利用手順マニュアル

astrollAnsible driver

*－*第1.0版*－*

Copyright © NEC Corporation 2019. All rights reserved.

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写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。

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おりません。

astrollの正式名称は「astroll IT Automation」になります。

目次

[はじめに 4](#_Toc6405859)

[1 Ansible driver概要 6](#_Toc6405860)

[1.1 Ansibleについて 6](#_Toc6405861)

[1.2 Ansible driverについて 7](#_Toc6405862)

[2 Ansible driverでの変数取り扱い 8](#_Toc6405863)

[2.1 変数の種類 8](#_Toc6405864)

[2.2 変数の抜出および具体値登録 8](#_Toc6405865)

[2.3 代入値登録による変数の扱い 8](#_Toc6405866)

[3 Ansible driver コンソールメニュー構成 9](#_Toc6405867)

[3.1 メニュー/画面一覧 9](#_Toc6405868)

[4 Ansible driver利用手順 11](#_Toc6405869)

[4.1 作業フロー 11](#_Toc6405870)

[4.1.1 Ansible-Legacy作業フロー 12](#_Toc6405871)

[4.1.2 Ansible-Legacy Role作業フロー 14](#_Toc6405872)

[4.1.3 Ansible-Pioneer作業フロー 16](#_Toc6405873)

[5 Ansible driver機能・操作方法説明 19](#_Toc6405874)

[5.1 基本コンソール 19](#_Toc6405875)

[5.1.1 機器一覧 19](#_Toc6405876)

[5.1.2 紐付対象メニュー 22](#_Toc6405877)

[5.1.3 投入オペレーション一覧 23](#_Toc6405878)

[5.2 Ansible 共通コンソール 24](#_Toc6405879)

[5.2.1 インタフェース情報 24](#_Toc6405880)

[5.2.2 グローバル変数管理 26](#_Toc6405881)

[5.3 Ansibel-Legacy／Legacy Role／Pioneer　コンソール 28](#_Toc6405882)

[5.3.1 Movement一覧 28](#_Toc6405883)

[5.3.2 プレイブック素材集（Ansible-Legacyのみ） 29](#_Toc6405884)

[5.3.3 ロールパッケージ管理（Ansible-Legacy Roleのみ） 31](#_Toc6405885)

[5.3.4 対話種別リスト（Ansible-Pioneerのみ） 32](#_Toc6405886)

[5.3.5 対話ファイル素材集（Ansible-Pioneerのみ） 33](#_Toc6405887)

[5.3.6 Movement詳細 35](#_Toc6405888)

[5.3.7 テンプレート管理（Ansible-Legacy/Ansible-Pioneer） 37](#_Toc6405889)

[5.3.8 ファイル管理 38](#_Toc6405890)

[5.3.9 多段変数最大繰返数管理（Ansible-Legacy Roleのみ） 39](#_Toc6405891)

[5.3.10 代入値自動登録設定 40](#_Toc6405892)

[5.3.11 作業対象ホスト 41](#_Toc6405893)

[5.3.12 代入値管理 42](#_Toc6405894)

[5.3.13 作業状態確認 44](#_Toc6405895)

[5.3.14 作業管理 46](#_Toc6405896)

[5.3.15 作業実行 47](#_Toc6405897)

[6 構築コード記述方法 48](#_Toc6405898)

[6.1 Playbook（Ansible-Legacy）の記述 48](#_Toc6405899)

[6.2 対話ファイル（Ansible-Pioneer）の記述 48](#_Toc6405900)

[6.3 ロールパッケージ（Ansible-Legacy Role）の記述 58](#_Toc6405901)

[6.4 astrollreadme（Ansible-Legacy Roleのみ）の記述 58](#_Toc6405902)

[6.5 読替表（Ansible-Legacy Roleのみ）の記述 58](#_Toc6405903)

[6.6 BackYardコンテンツ 58](#_Toc6405904)

[6.7 Ansible利用ガイドラインastroll追加ルール 58](#_Toc6405905)

[7 運用上の注意点 59](#_Toc6405906)

[7.1 ログレベルの変更 59](#_Toc6405907)

[7.2 起動周期の変更 61](#_Toc6405908)

[7.3 メンテナンス方法について 63](#_Toc6405909)

[7.3.1 Ansible driver 独立型プロセスの起動/停止/再起動 63](#_Toc6405910)

[8 トラブルシューティング 64](#_Toc6405911)

[9 付録 65](#_Toc6405912)

[9.1 Ansible実行時に使用される投入データとastrollメニューの紐づけ 65](#_Toc6405913)

[9.1.1 Ansible-Legacy投入データ 65](#_Toc6405914)

[9.1.2 Ansible-Pioneer投入データ 67](#_Toc6405915)

[9.1.3 Ansible-LegacyRole投入データ 69](#_Toc6405916)

はじめに

本書では、astrollの機能および操作方法について説明します。

**関連マニュアル**

* 本製品におけるマニュアル構成は以下になります。

Ansible driverを初めて利用される方は、まずファーストステップガイドで全体の操作の流れを把握し、

利用手順マニュアル（astroll基本コンソール）で共通画面、機器情報登録画面などの操作方法を参照して

ください。

また、構築作業を定義するプレイブックは、作成において記述規定がありますので、「Ansible利用ガイド

ライン」を参照ください。

| **No.** | **マニュアル名** | **説明** |
| --- | --- | --- |
| 1 | インストールマニュアル  astroll | astrollのシステム構成、動作環境と、環境構築、インストール手順の概要を説明したドキュメントです。 |
| 2 | RHEL6.x\_環境構築マニュアル | インストールマニュアルの別冊資料です。 RHEL 6.x環境への環境構築、インストール手順を説明します。 |
| 3 | RHEL7.x\_環境構築マニュアル | インストールマニュアルの別冊資料です。  RHEL 7.x環境への環境構築、インストール手順を説明します。 |
| 4 | ファーストステップガイド | astrollの全体の概要、機能、操作の概要について説明したドキュメントです。astrollを初めて利用される方は、本ドキュメントで全体の操作の流れを参照してください。 |
| 5 | 利用手順マニュアル  astroll基本コンソール | astrollの基本機能である基本コンソールの機能、操作方法について説明したドキュメントです。機器情報の登録、ワークフローの作成、実行などについて説明しています。各Driverをご利用の方は、本ドキュメントも合わせて参照してください。 |
| 6 | 利用手順マニュアル  astroll管理コンソール | astrollの管理機能である管理コンソールの機能、操作方法について説明したドキュメントです。 ユーザー管理、権限管理、astrollシステムの設定機能などを説明します。 |
| 7 | 【本書】  利用手順マニュアル  astrollAnsible driver | Ansible driver の機能、操作方法について説明したドキュメントです。 |
| 8 | 利用手順マニュアル  astrollAnsible driver  別紙　Ansible利用ガイドライン  astroll追加ルール | Ansible driver利用手順マニュアルの補足資料です。 astrollからAnsibleを利用するときの注意、制限事項と、エラーメッセージとその対処方法などについて説明します。 |
| 10 | 利用手順マニュアル  astrollCobbler driver | Cobbler driver の機能、操作方法について説明したドキュメントです。 |
| 11 | 構成管理メニュー作成ガイド | 独自の構成管理画面を作成し、astrollのメニューに追加する手順を説明したドキュメントです。 |
| 12 | 利用手順マニュアル astroll\_Ansilble系ドライバー共通 | Ansible系統のドライバーにおける共通機能および操作方法について説明したドキュメントです。 |

# Ansible driver概要

本章ではAnsibleおよびAnsible driverについて説明します。

## Ansibleについて

Ansibleとは、多数の構築管理対象に対して、アプリーケーション/システムのデプロイ作業を容易にするPF構築自動化ツールです。

Ansibleは、PlaybookというYAML形式のテキストファイルに定型処理をタスクとして記述し、それをAnsibleに実行させることにより、さまざまな処理を実現できます。

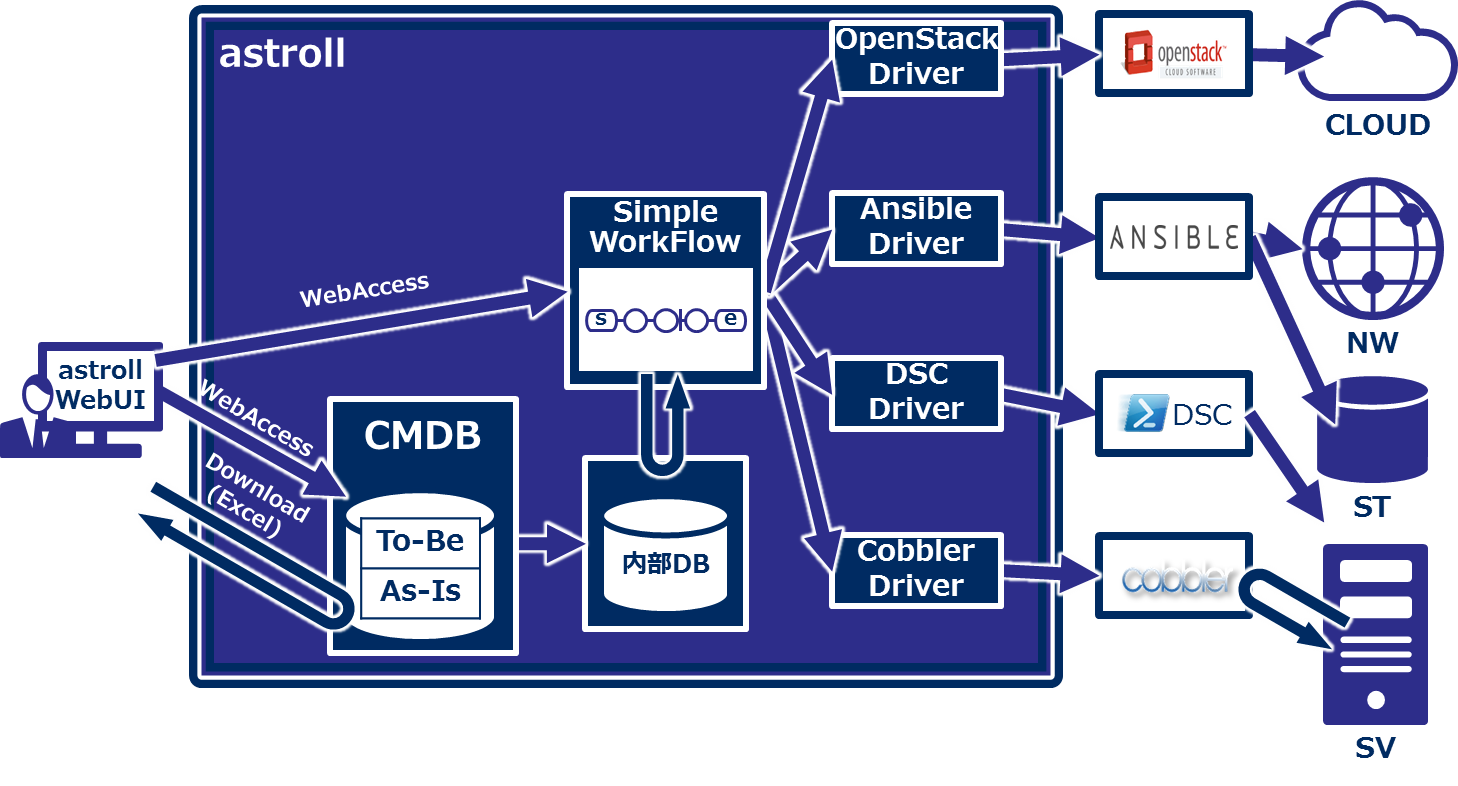
タスクはモジュールと呼ばれる処理プログラムと紐付いており、さまざまな機器に対する制御を行うことができます。

Ansibleの詳細情報については、Ansibleの製品マニュアルを参照してください。

Ansibleのバージョンは2.2.1.0 となります。 最新の Ansible に対応した記法は使えないことがありますので、注意してください。

## Ansible driverについて

Ansible driverは、astrollシステムのオプションとして機能し、astrollシステムで登録した構築対象のサーバ・ストレージ・ネットワーク各機器に対し、Ansibleを用いて実際の運用設定を自動的に行います。



Ansible driverには用途に応じて以下3つのモードを用意しています。

1. **Legacy モード**  
   Ansible標準の機能を用いて各種ホストへ設定を投入します。  
   構築コードを単体YAMLファイルとして登録し、作業パターンをその組み合わせで構成します。  
   OS,NWの環境設定などの作業用に使われることを想定します。
2. **Legacy Role モード**  
   Legacyモードと同じく、Ansible標準の機能を用いて各種ホストへ設定を投入します。  
   構築コードをパッケージとして登録し、作業パターンをRoleの組み合わせで構成します。  
   製品部門などが提供するRoleパッケージを用いて、製品のインストール、環境構築などを行う際に使われることを想定します。
3. **Pioneer モード**  
   Ansibleに独自モジュールを追加し、対話形式による設定投入を可能とします。  
   サーバ、ストレージ、ネットワークを問わず、Telnet, SSH でログイン可能なあらゆる機器に対応しています。対象機器と直接やり取りが必要となるため、相応のＩＴスキルが必要となります。

また、Ansible driverは、Playbook中の変数を画面から設定することができます。詳細は本書「2Ansible driverでの変数取り扱い」をご参照ください。

# Ansible driverでの変数取り扱い

## 変数の種類

Ansible driverでは、Playbook中の変数をastrollの設定画面から指定することができます。

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

## 変数の抜出および具体値登録

各モードとも、astrollにアップロードしたPlaybook等の資材から変数を抜き出します。  
詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

変数の抜出方法は以下のとおりです。

1. **Ansible-Legacy**  
   「プレイブック素材集 (本書：5.3.2プレイブック素材集（Ansible-Legacyのみ）)」でアップロードしたPlaybookより、以下の書式の変数定義を抜出します。

{{△VAR\_xxx△}}　または　{%△VAR\_xxx△%}

※ △:半角スペース　xxx: 半角英数字とアンダースコア（ \_ ）

Ansible-Legacyでは具体値の登録の仕方で通常変数か複数具体値変数かを決定します。詳しくは「5.3.12代入値管理」を参照してください。

1. **Ansible-Pioneer**  
   「対話ファイル素材 (本書：5.3.5対話ファイル素材集（Ansible-Pioneerのみ）」でアップロードした対話ファイルより、Ansible-Legacyと同様の抜出を行います。
2. **Ansible-Legacy Role**  
   「ロールパッケージ管理 (本書：5.3.3ロールパッケージ管理（Ansible-Legacy Roleのみ）」でアップロードしたロールパッケージ内のPlaybookより、Ansible-Legacyと同様の抜出を行います。  
   詳しくは「ロールパッケージの記述(本書：5.3.3ロールパッケージ管理（Ansible-Legacy Roleのみ）」を参照してください。

また、読替表を作成することでdefaults変数定義ファイルまたはastrollreadmeに定義されている「VAR\_xxx」以外の変数をastrollで扱うことが出来ます。詳しくは「[6.5 読替表の記述](#_読替表（Ansible_Legacy_Roleのみ）の記述)」を参照して下さい。

## 代入値登録による変数の扱い

Playbookで定義した変数の値は代入値登録機能により上書きすることができます。

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

# Ansible driver コンソールメニュー構成

本章では、astrollコンソールのメニュー構成について説明します

なお、Webコンソールへのログイン方法、およびメニュー画面の構成要素/基本的な操作については、  
「利用手順マニュアル(astrollシステム)」を参照してください。

## メニュー/画面一覧

1. **astroll基本コンソールのメニュー**

Ansible driverで利用するastroll基本コンソールのメニュー一覧を以下に記述します。

表 3.1‑1基本コンソール メニュー/画面一覧

|  |  |  |  |
| --- | --- | --- | --- |
| No | メニューグループ | メニュー・画面 | 説明 |
| 1 | astroll 基本コンソール | 機器一覧 | 作業対象システム一覧をメンテナンス(閲覧/登録/更新/廃止)します |
| 2 | 紐付対象メニュー | 代入値自動登録設定と連携するCMDBを管理します |
| 3 | 投入オペレーション  一覧 | 投入オペレーション一覧をメンテナンス(閲覧/登録/更新/廃止)できます |

1. **Ansible共通コンソールのメニュー**

Ansible共通コンソールのメニュー一覧を以下に記述します。

表 3.1‑2共通コンソール メニュー/画面一覧

|  |  |  |  |
| --- | --- | --- | --- |
| No | メニューグループ | メニュー・画面 | 説明 |
| 1 | Ansible  共通コンソール | インタフェース情報 | astrollシステム・Ansible driverサーバとAnsibleサーバが共有するディレクトリのパスおよびAnsibleサーバへの接続インタフェース情報を管理します |
| 2 | グローバル変数管理 | Playbookや対話ファイルなどで共通利用する変数（以降、グローバル変数と称す）と具体値を管理します |

1. **Ansibleコンソールのメニュー**

各Ansibleコンソールに対応するメニュー一覧を以下に記述します。

表 3.1‑3 Ansible driverコンソール メニュー/画面一覧

| No | メニューグループ | | | メニュー・画面 | 説明 |
| --- | --- | --- | --- | --- | --- |
| **Ansible コンソール** | | |
| Legacy | Legacy Role | Pioneer |
| 1 | **○** | **○** | **○** | Movement一覧 | Symphonyに登録するMovementの一覧を管理します |
| 2 | **○** |  |  | プレイブック素材集 | Playbookファイルを管理します |
| 3 |  | **○** |  | ロールパッケージ管理 | ロールパッケージを管理します |
| 4 |  |  | **○** | 対話種別リスト | 同一目的の対話ファイルを対話種別としてまとめる種別を管理します |
| 5 |  |  | **○** | 対話ファイル素材集 | 対話種別に紐づけるOS種別とastrollシステム独自フォーマットの作業手順ファイル(以降、対話ファイルと称す。)を管理します |
| 6 | **○** | **○** | **○** | Movement詳細 | Movementとプレイブック素材の関連付けを管理します |
| 7 | **○** |  | **○** | テンプレート管理 | Playbook内のtemplateモジュールで使用するテンプレートファイルと埋め込み変数を管理します |
| 8 | **○** | **○** | **○** | ファイル管理 | Playbook内の各モジュールで使用する素材ファイルと埋め込み変数を管理します |
| 9 |  | **○** |  | 多段変数最大繰返数管理 | 多段変数が繰返配列で構成されている場合の最大繰返配列数を管理します |
| 10 | **○** | **○** | **○** | 代入値自動登録設定 | CMDBのメニューに登録されているオぺレーションとホスト毎の項目値を紐付けるMovementと変数を管理します |
| 11 | **○** | **○** | **○** | 作業対象ホスト | Movementで使用するホストを管理します |
| 12 | **○** | **○** | **○** | 代入値管理 | 変数の代入値を管理します |
| 13 | **○** | **○** | **○** | 作業実行 | 作業実行するMovementとオペレーションを選択し実行を指示します |
| 14 | **○** | **○** | **○** | 作業状態確認 | 作業実行状態を表示します |
| 15 | **○** | **○** | **○** | 作業管理 | 作業実行履歴を管理します |

