



# OASE Operation Autonomy Support Engine

## Base 【実習】

※本書では「Operation Autonomy Support Engine」を「OASE」として記載します。

Exastro Operation Autonomy Support Engine Version 1.3.1  
Exastro developer

# 目次

1. はじめに
    - 1.1 Base【実習】について
  2. シナリオ説明
    - 2.1 本書のシナリオ
  3. 事前設定
    - 3.1 グループ作成
    - 3.2 ユーザ作成、新規ユーザでのログイン
    - 3.3 トークン払い出し
    - 3.4 アクション設定 (ITAドライバ)
    - 3.5 ディシジョンテーブル作成
  4. 作業実行
    - 4.1 ディシジョンテーブルファイル作成
    - 4.2 ルール登録 (アップロード、テ스트リクエスト)
    - 4.3 ルール判定 (curlコマンドによるリクエスト送信)
    - 4.4 アクション実行結果の確認
- A 付録  
サンプル1

# 1. はじめに



# 1.1 Base 【実習】について (1/3)

## まえがき

- 本稿は、Exastro Operation Autonomy Support Engine (OASE) を利用する上で、基本的な利用手順の理解を支援するための資料です。
- 概要を理解したい場合は、< [Exastro OASE Base 【座学】](#) >もありますので、そちらをご参照ください。
- 包括的な内容としては、Exastro OASE の公式マニュアル集である< [OASE docs](#) >をご参照ください。

# 1.1 Base【実習】について(2/3)

## 前提

- アクションとしてITAを実行するため、以下が導入済みであること

① ITAインストールについては以下を参照

- <[ITA オンラインインストールマニュアル](#)>

② ITA BASE【実習】については以下を参照

- <[IT Automation BASE【実習】](#)>
- 上記の設定を使用するため実施する
- 「シナリオ」スライド以降の内容を使用する

③ OASEインストールについては以下を参照

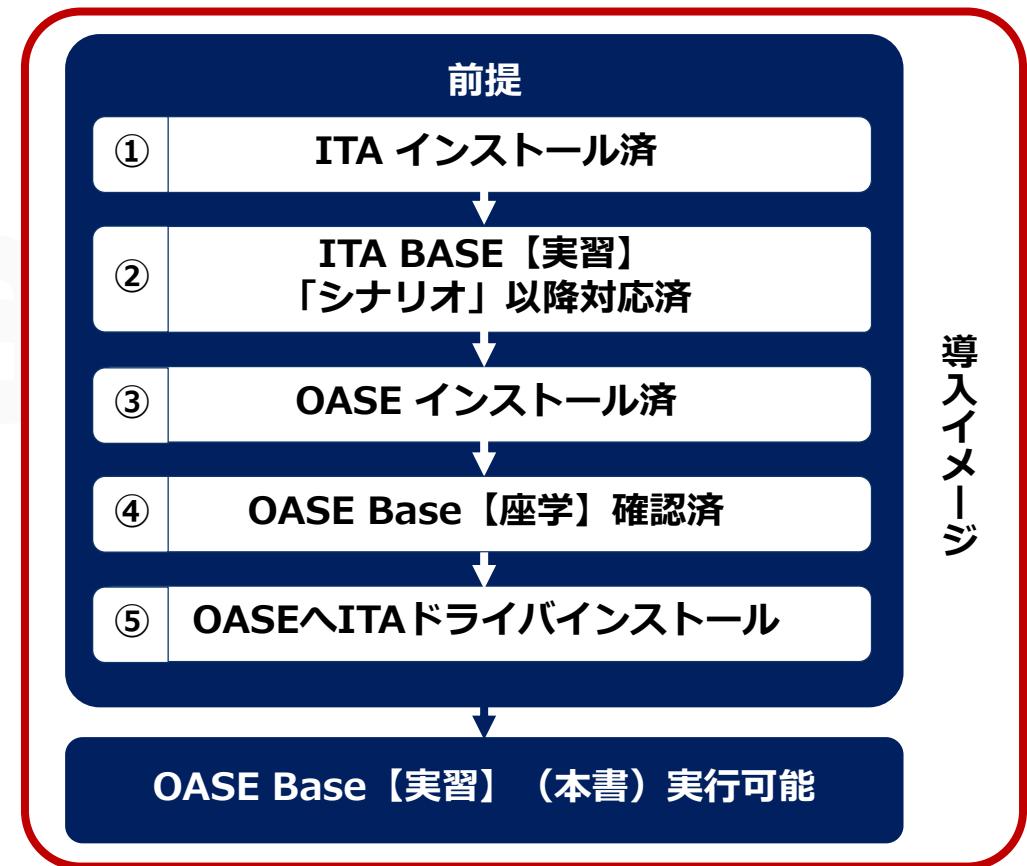
- <[OASE オンラインインストールマニュアル](#)>
- 一般ユーザ作成時のパスワード通知のため、インストール時にメールサーバの設定が必要

