



# IT Automation

## オフラインインストール

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

第1.6.1版

Exastro developer

# 目次

- 1. はじめに
  - 1.1 本資料について
- 2. システム構成
  - 2.1 環境構築
  - 2.2 動作環境・条件 (1/4)
  - 2.3 動作環境・条件 (2/4)
  - 2.4 動作環境・条件 (3/4)
  - 2.5 動作環境・条件 (4/4)
- 3. ITA環境構築手順
  - 3.1 オフラインインストール
  - 3.2 事前準備
  - 3.3 ITA環境構築フロー
  - 3.4 環境構築 (1/11)
  - 3.5 環境構築 (2/11)
  - 3.6 環境構築 (3/11)
  - 3.7 環境構築 (4/11)
  - 3.8 環境構築 (5/11)
  - 3.9 環境構築 (6/11)
  - 3.10 環境構築 (7/11)
  - 3.11 環境構築 (8/11)
  - 3.12 環境構築 (9/11)
  - 3.13 環境構築 (10/11)
  - 3.14 環境構築 (11/11)
- 4. ITA動作確認
  - 4.1 動作確認 (1/6)
  - 4.2 動作確認 (2/6)
  - 4.3 動作確認 (3/6)
  - 4.4 動作確認 (4/6)
  - 4.5 動作確認 (5/6)
  - 4.6 動作確認 (6/6)
- 5. 参考
  - 5.1 参考 (1/2)
  - 5.2 参考 (2/2)

# 1. はじめに

## 1.1 本資料について

### ■ 本資料について

- 本資料では、ITAサーバをオフライン環境でご利用いただく場合の、構築手順について記載しています。



## 2. システム構成

## 2.1 連携実行機能

### 連携実行機能について

- ITAでは、さまざまな構築ツール等と連携することができ、以下のツールとの連携機能をサポートしています。

ドライバ名	連携ツール (オーケストレーター)	機能	内容	環境構築ツール での連携ツール インストール	本資料での インストール 対象
Material	git	構築資材管理	標準構築ツールに登録された資材の払出 / 払戻と、Gitを介して対象資材のバージョン管理を行います。	○	×
Create_param	-	メニュー作成	メニューを作成・管理します。	-	○
Hostgroup	-	ホストグループ	ホスト群を論理的な単位（機能・役割）でまとめたグループにして、投入するパラメータを管理します。	-	○
Ansibleドライバー	Ansible	システム構築	Red Hat社が提供するOSSのPF構築ツールです。 Playbookと呼ばれる構築コードをもとに、ネットワークで接続された機器に対して、ソフトウェアのインストール、各種設定、ファイル転送、パッチの適用などを行います。	○	○
	AnsibleTower	システム構築	PF構築自動化ツールであるAnsibleにアクセスコントロール、ジョブスケジューリング、タスクの可視化などの機能を拡張した管理プラットフォームです。	×	
Cobblerドライバー	Cobbler	OSインストール	OSSのインストール自動化ツールです。 あらかじめ作成したテンプレートを元に、ネットワークで接続された機器に対して、OSのインストールを行うことができます。	×	×
Terraformドライバー	Terraform	システム構築	HashiCorp社が提供するインフラストラクチャを効率化するオーケストレーションツールです。 HCL(HashiCorp Configuration Language)という言語でコード化したインフラストラクチャ構成について、実行計画を生成したうえで構築を実行します。また、Policy as Codeによるアクセスポリシーをコード化して管理することが可能です。	×	×

### ■ ITAをご利用いただくための環境について

- 「Exastro-ITA\_システム構成／環境構築ガイド\_基本編」を参照してください。



### ■ ライブラリ収集を実行する場合の前提条件

- ライブラリ収集を実行する場合には、ライブラリ収集用サーバ（オンライン環境） / ITAサーバ（オフライン環境）、両サーバの構築状態（OSのバージョン、インストール済のパッケージ）を、合わせる必要があります。
- ライブラリ収集用サーバ（オンライン環境）は、以下のリポジトリが参照できる状態である必要があります。