※ Ansible RestAPIは、設定等ユーザー操作を行わないため、ここでの説明は省略します。

# Ansible driver利用手順

各Ansibleコンソールの利用手順について説明します

## 作業フロー

各Ansibleコンソール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。

astroll基本コンソールの利用方法は、「利用手順マニュアル(astroll\_基本コンソール)」を参照してください。

### Ansible-Legacy作業フロー

以下は、Ansible-Legacyで作業を実行するまでの流れです。

**①　機器情報にAnsible利用情報を設定**

**③ 作業パターン(Movement)の登録**

**④ プレイブックの登録**

**⑧ 作業対象ホストの指定 ※**

**⑨ 変数値の設定**

**⑩ 作業実行**

**⑪ 実行状態確認**

**⑥ 素材ファイルの登録**

**⑤ テンプレートファイルの登録**

**⑫ 作業履歴確認**

**※一度Movementを設定してあれば、次回より**

**本手順以降から始めることができます。**

**必須タスク**

**任意タスク**

**【凡例】**

**⑦　Movementにプレイブック素材を指定**

**②　投入オペレーション名の登録**

* **作業フロー詳細と参照先**

1. **機器情報にAnsible利用情報を設定**astroll基本コンソールの機器一覧の画面から、各機器に対してAnsible利用情報を設定します。  
   詳細は 5.1.1機器一覧 を参照してください。
2. **投入オペレーション名の登録**

astroll基本コンソールの投入オペレーション一覧の画面から、作業用の投入オペレーション名を登録します。

詳細は5.1.3投入オペレーション一覧 を参照してください。

1. **作業パターン(Movement)の登録**Ansible-LegacyコンソールのMovement一覧の画面から、作業用のMovementを登録します。  
   詳細は 5.3.1Movement一覧 を参照してください。
2. **Playbookの登録**Ansible-Legacyコンソールのプレイブック素材集の画面から、作業で使用するPlaybookを登録します。  
   詳細は 5.3.2プレイブック素材集（Ansible-Legacyのみ） を参照してください。
3. **テンプレートファイルの登録（必要に応じて実施）**

Ansible-Legacyコンソールのテンプレート管理の画面から、Playbook内で定義しているtemplateモジュールのtemplateファイル(src)とtemplate埋め込み変数の登録／更新／廃止を行います。

詳細は5.3.7テンプレート管理（Ansible-Legacy/Ansible-Pioneer） を参照してください。

1. **素材ファイルの登録 （必要に応じて実施）**Ansible-Legacyコンソールのファイル管理の画面から、作業対象サーバに配置するファイルを登録します。

詳細は 5.3.8ファイル管理 を参照してください。

1. **Movementにプレイブック素材を指定**Ansible-LegacyコンソールのMovement詳細の画面から、登録したMovementにプレイブック素材を指定します。

詳細は 5.3.6Movement詳細 を参照してください。

1. **作業対象ホストの指定**Ansible-Legacyコンソールの作業対象ホストの画面から、作業対象ホストを指定します。  
   詳細は 5.3.11作業対象ホスト を参照してください。
2. **変数値の設定（必要に応じて実施）**Ansible-Legacyコンソールの代入値管理の画面から、Movementに登録したPlaybook内で定義した変数の値を設定します。変数を利用していない場合、設定は不要です。  
   詳細は 5.3.12代入値管理 を参照してください。
3. **作業実行**Ansible-Legacyコンソールの作業実行の画面から、実行日時、投入オペレーションを選択して設定して処理の実行を指示します。

詳細は 5.3.15作業実行 を参照してください。

1. **作業状態確認**Ansible-Legacyコンソールの作業状態確認の画面では、実行した作業の状態がリアルタイムで表示されます。また、作業の緊急停止や、実行ログ、エラーログを監視することができます。  
   詳細は 5.3.13作業状態確認 を参照してください。
2. **作業履歴確認**Ansible-Legacyコンソールの作業管理の画面では、実行した作業の一覧が表示され履歴が確認できます。

詳細は 5.3.14作業管理 を参照してください。

### Ansible-Legacy Role作業フロー

以下は、Ansible-Legacy Roleで作業を実行するまでの流れです。

**① 機器情報にAnsible利用情報を設定**

**③ 作業パターン(Movement)の登録**

**④ ロールパッケージの登録**

**⑧ 作業対象ホストの指定**

**⑩ 作業実行**

**⑪ 実行状態確認**

**⑥ Movementにロールパッケージを指定**

**⑫ 実行履歴確認**

**必須タスク**

**任意タスク**

**【凡例】**

**※一度Movementを設定してあれば、次回より**

**本手順以降から始めることができます。**

**⑨ 変数値の設定**

**⑦ 多段変数の最大繰返数を指定 ※**

**② 投入オペレーション名の登録**

**⑤ 素材ファイルの登録**

* **作業フロー詳細と参照先**

1. **機器情報にAnsible利用情報を設定**astroll基本コンソールの機器一覧の画面から、各機器に対してAnsible利用情報を設定します。  
   詳細は 5.1.1機器一覧 を参照してください。
2. **投入オペレーション名の登録**

astroll基本コンソールの投入オペレーション一覧の画面から、作業用の投入オペレーション名を登録します。

詳細は 5.1.3投入オペレーション一覧 を参照してください。

1. **作業パターン(Movement)の登録**Ansible-Legacy RoleコンソールのMovement一覧の画面から、作業用のMovementを登録します。  
   詳細は 5.3.1Movement一覧 を参照してください。
2. **ロールパッケージの登録**Ansible-Legacy Roleコンソールのロールパッケージ管理の画面から、作業で使用するロールパッケージを登録します。  
   詳細は 5.3.3ロールパッケージ管理（Ansible-Legacy Roleのみ） を参照してください。
3. **素材ファイルの登録 （必要に応じて実施）**

Ansible-Legacy Roleコンソールのファイル管理の画面から、作業対象サーバに配置するファイルを登録します。

詳細は 5.3.8ファイル管理 を参照してください。

1. **Movementにロールパッケージを指定**

Ansible-Legacy RoleコンソールのMovement詳細の画面から、登録したMovementにプレイブック素材を指定します。

詳細は 5.3.6Movement詳細 を参照してください。

1. **多段変数の最大繰返数を指定**Ansible-Legacy Roleコンソールの多段変数最大繰返管理の画面から、多段変数で配列定義されているメンバー変数の配列の最大繰返数を指定します。

詳細は 5.3.9多段変数最大繰返数管理（Ansible-Legacy Roleのみ） を参照してください。

1. **作業対象ホストの指定**Ansible-Legacy Roleコンソールの作業対象ホストの画面から、作業対象ホストを指定します。  
   詳細は 5.3.11作業対象ホスト を参照してください。
2. **変数値の設定**Ansible-Legacy Roleコンソールの代入値管理の画面から、Movementに登録したPlaybook内で定義した変数の値を設定します。変数を利用していない場合、設定は不要です。  
   詳細は 5.3.12代入値管理 を参照してください。
3. **作業実行**Ansible-Legacy Roleコンソールの作業実行の画面から、実行日時、投入オペレーションを選択して設定して処理の実行を指示します。

詳細は 5.3.15作業実行 を参照してください。

1. **作業状態確認**Ansible-Legacy Roleコンソールの作業状態確認の画面から、実行した作業の状態がリアルタイムで表示されます。また、作業の緊急停止や、実行ログ、エラーログを監視することができます。  
   詳細は 5.3.13作業状態確認 を参照してください。
2. **作業履歴確認**Ansible-Legacy Roleコンソールの作業管理の画面から、実行した作業の一覧が表示され履歴が確認できます。

詳細は 5.3.14作業管理 を参照してください。

### Ansible-Pioneer作業フロー

以下は、Ansible-Pioneerで作業を実行するまでの流れです。

**①　機器情報にAnsible利用情報を設定**

**③ 作業パターン(Movement)の登録**

**⑤ 対話ファイルの登録**

**⑨ 作業対象ホストの指定 ※**

**⑩ 変数値の設定**

**⑪ 作業実行**

**⑫ 実行状態確認**

**⑦ 素材ファイルの登録**

**⑥ テンプレートファイルの登録**

**⑬ 作業履歴確認**

**※一度Movementを設定してあれば、次回より**

**本手順以降から始めることができます。**

**必須タスク**

**任意タスク**

**【凡例】**

**⑧　Movementに対話ファイルを指定**

**②　投入オペレーション名の登録**

**④ 対話種別の登録**

* **作業フロー詳細と参照先**

1. **機器情報にAnsible利用情報を設定**astroll基本コンソールの機器一覧の画面から、各機器に対してAnsible利用情報を設定します。  
   詳細は 5.1.1機器一覧 を参照してください。
2. **投入オペレーション名の登録**

astroll基本コンソールの投入オペレーション一覧の画面から、作業用の投入オペレーション名を登録します。

詳細は 5.1.3投入オペレーション一覧 を参照してください。

1. **作業パターン(Movement)の登録**Ansible-PioneerコンソールのMovement一覧の画面から、作業用のMovementを登録します。  
   詳細は 5.3.1Movement一覧 を参照してください。
2. **対話種別の登録**Ansible-Pioneerコンソールの対話種別リストの画面から、対話種別を登録します。  
   OS種別ごとの差異を対話ファイルごとに定義し、同一目的の対話ファイルを対話種別として纏めて機器差分を吸収(抽象化)します  
   詳細は 5.3.4対話種別リスト（Ansible-Pioneerのみ） を参照してください。
3. **対話ファイルの登録**

Ansible-Pioneerコンソールの対話ファイル素材集の画面から、対話種別とOS種別の組み合わせに対して対話ファイルを登録します。

詳細は 5.3.5対話ファイル素材集（Ansible-Pioneerのみ） を参照してください。

1. **テンプレートファイルの登録(必要に応じて実施)**

Ansible-Pioneerコンソールのテンプレート管理の画面から、対話ファイル内で定義しているtemplateモジュールのtemplateファイルとtemplate埋め込み変数の登録／更新／廃止を行います。

詳細は5.3.7テンプレート管理（Ansible-Legacy/Ansible-Pioneer） を参照してください。

1. **素材ファイルの登録(必要に応じて実施)**

Ansible-Pioneerコンソールのファイル管理の画面から、作業対象サーバに配置するファイルを登録します。

詳細は 5.3.8ファイル管理 を参照してください。

1. **Movementに対話ファイルを指定**Ansible-PioneerコンソールのMovement詳細の画面から、登録したMovementに対話ファイルに対応した対話種別リストを指定します。

詳細は 5.3.6Movement詳細 を参照してください。

1. **作業対象ホストの指定**Ansible-Pioneerコンソールの作業対象ホストの画面から、作業対象ホストを指定します。  
   詳細は 5.3.11作業対象ホスト を参照してください。
2. **変数値の設定**Ansible-Pioneerコンソールの代入値管理の画面から、Movementに登録したPlaybook内で定義した変数の値を設定します。変数を利用していない場合、設定は不要です。  
   詳細は 5.3.12代入値管理 を参照してください。
3. **作業実行**Ansible-Pioneerコンソールの作業実行の画面から、実行日時、投入オペレーションを選択して設定して処理の実行を指示します。

詳細は 5.3.15作業実行 を参照してください。

1. **作業状態確認**Ansible-Pioneerコンソールの作業状態確認の画面では、実行した作業の状態がリアルタイムで表示されます。また、作業の緊急停止や、実行ログ、エラーログを監視することができます。  
   詳細は 5.3.13作業状態確認 を参照してください。
2. **作業履歴確認**Ansible-Pioneerコンソールの作業管理の画面では、実行した作業の一覧が表示され履歴が確認できます。

詳細は 5.3.14作業管理 を参照してください。

**■登録画面項目一覧凡例**

次項に記載の登録画面項目一覧表の内容について説明します。

**②②**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **②** | **③** | **④** | **⑤** |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
|  |  |  |  |  |

**①項目**

　・サブメニュー内の項目名です

**②説明**

　・項目に対する説明です

**③入力必須**

　・○：項目に対する内容の入力が必須の項目

　・‐ ：項目に対する内容の入力が任意の項目

**④入力形式**

　・手動入力：手動での入力が必要な項目

　・自動入力：自動で内容が入力される項目

　・チェックボックス：チェックボックス形式の項目

　・ボタン：ラジオボタン形式の項目

　・リスト選択：リストボックス形式の項目

**⑤制約事項**

　・項目に対する制約事項(文字数制限など)です

# Ansible driver機能・操作方法説明

本章では、Ansible driverで利用する各コンソールの機能について説明します。

## 基本コンソール

本節では、astroll基本コンソールでの操作について記載します。

本作業はastroll基本コンソールマニュアルを参照して、astroll基本コンソール画面内で作業を実施してください。

### 機器一覧

1. [機器一覧]では、作業対象ホストの情報を登録／更新／廃止を行います。  
   本書では、Ansible driverの動作に必要となる機器一覧の項目(赤枠)について説明します。  
   他の項目についての説明は、「利用手順マニュアル\_astroll\_基本コンソール）」を参照してください。

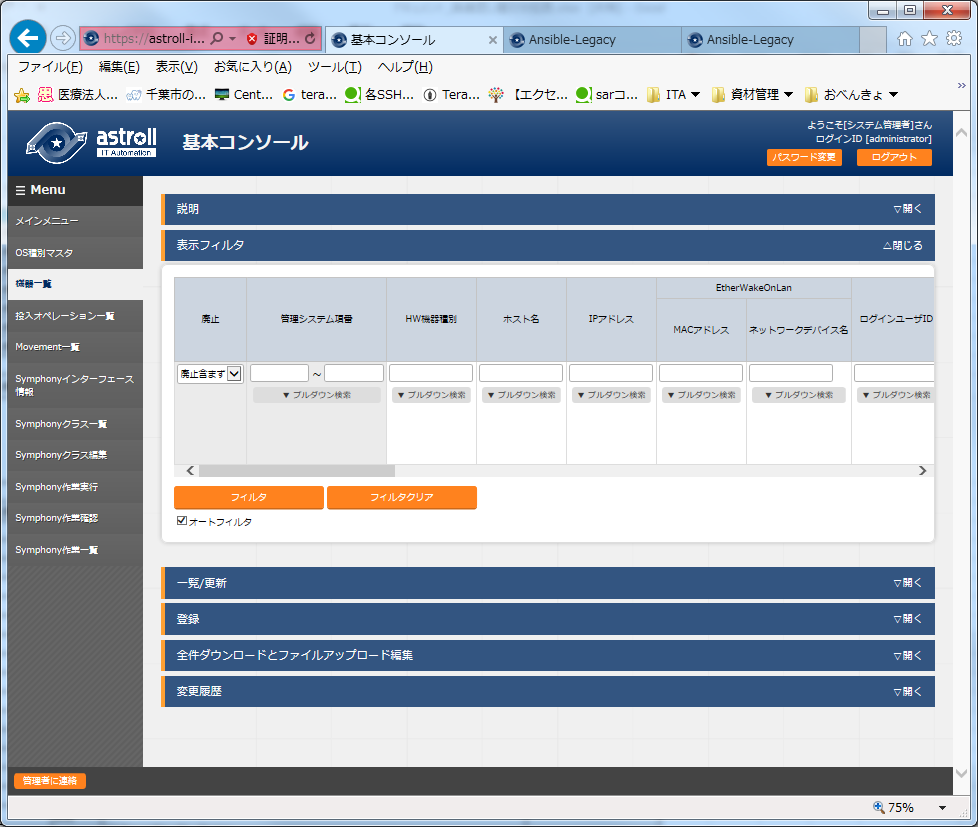


図 5.1‑1サブメニュー画面（機器一覧）

1. 「登録」-「登録開始」ボタンより、機器情報の登録を行います。



図 5.1‑2 登録画面（機器一覧 - 共通項目）



図 5.1‑3 登録画面（機器一覧 - Ansible利用情報）

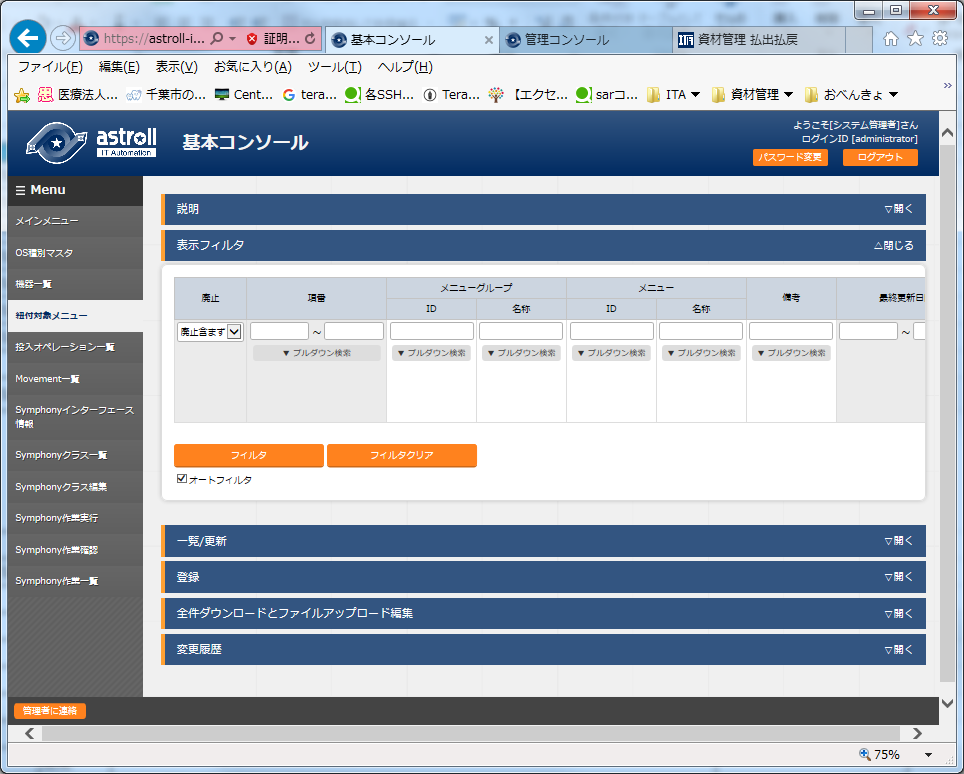
1. 登録画面の共通項目一覧は以下のとおりです。  
   Web画面のカラム名の後ろに赤のアスタリスク（＊）が付いているカラムが必須入力になりますが、Ansible　driverを利用する場合には、Ansible利用情報も必須入力になります。  
   未入力で作業実行した場合、想定外エラーとなる場合があります。