④ OASE Base【座学】については以下を参照

- <[OASE Base【座学】](#)>

⑤ ITAドライバインストールについては以下を参照

- <[環境構築マニュアル - ドライバインストール編](#)->

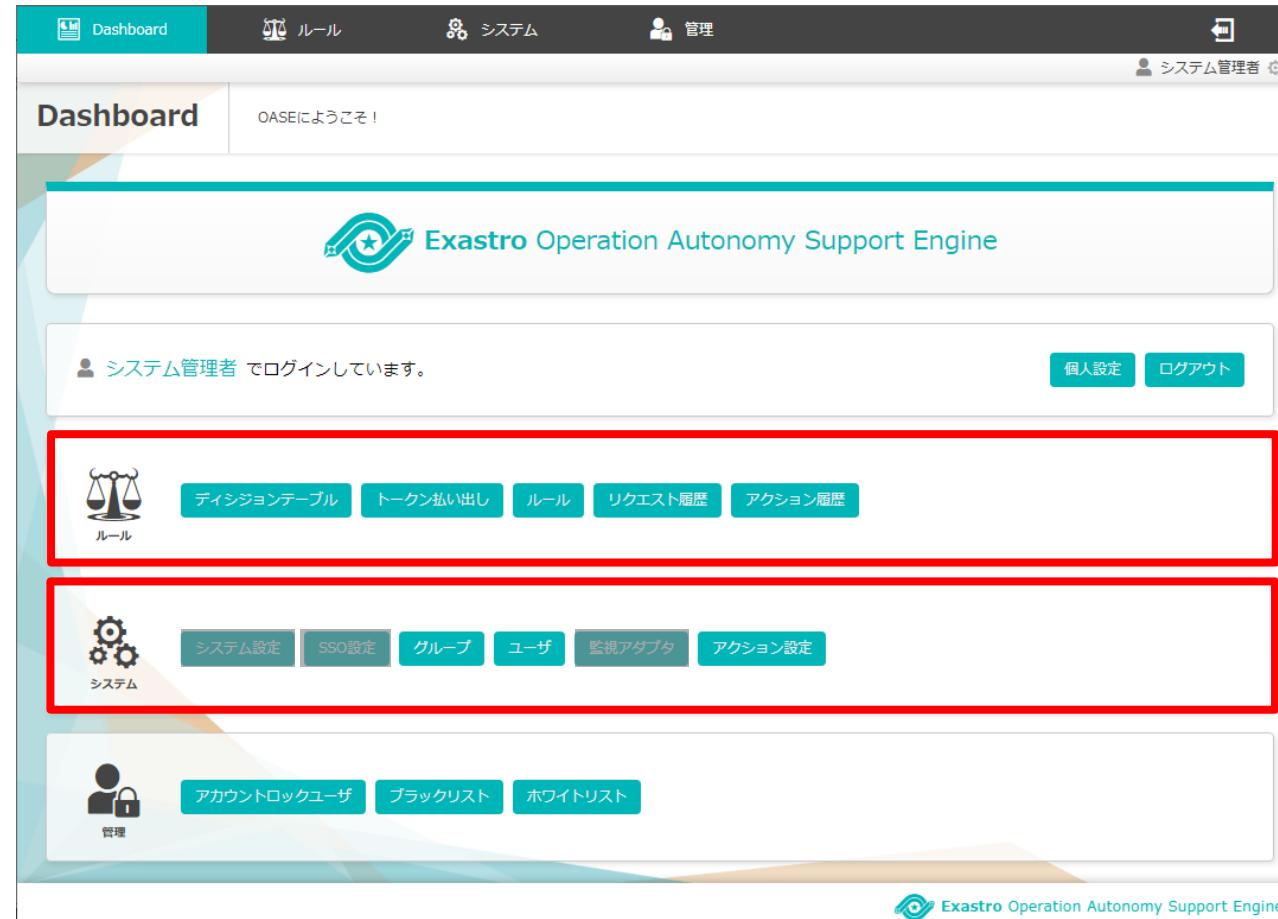


※本書では「Exastro IT Automation」を「ITA」として記載します。

# 1.1 Base 【実習】について (3/3)

BaseではOASEの以下機能（画面）を用いる