（※次頁に記載）





## 2.4 動作環境・条件 3/4

### ● 参照するリポジトリ一覧

OS	リポジトリ
RHEL7	<a href="https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm">https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm</a>
	<a href="https://downloads.mariadb.com/MariaDB/mariadb_repo_setup">https://downloads.mariadb.com/MariaDB/mariadb_repo_setup</a>
	<a href="http://rpms.remirepo.net/enterprise/remi-release-7.rpm">http://rpms.remirepo.net/enterprise/remi-release-7.rpm</a>
	rhel-7-server-optional-rpms
RHEL8	<a href="https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm">https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm</a>
	codeready-builder-for-rhel-8- <b>xxxxxx</b> -rpms
CentOS7	epel-release
	<a href="https://downloads.mariadb.com/MariaDB/mariadb_repo_setup">https://downloads.mariadb.com/MariaDB/mariadb_repo_setup</a>
	<a href="http://rpms.remirepo.net/enterprise/remi-release-7.rpm">http://rpms.remirepo.net/enterprise/remi-release-7.rpm</a>
CentOS8	epel-release
	PowerTools

**xxxxxx** : アーキテクチャ

## 2.5 動作環境・条件 4/4

### ● 参照するリポジトリ一覧（クラウド環境の場合）

OS	リポジトリ
RHEL7	<a href="https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm">https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm</a>
	<a href="https://downloads.mariadb.com/MariaDB/mariadb_repo_setup">https://downloads.mariadb.com/MariaDB/mariadb_repo_setup</a>
	<a href="http://rpms.remirepo.net/enterprise/remi-release-7.rpm">http://rpms.remirepo.net/enterprise/remi-release-7.rpm</a>
	rhui-rhel-7-server-rhui-optional-rpms
RHEL7 (AWS/RHUI2)	<a href="https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm">https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm</a>
	<a href="https://downloads.mariadb.com/MariaDB/mariadb_repo_setup">https://downloads.mariadb.com/MariaDB/mariadb_repo_setup</a>
	<a href="http://rpms.remirepo.net/enterprise/remi-release-7.rpm">http://rpms.remirepo.net/enterprise/remi-release-7.rpm</a>
	rhui-REGION-rhel-server-optional
RHEL7 (AWS/RHUI3)	<a href="https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm">https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm</a>
	<a href="https://downloads.mariadb.com/MariaDB/mariadb_repo_setup">https://downloads.mariadb.com/MariaDB/mariadb_repo_setup</a>
	<a href="http://rpms.remirepo.net/enterprise/remi-release-7.rpm">http://rpms.remirepo.net/enterprise/remi-release-7.rpm</a>
	rhel-7-server-rhui-optional-rpms
RHEL8_AWS	<a href="https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm">https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm</a>
	codeready-builder-for-rhel-8-rhui-rpms

※RHEL7(AWS/RHUI2) : AWS上のRHEL7 (RHUI2を使用)

RHEL7(AWS/RHUI3) : AWS上のRHEL7 (RHUI3を使用)