**表 5.1‑1　登録画面項目一覧（機器一覧）**

| **項目** | | | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| --- | --- | --- | --- | --- | --- | --- |
| 管理システム項番 | | | 登録情報を識別する一意のＩＤが自動入力されます | - | 自動入力 | - |
| ホスト名 | | | ホスト名を記入します | ○ | 手動入力 | 最大長128バイト |
| IPアドレス | | | IPアドレス(xxx.xxx.xxx.xxx形式)を記入します | ○ | 手動入力 | 最大長15バイト |
| EtherWake  OnLan | MAC  アドレス | | MACアドレスを記入します | - | 手動入力 | 最大長17バイト |
| ネットワークデバイス名 | | ネットワークデバイス名を記入します | - | 手動入力 | 最大長32バイト |
| ログインユーザID | | | ログインユーザIDを記入します | ○ | 手動入力 | 最大長30バイト |
| ログインパスワード | 管理 | | astrollでパスワードを管理する場合「●」を選択します | ○ | リスト選択 | - |
| ログイン  パスワード | | パスワードを指定します | ○ | 手動入力 | 最大長30バイト |
| ssh認証鍵ファイル | | | ssh認証鍵ファイルを指定して鍵認証する場合のファイルを入力します。  認証方式が鍵方式の場合でssh認証鍵ファイルを指定する場合に必須入力となります。 | - | ファイル選択 | 最大サイズ10Kバイト |
| Legacy/Role  利用情報 | 認証方式 | | Ansibleからホストへ接続する際の認証方式を選択します。  ・パスワード方式の場合、ログインパスワードの管理は、「●」である必要があります。  ・鍵方式の場合、ログインユーザでのsudo権限を/etc/sudoersに設定しておく必要があります。また、公開鍵を交換しておく必要があります。 | ○ | リスト選択 | 説明欄記載のとおり |
| WinRM接続情報 | ポート番号 | WindowsServerにWinRM接続する際のポート番号を入力します。  未入力の場合はデフォルト(5985)でのWinRM接続となります。 | - | 手動入力 | 説明欄記載のとおり |
| ｻｰﾊﾞ証明書 | WinRM接続ポートでhttpsのポート番号を指定した場合にｻｰﾊﾞ証明書を入力します。  サーバ証明書の認証を省く場合、インベントリファイル追加オプションに下記を追記して下さい。  ansible\_winrm\_server\_cert\_validation=ignore | - | ファイル選択 | 最大サイズ10Kバイト |
| Pioneer  利用情報 | プロトコル | | 対象機器にログインする際のプロトコル(ssh/telnet)を選択します。 | ○ | リスト選択 | - |
| OS種別 | | 対象機器のOSを選択します。 OS種別マスタで登録されているOS種別がリスト表示されます。 | ○ | リスト選択 | - |
| 接続オプション | | | 1. （ssh接続の場合）   /etc/ansible.cfg/ssh\_argsに設定しているsshオプション以外のオプションを設定したい場合、設定したいオプションを入力します。  （telnet接続の場合）  telnet接続時のオプションを設定したい場合、設定したいオプションを入力します。 | - | 手動入力 | 最大長512バイト |
| インベントリファイル追加オプション | | | astrollが設定しないインベントリファイルのオプションを設定したい場合に追加するオプションを入力します  Exp)  Pythonのバージョンが2.7以降でhttpsのサーバ証明書の検証を行わない場合。  ansible\_winrm\_server\_cert\_validation=ignore | - | 手動入力 | 最大長512バイト |
| 備考 | | | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

### 紐付対象メニュー

1. [紐付対象メニュー]では、代入値自動登録設定で連携するCMDBのメニューを登録／更新／廃止を行います。  
   ※CMDBをカスタマイズしたときに、連携できるオプションの機能です。デフォルトでは利用しません。

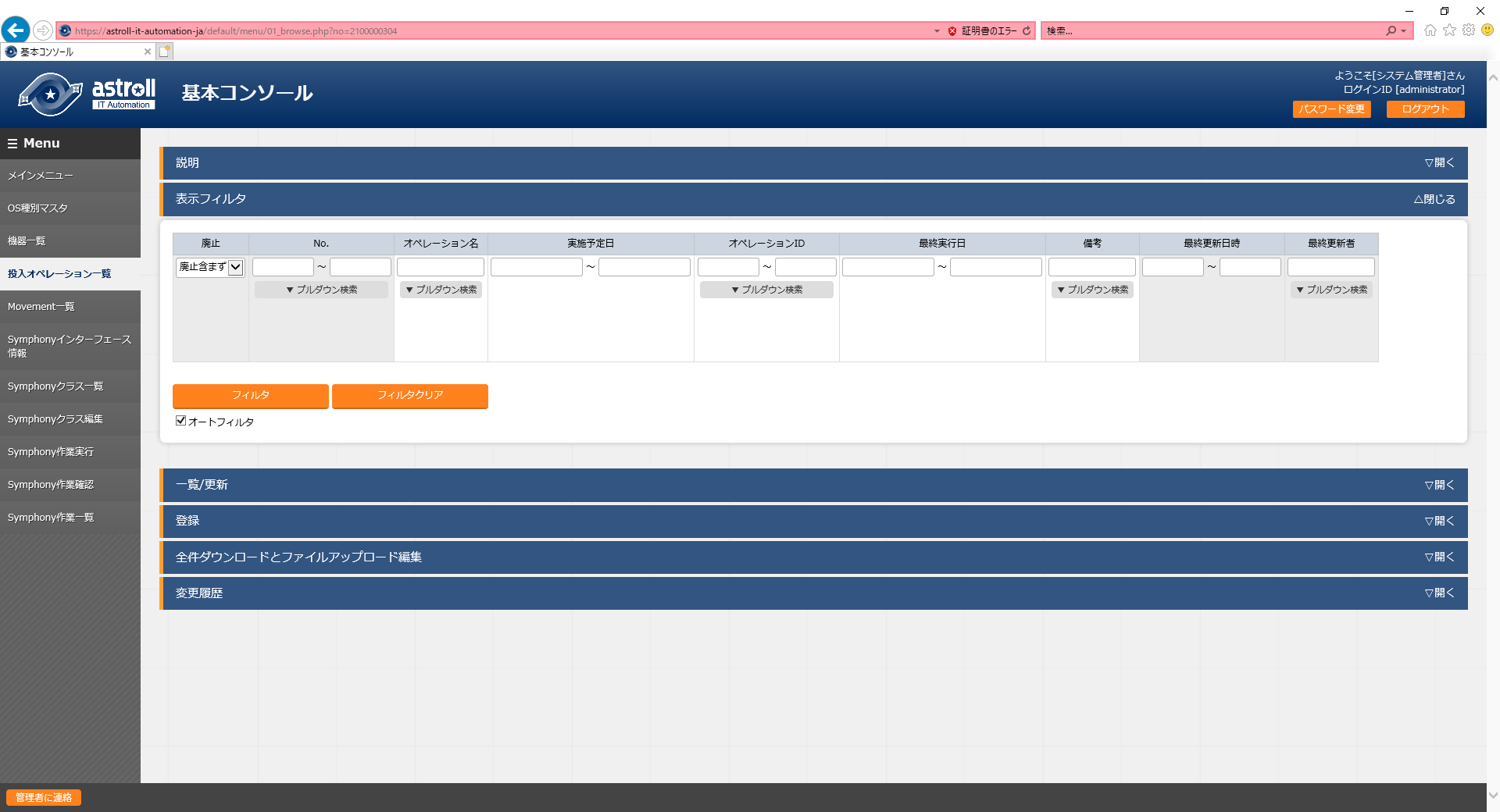


**図 5.1‑4サブメニュー画面（紐付対象メニュー）**

登録方法の詳細は、関連マニュアルの『利用手順マニュアル\_astroll\_基本コンソール』をご参照下さい。

### 投入オペレーション一覧

1. [投入オペレーション一覧]画面では、オーケストレータで実行する作業対象ホストに対するオペレーションを管理します。作業はastroll基本コンソール内メニューより選択します。



**図 5.1‑5サブメニュー画面（投入オペレーション一覧）**

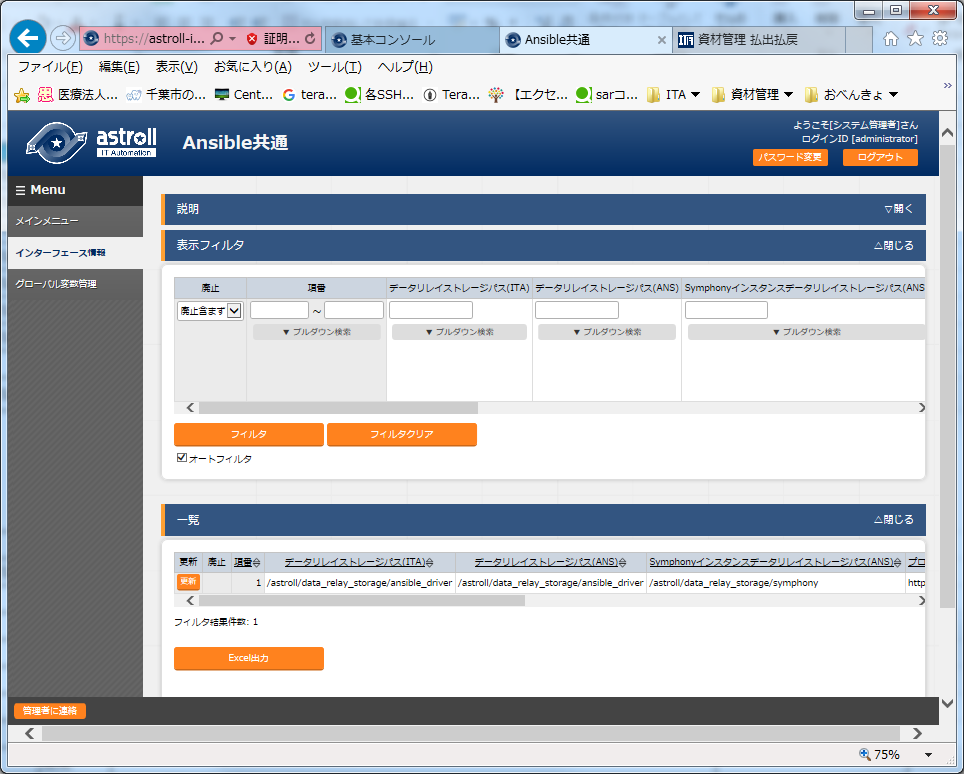
登録方法の詳細は、関連マニュアルの『利用手順マニュアル\_astroll\_基本コンソール』をご参照下さい。

## Ansible 共通コンソール

本節では、Ansible共通コンソールでの操作について記載します。

### インタフェース情報

1. [インタフェース情報]では、astrollシステム・Ansible driverサーバとAnsibleサーバが共有するディレクトリのパスのおよびAnsibleサーバへの接続インタフェース情報を登録／更新／廃止を行います。  
   ※なお、インストール時に必要な値が設定されるため、基本的には設定は必要ありません。



**図 5.2‑1サブメニュー画面（インタフェース情報）**

1. 「一覧」-「更新」ボタンより、オペレーション情報の登録を行います。

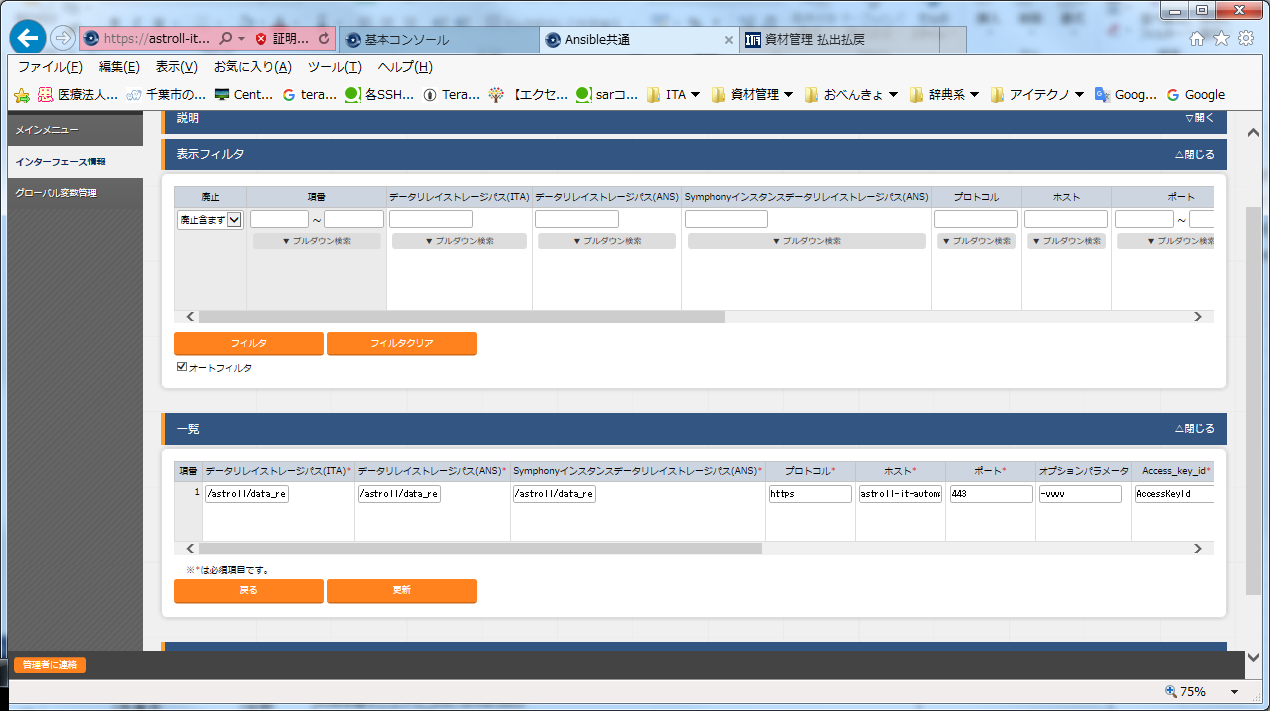


図 5.2‑2 登録画面（インタフェース情報）

1. インタフェース情報画面の項目一覧は以下のとおりです。  
   インタフェース情報が未登録または、複数レコード登録されている状態で作業実行した場合、**作業実行は想定外エラーとなります**。

**表 5.2‑1　登録画面項目一覧（インタフェース情報）**

| **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(astroll)※1 | astrollシステム・Ansible driverサーバから見たディレクトリを入力します。 | ○ | 手動入力 | 最大長128バイト |
| データリレイストレージパス(ANS) | Ansible RestAPIサーバから見たディレクトリを入力します。 | ○ | 手動入力 | 最大長128バイト |
| Symphonyインスタンス  データリレイストレージパス(ANS) | Symphony実行時、各Movementで共有するディレクトリを、Ansible RestAPIサーバで共有するディレクトリで入力します。  astrollシステムから見たパスはSymphonyインターフェース情報より設定します。Symphonyインターフェース情報については「利用手順マニュアル astroll基本コンソール」を参照して下さい。 | ○ | 手動入力 | 最大長128バイト |
| プロトコル | http / https　のどちらかを入力します。 | ○ | 手動入力 | - |
| ホスト | Ansibleサーバのホスト名（またはIPアドレス）を入力します。  HTTPS通信の場合はホスト名が推奨です。 | ○ | 手動入力 | 最大長128バイト |
| ポート | Ansibleサーバの接続ポート(80/443)を入力します。通常はHTTPS(443)です。 | ○ | 手動入力 | - |
| オプションパラメータ | ansible-playbookコマンドのオプションパラメータを入力します。デフォルトは –vvvv が入力されています。  -i オプションはastrollが設定します。  -f オプションは「[5.4.6 Movement一覧](#_Movement詳細)」の並列実行数での設定を推奨します。  --checkオプションは「[6.2.1 作業実行](#_作業実行)」のドライランボタンの使用を推奨します。 | - | 手動入力 | 最大長512バイト |
| ACCESS\_KEY\_ID | Ansibleサーバ接続時の認証に使用するアクセスキーを入力します。 | ○ | 手動入力 | 最大長64バイト |
| SECRET\_ACCESS\_KEY | Ansibleサーバ接続時の認証に使用するシークレットアクセスキーを入力します。 | - | 手動入力 | 最大長64バイト |
| 状態監視周期（単位ミリ秒） | 「5.3.13作業状態確認」で表示されるログのリフレッシュ間隔を入力します。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 |
| 進行状態表示行数 | 「5.3.13作業状態確認」での進行ログ・エラーログの最大表示行数を入力します。通常は1000行程度が推奨値です。 | ○ | 手動入力 | - |
| NULL連携 | 代入値自動登録設定でパラメータシートの具体値がNULL(空白)の場合に、代入値管理への登録をNULL(空白)の値で行うか設定します。  代入値自動登録設定メニューの「NULL連携」が空白の場合この値が適用されます。  ・「有効」の場合、パラメータシートの値がどのような値でも代入値管理への登録が行われます。  ・「無効」の場合、パラメータシートに値が入っている場合のみ代入値管理への登録が行われます。 | ○ | リスト選択 | - |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

※1 データリレイストレージパスは、それぞれ異なるサーバで運用される場合、ディレクトリパス名が異なる可能性があるため、別々に管理します。詳細は「システム構成／環境構築ガイド（Ansible driver編）」を参照してください。