- Dashboard画面



## カテゴリ：ルール

画面名称
ディシジョンテーブル
トークン払い出し
ルール
リクエスト履歴
アクション履歴

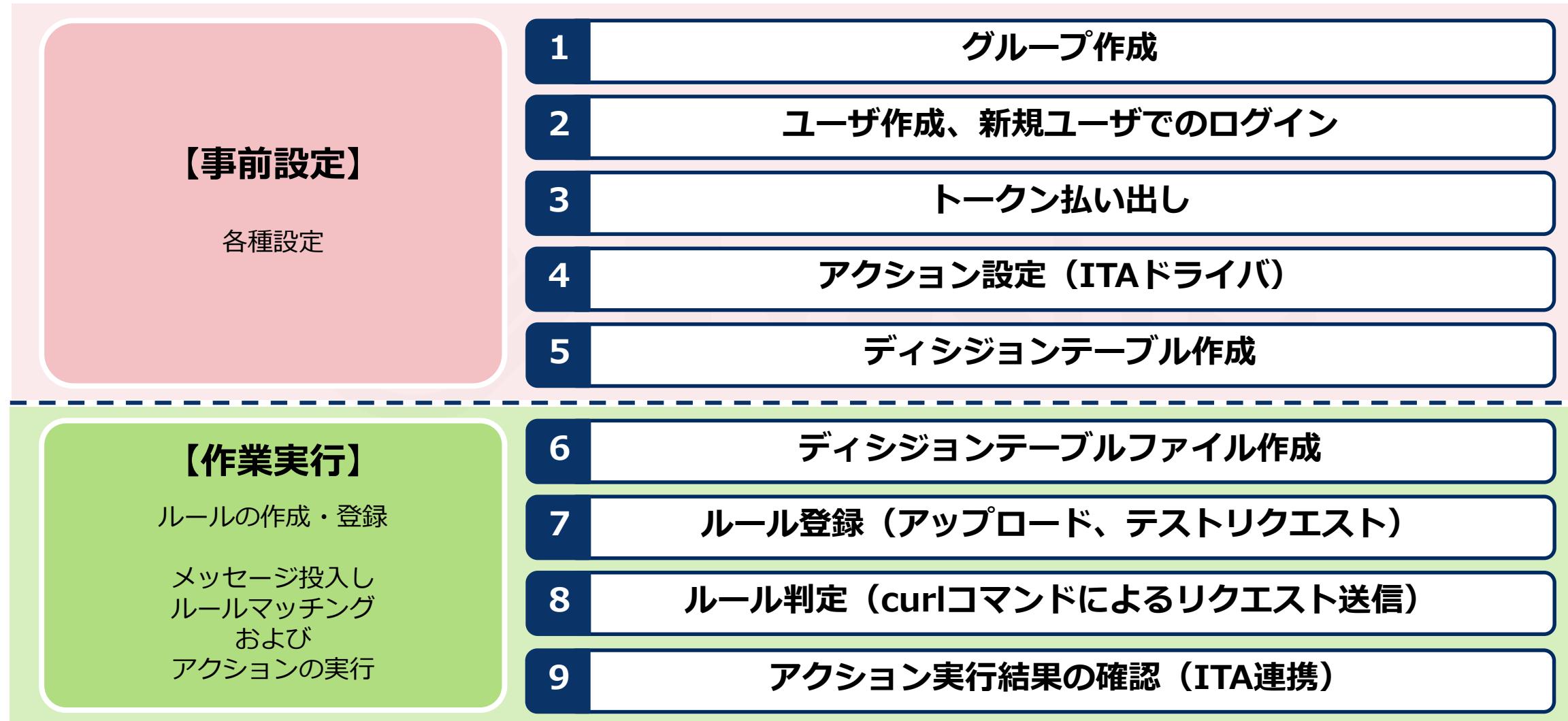
## カテゴリ：システム

画面名称
グループ
ユーザ
アクション設定

## 2. シナリオ説明

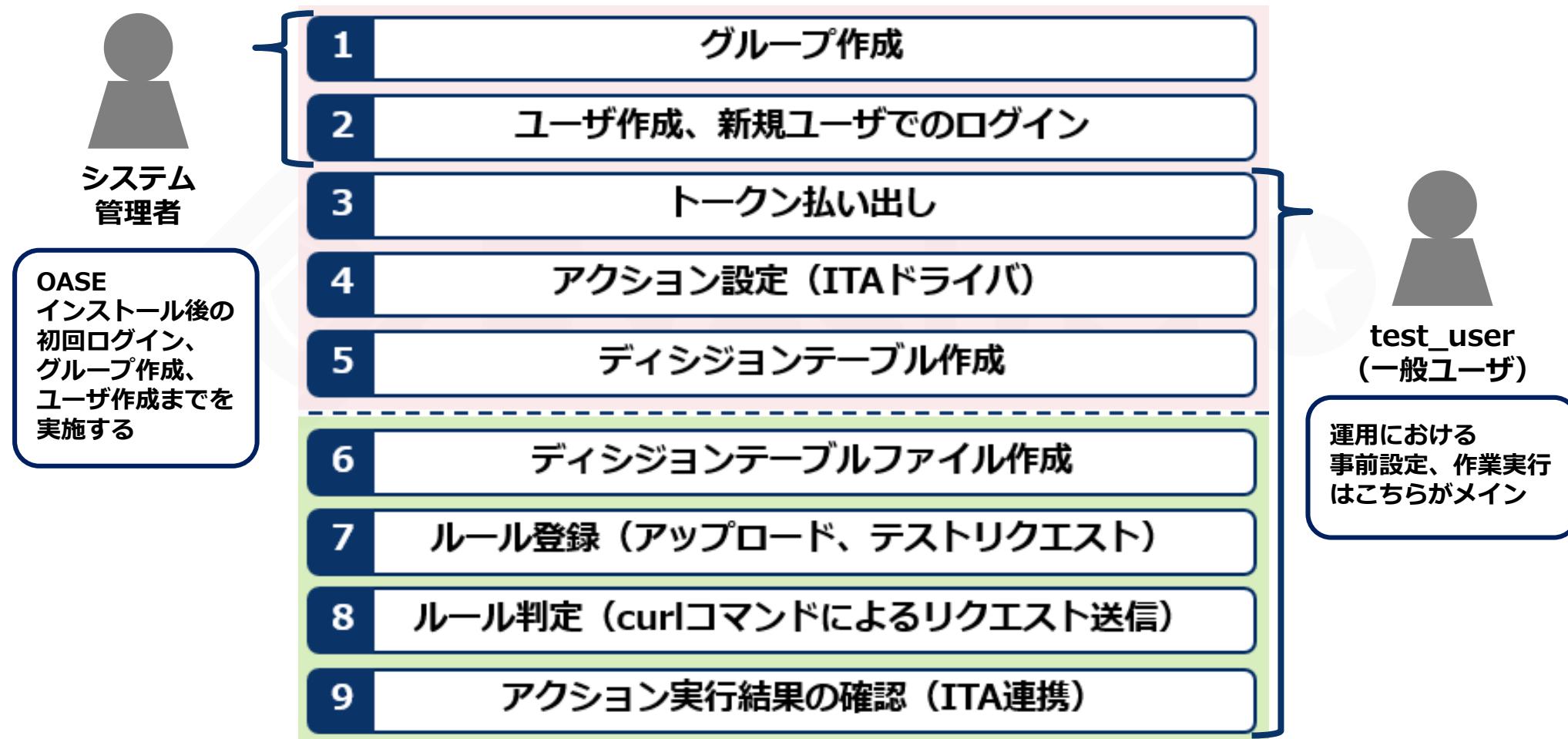
## 2.1 本書のシナリオ (1/2)