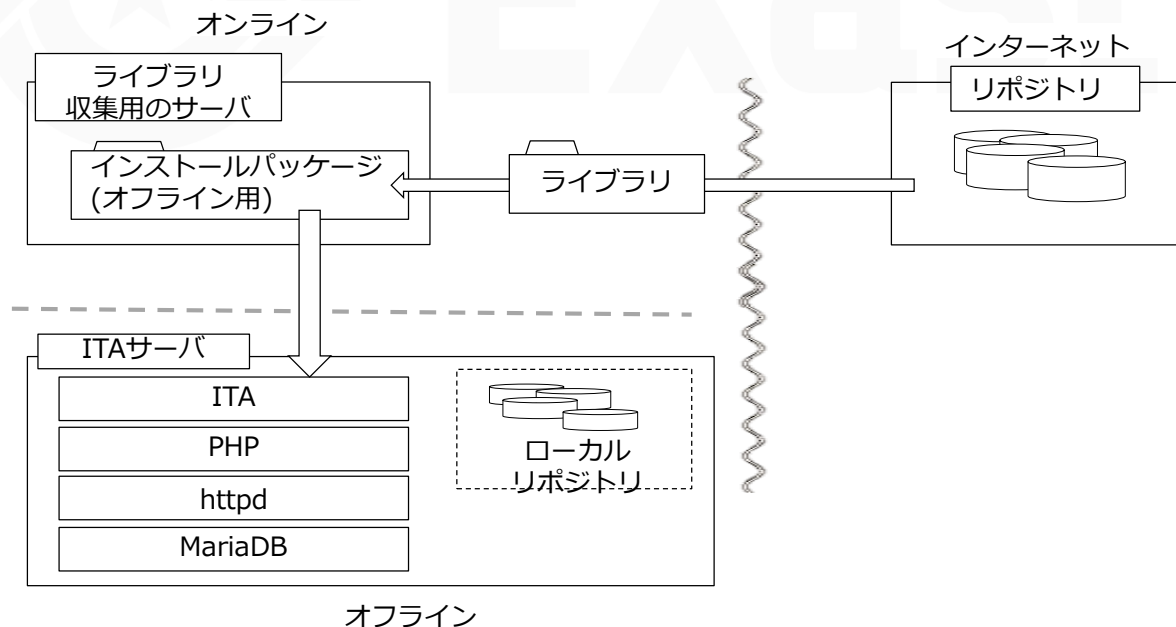
### 3. ITA環境構築手順

## 3.1 オフラインインストール

### インストール手順について

ITAサーバがオフライン環境の場合、以下の手順で環境構築を行います。

- ライブラリ収集用のサーバ(オンライン)にてインターネット経由で必要なライブラリを収集し、ITAインストールパッケージとライブラリを一つに圧縮し、インストールパッケージ(オフライン用)を作成します。
- 記憶媒体等でITAサーバに、インストールパッケージ(オフライン用)を移動します。
- インストールパッケージ(オフライン用)からローカルリポジトリを作成し、必要なライブラリのインストールと、ITAインストーラーの実行を行います。



## 3.2 事前準備

### ITA環境構築ツール一覧

- ITA環境構築ツール一覧は以下となります。

説明	ファイル	格納先
ITAインストーラー	ita_installer.sh	/(インストール資材展開先)/ita_install_package/install_scripts/
アンサーファイル	ita_answers.txt	/(インストール資材展開先)/ita_install_package/install_scripts/