### グローバル変数管理

1. [グローバル変数管理]では、Playbookや対話ファイルなどで利用するグローバル変数名を登録／更新／廃止を行います。



**図 5.2‑3サブメニュー画面（グローバル変数管理）**

1. 「登録」-「登録開始」ボタンより、オペレーション情報の登録を行います。



**図 5.2‑4 登録画面（グローバル変数管理）**

1. グローバル変数管理画面の項目一覧は以下のとおりです。

**表 5.2‑2　登録画面項目一覧（グローバル変数管理）**

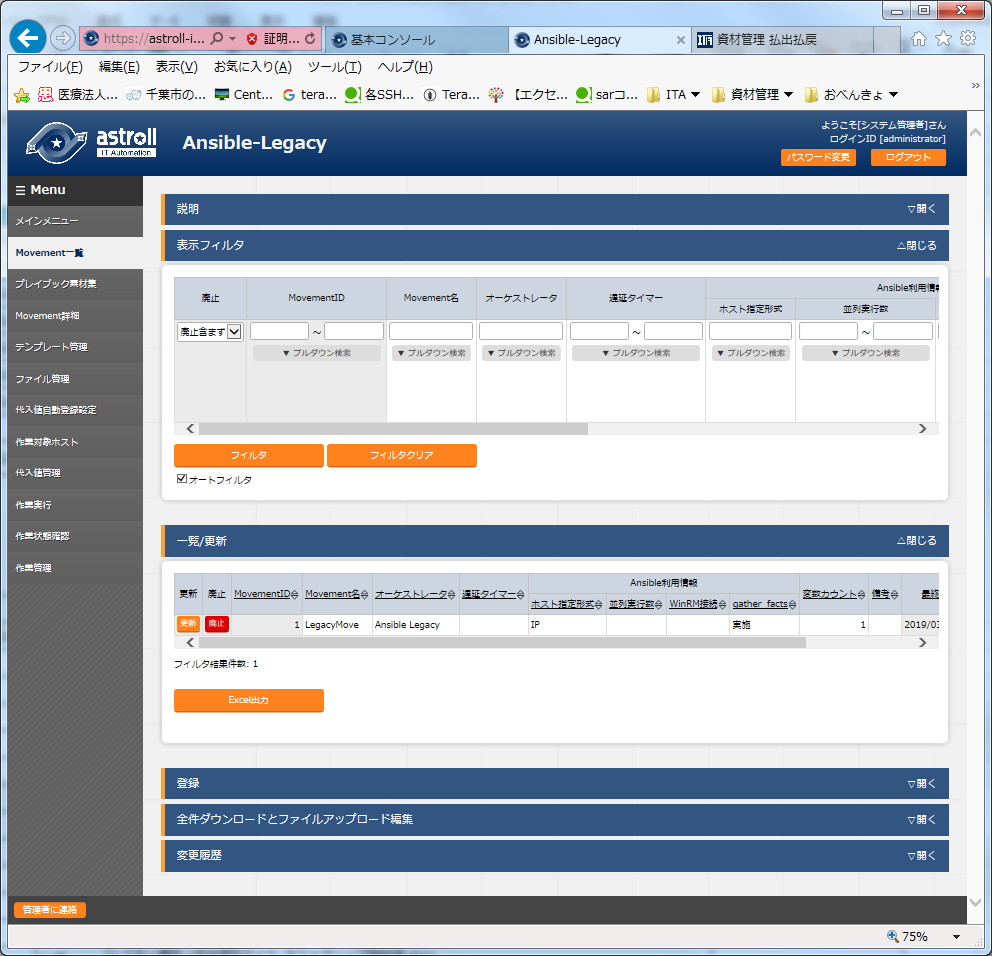
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| グローバル変数名 | 変数名を入力します。  変数名は、「GBL\_\*\*\*\*」形式で入力します。  \*\*\*\*：半角英数字とアンダースコア（ \_ ）が利用可能です。(最小値:1バイト、最大値:128バイト) | ○ | 手動入力 | 説明欄記載のとおり |
| 具体値 | 具体値を入力します。  具体値にファイル埋込変数「CPF\_」とテンプレート埋込変数「TPF\_」が入力出来ます。変数を記述する場合、Playbookに変数を記述する場合と同様、変数名を{{ }}で囲みます。  入力例)  具体値にTPF\_sampleを入力する場合  '{{△TPF\_sample△}}' △:半角スペース  ‘:任意  各モードで具体値にファイル埋込変数とテンプレート埋込変数が入力されている場合の扱い。  ・Ansible-Legacy  　ファイル埋込変数とテンプレート埋込変数が各変数として扱われます。  ・Ansible-Legacy-Role  　ファイル埋込変数のみが変数として扱われます。  　テンプレート埋込変数は単なる具体値として扱われます。  ・Ansible-pioneer  　単なる具体値として扱われます。 | ○ | 手動入力 | 最大長1024バイト |
| 変数名説明 | 変数に対する説明・コメントを入力します | - | 手動入力 | 最大長128バイト |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

## Ansibel-Legacy／Legacy Role／Pioneer　コンソール

Ansibel-Legacy／Legacy Role／Pioneer　コンソールの操作です。

### Movement一覧

1. [Movement一覧]ではMovement名の登録／更新／廃止を行います。



**図 5.3‑1サブメニュー画面（Movement一覧）**

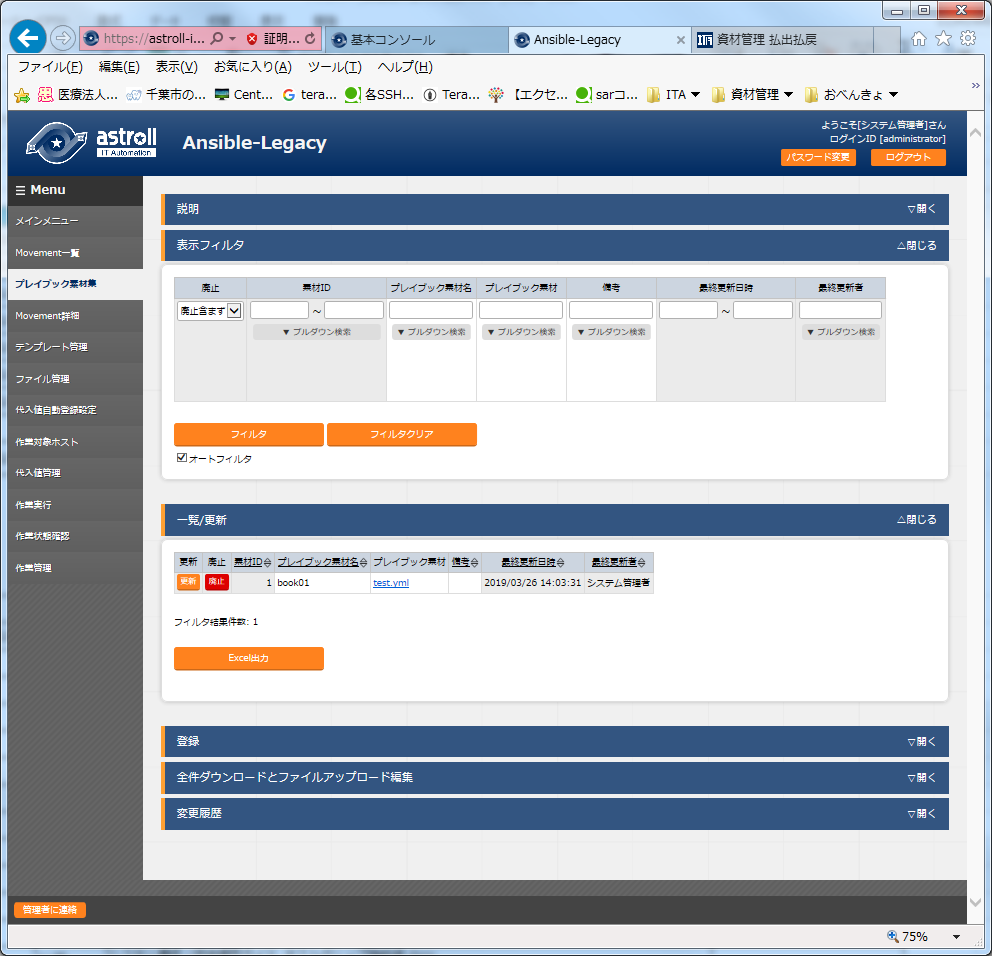
登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

ご参照下さい。

### プレイブック素材集（Ansible-Legacyのみ）

1. [プレイブック素材集]ではユーザーが作成したPlaybookの登録／更新／廃止を行います。  
   ※本メニューはAnsible-Legacy コンソールにのみ存在します。

Playbookの記述など関しては、「6.1Playbook（Ansible-Legacy）の記述」を参照してください。



**図 5.3‑2サブメニュー画面（プレイブック素材集）**

1. 「登録」-「登録開始」ボタンより、Playbookの登録を行います。



図 5.3‑3 登録画面（プレイブック素材集）

1. 登録画面の項目一覧は以下のとおりです。

**表 5.3‑1　登録画面項目一覧（プレイブック素材集）**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| プレイブック素材名 | astrollで管理するプレイブック素材名を入力します。 | ○ | 手動入力 | 最大長32バイト |
| プレイブック素材 | 作成したPlaybookファイルをアップロードします。 | ○ | ファイル選択 | 最大サイズ20Mバイト |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

「登録」の前に、「プレイブック素材」を「事前アップロード（①）」してください。「アップロード状況（②）」にPlaybookのファイル名が表示されたのを確認してから、「登録」ボタンを押してください。



内部の処理でPlaybookファイル内に定義している変数を抜出します。抜出した変数は、「5.3.12代入値管理代入値自動登録設定」で具体値の登録が可能になります。

抜出する タイミングはリアルタイムではありませんので、「5.3.10代入値自動登録設定」や「5.3.12代入値管理」で変数が扱えるまでに時間がかかる※4場合があります。

※4 「本書：7.2起動周期の変更」の起動周期に依存します。

### ロールパッケージ管理（Ansible-Legacy Roleのみ）

1. ユーザーが作成したロールパッケージファイルの登録／更新／廃止を行います。

※本メニューはAnsible-Legacy Role コンソールにのみ存在します。

ロールパッケージファイルは、「roles」のある階層のディレクトリをzipにて圧縮したものを登録してください。ロールパッケージディレクトリ構成などは「6.3ロールパッケージ（Ansible-Legacy Role）の記述」を参照してください。



**図 5.3‑4サブメニュー画面（ロールパッケージ管理）**

登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

ご参照下さい。

### 対話種別リスト（Ansible-Pioneerのみ）

1. [対話種別リスト]では、対話種別の登録／更新／廃止を行います。

※本メニューはAnsible-Pioneer コンソールにのみ存在します。

Ansible-Pioneerでは、「OS種別」ごとの差異を対話ファイルごとに定義し、同一目的の対話ファイルを「対話種別」として纏めて機器差分を吸収(抽象化)します。



**図 5.3‑5サブメニュー画面（対話種別リスト）**

1. 「登録」-「登録開始」ボタンより、オペレーション情報の登録を行います。



図 5.3‑6 登録画面（対話種別リスト）

1. 登録画面の項目一覧は以下のとおりです。

**表 5.3‑2　登録画面項目一覧（対話種別リスト）**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| 対話種別名 | 対話種別名を入力します | ○ | リスト選択 | 最大長32バイト |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

### 対話ファイル素材集（Ansible-Pioneerのみ）

1. [対話ファイル素材集]では、ユーザーが作成した対話ファイルの登録／更新／廃止を行います。

※本メニューは、Ansible-Pioneer コンソールにのみ存在します。

対話ファイルの記述などに関しては、「5.3.4対話種別リスト（Ansible-Pioneerのみ）」を参照してください。

「対話種別」と「OS種別」の組み合わせごとに対話ファイルを登録します。

１つの「対話種別」で複数のOSに対応させたい場合は、同じ「対話種別」で、「OS種別」それぞれについて対話ファイルを登録してください。



**図 5.3‑7サブメニュー画面（対話ファイル素材集）**

1. 「登録」-「登録開始」ボタンより、対話ファイル素材の登録を行います。



図 5.3‑8 登録画面（対話ファイル素材集）

1. 登録画面の項目一覧は以下のとおりです。

**表 5.3‑3　登録画面項目一覧（対話ファイル素材集）**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| 対話種別 | 対話種別リストに登録されている対話種別が表示されます。  登録する対話ファイルの対話種別を選択します。 | ○ | リスト選択 | - |
| OS種別 | OS種別マスタに登録されているOS種別が表示されます。  登録する対話ファイルのOS種別を選択します。 | ○ | リスト選択 | - |
| 対話ファイル素材 | 対話種別とOS種別に対応する対話ファイルをアップロードします。 | ○ | ファイル登録 | 最大サイズ20Mバイト |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

「登録」の前に、「対話ファイル」を「事前アップロード（①）」してください。「アップロード状況（②）」に対話ファイルのファイル名が表示されたのを確認してから、「登録」ボタンを押してください。



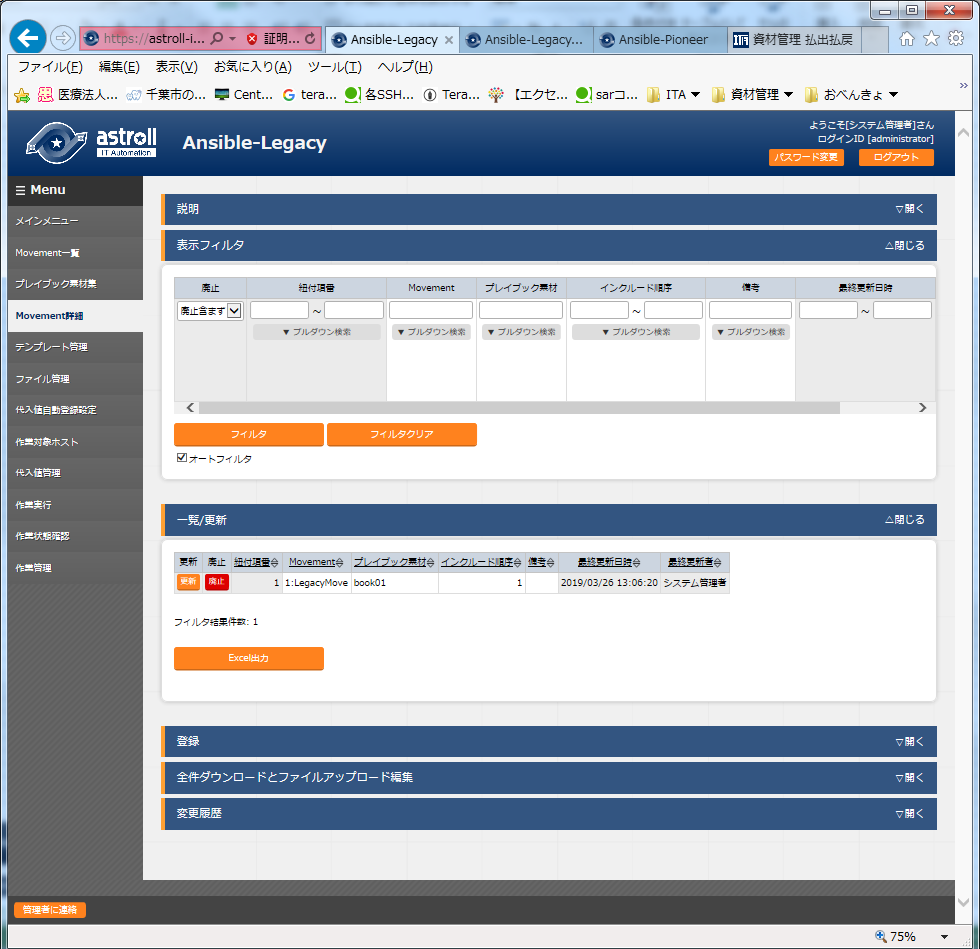
内部の処理で対話ファイル内に定義している変数を取り出します。取り出した変数は、「5.3.10代入値自動登録設定」や「5.3.12代入値管理」で具体値の登録が可能になります。

抜出する タイミングはリアルタイムではないので、「5.3.10代入値自動登録設定」や「5.3.12代入値管理」で変数が扱えるまでに時間がかかる※6場合があります。

※6 「本書：7.2起動周期の変更」の起動周期に依存します。

### Movement詳細

1. [Movement詳細]では、Movementで実行する素材の登録／更新／廃止を行います。



**図 5.3‑9サブメニュー画面（Movement詳細）**※画面はAnsible-Legacyのものです。

1. 「登録」-「登録開始」ボタンより、Movement詳細の登録を行います。



図 5.3‑10 登録画面（Movement詳細）

1. 登録画面の項目一覧は以下のとおりです。

* **Ansible-Legacy の場合**

**表 5.3‑4　登録画面項目一覧（Movement詳細 Ansible-Legacy の場合）**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| Movement | Movement一覧で登録したMovementが表示されます。  Movementを選択します。 | ○ | リスト選択 | - |
| プレイブック素材 | 「5.3.2プレイブック素材集（Ansible-Legacyのみ）」で登録したプレイブック素材が表示されます。プレイブック素材を選択します。 | ○ | リスト選択 | - |
| インクルード順序 | プレイブック素材の実行順序(1～:一意値)を入力します。  入力されたインクルード順序(昇順)でプレイブック素材が実行されます。 | ○ | 手動入力 | 半角整数 |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

* **Ansible-Legacy Role の場合**

**表 5.3‑5　登録画面項目一覧（Movement詳細 Ansible-Legacyの場合）**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| Movement | Ansible-Legacyと同様。 | ○ | リスト選択 | - |
| ロールパッケージ名 | ロールパッケージ管理で登録したロールパッケージが表示されます。実行するロールパッケージを選択します。 | ○ | リスト選択 | - |
| ロール名 | ロールパッケージ名で選択したロールパッケージに含まれているロール名が表示されます。実行するロールパッケージ内のロールを選択します。 | ○ |  | - |
| インクルード順序 | Ansible-Legacyと同様。 | ○ | 手動入力 | 半角整数 |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