### OASEインストール後からアクション実行するまでのシナリオ



## 2.1 本書のシナリオ (2/2)

本シナリオはグループ「test\_group」およびユーザ「test\_user」を新規作成し、一般ユーザの「test\_user」が一連の操作を実施する内容となる



### 3. 事前設定



### 3.1 グループ作成

#### グループ「test\_group」の新規追加

- ① システム管理者で初回ログインする
- ② グループ画面に遷移し「test\_group」を新規追加する
- ③ 「test\_group」のアクセス権限を設定する

The screenshot shows the 'Groups' page in the Exastro UI. A new group named 'test\_group' is being created. The 'Access Rights' dialog is open, showing the access rights configuration for the group.

**1. ユーザ作成、新規ユーザでのログイン**

**2. 以下の値を入力する**

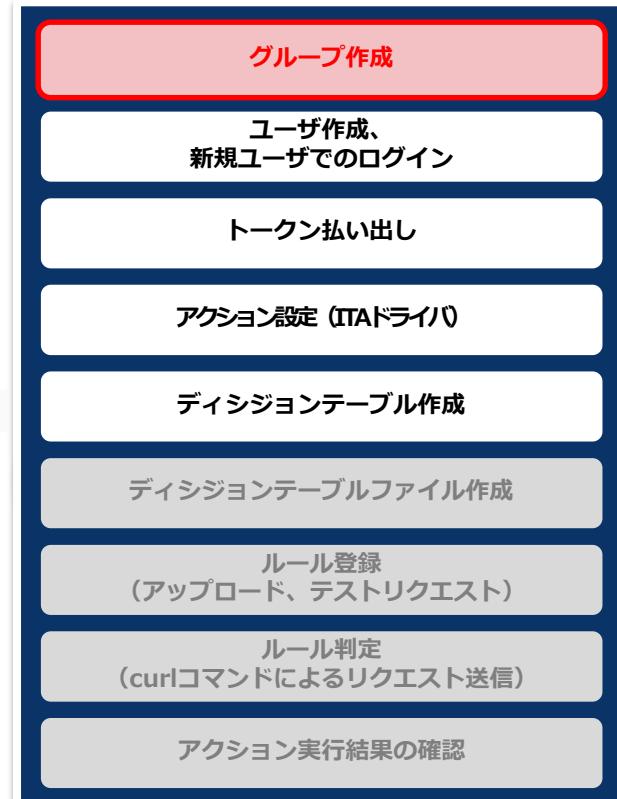
項目	設定値
グループ名	test_group

**3. 以下の値を入力する**

項目	設定値
アクセス権限	すべて「更新可能」を選択

**Access Rights Configuration (Access Rights dialog):**

権限の設定						
グループ名	test_group	概要				
カテゴリ	画面	機能	X	参照なし	参照のみ	更新可能
ルール	ディシジョンテーブル	新規追加	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
	トークン払い出し	—	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
システム	システム設定	—	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
	SSO設定	—	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
システム	グループ	—	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
	ユーザ	—	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
	アクション設定	—	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>



### 3.2 ユーザ作成、新規ユーザでのログイン

#### ユーザ「test\_user」の新規追加

- ① システム管理者で「ユーザ」画面に遷移
- ② 「ユーザ」画面で「test\_user」を新規追加する
- ③ システム管理者からログアウトし「test\_user」でログインする