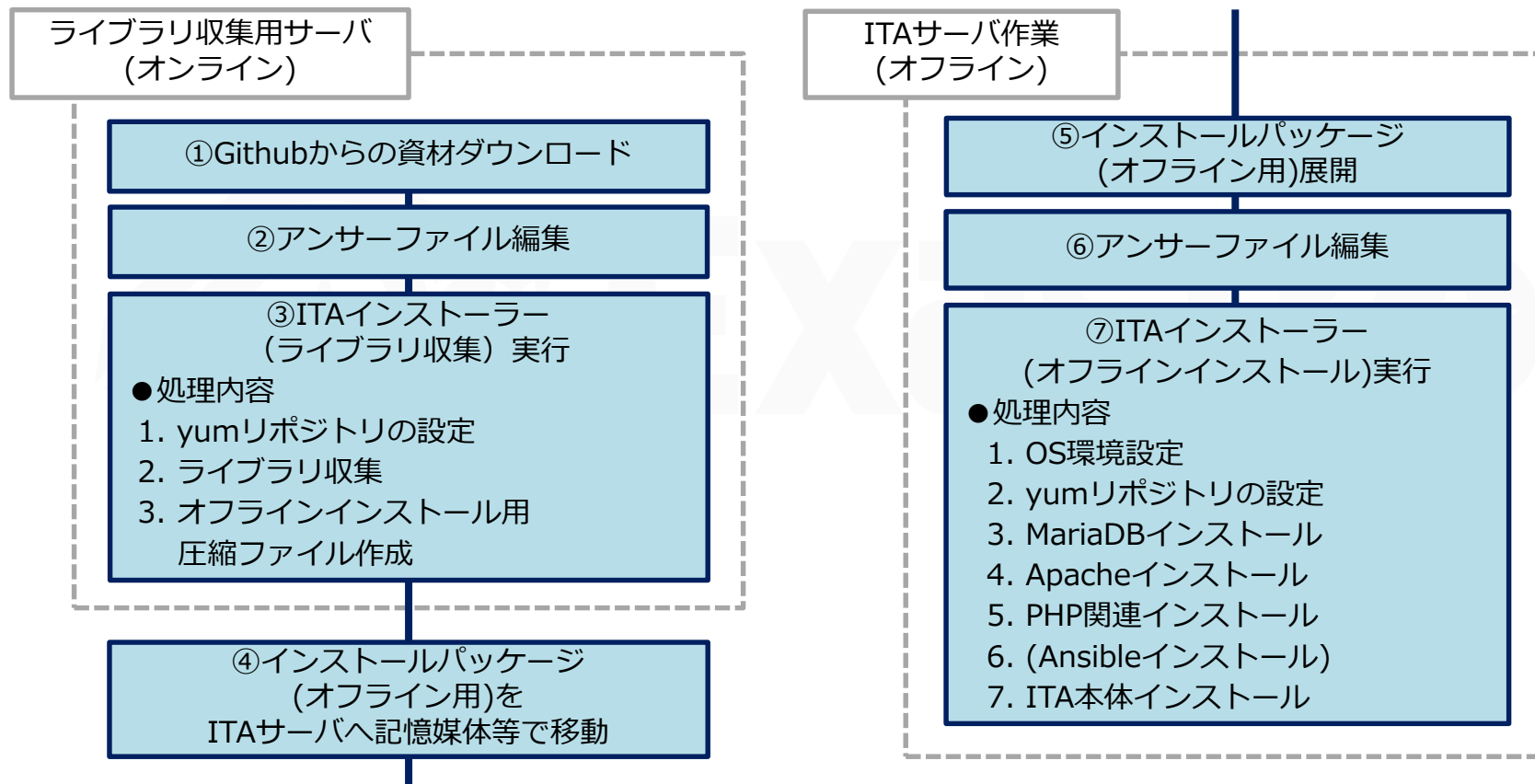
### RHELサブスクリプションについて

- クラウド環境以外のRHEL7・RHEL8のOSの環境でライブラリ収集をする場合は、その環境へのサブスクリプション登録を事前に完了させてください。

## 3.3 ITA環境構築フロー

### 環境構築フロー（オフライン）

- 環境構築は以下のフローとなっています。



## 3.4 環境構築 (1/11)

※ **オンライン環境**で実施します

※環境構築ユーザーはrootユーザーで実施すること。

### Githubからの資材ダウンロード

- 以下のコマンドで資材をDLします。

```
# wget https://github.com/exastro-suite/it-automation/releases/download/vx.x.x/exastro-it-automation-x.x.x.tar.gz
```

※ wgetコマンドは事前にインストールしてください。

※ **バージョン(x.x.x)**は適宜変更してください。

### 資材の展開

- .tar.gz ファイルを解凍します。

```
# tar xzf exastro-it-automation-x.x.x.tar.gz
```

### ディレクトリ移動

- 環境構築を設定を行うセッティングファイルとシェルのあるディレクトリに移動します。

```
# cd it-automation-x.x.x/ita_install_package/install_scripts
```

## 3.5 環境構築 (2/11)

### ■ アンサーファイル(ita\_answers.txt)を編集

- ライブラリ収集を行う前にアンサーファイルを事前に作成してください。
- ライブラリ収集を行う場合は「install\_mode」の設定値を「Gather\_Library」にしてください。
- ライブラリ収集の時点でのアンサーファイルの必須項目は「install\_mode」と「linux\_os」です。

種目	必須	初期値	説明
install_mode	○	Install_Online	インストールモードの設定 ・ Install_Online : オンラインインストール ・ Install_Offline : オフラインインストール ・ Gather_Library : ライブラリ収集 ・ Install_ITA : ITA本体のインストール ・ Versionup_All : ITA本体のバージョンアップ (ライブラリのインストールあり) ・ Versionup_ITA : ITA本体のバージョンアップ (ライブラリのインストールなし) ・ Uninstall : ITA本体のアンインストール ※詳細は参考参照
ita_directory	—	—	インストールディレクトリ ITAをインストールするディレクトリを絶対パスで指定してください。 ディレクトリが無い場合作成されます。 全ユーザーが参照可能なディレクトリを指定してください。 ディレクトリが無い場合作成されます。 ※ITAをインストールするディレクトリの親ディレクトリ全てに、 その他ユーザの実行権限を与えてください。
ita_language	—	en_US	ITA画面表示の言語 (日本語 (ja_JP) / 英語 (en_US) )
linux_os	○	—	ライブラリ収集サーバのOS ("CentOS7","CentOS8","RHEL7","RHEL8")
private_key_path	—	—	ユーザ指定のSSL秘密鍵に使用するファイルのファイルパスを指定 (ユーザ指定のSSL秘密鍵使用時のみ入力。絶対パスで指定してください。)



## 3.6 環境構築 (3/11)

### ITAインストーラー（ライブラリ収集）の実行

- 以下のコマンドで、ITAインストーラー（ライブラリ収集）を実行します。

```
# sh ita_installer.sh
```

### 処理の確認

- ライブラリ収集スクリプトを実行すると、ita\_gather.log に処理内容が出力されます。
- ログ格納パス  
/(インストール資材展開先)/ita\_install\_package/install\_scripts/log/