**【注意】**

**同一Movementに複数のロールパッケージを登録しないでください。**

**複数のロールパッケージが登録されているMovementを作業実行した場合、想定外エラーとなります。**

* **Ansible-Pioneer の場合**

**表 5.3‑6　登録画面項目一覧（Movement詳細 Ansible-Pioneerの場合）**

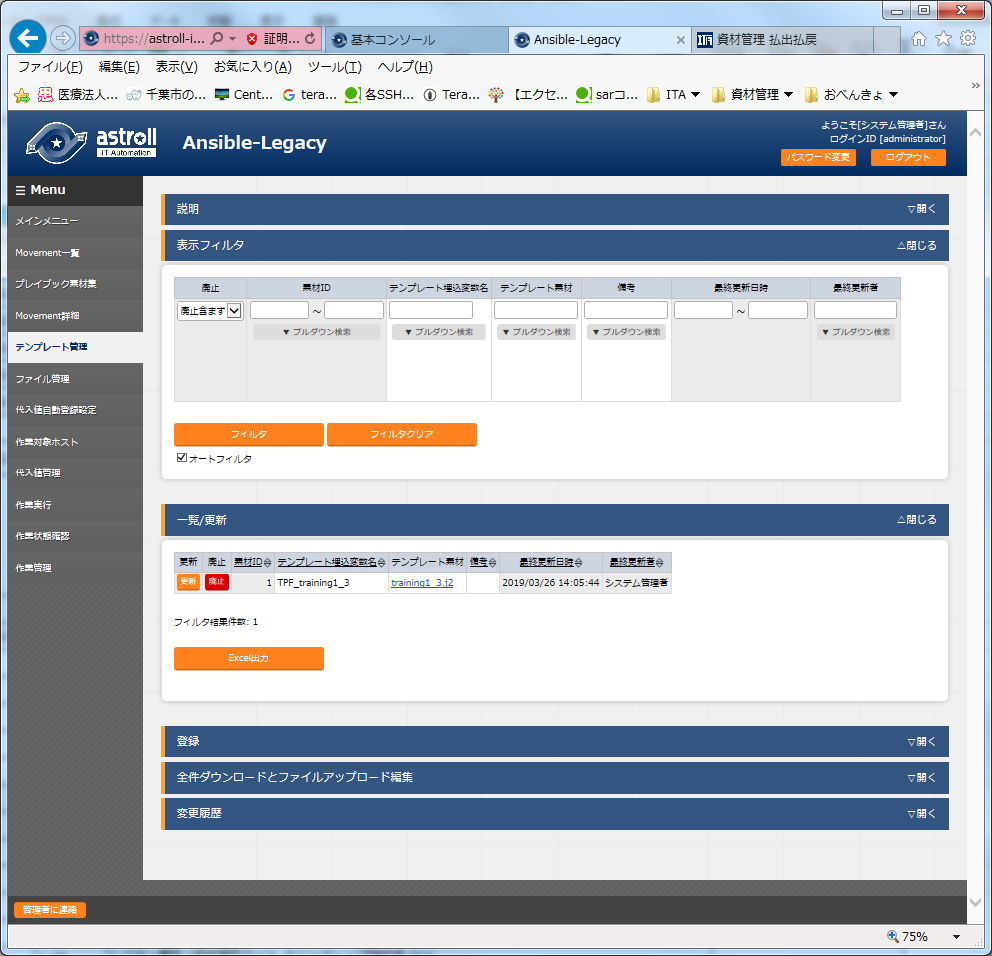
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **項目** | **説明** | **入力**  **必須** | **入力形式** | **制約事項** |
| Movement | Ansible-Legacyと同様。 | ○ | リスト選択 | - |
| 対話種別 | 「5.3.4対話種別リスト（Ansible-Pioneerのみ）」で登録した対話種別が表示されます。実行する対話種別を選択します。  ホスト毎にOS種別と対話種別に関連付く対話ファイルが実行対象となります。 | ○ | リスト選択 | - |
| インクルード順序 | Ansible-Legacyと同様。 | ○ | 手動入力 | 半角整数 |
| 備考 | 自由記述欄です。 | - | 手動入力 | 最大長4000バイト |

### テンプレート管理（Ansible-Legacy/Ansible-Pioneer）

1. [テンプレート管理]では、Playbookで定義しているtemplateモジュールやios\_configモジュールなどのパラメータで使用するJinja2テンプレートファイルとtemplate埋め込み変数の登録／更新／廃止を行います。

※本メニューは、Ansible-LegacyとAnsible-Pioneer コンソールに存在します。

Ansible-LegacyとAnsible-Pioneerでは、テンプレート管理でテンプレートモジュールを登録しておくことで、Playbook内で定義しているtemplateモジュールで使用するtemplateファイルをtemplate埋め込み変数で指定することが出来ます。



**図 5.3‑11サブメニュー画面（テンプレート管理）**

登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

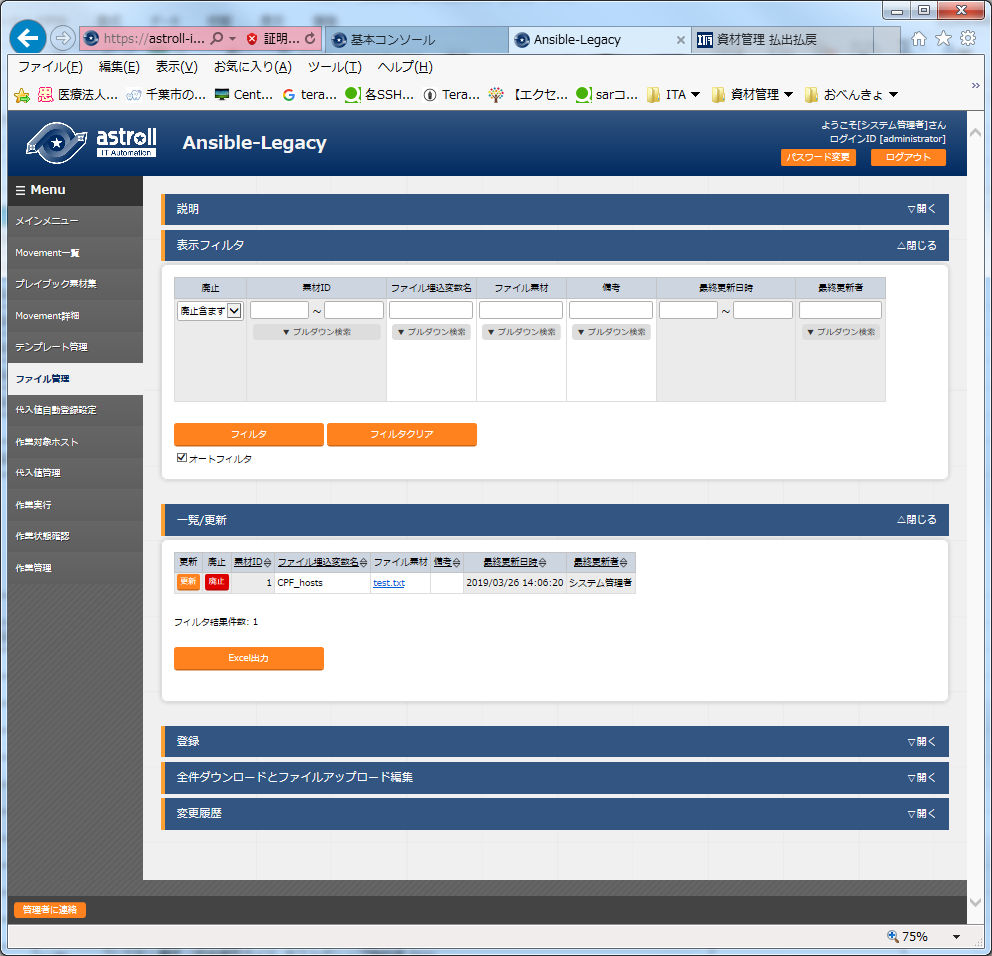
ご参照下さい。

### ファイル管理

1. [ファイル管理]では、Playbook内で定義している各モジュールで使用するファイルとファイル埋め込み変数の登録／更新／廃止を行います。

※本メニューは、Ansible-Legacy/ Ansible-Legacy Role/ Ansible-Pioneerコンソールに存在します。

ファイル管理でファイル素材を登録しておくことで、Playbook内で定義している各モジュールで使用するファイルをファイル埋め込み変数名で指定することが出来ます。



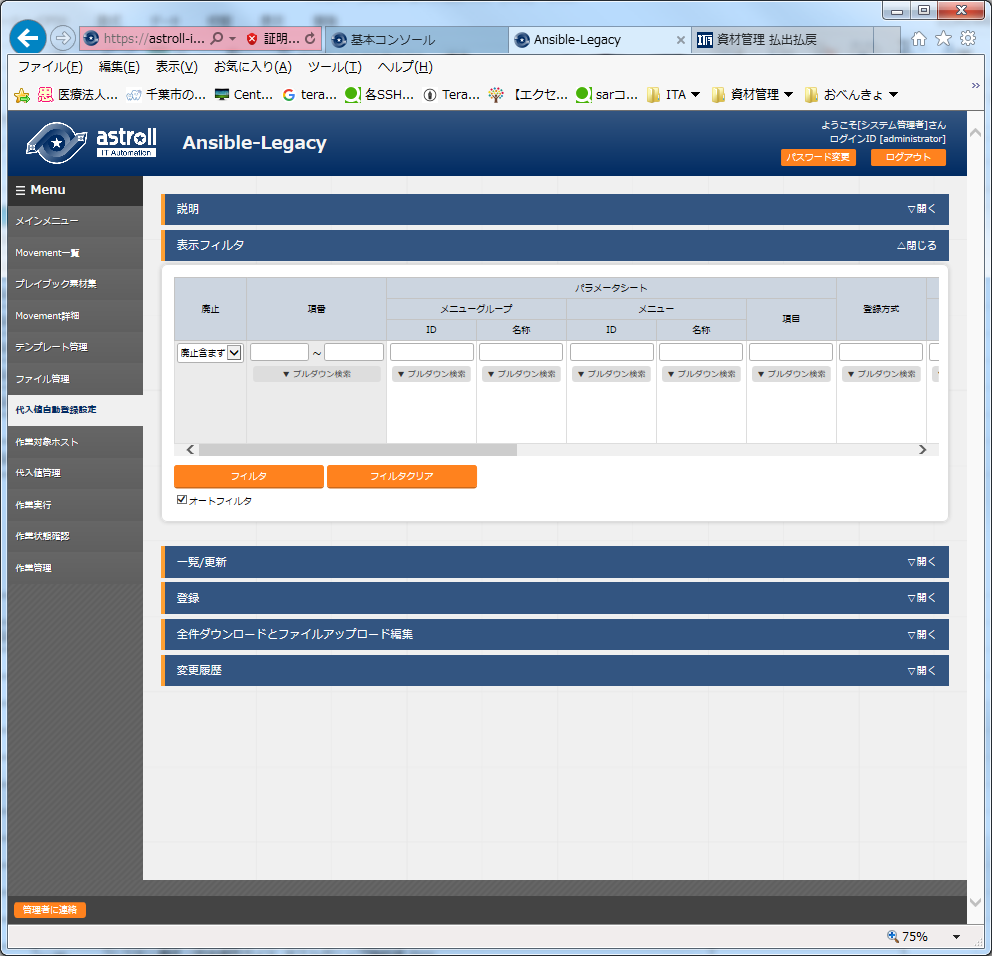
**図 5.3‑12サブメニュー画面（ファイル管理）**

登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

ご参照下さい。

### 多段変数最大繰返数管理（Ansible-Legacy Roleのみ）

1. [多段変数最大繰返管理]では、「5.3.3ロールパッケージ管理（Ansible-Legacy Roleのみ）」で登録したロールパッケージで定義されている多段変数内で繰返配列定義されているメンバー変数の配列の最大繰返数の更新が行えます。変更したいメンバー変数の更新ボタンをクリックし最大繰返数を更新します。



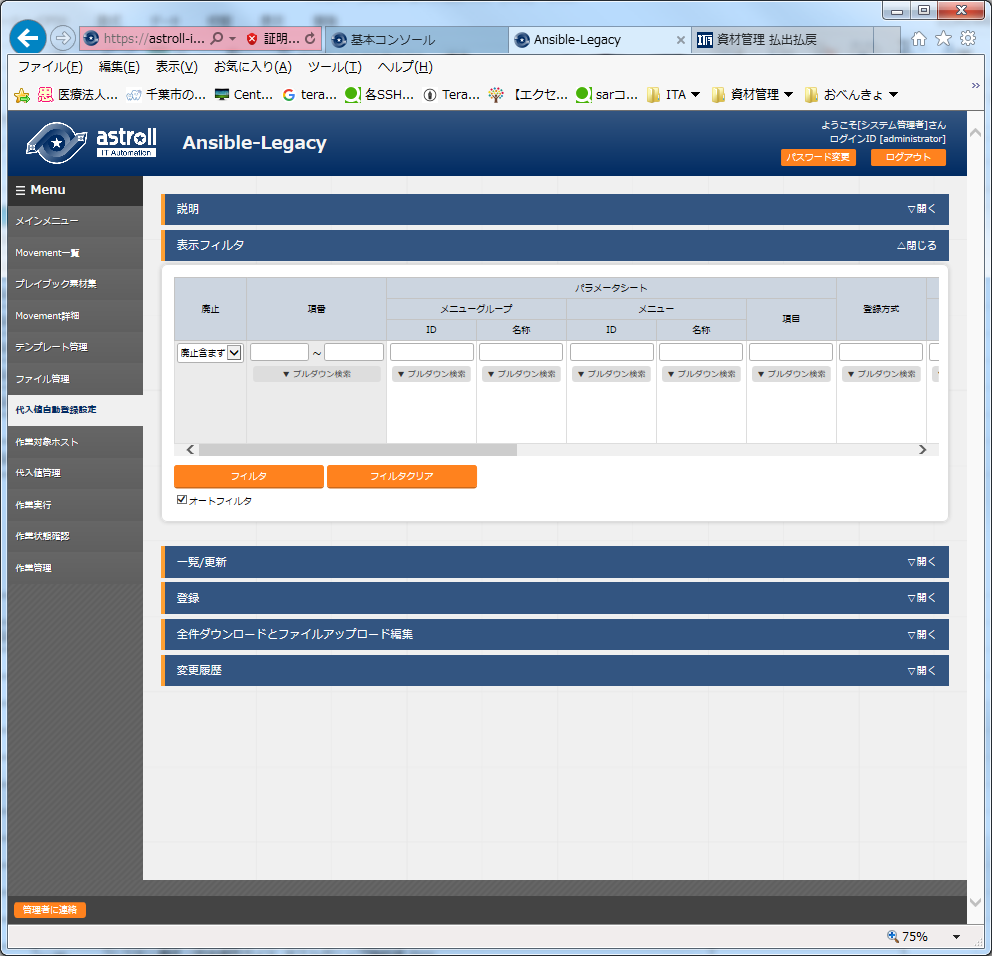
**図 5.3‑13サブメニュー画面（多段変数最大繰返数管理）**

登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

ご参照下さい。

### 代入値自動登録設定

1. 基本コンソールの「紐付対象メニュー」で連携対象としたCMDBのオぺレーションとホスト毎の項目の設定値を紐付けるMovementと変数の登録／更新／廃止を行います。登録した情報は内部の処理により代入値管理と作業対象ホストに反映されます。  
   ※CMDBをカスタマイズしたときに、連携できるオプションの機能。 デフォルトでは利用しません。



**図 5.3‑14サブメニュー画面（代入値自動登録設定）**

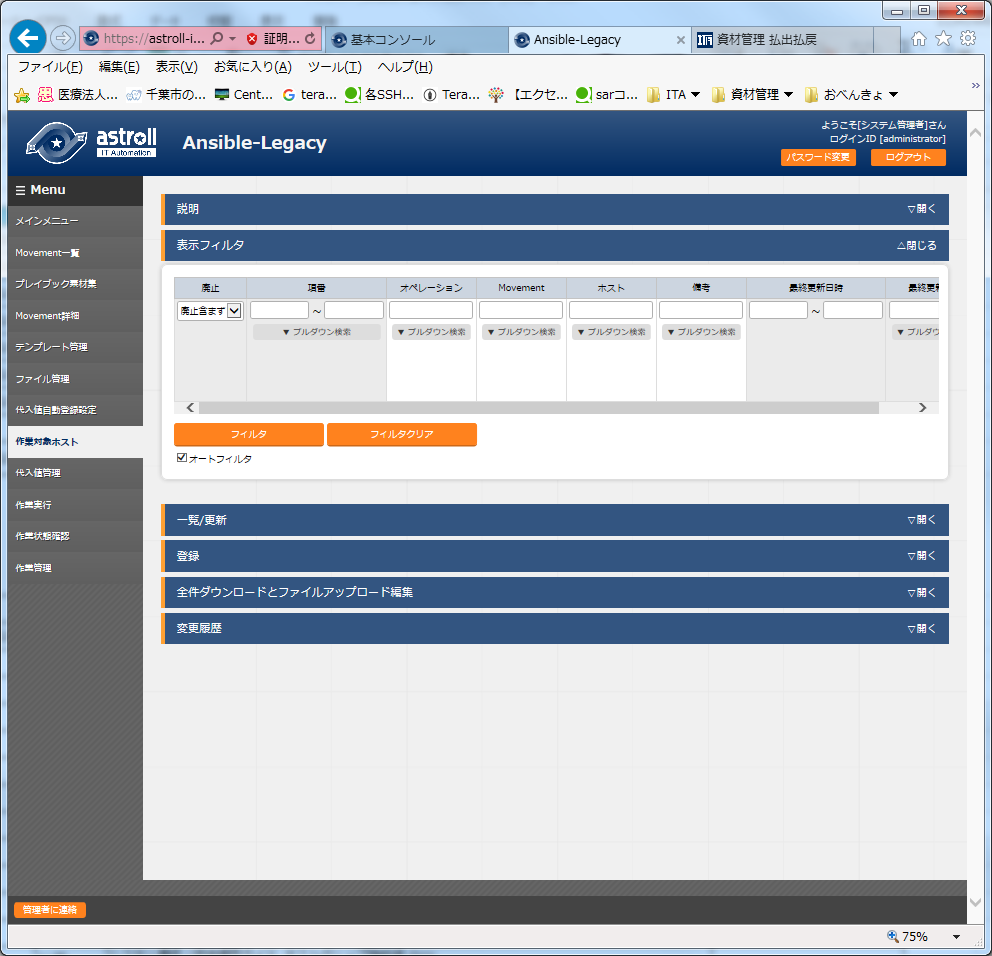
※画面はAnsible-Legacy Roleのものです

登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

ご参照下さい。

### 作業対象ホスト

1. [作業対象ホスト]では、オペレーションに関連付くMovementとホストの登録／更新／廃止を行います。



**図 5.3‑15サブメニュー画面（作業対象ホスト）**

登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

ご参照下さい。

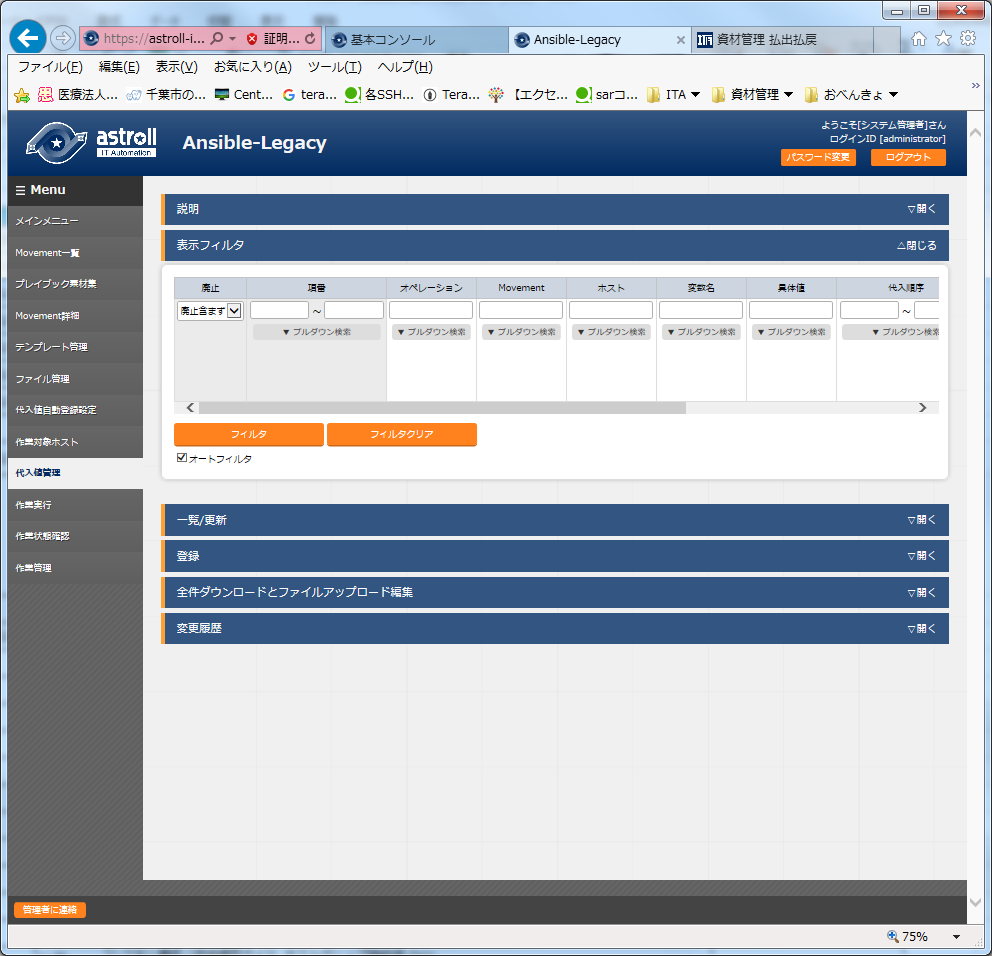
### 代入値管理

1. 変数への代入値の登録／更新／廃止を行います。

オペレーションごとに、対象のMovementで利用されるPlaybookやテンプレートファイル内の変数「VAR\_」に代入する具体値をメンテナンス(閲覧/登録/更新/廃止)できます。