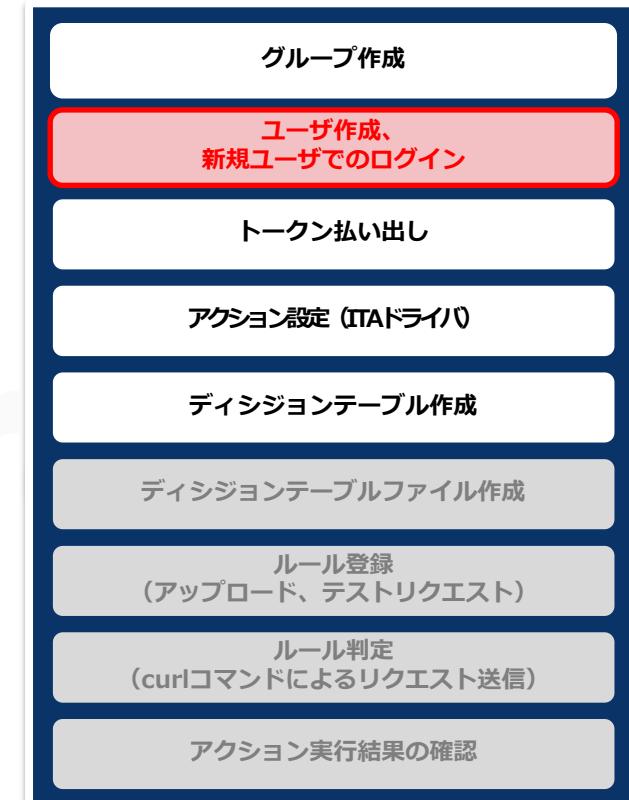
1 システム管理者

2 以下の値を入力する

項目	設定値
ユーザ名	test_user
ログインID	test_user
メールアドレス	(受信可能なメールアドレス)
グループ	test_group

#### POINT

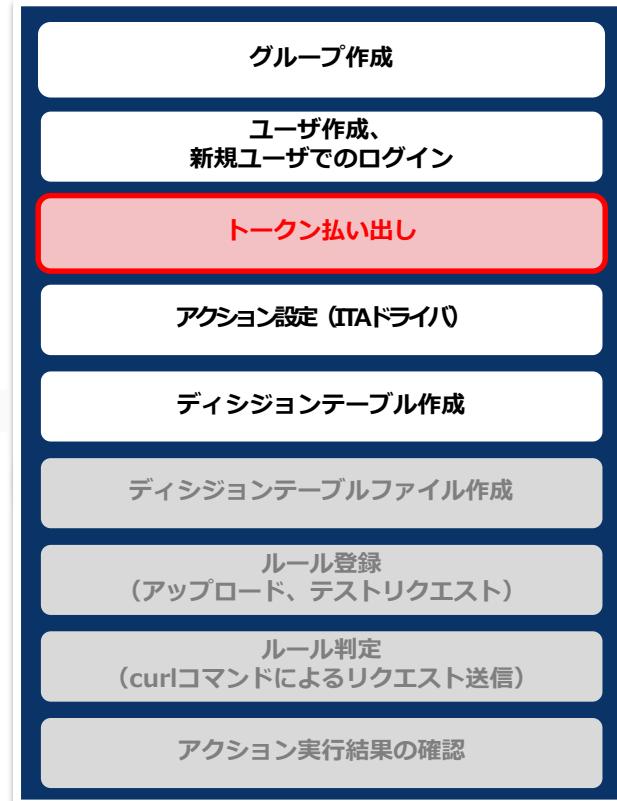
ユーザ登録時に入力したメールアドレス宛に「ログインID通知」「パスワード通知」のメール2通が届きます。  
記載されているIDとパスワードでOASEにログインしてください。



### 3.3 トークン払い出し

#### 新規トークンの払い出しを実施

- ① 「新規トークン払い出し」ボタンを押下
- ② 「新規トークン払い出し」画面で必要情報を入力
- ③ 「トークン払い出し」ボタンを押下
- ④ 「トークン」画面に表示されるトークンをコピーして保持する
- ⑤ 「閉じる」ボタンを押下



#### POINT

トークンは後述する<[4.3 ルール判定 \(curlコマンドによるリクエスト送信\)](#)>時に使用するため設定が必要です。

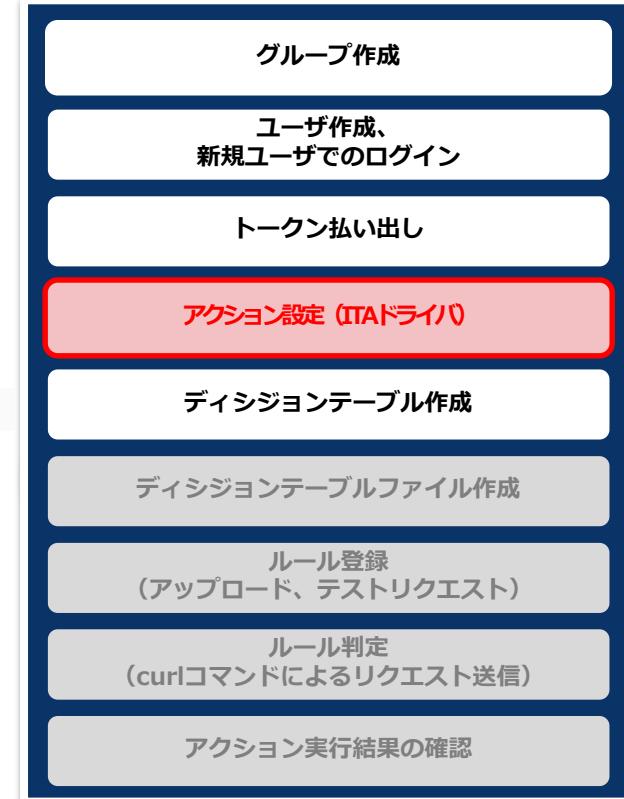
### 3.4 アクション設定 (ITAドライバ) (1/4)