### ファイルの移動

- インストールパッケージ（オフライン用）をITAサーバへ記憶媒体等で移動します。

※以降の手順は、ITAサーバ（**オフライン環境**）で実施します

### インストールパッケージ（オフライン用）展開

- ITAサーバ上で、インストールパッケージ（オフライン用）を展開します。

```
# tar zxf ita_Verx.x_offline_yyyymmddhhmmss.tar.gz
```

## 3.7 環境構築 (4/11)

### ■ アンサーファイル(ita\_answers.txt)を編集

- ITAのインストール設定を行うアンサーファイルを編集してください。
- オフラインインストールを行う場合は「install\_mode」の設定値を「Install\_Offline」にしてください。
- アンサーファイル(ita\_answers.txt)の項目一覧 (1/2)

種目	必須	初期値	説明
install_mode	○	Install_Online	インストールモードの設定 ・ Install_Online : オンラインインストール ・ Install_Offline : オフラインインストール ・ Gather_Library : ライブラリ収集 ・ Install_ITA : ITA本体のインストール ・ Versionup_All : ITA本体のバージョンアップ (ライブラリのインストールあり) ・ Versionup_ITA : ITA本体のバージョンアップ (ライブラリのインストールなし) ・ Uninstall : ITA本体のアンインストール ※詳細は参考参照
ita_directory	○	—	インストールディレクトリ ITAをインストールするディレクトリを絶対パスで指定してください。 全ユーザーが参照可能なディレクトリを指定してください。 ディレクトリが無い場合作成されます。 ※ITAをインストールするディレクトリの親ディレクトリ全てに、 その他ユーザの実行権限を与えてください。
ita_language	○	en_US	ITA画面表示の言語 (日本語 (ja_JP) / 英語 (en_US) )
linux_os	○	—	ITAサーバのOS ("CentOS7","CentOS8","RHEL7","RHEL8")
db_root_password	○	—	MariaDBのrootパスワード
db_name	○	—	MariaDBのDB名
db_username	○	—	MariaDBのDBユーザー名
db_password	○	—	MariaDBのDBパスワード

## 3.8 環境構築 (5/11)

- 「ita\_base」から「terraform\_driver」までの項目はITA本体や機能、連携ドライバのインストール設定の項目です。インストールする場合は設定値を「yes」、インストールしない場合は「no」としてください。

### ・アンサーファイル(ita\_answers.txt)の項目一覧 (2/2)

種目	必須	初期値	説明
ita_base	○	yes	ITA本体のインストール（"yes"のみ）
Material	○	no	構築資材管理機能のインストール有無指定
create_param	○	yes	メニュー作成機能のインストール有無指定
Hostgroup	○	yes	ホストグループ機能のインストール有無指定
ansible_driver	○	yes	Ansible driverのインストール有無指定
cobbler_driver	○	no	Cobbler driverのインストール有無指定
terraform_driver	○	no	Terraform driverのインストール有無指定
ita_domain	○	exastro-it-automation.local	ITAのドメイン名の指定（ITAインストーラーが自己証明書を作成する時はこちらの値を使用）
certificate_path	任意	—	ユーザ指定のSSLサーバ証明書に使用するファイルのファイルパスを指定（ユーザ指定のSSL証明書使用時のみ入力。絶対パスで指定してください。）
private_key_path	任意	—	ユーザ指定のSSL秘密鍵に使用するファイルのファイルパスを指定（ユーザ指定のSSL秘密鍵使用時のみ入力。絶対パスで指定してください。）

## 3.9 環境構築 (6/11)

### ユーザ指定サーバ証明書・秘密鍵について

- サーバ証明書と秘密鍵にユーザが用意したファイルを使用することができます。使用する場合は、サーバ証明書と秘密鍵の両方を用意し、アンサーファイル(ita\_answers.txt)の「certificate\_path」と「private\_key\_path」の両方にファイルパスを入力してください。証明書と秘密鍵どちらか片方のみの使用はできません。
- サーバ証明書に中間証明書が付属している場合は、サーバ証明書に中間証明書を連結してファイルを作成し、「certificate\_path」に作成したファイルのパスを指定してください。

