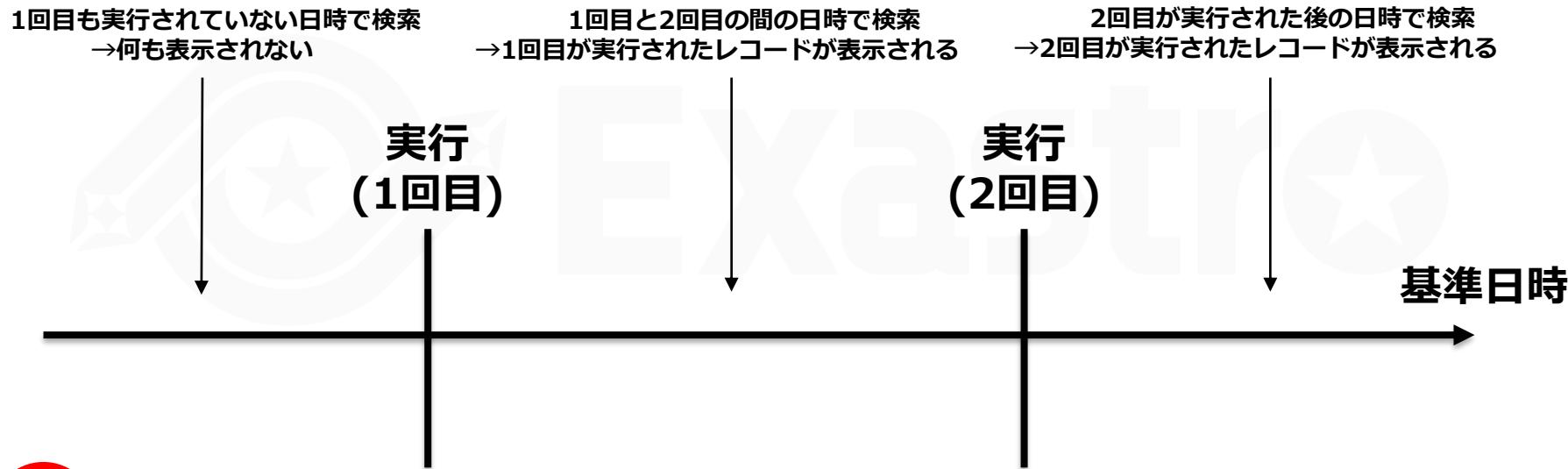
実行結果の確認

6. CMDBパラメータの履歴確認

6.1 作業実行と履歴管理

履歴管理と本シナリオのポイント

- ITAはCMDBに「誰が・いつ・何をしたのか？」を履歴管理し、その時の時点でシステムのパラメータはどうなっているのかを抽出できる機能があります。パラメータの履歴管理することにより、設計者や運用者がストレスなくシステム更改を行うことができます。



Point

今回のクイックスタートではパラメータの履歴管理を体験していただくために、実行を2回行いました。

6.2 CMDBパラメータの履歴を確認する(1/3)

履歴確認

- 実際にパラメータが管理できているかどうか確認をしていきます。
「参照用」メニューグループ >> 「インストールパッケージ一覧」メニューをクリック。
まずは基準日付を入力せずにフィルタをかけます。(次ページへ続く)

The screenshot shows the Exastro IT Automation interface. The top navigation bar includes the Exastro logo, user information (ログインID [administrator]), and links for ロール (Role), パスワード変更 (Change Password), and ログアウト (Logout). The main menu on the left has items like メインメニュー (Main Menu), インストールパッケージ一覧 (Install Package List), and so on. The current page is titled '説明' (Description) under the 'Install Package List' section. A search panel is open, with step 1 pointing to the 'インストールパッケージ一覧' (Install Package List) menu item. Step 2 points to the 'フィルタ' (Filter) button in the search bar. Step 3 points to the 'フィルタ' (Filter) button in the search panel. The main content area shows a table titled '一覧' (List) with columns for 製品 (Product), No (No), ホスト名 (Host Name), ID (ID), オペレーション名 (Operation Name), 基準日時 (Reference Date/Time), 実施予定日時 (Planned Execution Date/Time), 最終実行日時 (Last Execution Date/Time), オペレーション (Operation), パラメータ (Parameters), アクセス (Access), 最終更新日時 (Last Update Date/Time), and 最終更新者 (Last Update User). A single row is highlighted with a red box, showing hostA, operation 22, and various parameter values. Step 4 points to this row, indicating that the latest data is displayed.