#### アクション先の追加

- ① 「アクション設定」画面上の「アクション先の追加」ボタンを押下
- ② 「アクション先の選択」欄で「ITA Driver ver1」を選択



※ ドライバをインストールしていない場合、上記の画面は表示されません。



#### POINT

事前に<[環境構築マニュアル - ドライバインストール編](#)>を参照のうえメールドライバをインストールしてください。

### 3.4 アクション設定 (ITAドライバ) (2/4)

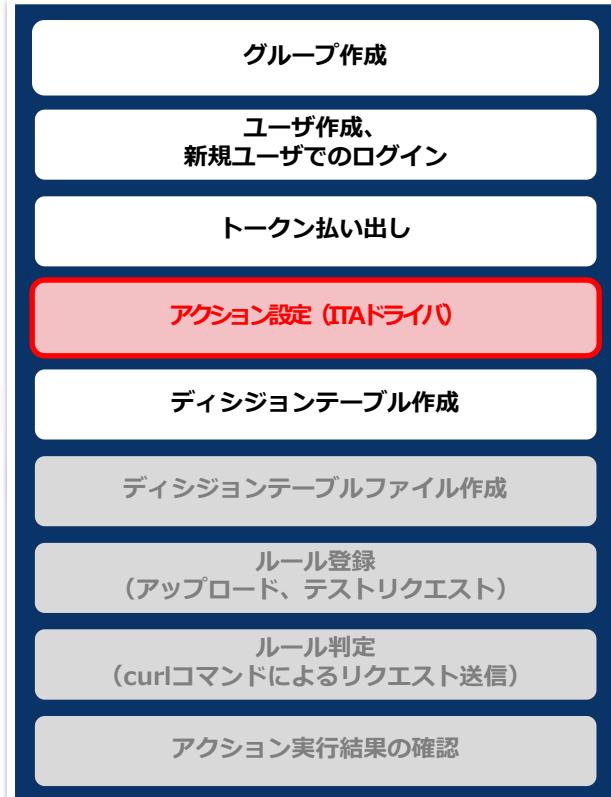
#### アクション先の設定

- ① 「ITA Driver ver1」画面の必要情報を入力
- ② 「保存」ボタンを押下



1 以下の値を入力する

項目	設定値
名前	(任意の文字列)
バージョン	1.7.1
プロトコル	http
ホスト/IP	(ITAのホストおよびIPアドレス)
ポート	80
ユーザ名	administrator
パスワード	(ITAにログインできるパスワード)
権限の設定	test_group : 「更新可能」



#### POINT

「名前」は後述する<[ディシジョンテーブルファイル作成](#)>時に「どのアクション先に対してアクション実行するのか」指定するために必要です。

### 3.4 アクション設定 (ITAドライバ) (3/4)

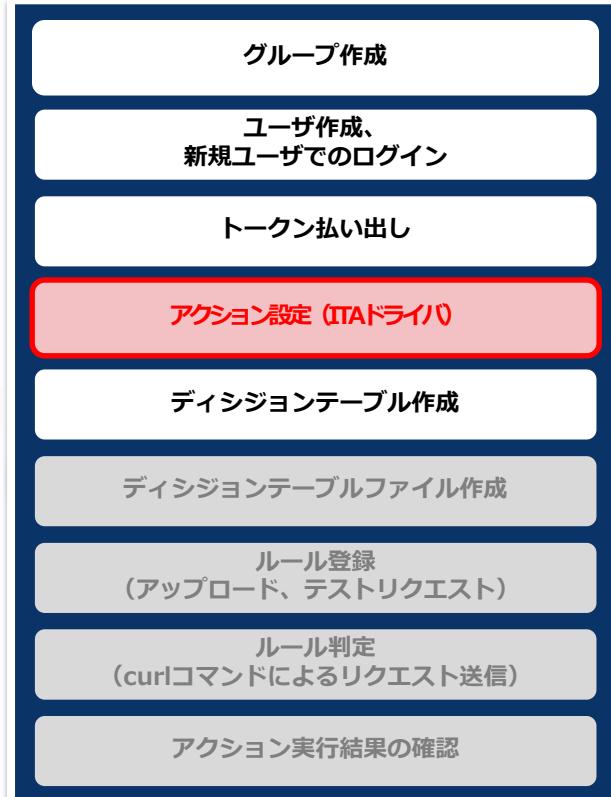
#### ITAの登録内容を変更する

- ① ITAにログインし<[Learn ITA BASE【実習】](#)>のスライド「シナリオ」から  
「Symphony完了確認」まで実施する  
※章番号等は変更される場合があります。

The diagram illustrates the workflow for scenario registration. It starts with 'Scenario registration' (3.1) and follows a path through various steps: 'Job target host registration' (4.1), 'Operation registration' (4.2), 'IaC registration' (4.3), 'Movement registration' (4.4), 'Movement detail registration' (4.5), 'Operation related Movement and host registration' (4.6), 'Symphony registration' (4.7), 'Symphony execution' (4.8), and finally 'Symphony completion confirmation' (4.9). A note at the bottom indicates the use of Ansible-Legacy's menu for steps 4.3 to 4.7.

3.1 シナリオ

4.10 Symphony完了確認



- ② ITAの「Ansible-Legacy」メニュー群 > 「代入値管理」メニュー > 「一覧/更新」サブメニューの「具体値」を変更する

The screenshot shows the Ansible-Legacy interface with the '代入値管理' (Input Value Management) sub-menu highlighted in red. The main window displays a table for managing input values, with the '具体値' (Specific Value) column highlighted in red.

2 以下の値を入力する

項目	設定値
具体値	(任意の文字列)