作成コマンド例

```
# cat [サーバ証明書ファイル] [中間証明書ファイル] > [連結済サーバ証明書ファイル]
```

- 「certificate\_path」と「private\_key\_path」に入力がない場合は、ITAインストーラーがアンサーファイルの「ita\_domain」の値を使用して自己証明書を作成・設置します。  
(※「ita\_domain」の値を自己証明書作成時のコモンネーム、ならびに自己証明書と秘密鍵のファイル名に使用します)

## 3.10 環境構築 (7/11)

- インストール時にサーバ証明書と秘密鍵は「/etc/pki/tls/certs」のディレクトリに設置され、アンインストール時にはそのディレクトリから削除されますので、ユーザ指定のサーバ証明書と秘密鍵を使用する場合は、オリジナルのサーバ証明書・秘密鍵ファイルの管理に注意してください。
- アンインストールでは、アンサーファイル(ita\_answers.txt)の「certificate\_path」と「private\_key\_path」の両方にファイル指定がある場合は、それらの指定されたファイルの削除を行い、ファイル指定がない場合は、アンサーファイルの「ita\_domain」に指定されている名前を使用したファイルを削除します。

## 3.11 環境構築 (8/11)

### ■ アンサーファイル(ita\_answers.txt)のサンプル

- アンサーファイル(ita\_answers.txt)のサンプルを以下に示します

#### ・ アンサーファイル(ita\_answers.txt)のサンプル(1/2)

```
# Select install mode.
# ("Install_Online","Install_Offline","Gather_Library","Install_ITA",
# "Versionup_All","Versionup_ITA","Uninstall")
# e.g) install_mode:Install_Online
# *This installer operates according to the inputted values below.
# Install_Online: ITA will be installed after the necessary libraries
#                has been installed via internet.
# Install_Offline: ITA will start installing using the package created
#                in Gather Library.
# Gather_Library: Gathers the necessary libraries via internet and creates
#                the package necessary to execute Install_Offline.
#                (Execute this before executing Install_Offline)
# Install_ITA: Installs ITA without installing any libraries.
# Versionup_All: Installs ITA after installing the necessary libraries for
#                the desired ITA version via internet.
# Versionup_ITA: Updates ITA without installing any libraries.
# Uninstall: ITA Uninstalls ITA.(Libraries will not be uninstalled)
install_mode:Install_Offline

# Enter install directory.
# e.g) ita_directory:/exastro
ita_directory:/exastro

# Select language. ("en_US" or "ja_JP")
# e.g) ita_language:en_US
ita_language:en_US

# Select Operation System. ("CentOS7","CentOS8","RHEL7","RHEL8")
# e.g) linux_os:RHEL8
# * If registering a subscription is needed in order to acquire
#   the RHEL7 and RHEL8 libraries, please do so in advance.
linux_os:CentOS7
```

#### POINT

**アンサーファイル  
(ita\_answers.txt)ではど  
の項目にも全角文字が使用  
できません。**

## 3.12 環境構築 (9/11)

- アンサーファイル(ita\_answers.txt)のサンプルを以下に示します
  - ・ アンサーファイル(ita\_answers.txt)のサンプル(2/2)

```
# Enter the MariaDB root user's password
# e.g) db_root_password:sample_root_password
db_root_password:sample_root_password

# Decide the database name, username, and password for ITA.
# e.g) db_name:sample_db_name
db_name:sample_db_name
# e.g) db_username:sample_db_username
db_username:sample_db_username
# e.g) db_password:sample_db_password
db_password:sample_db_password

# Select the target you need to install.
# yes : need
# no  : no need
ita_base:yes
material:no
createparam:yes
hostgroup:yes
ansible_driver:yes
cobbler_driver:no
terraform_driver:no

# Enter the ita domain name.
# e.g) ita_domain:exastro-it-automation.local
ita_domain:exastro-it-automation.local

#####
# Enter when using user-specified certificates and private keys.
# If no file path is entered for both "certificate_path" and "private_key_path",
# the ITA installer creates and installs a self-certificate and private key
# using the values entered in "ita_domain".

# Enter the file path where the certificate to be install.
# e.g) certificate_path:/temp/etc_pki_tls_certs/exastro-it-automation.crt
certificate_path:

# Enter the file path where the private key to be install.
# e.g) private_key_path:/temp/etc_pki_tls_certs/exastro-it-automation.key
private_key_path:

#####
```