また、読替表の定義により「VAR\_」以外の変数「LCA\_」に対して代入する具体値をメンテナンスできます。詳しくは「[6.5 読替表の記述](#_読替表（Ansible_Legacy_Roleのみ）の記述)」を参照してください。

登録した変数の情報は作業実行時にホスト変数ファイル(host\_vars/配下)に出力されます。



**図 5.3‑16サブメニュー画面（代入値管理）**

※画面はAnsible-Legacy Roleのものです。

登録方法の詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』を

ご参照下さい。

1. **代入順序の入力**

Ansible-Legacyでは、代入順序が未入力の場合は、通常変数として扱います。  
代入順序が入力されている場合は、複数具体値変数として扱います。複数具体値変数の場合は複数の具体値が必要ない場合(具体値が1個でよい)でも代入順序は入力してください。  
Ansible-Legacy Roleでは、変数名またはメンバー変数名を選択することで、複数具体値変数の場合のみ代入順序が入力可能となります。複数具体値変数の場合に入力してください。

Ansible-Pioneerでは、代入順序が未入力の場合は、通常変数として扱います。  
代入順序が入力されている場合は、複数具体値変数として扱います。複数具体値変数の場合は、

複数の具体値が必要ない場合(具体値が1個でよい)でも代入順序を入力してください。

各モードとも、特定の複数具体値変数に対して代入順序が連続していなくても問題ありません。

Exp)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **代入値管理の登録** | |  |  |  | **HOST\_Aのホスト変数ファイルへの出力内容** |
| ホスト | 変数 | 具体値 | 代入  順序 |  | VAR\_std: value1 VAR\_list\_a:  - value2 VAR\_list\_b:  - value3  - value4 |
| HOST\_A | VAR\_std | value1 |  |  |
| HOST\_A | VAR\_list\_a | value2 | 10 |  |
| HOST\_A | VAR\_list\_b | value3 | 100 |  |
| HOST\_A | VAR\_list\_b | value4 | 200 |  |
|  |  |  |  |  |

1. **ホスト変数ファイルへの出力**

代入値管理で登録した変数の具体値はホスト変数ファイルへ出力されます。  
Ansible-Legacyと Ansible-Pioneerでは、作業実行時にPlaybookまたは対話ファイルで使用している  
変数の具体値が代入値管理に登録 されていないと作業実行が想定外エラーとなります。  
Ansible-Legacy Roleでは、代入値管理で具体値を登録した変数のみが作業実行時にホスト変数ファイルへ出力されます。多段変数も同様で具体値を登録しているメンバー変数のみとなります。  
Exp)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **変数定義**  VAR\_users:  - name: alice  authorized:  - /tmp/alice/onekey.pub  mysql:  password: mysql-password  hosts:  - "127.0.0.1"  - "localhost"  - name: bob  略 | **代入値管理の登録** | | | | |
| ホスト | 変数 | メンバー変数 | 具体値 | 代入順序 |
| HOST\_A | VAR\_users: | [0].name | value1 |  |
| HOST\_A | VAR\_users | [1].authorized | value2 |  |
| **HOST\_Aのホスト変数ファイルへの出力内容**  VAR\_users:  - name :value1  - .authorized: value2 | | | | |

1. **デフォルト値チェックオプション**

複数ロール間でデフォルト値が一致していない変数に対して具体値の登録した場合に、警告メッセージを表示して登録させないパラメータを「astroll管理コンソール システム設定」で設定することが出来ます。このパラメータはデフォルトでは未登録です。必要に応じて登録して下さい。

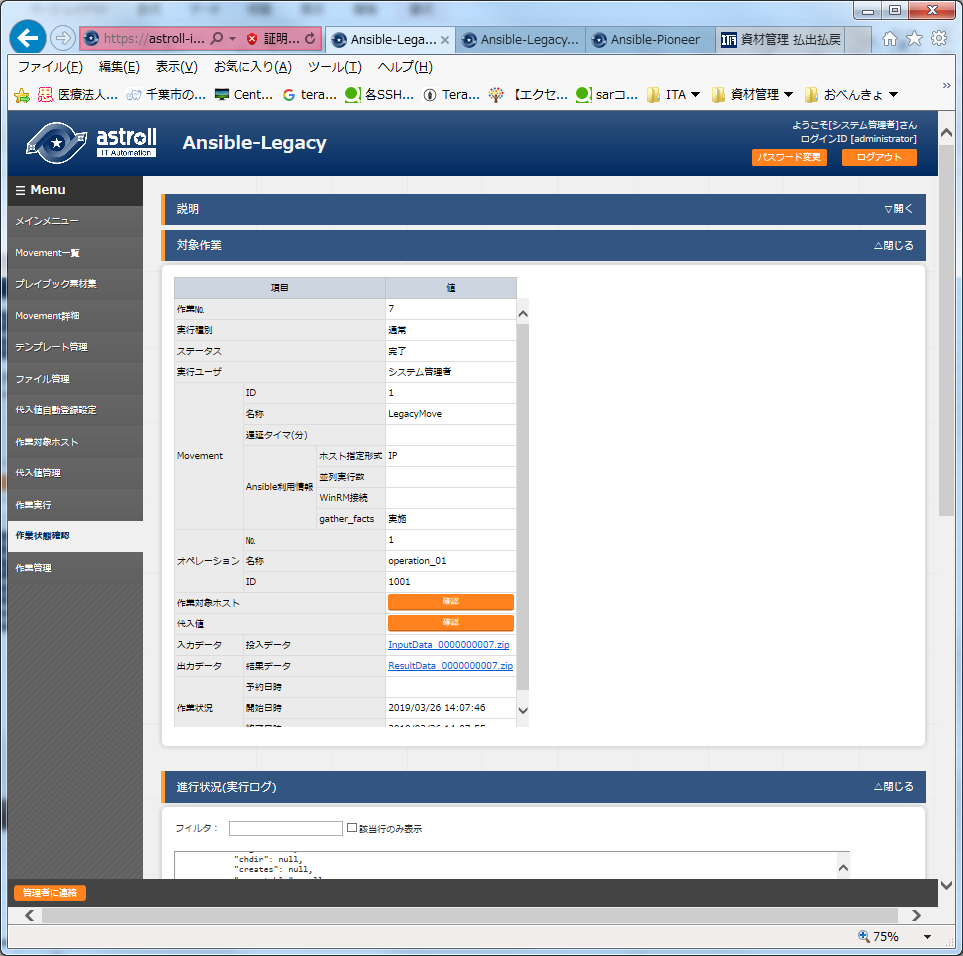
システム設定に登録する内容は以下の通りです。尚、システム設定については「利用手順マニュアル\_ astroll\_管理コンソール」を参照下さい。

**表 5.3‑7-2　システム設定登録内容**

| **項目** | **入力値** | **入力**  **必須** |
| --- | --- | --- |
| 識別ID | ANSIBLE\_DEF\_VAL\_CHK | ○ |
| 項目名 | 任意の文字列 | - |
| 設定値 | 1： パラメータ有効  1以外またはレコード未登録 ： パラメータ無効 | ○ |
| 備考 | 任意の文字列 |  |

### 作業状態確認

1. 作業の実行状態を監視します。



**図 5.3‑17サブメニュー画面（作業状態確認）**

1. **実行状態表示**実行状況に即し、「ステータス」が表示されます。

また、実行ログ、エラーログに実行状況の詳細が表示されます。

「実行種別」には、ドライランの場合は「ドライラン」、それ以外は「通常」が表示されます。  
ステータスが想定外エラーで終了した場合、Webコンテンツの登録不備が原因であれば、エラーログにメッセージが表示されます。  
また、「5.2.1インタフェース情報」の登録不備等で、Ansible RestAPIとの通信に失敗した場合にはエラーログにメッセージが表示されません。この場合は、アプリケーションログにエラー情報が記録されます。必要に応じてアプリケーションログを確認ください。

「実行ユーザ」には、作業実行メニューより「実行」ボタンまたは「ドライラン」ボタンを押下した際のログインユーザが表示されます。

1. **作業対象ホスト確認**

「確認」ボタンで「[5.4.11 作業対象ホスト](#_作業対象ホスト)」が表示され、作業対象のオペレーションとMovementに絞り込んだホストが表示されます。

1. **代入値確認**

「確認」ボタンで「[5.4.12代入値管理](#_代入値管理)」が表示され、作業対象のオペレーションとMovementに絞り込んだ代入値が表示されます。

1. **緊急停止/予約取り消し**

「緊急停止」ボタンで構築作業を停止させることができます。

また、実行前の「予約実行」の作業の場合は、「予約取消」ボタンが表示されます。「予約取消」ボタンで予約実行が取り消せます。

1. **ログ検索**実行ログ、エラーログは、フィルタリングができます。各ログのフィルタのテキストボックスに検索したい文字列を入力し、「該当行のみ表示」のチェックボックスをチェックすることで該当する行だけが表示されます。  
   実行ログ、エラーログのリフレッシュ表示間隔と最大表示行数を、「5.2.1インタフェース情報」の「状態監視周期（単位ミリ秒）」と「進行状態表示行数」で設定できます。
2. **投入データ**実行したPlaybookなどをダウンロードすることができます。
3. **結果データ**実行ログ、エラーログなどをダウンロードすることができます。

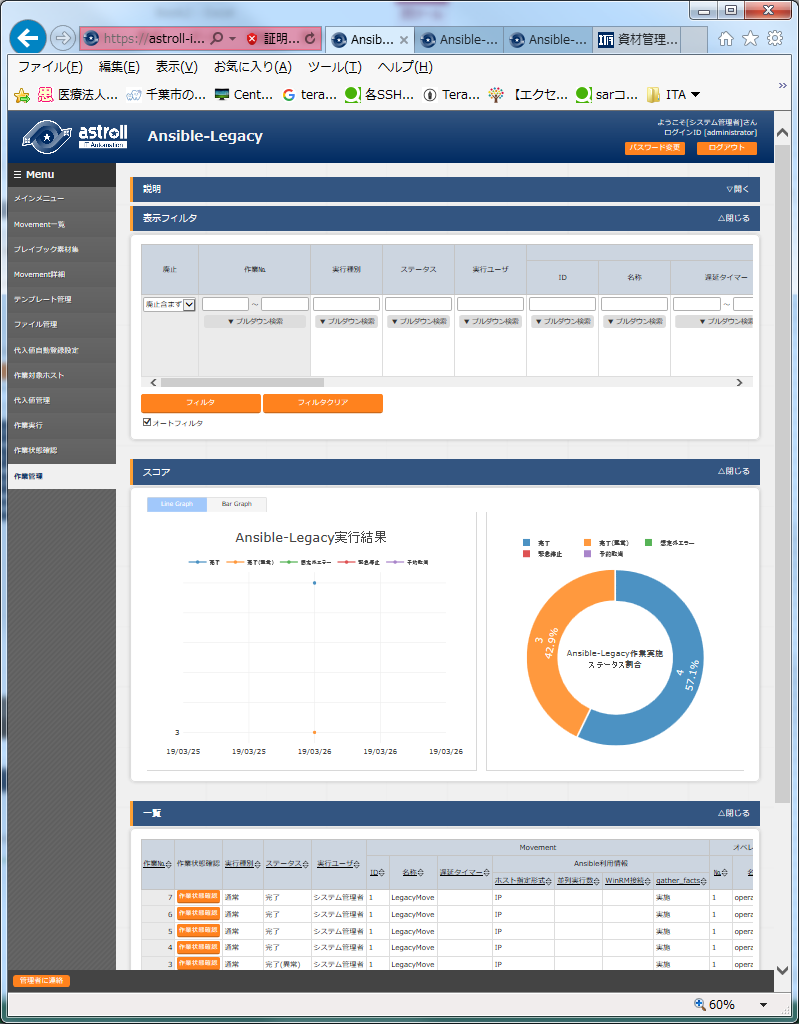
### 作業管理

1. 作業の履歴を閲覧できます。

条件を指定し「フィルタ」ボタンをクリックすると、作業一覧テーブルとグラフを表示します。

それぞれのグラフにマウスカーソルを合わせると、グラフのダウンロードボタンが右上に表示されます。表示されたグラフのダウンロードボタンをクリックすることで、グラフのダウンロードが可能です。

「作業状態確認」ボタンで、「5.3.13作業状態確認」に遷移し、実行状態の詳細を見ることができます。



**図 5.3‑18サブメニュー画面（作業管理）**

### 作業実行

1. 作業の実行を指示します。Movement一覧、オペレーション一覧からそれぞれラジオボタンで選択し、実行ボタンを押すと、「5.3.13作業状態確認」に遷移し、実行されます。



**図 5.3‑19サブメニュー画面（作業実行）**

※画面はAnsible-Legacyのものです。

1. **ドライラン**「ドライラン」ボタンをクリックすると、実際に対象機器に対して構築作業をせず、ドライランを行うことができます。ドライランを行った場合の、モード毎の動作は以下のとおりです。

|  |  |
| --- | --- |
| Driver | 動作 |
| Ansible-Legacy | Ansible-Playbookコマンドの—checkパラメータを指定しPlaybookを実行します。 |
| Ansible-Legacy Role | Ansible-Playbookコマンドの—checkパラメータを指定しroleを実行します。 |
| Ansible-Pioneer | 対象機器への接続チェックのみを行います。 |

1. **予約日時の指定**  
   「予約日時」を入力することで、実行を予約することがきます。  
   「予約日時」には、未来の日時のみ登録可能です。

# 構築コード記述方法

## Playbook（Ansible-Legacy）の記述

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

## 対話ファイル（Ansible-Pioneer）の記述

対話ファイルのAnsible-Pioneerではastroll独自モジュールをAnsibleに組込んでいます。

対話ファイルはastroll独自書式となります。

文字コードは、UTF-8で作成してください。

1. 対話ファイルの構成

対話ファイルは2種類のセクションにより構成されます。

|  |  |
| --- | --- |
| セクション名 | 用途 |
| Conf | timeoutパラメータによりタイムアウト値を指定します。  タイムアウト値:1～3600(単位:秒) |
| exec\_list | 3種類の対話コマンドにより作業対象ホストの構築を行います。 |

対話ファイルの先頭にtimeoutパラメータを記述。以降に対話コマンドを記述します。

コメントはAnsibleの基本書式と同様の記述が出来ます。

Exp)

# コメント

conf:

△△timeout: 10

exec\_list:

**※△:半角スペース**

**timeout:の記述の前に半角スペース2文字を付与してください。**

1. 対話コマンド

対話コマンドは以下の4種類があります。

|  |  |
| --- | --- |
| モジュール | 用途 |
| exec | 作業対象ホストにコマンドを投入します。 |
| expect | 作業対象ホストが標準出力に出力する内容より、期待する文字列(プロンプト)の出力を待ち合せます。 |
| state | 作業対象ホストにコマンドを投入し、標準出力にプロンプトを出力するまでの標準出力の内容を外部Shellで解析し結果判定をします。 |
| command | 作業対象ホストにコマンドを投入する前後において、繰り返しや条件分岐を行うことができます。 |

1. expectモジュール

作業対象ホストが標準出力に出力する内容より、期待する文字列(プロンプト)の出力を待ち合せます。期待する文字列は正規表記で記述できます。

期待する文字列を受取ると次へ進みます。また、timeoutパラメータで指定された時間内に受取れない場合は対話ファイルを異常終了します。

Exp)　　telnet接続でパスワード入力のプロンプトを待ち合せます。

△△-△expect:△’Password’

**※△:半角スペース**

**- expect: の記述の前に半角スペース2文字を付与してください。**

**待ち合わせる文字列をコーテーションで囲むことを推奨します。**

1. execモジュール

作業対象ホストにコマンドを投入します。

execモジュールとexpectモジュールは対で使用します。

Exp) telnet接続でパスワード入力のプロンプトを待ち合せてパスワードを投入します。

△△-△expect:△’Password’