#### POINT

「具体値」に入力した名称でディレクトリが作成されます。

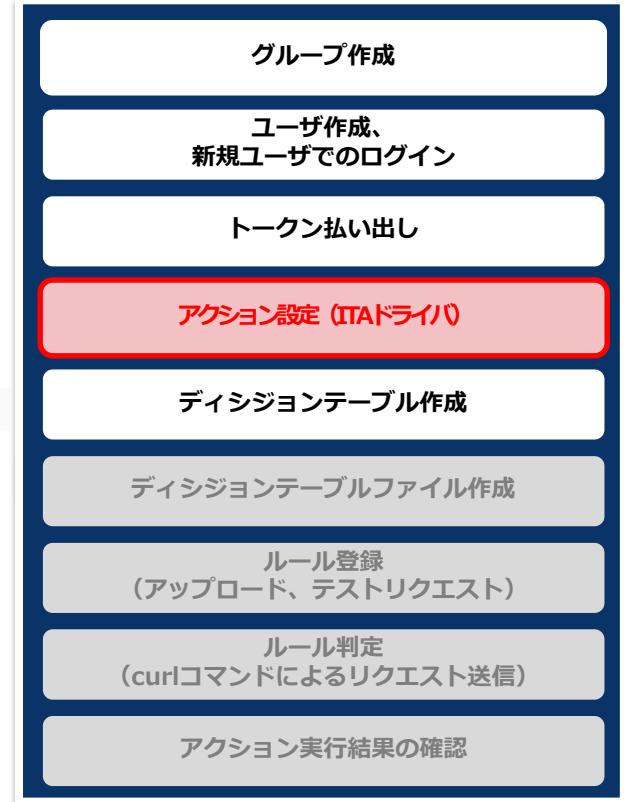
### 3.4 アクション設定 (ITAドライバ) (4/4)

#### OASE-ITA連携に必要な以下の設定を実施する

- 「管理コンソール」メニュー群 > 「ロール・メニュー紐付管理」メニュー > 「一覧/更新」サブメニューより、「Symphony紐付Movement一覧」が廃止されているのを復活させる

「復活」ボタンを押下する

Symphony紐付Movement一覧



### 3.5 ディシジョンテーブル作成 (1/2)

#### ディシジョンテーブルの作成

- ① 「ディシジョンテーブル」画面の「新規追加」ボタンを押下
- ② 「新規追加」画面の「基本情報・権限」タブに必要情報を入力
- ③ 「条件式の設定へ」ボタンを押下



2 以下の値を入力する

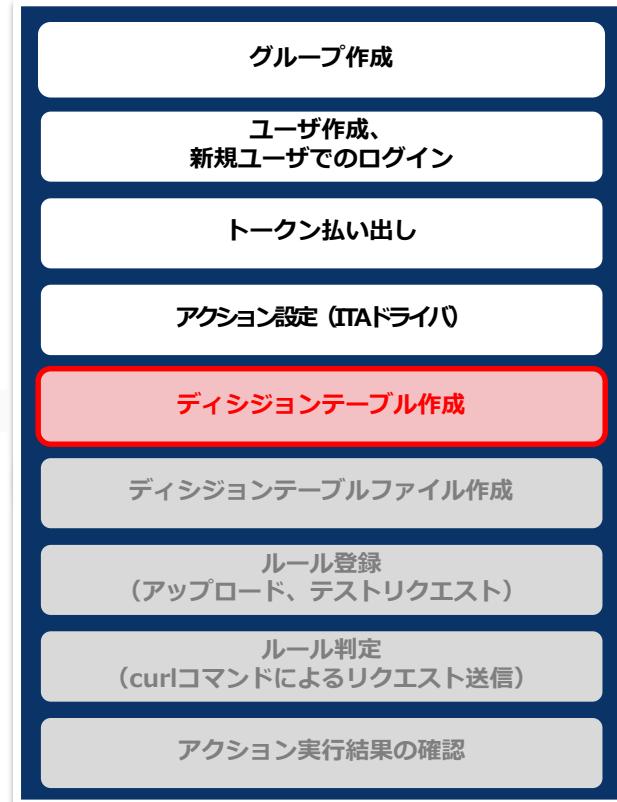
#### 「基本情報・権限」タブ

項目	設定値
ディシジョンテーブル名	(任意の文字列)
権限の設定	test_group : 全て「更新可能」

3

#### POINT

「権限の設定」では最低でも1グループは必ず「更新可能」を設定してください。  
ディシジョンテーブルの更新ができなくなります。



### 3.5 ディシジョンテーブル作成 (2/2)

#### ディシジョンテーブルの作成

- ④ 「新規追加」画面の「条件式」タブに必要情報を入力
- ⑤ 「未知事象通知の設定へ」ボタンを押下
- ⑥ 「新規追加」画面の「未知事象通知」タブに必要情報を入力
- ⑦ 「保存」ボタンを押下

「条件式」タブ	
項目	設定値
条件名	(任意の文字列)
条件式	プルダウン選択

上記の値を入力する



4

「未知事象通知」タブ	
項目	設定値
未知事象通知	「メールで通知する」を選択
連絡先メールアドレス	(受信可能なメールアドレス)