### POINT

**MariaDBのデータベース名、ユーザ名、パスワードはアンサーファイルで定義します。**

**※パスワードに使える文字は半角英数字と半角記号です。**

### POINT

**ユーザ指定のSSL証明書と秘密鍵の両方を使用する時のみ入力してください。どちらか片方のみの使用はできません。**

## 3.13 環境構築 (10/11)

### ITAインストーラー(オフラインインストール)実行

- 以下のコマンドで、ITAインストーラー(オフラインインストール)を実行します。

```
# sh ita_installer.sh
```

### 処理の確認

- 環境構築ツールを実行すると、ita\_builder.log、ita\_installer.log に処理内容が出力されます。
- ログ格納パス  
/(インストール資材展開先)/ita\_install\_package/install\_scripts/log/

### 終了ステータスについて

- ITAインストーラーは、シェルの処理終了時に終了の状態によって以下の終了ステータスを返します。  
正常終了時 : 0  
異常終了時 : 1



## 3.14 環境構築（11/11）

### 環境構築時にインストールされるライブラリの一覧

- ITAインストーラー(オフラインインストール)を実行することでインストールされるライブラリは、以下となります。

インストールドライバ	ライブラリ概要	ライブラリ名
ita_base	インストールツール	yum-utils(*), createrepo(*)
ita_base	ITA共通	zip, telnet, mailx, unzip, sudo, crontabs
ita_base	MariaDB	MariaDB, MariaDB-server, expect
ita_base	httpd	httpd, mod_ssl
ita_base	php	php, php-bcmath, php-cli, php-ldap, php-mbstring, php-mcrypt, php-mysqlnd, php-pear, php-pecl-zip, php-process, php-snmp, php-xml, php-json, php-gd, Python3, php-devel, libyaml, libyaml-devel, make
ita_base	phpプラグイン	php-yaml, HTML_AJAX-beta, PhpSpreadsheet
material	git	Git
ansible_driver	ansible	ansible, sshpass, pexpect, pywinrm, boto3, nmap-ncat, paramiko

※ RHEL7,CentOS7

## 4. ITA動作確認

## 4.1 動作確認 (1/4)

### メインメニューの表示による確認

- インストール処理終了後、自端末のWindowsPCから下記の手順により、ITAシステムメインメニューにアクセスし、ITA本体、各ドライバーが正常に表示されたことを確認してください。

### URL接続

- 以下のURLより、ログイン画面にアクセスしてください。
- URL : [http:// \(サーバのIPアドレス\)](http://(サーバのIPアドレス))

※インストール後は、HTTPとHTTPSの両方のアクセスが可能です。  
HTTPはセキュリティ的に脆弱なので、HTTPSでのアクセスを推奨します。  
HTTPSでのアクセス方法は、動作確認 (4/6) 以降を確認してください。


### ログイン

- ITAのログイン画面が表示されたら、指定のログインID、初期パスワードを入力して、[ログイン]ボタンをクリックしてください。
  - ・ ログインID : administrator
  - ・ 初期パスワード : password
- インストール後に初めてログインした場合は、「パスワード変更画面」に遷移します。
- パスワード変更画面から、初期パスワードを変更してください。