△△△△exec:△itapassword

**※△:半角スペース**

**- exec: の記述の前に半角スペース4文字を付与してください。**

必要に応じコーテーションで囲むことを推奨します。

1. stateモジュール

作業対象ホストにコマンドを投入し、標準出力にプロンプトを出力するまでの標準出力の内容を外部Shellで解析し結果判定をします。

stateモジュールの書式

| パラメータ | 必須/  任意 | 説明 |
| --- | --- | --- |
| △△－△state:△xxx | 必須 | 投入するコマンドを指定します。 |
| △△△△prompt:△xxx | 必須 | 待受けプロンプトを指定します。正規表記で記述できます。 |
| △△△△shell:△xxx | 任意 | 作成したshellで結果を確認する場合に、shellファイル名を指定します。  作成したshellのexitコードが0の場合は正常、他は異常と判定します。  デフォルトのshellで結果を確認する場合、本パラメータは不要となります。デフォルトのshellはparameter(-)で指定された文字列で標準出力の内容をgrepします。マッチする行が1行でもあれば正常とし、マッチする行がなければ異常と判定します。また、parameterを指定しなかった場合、異常と判定されます。コマンドの結果(標準出力)をstdout\_fileで指定したファイルに退避したい目的で使用する場合、ignore\_errorsで「yes」を指定して下さい。 |
| △△△△parameter:  △△△△△△-△xxx  △△△△△△-△xxx | 任意 | 投入するコマンドの結果(標準出力)を検索する文字列を指定します。  shellを指定している場合、作成したshellの実行時パラメータとなります。複数ある場合は検索文字列を列挙します。 |
| △△△△stdout\_file:△xxx | 任意 | 投入するコマンドの結果(標準出力)を退避するファイルです。  shell:を指定している場合は、このファイルからコマンドの結果を取得してください。 |
| △△△△success\_exit:△ xxx | 任意 | 検索結果が正常の場合で対話ファイルを正常終了する場合に「yes」を指定します。「no」の場合は正常の場合は次に進みます。デフォルトは「no」。 |
| △△△△ignore\_errors:△ xxx  ※△:半角スペース | 任意 | 検索結果が異常でも次に進む場合に「yes」を指定します。  「no」の場合は、異常の場合に対話ファイルを異常終了とします。デフォルトは「no」。 |

Exp3)

hostsファイルをcatし、表示結果をparameter値でgrepしている。139.0.0.1、lalhostを含む行あれば正常と判定し次に進みます。行がなければ異常と判定しますがignore\_errors: yes の設定により次に進みます。

exec\_list:

- state: cat /etc/hosts

prompt: root@{{ \_\_loginhostname\_\_ }}

parameter:

- 139.0.0.1

- lalhost

ignore\_errors: yes

Exp2)

hostsファイルをcatし、表示結果をparameter値でgrepしている。139.0.0.1、lalhostを含む行あれば正常と判定しますがsuccess\_exit: yesの設定により対話ファイルを正常終了します。行がなければ異常と判定し対話ファイルを異常終了します。

exec\_list:

- state: cat /etc/hosts

prompt: root@{{ \_\_loginhostname\_\_ }}

parameter:

- 139.0.0.1

- lalhost

success\_exit: yes

Exp1)

hostsファイルをcatし、表示結果をparameter値でgrepしている。139.0.0.1、lalhostを含む行あれば正常と判定し次に進みます。行がなければ異常と判定し対話ファイルを異常終了します。

exec\_list:

- state: cat /etc/hosts

prompt: root@{{ \_\_loginhostname\_\_ }}

parameter:

- 139.0.0.1

- lalhost

Exp5)

hostsファイルをcatし、表示結果をstdout\_file で指定したファイルに保存し次に進みます。

デフォルトのshellはparameterの設定がないと異常と判定します。次に進める為にignore\_errors: yesを設定します。

exec\_list:

- state: cat /etc/hosts

prompt: root@{{ \_\_loginhostname\_\_ }}

stdout\_file: {{ \_\_symphony\_workflowdir\_\_ }}/hosts

ignore\_errors: yes

Exp4)

hostsファイルをcatし、ユーザー作成のshellで表示結果をparameter値でgrepしている。139.0.0.1、lalhostを含む行あれば正常と判定し次に進みます。行がなければ異常と判定し対話ファイルを異常終了します。

exec\_list:

- state: cat /etc/hosts

prompt: root@{{ \_\_loginhostname\_\_ }}

shell: /tmp/grep.sh

stdout\_file: /tmp/stdout.txt

parameter:

- 139.0.0.1

- lalhost

ユーザー作成 shell(/tmp/grep.sh)

#!/bin/bash

STDOUT=/tmp/STDOUT.tmp

STDERR=/tmp/STDERR.tmp

cat /tmp/stdout.txt|grep $1|grep $2 | wc -l >${STDOUT} 2>${STDERR}

RET=$?

if [ $RET -ne 0 ]; then

EXIT\_CODE=$RET

else

if [ -s ${STDERR} ]; then

EXIT\_CODE=1

else

CNT=`cat ${STDOUT}`

if [ ${CNT} -eq 0 ]; then

EXIT\_CODE=1

else

EXIT\_CODE=0

fi

fi

fi

/bin/rm -rf ${STDOUT} ${STDERR} >/dev/null 2&>1

exit ${EXIT\_CODE}

1. commandモジュール

作業対象ホストにコマンドを投入する前後において、繰り返しや条件分岐を行うことができます。

commandモジュールの書式

| パラメータ | 必須/  任意 | 説明 |
| --- | --- | --- |
| △△－△command:△xxx | 必須 | 投入するコマンドを指定します。 |
| △△△△prompt:△xxx | 必須 | 待受けプロンプトを指定します。正規表記で記述できます。 |
| △△△△timeout:△xxx | 任意 | コマンドを送ってからのプロンプト待ちタイマを指定します。  省略されている場合は、conf->timeoutを使用します。 |
| △△△△register:△xxx | 任意 | コマンドを送信後に標準出力の情報を退避する任意の文字列です。  with\_itemsでループしている場合は、最後のコマンド送信後の標準出力の情報を退避。この変数で条件判定が出来ます。(使用できるのは条件判定のみ)  ただし、変数名ごとに標準出力の情報を退避することはできません。  前の情報を上書きします。 |
| △△△△with\_items:  △△△△△△-△xxx  △△△△△△-△xxx | 任意 | with\_itemsに設定する変数は複数具体値変数です。  with\_itemsに設定する変数の具体値数は同じでなくても良いです。  同じでない場合は、各変数の具体値数の最大値数でループします。  具体値が不足している変数の具体値は空として扱います。  ただし、promptまたはtimeoutの場合は上記に当てはまりません。  prompt、timeoutで具体値が不足していると、エラーになります。  各変数のスコープはitem.X(Xは0から99)とします。  変数(item.X)の適用範囲はregister/when以外です。  prompt、timeoutでwith\_itemsを利用する場合の変数名は下記の通りにしてください。  prompt： {{△VAR\_prompt\_XXX△}}  timeout： {{△VAR\_timeout\_XXX△}}  (△は半角スペース。XXXは任意の半角英数字とアンダースコア) |
| △△△△when:  △△△△△△-△xxx  △△△△△△-△xxx | 任意 | command実行前の条件判定です。  条件にマッチしてcommand実行します。  条件にマッチしていなければ次のcommand行に移ります。  条件式  変数定義判定  VAR\_xx is define 変数が定義されている true  VAR\_xx is undefine 変数が未定義 true  ※define/undefineはastrollの変数(VAR\_xx)のみ指定可能  変数具体値判定  VAR\_xx/register変数 比較演算子 文字列  VAR\_xx/register変数 比較演算子 VAR\_xx  VAR\_xx/register変数 match(正規表記文字列/VAR\_xx)  VAR\_xx/register変数 no match(正規表記文字列/VAR\_xx)  ※比較演算子は「==」、「!=」、「>」、「>=」、「<」、「<=」  ※比較演算子の「>」、「>=」、「<」、「<=」は数値を想定しています。  and/orによる複合条件 |
| △△△△exec\_when:  △△△△△△-△xxx  △△△△△△-△xxx | 任意 | ループ毎の条件判定です。(continue条件)  条件にマッチしていれば該当ループのcommandを実行します。  マッチしていなければ次のループへ移ります。  条件式  変数定義判定  VAR\_xx is define 変数が定義されている true  VAR\_xx is undefine 変数が未定義 true  ※define/undefineはastrollの変数(VAR\_xx)のみ指定可能  変数具体値判定  VAR\_xx/register変数 比較演算子 文字列  VAR\_xx/register変数 比較演算子 VAR\_xx  VAR\_xx/register変数 match(正規表記文字列/VAR\_xx)  VAR\_xx/register変数 no match(正規表記文字列/VAR\_xx)  ※比較演算子は「==」、「!=」、「>」、「>=」、「<」、「<=」  ※比較演算子の「>」、「>=」、「<」、「<=」は数値を想定しています。  and/orによる複合条件 |
| △△△△failed\_when:  △△△△△△-△xxx  △△△△△△-△xxx  ※△:半角スペース | 任意 | command実行後(ループ毎)のstdoutの内容に対する  条件判定です。  条件にマッチしていれば正常とします。  マッチしていなければ異常とし、対話ファイルを異常終了させます。  条件式  変数具体値判定  stdout 比較演算子 文字列  stdout 比較演算子 VAR\_xx  stdout match(正規表記文字列/VAR\_xx)  stdout no match(正規表記文字列/VAR\_xx)  ※比較演算子は「==」、「!=」、「>」、「>=」、「<」、「<=」  ※比較演算子の「>」、「>=」、「<」、「<=」は数値を想定しています。  and/orによる複合条件 |

Exp1)

conf:

timeout: 30

exec\_list:

# プロンプト以外の文字列で待合せが必要な場合は、expect/execの組合せでする。

# パスワードが必要な場合

- expect: 'password:'

exec: '{{ \_\_loginpassword\_\_ }}'

# VAR\_hosts\_makeというastroll変数がホスト変数ファイルに記載されている場合、

# hostsファイルをcatします。記載されていない場合は、スキップします。

- command: cat /etc/hosts

prompt: root@{{ \_\_loginhostname\_\_ }}

when:

- VAR\_hosts\_make is define

Exp2)

conf:

timeout: 30

exec\_list:

# プロンプト以外の文字列で待合せが必要な場合は、expect/execの組合せでする。

# パスワードが必要な場合

- expect: 'password:'

exec: '{{ \_\_loginpassword\_\_ }}'

# VAR\_hosts\_makeというastroll変数がホスト変数ファイルに記載されている場合、

# hostsファイルをcatします。記載されていない場合は、スキップします。

# catにより、標準出力されたhostsファイルの内容をresult\_stdoutに退避します。

- command: cat /etc/hosts

prompt: root@{{ \_\_loginhostname\_\_ }}

register: result\_stdout

when:

- VAR\_hosts\_make is define

# VAR\_hosts\_makeというastroll変数がホスト変数ファイルに記載されている場合、

# command実行します。記載されていない場合は、スキップします。

# with\_itemsの複数具体値変数に設定されている具体値数分command実行します。

# ループ毎の条件判定として、hostsファイルに「ipアドレス ホスト名」が該当しない場合

# command実行します。

# hostsファイルの最終行にechoによる、「IPアドレス ホスト名」を追記します。

- command: 'echo {{ item.0 }} {{ item.1 }} >> /etc/hosts'

prompt: 'root@{{ \_\_loginhostname\_\_ }}'

when:

- VAR\_hosts\_make is define

with\_items:

- '{{ VAR\_hosts\_ip }}' # item.0

- '{{ VAR\_hosts\_name }}' # item.1

exec\_when:

- result\_stdout no match({{ item.0 }} \*{{ item.1 }})

Exp3)

conf:

timeout: 30

exec\_list:

# プロンプト以外の文字列で待合せが必要な場合は、expect/execの組合せでする。

# パスワードが必要な場合

- expect: 'password:'

exec: '{{ \_\_loginpassword\_\_ }}'

# with\_itemsの複数具体値変数に設定されている具体値数分command実行します。

# 自動起動設定を実行します。

- command: 'systemctl enable {{ item.0 }}'

prompt: 'root@{{ \_\_loginhostname\_\_ }}'

with\_items:

- '{{ VAR\_service\_name\_list }}' # item.0

# with\_itemsの複数具体値変数に設定されている具体値数分command実行します。

# サービスの起動を実行します。

- command: 'systemctl start {{ item.0 }}'

prompt: 'root@{{ \_\_loginhostname\_\_ }}'

with\_items:

- '{{ VAR\_service\_name\_list }}' # item.0

# with\_itemsの複数具体値変数に設定されている具体値数分command実行します。

# サービスのステータスを標準出力します。

# 標準出力された結果の内容に、item.1の正規表現がある場合、正となります。

# 例えば、VAR\_service\_status\_listの具体値をrunningと設定し、サービスが起動している場合、

# 「Active: active (running)」のrunnigが一致するので正となります。(次のループに移ります)

# そうでない場合は、異常と判断し、対話ファイルは異常終了となります。

- command: 'systemctl status {{ item.0 }}'

prompt: 'root@{{ \_\_loginhostname\_\_ }}'

with\_items:

- '{{ VAR\_service\_name\_list }}' # item.0

- '{{ VAR\_service\_status\_list }}' # item.1

failed\_when:

- stdout match({{ item.1 }})

Exp4)

conf:

timeout: 30

exec\_list:

# プロンプト以外の文字列で待合せが必要な場合は、expect/execの組合せでする。

# パスワードが必要な場合

- expect: 'password:'

exec: '{{ \_\_loginpassword\_\_ }}'

# with\_itemsの複数具体値変数に設定されている具体値数分command実行します。

# commandに「{{ item.0 }}」のみの記述をする場合は、ダブルクォーテーションで囲みます。

# promptやtimeoutでwith\_itemsを利用する場合、具体値数に注意が必要です。

# prompt→command→prompt→command→prompt ・・・(以下ループ)となり、command数+1

# 設定する必要があります。(timeoutも同様)

- command: "{{ item.0 }}"

prompt: '{{ item.1 }}'

timeout: '{{ item.2 }}'

with\_items:

- '{{ VAR\_command\_list }}' # item.0

- '{{ VAR\_prompt\_list }}' # item.1

- '{{ VAR\_timeout\_list }}' # item.2

Exp5)

conf:

timeout: 30

exec\_list:

# プロンプト以外の文字列で待合せが必要な場合は、expect/execの組合せでする。

# パスワードが必要な場合

- expect: 'password:'

exec: '{{ \_\_loginpassword\_\_ }}'

# with\_itemsの複数具体値変数に設定されている具体値数分command実行します。

# 代入値管理の具体値に{{ item.X }} を設定することができます。その際は対話ファイルに記載する

# item.Xより具体値に記載するitem.Xの数値が大きくなるようにしてください。

# 今回の例で実行するcommandは「systemctl status ky\_pioneer\_execute-workflow.service」

- command: "{{ item.0 }}"

prompt: 'root@{{ \_\_loginhostname\_\_ }}'

with\_items:

- '{{ VAR\_command\_list }}' # item.0

- '{{ VAR\_service\_name\_list }}' # item.1

|  |  |
| --- | --- |
| 変数名 | 具体値 |
| VAR\_command\_list | systemctl status {{ item.1 }} |
| VAR\_service\_name\_list | ky\_pioneer\_execute-workflow.service |

and条件

Exp6)

conf:

timeout: 30

exec\_list:

# プロンプト以外の文字列で待合せが必要な場合は、expect/execの組合せでする。

# パスワードが必要な場合

- expect: 'password:'

exec: '{{ \_\_loginpassword\_\_ }}'

# and/orによる複合条件の記述例です。

# or条件を行いたい場合、if文を横に記述することができます。

# and条件を行いたい場合、複数行に分けて記述するとand条件になります。

# 今回、whenを例にしていますが、exec\_when、failed\_whenも同様です。

- command: echo aaa

prompt: 'root@{{ \_\_loginhostname\_\_ }}'

when:

- 10 == 9 OR 10 != 9 OR 10 >= 9

- 10 > 9 OR 10 <= 9 OR 10 < 9

or条件

1. 正規表記

下記のコマンド及びパラメータに記述された文字列は正規表記で評価されます。

・expectモジュール

・stateモジュールのpromptパラメータ

・commandモジュールのpromptパラメータ

ですので正規表記での記述が可能です。また、記述された文字列にメタ文字「(){}.など」を含む場合、メタ文字の前にエスケープ文字「\」を挿入する必要があります。

Exp1)

以下のような文字列を待ち受ける場合、赤字がメタ文字となります。

XAMPP Developer Files **[**Y**/**n] exec\_list:

　メタ文字の前にエスケープ文字「\」を挿入する必要があります。

　　XAMPP Developer Files \**[**Y\**/**n\] exec\_list:

## ロールパッケージ（Ansible-Legacy Role）の記述

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

## astrollreadme（Ansible-Legacy Roleのみ）の記述

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

## 読替表（Ansible-Legacy Roleのみ）の記述

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

## BackYardコンテンツ

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

1. **作業インスタンス履歴削除**

astroll基本コンソールの投入オペレーション一覧に登録されてるオペレーションで実施予定日が一定

期間経過しているオペレーションIDを「5.3.11作業対象ホスト」と「5.3.12代入値管理」で使用している

レコードがある場合、経過日数に応じて削除(物理削除または廃止)します。

下記ファイルに経過日数を登録しています。この日数に応じてレコードが削除(物理削除または廃止)さ

れます。

●経過日数登録ファイル

~/ita-root/confs/backyardconfs/ansible\_driver/keep\_day\_length.txt

●ファイルフォーマット

p1,p2　　　　　p1:廃止までの日数 p2:物理削除までの日数

Exp)

廃止までの日数: 30日　　物理削除までの日数: 　60日の場合

30,60

## Ansible利用ガイドラインastroll追加ルール

詳細は、関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』をご参照下さい。

# 運用上の注意点

astrollシステムを活用する操作はクライアントPCのブラウザ画面からのユーザー利用による入力だけでは無く、システム運用・保守による操作もあります。

用意している運用・保守の操作は次のとおりです。

●ログレベルの変更

●メンテナンス

## ログレベルの変更

astrollシステム のプロセスのログレベルの変更方法は次のとおりです。  
ログレベルを変更できるプロセスには、通常の独立監視プロセスと、RestAPIのプロセスがあります。

1. **通常の独立監視プロセス**
   1. RHEL 6.xの場合の対象ファイル

~/ita-root/backyards/ansible\_driver/

(Ansible操作)

ky\_legacy\_checkcondition-workflow  
ky\_legacy\_execute-workflow  
ky\_legacy\_role\_checkcondition-workflow

ky\_legacy\_role\_execute-workflow

ky\_pioneer\_checkcondition-workflow

ky\_pioneer\_execute-workflow

(変数自動登録)

ky\_legacy\_varsautolistup-workflow  
ky\_legacy\_role\_varsautolistup-workflow

ky\_pioneer\_varsautolistup-workflow

(代入値自動登録設定)

ky\_legacy\_valautostup-workflow

ky\_legacy\_role\_valautostup-workflow

ky\_pioneer\_valautostup-workflow

(構成管理DB連携)

ky\_ansible\_cmdbmenuanalysis-workflow

* + 上記ファイルへのリンクファイルが /etc/init.d に作成されています。これらは削除しないでください。

NORMALレベル

「LOG\_LEVEL='NORMAL'」を有効にします。

# ログ出力レベル

# DEBUG ：解析レベルでログ出力

# NORMAL：クリティカルな場合のみログ出力

#LOG\_LEVEL='DEBUG'