製品	No	ホスト名	ID	オペレーション名	基準日時	実施予定日時	最終実行日時	オペレーション	パラメータ	アクセス	最終更新日時	最終更新者		
履歴	2	hostA	22	オペレーション2	2021/12/17 14:44	2021/12/16 13:25	2021/12/17 14:44	*	httpd MariaDB-server php perl python	*	*	*	2021/12/17 14:44:28	システム管理者

6.2 CMDBパラメータの履歴を確認する(2/3)

履歴確認

- 次に2回目の実行を行った基準日時より前の日付を入力してフィルタをかけます。
(次ページへ続く)

The screenshot shows the Exastro IT Automation interface with the following steps highlighted:

- 1** In the "Filter" dialog, the "Baseline Date" field is set to "2021/12/17 14:35". A red box and callout indicate to input the date before the second execution and click the filter button.
- 2** The "Filter" button is highlighted with a red box.
- 3** The results table shows one item: "1 hostA 21 オペレーション 1 2021/12/17 14:32 2021/12/16 12:35 2021/12/17 14:32 * * * * 2021/12/16 13:08:52 システム管理者". A red box and callout indicate to confirm the execution result from the first run is displayed.

履歴	No	ホスト名	ID	オペレーション名	基準日時	実施予定日時	最終実行日時	オペレーション	パラメータ	アクセス	最終更新日時	最終更新者
履歴	1	hostA	21	オペレーション 1	2021/12/17 14:32	2021/12/16 12:35	2021/12/17 14:32	*	*	*	2021/12/16 13:08:52	システム管理者

6.2 CMDBパラメータの履歴を確認する(3/3)

履歴確認

- 最後に1回目の実行を行った基準日時より前の日付を入力してフィルタをかけます。

The screenshot shows the Exastro IT Automation interface with the following steps highlighted:

- 1**: A red circle highlights the "基準日時" (Baseline Date) input field in the "表示フィルタ" (Display Filter) panel. A callout box with a red border contains the text: "1回目の実行を行った日時より前の日時を入力し フィルタをクリック".
- 2**: A red circle highlights the "フィルタ" (Filter) button in the "表示フィルタ" panel.
- 3**: A red circle highlights the "変更履歴" (Change History) button in the main content area. A callout box with a red border contains the text: "何も表示されないことを確認する".

The interface includes a top navigation bar with the Exastro logo, user information (ようこそ[システム管理者]さん), and links for ロール (Role), パスワード変更 (Change Password), and ログアウト (Logout). The left sidebar has a menu with "Menu", "メインメニュー" (Main Menu), and "インストールパッケージ一覧" (List of Installed Packages).

A 付録



参考① 【Ansible-Legacy】 単体実行

作業実行

- Ansible-Legacyは「作業実行」メニューがあり Movementごとに個別実行や、ドライランが可能です。

The screenshot shows the Ansible-Legacy application interface. On the left, there's a sidebar with various menu items like 'Menu', 'Movement一覧', 'Playbook(素材集)', etc. The main area has three tabs: '説明' (Description), 'スケジューリング' (Scheduling), and 'Movement[一覧]' (List of Movements). The 'Movement[一覧]' tab is active, displaying a table of movements. A red box highlights the first row of the table, which corresponds to the movement selected in step 1. A red arrow points from this table to the 'Movement[一覧]' tab in the second section. The second section shows a table of operations (オペレーション) with two rows. A red box highlights the first operation, corresponding to the one selected in step 2. A red arrow points from this table to the 'オペレーション[一覧]' tab in the third section. The third section shows a detailed view of the selected movement and operation. It includes a 'ドライラン' (Dry Run) button and an '実行' (Execute) button. Red circles numbered 1, 2, and 3 point to the highlighted areas in each section, with corresponding text descriptions.

1 作成済みのMovementを選択

2 Movementと紐づいたオペレーションを選択

3 ドライラン
: プレイブックの接続確認/構文チェック
実行
: プレイブックを実行

参考② 【Ansible-Legacy】 実行確認

作業結果確認

- 実行(またはドライラン)すると画面が切替わり、実行ステータスやログが表示されます。

Point

実行ステータスや、
投入データが確認可能です。

Point

実行ログやエラーログを
リアルタイムで確認可能です。

Point

投入データや結果データを
ダウンロードすることも可能です。

The screenshot shows the Ansible-Legacy execution details. Key sections include:

- Point (Input/Output Data):** Shows input data as "InputData_0000000116.zip" and output data as "ResultData_0000000116.zip".
- Point (Logs/Error Logs):** Displays the execution log and error log. The log shows tasks like "Setting" and "Verifying" for multiple hosts, with status counts like 2/4, 1/4, etc. The error log is empty.
- Point (Download Options):** Buttons for "進行状況(実行ログ)" (Execution Log) and "進行状況(エラーログ)" (Error Log) allow users to download the respective logs.



Exastro