## 4.2 動作確認 (2/4)

### ITAログイン画面

- 正常にインストールされている場合、以下のようなログイン画面が表示されます。



ログインID : administrator

初期パスワード : password

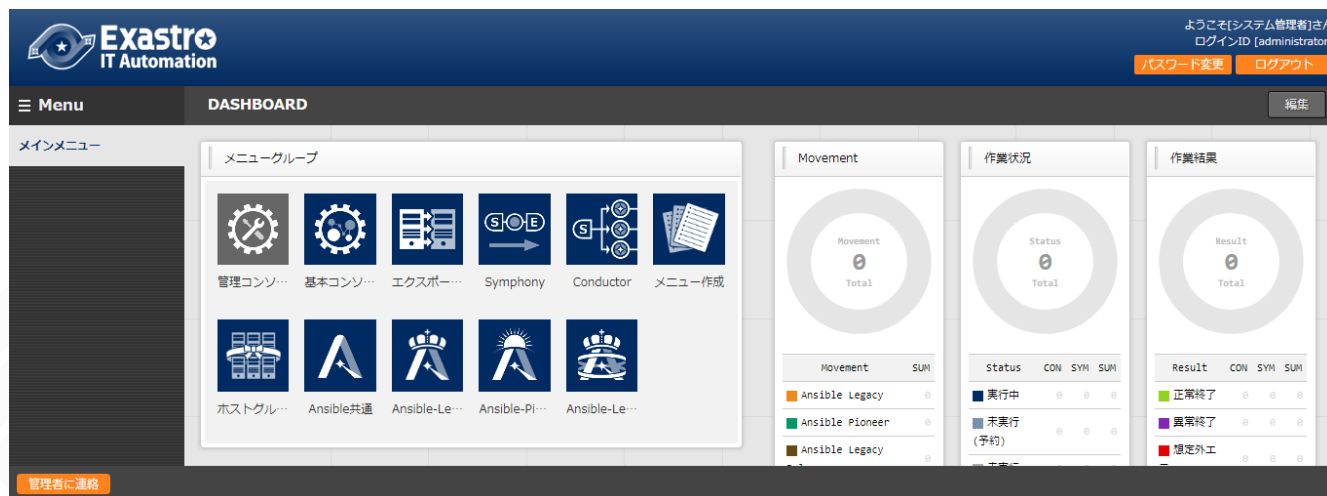
ログイン

管理者に連絡

## 4.3 動作確認 (3/4)

### 各メニューの表示による内容確認

- ログイン後、以下のメニューが正常に表示されていることを確認してください。



機能	メニュー
ITA本体	管理コンソール
	基本コンソール
	エクスポート/インポート
	Symphony
	Conductor
メニュー作成	メニュー作成
ホストグループ	ホストグループ管理
Ansible ドライバー	Ansible共通
	Ansible-Legacy
	Ansible-Pioneer
	Ansible-LegacyRole

## 4.4 動作確認 (4/4)

### HTTPSでアクセスするための準備作業

- アンサーファイルの「ita\_domain」に設定したホスト名をご使用の環境のDNSサーバまたは操作端末のhostsに登録してください。
- 操作端末（Windows）への証明書インポートを行います。  
ユーザ指定のサーバ証明書を使用していない場合、サーバ証明書はITAインストールパッケージの以下のパスに格納されています。

ディレクトリ	ファイル名
/etc/pki/tls/certs	[アンサーファイルのita_domainに設定したホスト名].crt

※ユーザ証明書を使用する場合はアンサーファイルの「certificate\_path」に設定した証明書ファイルを使用してください。

- Webブラウザに証明書のインポートをしてください。

### HTTPSでのURL接続

- 以下のURLより、ログイン画面にアクセスしてください。
- URL : **https://[アンサーファイルのita\_domainに入力したホスト名]**  
※ドメイン名の代わりに、サーバーのIPアドレスでアクセスすることも可能です。

接続後はHTTPの場合と同様となります。

## 5. 参考

## 5.1 参考 (1/2)

### ■ HTTPまたはHTTPSアクセスの制限

HTTPまたはHTTPSアクセスを制限する場合は、以下の手順を実施してください。

- ファイル「/etc/httpd/conf.d/vhosts\_exastro-it-automation.conf」を編集する。  
HTTPを制限する場合は、「<VirtualHost \*:80 >」から「</VirtualHost>」をコメントアウト(#)をする。  
HTTPSを制限する場合は、「<VirtualHost \*:443 >」から「</VirtualHost>」をコメントアウト(#)をする。
- 以下のコマンドによりApacheを再起動する。  
`systemctl restart httpd`



### ■ インストールモードについて

- バージョン1.6.0より、ITAインストール時に実行するシェルがita\_installer.shのみに統一され、アンサーファイル(ita\_answers.txt)の「install\_mode」の値によって、インストーラーの動作が以下のモードに分岐します。
  - Install\_Online : ITAに必要なライブラリのインストールをインターネット経由で行った後、ITA本体をインストールします。
  - Install\_Offline : Gather Libraryで作成したパッケージを使い、オフラインでライブラリのインストールとITA本体のインストールを行います。
  - Gather\_Library : ITAに必要なライブラリの収集をインターネット経由で行い、Install\_Offlineの実行に必要なパッケージを作成します。(Install\_Offlineを行う前に実行してください)
  - Install\_ITA : ライブラリのインストールは行わずに、ITA本体をインストールします。
  - Versionup\_All : バージョンアップで必要となるライブラリをインターネット経由で追加インストールした後、ITA本体をバージョンアップします。
  - Versionup\_ITA : ライブラリのインストールは行わずに、ITA本体をバージョンアップします。
  - Uninstall : ITA本体をアンインストールします。(ライブラリのアンインストールは行いません。)



**Exastro**