LOG\_LEVEL='NORMAL'

DEBUGレベル

「LOG\_LEVEL='DEBUG'」を有効にします。

# ログ出力レベル

# DEBUG ：解析レベルでログ出力

# NORMAL：クリティカルな場合のみログ出力

LOG\_LEVEL='DEBUG'

#LOG\_LEVEL='NORMAL'

* 1. RHEL 7.xの場合の対象ファイル

/etc/systemd/system/

(Ansible操作)  
ky\_legacy\_role\_execute-workflow.service

ky\_legacy\_role\_checkcondition-workflow.service

ky\_legacy\_execute-workflow.service

ky\_legacy\_checkcondition-workflow.service

ky\_pioneer\_execute-workflow.service

ky\_pioneer\_checkcondition-workflow.service

(変数自動登録)

ky\_legacy\_role\_varsautolistup-workflow.service

ky\_legacy\_varsautolistup-workflow.service  
ky\_pioneer\_varsautolistup-workflow.service

(代入値自動登録設定)

ky\_legacy\_role\_valautostup-workflow.service

ky\_legacy\_valautostup-workflow.service

ky\_pioneer\_valautostup-workflow.service

(構成管理DB連携)

ky\_ansible\_cmdbmenuanalysis-workflow

NORMALレベル

NORMALを設定にします。

ExecStart=/astroll/ita-root/backyards/common/ky\_loopcall-php-procedure.sh /usr/local/bin/php /usr/local/bin/php /astroll/ita-root/backyards/ansible\_driver/ky\_pioneer\_varsautolistup-workflow.php /astroll/ita-root/logs/backyardlogs 60 **NORMAL** > /dev/null 2>&1

DEBUGレベル

DEBUGを設定にします。

ExecStart=/astroll/ita-root/backyards/common/ky\_loopcall-php-procedure.sh /usr/local/bin/php /usr/local/bin/php /astroll/ita-root/backyards/ansible\_driver/ky\_pioneer\_varsautolistup-workflow.php /astroll/ita-root/logs/backyardlogs 60 **DEBUG** > /dev/null 2>&1

* + ログレベル変更は、プロセス再起動（Restart）後に有効になります。再起動については

次節「7.3メンテナンス方法について」を参照してください。

* 1. RHEL 6.xとRHEL 7.xで共通の対象ファイル

~/ita-root/backyards/ansible\_driver/ky\_ansible\_dataautoclean-workflow.sh

「②RHEL 6.xの場合の対象ファイル」の設定と同様です。

## 起動周期の変更

astrollシステム のプロセスの起動周期の変更方法は次のとおりです。

ただし、例外を除き起動周期はデフォルト値の使用をしてください。

起動周期を変更できるプロセスには、通常の独立監視プロセスのみです。

1. **通常の独立監視プロセス**
2. RHEL 6.xの場合の対象ファイル

~/ita-root/backyards/ansible\_driver/

(Ansible操作)

ky\_legacy\_checkcondition-workflow  
ky\_legacy\_execute-workflow  
ky\_legacy\_role\_checkcondition-workflow

ky\_legacy\_role\_execute-workflow

ky\_pioneer\_checkcondition-workflow

ky\_pioneer\_execute-workflow

(変数自動登録)

ky\_legacy\_varsautolistup-workflow  
ky\_legacy\_role\_varsautolistup-workflow

ky\_pioneer\_varsautolistup-workflow

(代入値自動登録設定)

ky\_legacy\_valautostup-workflow

ky\_legacy\_role\_valautostup-workflow

ky\_pioneer\_valautostup-workflow

(構成管理DB連携)

ky\_ansible\_cmdbmenuanalysis-workflow

上記ファイルへのリンクファイルが /etc/init.d に作成されています。これらは削除しないでください。

・起動周期の設定

INTERVALで起動周期を設定します。（単位:秒）

# 無限ループのインターバル(0以上を指定、180未満にすること)

INTERVAL=60

1. RHEL 7.xの場合の対象ファイル

/etc/systemd/system/

(Ansible操作)  
ky\_legacy\_role\_execute-workflow.service

ky\_legacy\_role\_checkcondition-workflow.service

ky\_legacy\_execute-workflow.service

ky\_legacy\_checkcondition-workflow.service

ky\_pioneer\_execute-workflow.service

ky\_pioneer\_checkcondition-workflow.service

(変数自動登録)

ky\_legacy\_role\_varsautolistup-workflow.service

ky\_legacy\_varsautolistup-workflow.service  
ky\_pioneer\_varsautolistup-workflow.service

(代入値自動登録設定)

ky\_legacy\_role\_valautostup-workflow.service

ky\_legacy\_valautostup-workflow.service

ky\_pioneer\_valautostup-workflow.service

(構成管理DB連携)

ky\_ansible\_cmdbmenuanalysis-workflow

・起動周期の設定

ExecStartの5番目のパラメータで設定します。（単位:秒）

ExecStart=/astroll/ita-root/backyards/common/ky\_loopcall-php-procedure.sh /usr/local/bin/php /usr/local/bin/php /astroll/ita-root/backyards/ansible\_driver/ky\_pioneer\_varsautolistup-workflow.php /astroll/ita-root/logs/backyardlogs **60** NORMAL > /dev/null 2>&1

## メンテナンス方法について

### Ansible driver 独立型プロセスの起動/停止/再起動

ky\_legacy\_checkcondition-workflowを例示します。

* + 1. RHEL 6.xの場合

●プロセス起動

＄ service ky\_legacy\_checkcondition-workflow start 

●プロセス停止

＄ service ky\_legacy\_checkcondition-workflow stop 

●プロセス再起動

＄ service ky\_legacy\_checkcondition-workflow restart 

* + 1. RHEL 7.xの場合

●プロセス起動

＄/usr/bin/systemctl start ky\_legacy\_checkcondition-workflow 

●プロセス停止

＄/usr/bin/systemctl stop ky\_legacy\_checkcondition-workflow 

●プロセス再起動

＄/usr/bin/systemctl restart ky\_legacy\_checkcondition-workflow 

同様に、各対象ファイル名に置き換えて起動/停止/再起動を行ってください。

# トラブルシューティング

関連マニュアルの『利用手順マニュアルastroll\_Ansible系ドライバー共通』にも、共通のQA事項について

記載をしています。そちらも併せてご参照ください。

|  |  |
| --- | --- |
| No | 内容 |
| Q-1 | 機器一覧のユーザーにrootユーザー以外を設定したホストに対してlegacyの作業実行をしたが、Ansibleのコマンドでエラーが発生する。 |
| A-1 | astrollでAnsibleのコマンドを実行する場合にsudoパラメータのデフォルトをyesに設定しています。  作業対象ホスト側で該当ユーザーに対するsudo権限を設定してください。  sudo権限を与えられない場合はsudoパラメータの設定をPlaybookに追記してください。 |
| Q-2 | 機器一覧のプロトコルをtelnetに設定したホストに対してpioneerの作業実行をしたが、作業対象ホストへの接続でエラーとなる。 |
| A-2 | 対話ファイルに記述しているパスワードおよびパスワードプロンプトの記述が正しいか確認してください。 |

# 付録

## Ansible実行時に使用される投入データとastrollメニューの紐づけ

各astrollメニューより情報を抜出してAnsible実行に必要な[投入データ]を作ります。

[投入データ]はZIP形式で[作業状態確認](#_作業状態確認)よりダウンロードが可能です。

各種データとastrollメニューの関係性は以下の通りです。

### Ansible-Legacy投入データ

【上位ディレクトリ】

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| **―** | child\_playbooks | ユーザーが作成したPlaybookが格納されているディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-Legacy | プレイブック素材集 | プレイブック素材 |
|  |  | Ansible-Legacy | Movement詳細 | インクルード順序 |
|  |  |  |  |  |
| ― | template\_files | 実行するPlaybook内で使用するtemplateファイルを格納するディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-Legacy | テンプレート管理 | テンプレート素材 |
|  |  | Ansible-Legacy | Movement詳細 | インクルード順序 |
|  |  |  |  |  |
| ― | copy\_files | 作業対象サーバに配置するファイルを格納されているディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-Legacy | ファイル管理 | ファイル素材 |
|  |  | Ansible-Legacy | Movement詳細 | インクルード順序 |
|  |  |  |  |  |
| ― | host\_vars | 作業実行の対象となるホストの情報と各種変数の定義ファイルが 格納されるディレクトリ | | |
|  |  |  | | |
|  |  | Ansible共通 | グローバル変数管理 | 変数名/具体値 |
|  |  | Ansible-Legacy | 代入値管理 | 変数名/具体値 |
|  |  | Ansible-Legacy | template管理 | テンプレート素材 |
|  |  | Ansible-Legacy | ファイル管理 | ファイル変数名 |
|  |  | Ansible-Legacy | Movement詳細 | インクルード順序 |
|  |  | Ansible-Legacy | インターフェース情報 | データリレイストレージパス(astroll) |
|  |  | Ansible-Legacy | インターフェース情報 | Symphonyインスタンスデータリレイストレージパス(ANS) |
|  |  | 基本コンソール | 機器一覧 | プロトコル |
|  |  | 基本コンソール | 機器一覧 | ログインユーザID |
|  |  | 基本コンソール | 機器一覧 | ログインパスワード |
|  |  | 基本コンソール | 機器一覧 | ホスト名 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| ― | ssh\_key\_files | 認証方式が鍵方式の場合、指定したssh認証鍵ファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | ssh認証鍵ファイル |
|  |  |  |  |  |
| ― | winrm\_ca\_files | WinRM接続する場合、接続情報を定義したファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | WinRM接続情報 |
|  |  |  |  |  |
|  | AnsibleExecOption.txt | AnsiblePlaybook実行時のパラメータ | | |
|  |  |  | | |
|  |  | Ansible共通 | インターフェース情報 | オプションパラメータ |
|  |  | Ansible-Legacy | Movement一覧 | 並列実行数 |
|  |  |  |  |  |
|  | hosts | 作業実行の対象となるホストの情報が記載されているファイル | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | ホスト名 |
|  |  | 基本コンソール | 機器一覧 | IPアドレス |
|  |  | 基本コンソール | 機器一覧 | ログインユーザID |
|  |  | 基本コンソール | 機器一覧 | ログインパスワード |
|  |  | 基本コンソール | 機器一覧 | 接続オプション |
|  |  |  |  | ※ansible\_ssh\_extra\_argsのパラメータ |
|  |  | 基本コンソール | 機器一覧 | ssh認証鍵ファイル |
|  |  | 基本コンソール | 機器一覧 | サーバー証明書 |
|  |  | 基本コンソール | 機器一覧 | インベントリファイル追加オプション |
|  |  |  |  |  |
|  | playbook.yml | Playbookやホストの情報を全て呼び出し、Ansibleを実行するファイル。 | | |
|  |  |  | | |
|  |  | Ansible-Legacy | プレイブック素材集 | プレイブック素材 |
|  |  | Ansible-Legacy | Movement詳細 | インクルード順序 |
|  |  | Ansible-Legacy | Movement詳細 | gather\_facts |
|  |  |  |  |  |

投入データを用いて以下のコマンドでAnsibleを直接実行することも可能です。

ansible-playbook (オプション) –i hosts playbook.yml

### Ansible-Pioneer投入データ

【上位ディレクトリ】

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| ― | template\_files | 実行するPlaybook内で使用するtemplateファイルを格納するディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-Pioneer | テンプレート管理 | テンプレート素材 |
|  |  | Ansible-Pioneer | Movement詳細 | インクルード順序 |
|  |  |  |  |  |
| ― | copy\_files | 作業対象サーバに配置するファイルを格納されているディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-Pioneer | ファイル管理 | ファイル素材 |
|  |  | Ansible-Pioneer | Movement詳細 | インクルード順序 |
|  |  |  |  |  |
| ― | ssh\_key\_files | 認証方式が鍵方式の場合、指定したssh 認証鍵ファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | ssh認証鍵ファイル |
|  |  |  |  |  |
| ― | winrm\_ca\_files | WインRM接続する場合は接続情報を定義したファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | WinRM接続情報 |
|  |  |  |  |  |
| ― | host\_vars | 作業実行の対象となるホストの情報と各種変数の定義ファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | Ansible共通 | インターフェース情報 | データリレイストレージパス(astroll) |
|  |  | Ansible共通 | インターフェース情報 | Symphonyインスタンスデータリレイストレージパス(ANS) |
|  |  | Ansible共通 | グローバル変数管理 | 変数名/具体値 |
|  |  | Ansible-Pioneer | 代入値管理 | 変数名/具体値 |
|  |  | Ansible-Pioneer | template管理 | テンプレート素材 |
|  |  | Ansible-Pioneer | Movement詳細 | インクルード順序 |
|  |  | Ansible-Pioneer | ファイル管理 | ファイル変数名 |
|  |  | Ansible-Pioneer | Movement詳細 | インクルード順序 |
|  |  | 基本コンソール | 機器一覧 | ログインパスワード |
|  |  | 基本コンソール | 機器一覧 | ホスト名 |
|  |  | 基本コンソール | 機器一覧 | 接続オプション |
|  |  | 基本コンソール | 機器一覧 | プロトコル |
|  |  | 基本コンソール | 機器一覧 | ログインユーザID |
|  |  |  |  |  |
| ― | dialog\_files | ユーザーが作成したPlaybookが格納されているディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-Pioneer | 対話ファイル素材集 | 対話ファイル素材 |
|  |  | Ansible-Pioneer | Movement詳細 | インクルード順序 |
|  |  |  | | |
|  | AnsibleExecOption.txt | AnsiblePlaybook実行時のパラメータ | | |
|  |  |  |  |  |
|  |  | Ansible共通 | インターフェース情報 | オプションパラメータ |
|  |  |  |  |  |
|  | hosts | 作業実行の対象となるホストの情報が記載されているファイル | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | ホスト名 |
|  |  | 基本コンソール | 機器一覧 | IPアドレス |
|  |  | 基本コンソール | 機器一覧 | ログインユーザID |
|  |  | 基本コンソール | 機器一覧 | ログインパスワード |
|  |  | 基本コンソール | 機器一覧 | ssh認証鍵ファイル |
|  |  | 基本コンソール | 機器一覧 | インベントリファイル追加オプション |
|  |  |  |  |  |
|  | playbook.yml | Playbookやホストの情報を全て呼び出し、Ansibleを実行するファイル。 | | |
|  |  |  | | |
|  |  | Ansible-Pioneer | インターフェース情報 | データリレイストレージパス(astroll) |

投入データを用いて以下のコマンドでAnsibleを直接実行することも可能です。

ansible-playbook (オプション) –i hosts playbook.yml

### Ansible-LegacyRole投入データ

【上位ディレクトリ】

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ― | copy\_files | 作業対象サーバに配置するファイルを格納されているディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-LegacyRole | ファイル管理 | ファイル素材 |
|  |  | Ansible-LegacyRole | Movement詳細 | インクルード順序 |
|  |  |  |  |  |
| ― | roles | ユーザーが作成したroleが格納されているディレクトリ | | |
|  |  |  | | |
|  |  | Ansible-LegacyRole | ロールパッケージ管理 | ロールパッケージファイル(ZIP形式) |
|  |  |  |  |  |
| ― | ssh\_key\_files | 認証方式が鍵方式の場合、指定したssh 認証鍵ファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | ssh認証鍵ファイル |
|  |  |  |  |  |
| ― | winrm\_ca\_files | WinRM接続する場合は接続情報を定義したファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | 基本コンソール | 機器一覧 | WinRM接続情報 |
|  |  |  |  |  |
| ― | host\_vars | 作業実行の対象となるホストの情報と各種変数の定義ファイルが格納されるディレクトリ | | |
|  |  |  | | |
|  |  | Ansible共通 | インターフェース情報 | データリレイストレージパス(astroll) |
|  |  | Ansible共通 | インターフェース情報 | Symphonyインスタンスデータリレイストレージパス(ANS) |
|  |  | Ansible共通 | グローバル変数管理 | 変数名/具体値 |
|  |  | Ansible-LegacyRole | 代入値管理 | 変数名/具体値 |
|  |  | Ansible-LegacyRole | template管理 | テンプレート素材 |
|  |  | Ansible-LegacyRole | Movement詳細 | インクルード順序 |
|  |  | Ansible-LegacyRole | ファイル管理 | ファイル変数名 |
|  |  | Ansible-LegacyRole | Movement詳細 | インクルード順序 |
|  |  | 基本コンソール | 機器一覧 | プロトコル |
|  |  | 基本コンソール | 機器一覧 | ログインユーザID |
|  |  | 基本コンソール | 機器一覧 | ログインパスワード |
|  |  | 基本コンソール | 機器一覧 | ホスト名 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | |  |
|  | AnsibleExecOption.txt | AnsiblePlaybook実行時のパラメータ | | | |
|  |  |  | | | |
|  |  | Ansible共通 | インターフェース情報 | | オプションパラメータ |
|  |  | Ansible-LegacyRole | Movement一覧 | | 並列実行数 |
|  |  |  |  | |  |
|  | hosts | 作業実行の対象となるホストの情報が記載されているファイル | | | |
|  |  |  | | | |
|  |  | 基本コンソール | 機器一覧 | | ホスト名 |
|  |  | 基本コンソール | 機器一覧 | | IPアドレス |
|  |  | 基本コンソール | 機器一覧 | | ログインユーザID |
|  |  | 基本コンソール | 機器一覧 | | ログインパスワード |
|  |  | 基本コンソール | 機器一覧 | | 接続オプション |
|  |  |  |  | ※ansible\_ssh\_extra\_argsのパラメータ | |
|  |  | 基本コンソール | 機器一覧 | | ssh認証鍵ファイル |
|  |  | 基本コンソール | 機器一覧 | | サーバー証明書 |
|  |  | 基本コンソール | 機器一覧 | | インベントリファイル  追加オプション |
|  |  |  |  | |  |
|  | site.yml | Playbookやホストの情報を全て呼び出し、Ansibleを実行するファイル。 | | | |
|  |  |  | | | |
|  |  | Ansible-Legacy | プレイブック素材集 | | プレイブック素材 |
|  |  | Ansible-Legacy | Movement詳細 | | インクルード順序 |
|  |  | Ansible-Legacy | Movement詳細 | | gather\_facts |
|  |  |  |  | |  |

投入データを用いて以下のコマンドでAnsibleを直接実行することも可能です。

ansible-playbook (オプション) –i hosts site.yml