上記の値を入力する

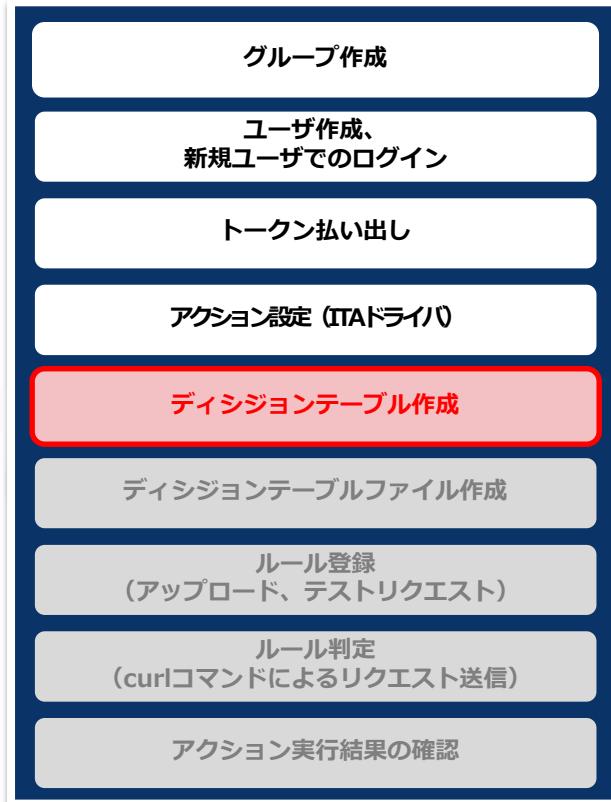


6



5

7



#### POINT

設定した条件式には、後述するディシジョンテーブルファイルの「条件部」で具体値を設定します。

## 4. 作業実行



## 4.1 ディシジョンテーブルファイル作成 (1/3)

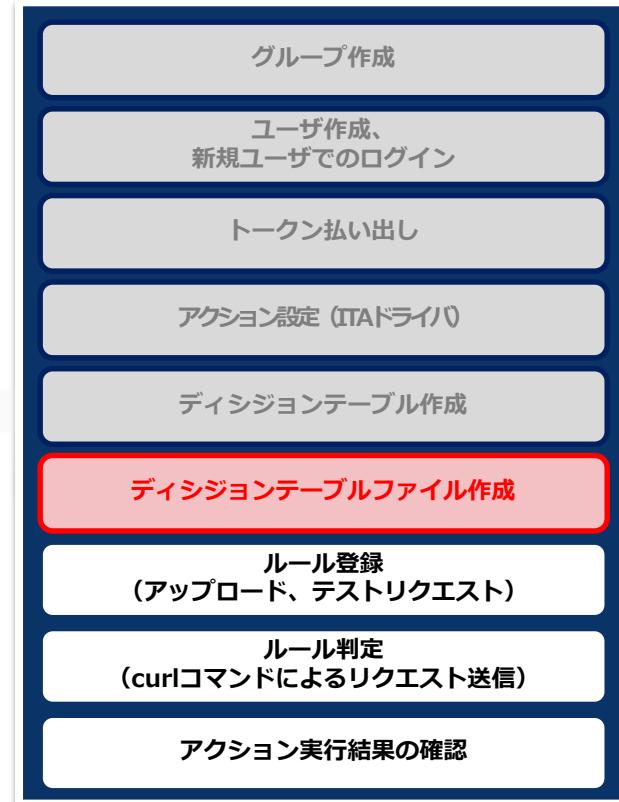
### ディシジョンテーブルファイルのダウンロードおよび作成

- 「3.2 ディシジョンテーブルの作成」で作成したディシジョンテーブルの「ダウンロードボタン」を押しディシジョンテーブルファイルをダウンロードする

操作	ディシジョンテーブル名	最終更新者	最終更新日時
	test_dt	test_user	2021年 06月 24日 22:32
	test_dt_01	test_user	2021年 06月 24日 22:33
	test_dt_02	test_user	2021年 06月 24日 22:34

件数 3 表示する件数 50 / 1 + ▲ ▼

Exastro Operation Autonomy Support Engine



### POINT

ディシジョンテーブルファイルの名前は自動生成されます（例「id0000000000.xlsx」）。  
先述の「ディシジョンテーブル名」とは異なります。  
各項目の記述内容については次のページで説明します。

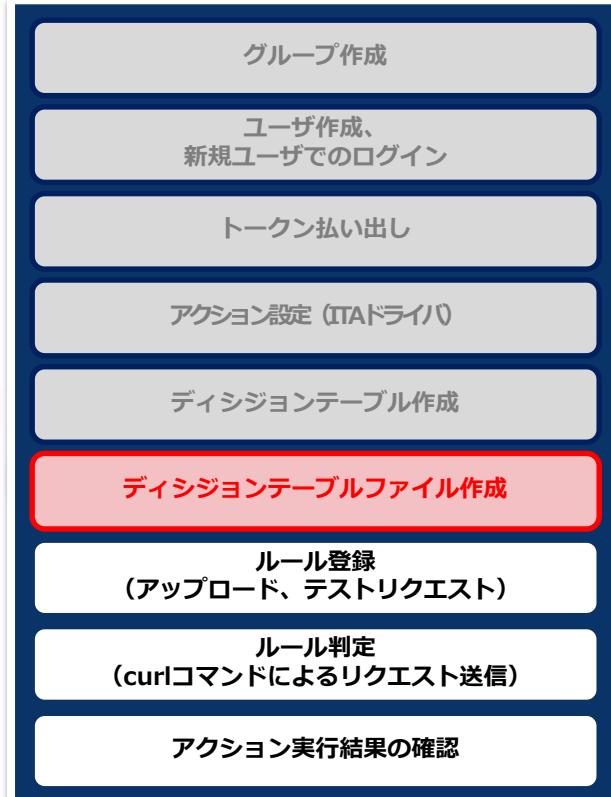
## 4.1 ディシジョンテーブルファイル作成 (2/3)

### ディシジョンテーブルファイルに以下の内容を記述作成

\*具体的なディシジョンテーブルファイルの記述例は後述の[付録 サンプル1](#)を参照

A	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
11	①	②	対象(正規表現可一致)	ルール名(必須)	アクション種別(必須)	アクションパラメータ情報(必須) ※ダブルクオーテーションは使用不可	アクション実行前パラメータ情報(必須) ※ダブルクオーテーションは使用不可	リトライ回数(必須)	リトライ間隔(秒)(必須)	抑止回数(必須)	件数(必須)	条件期間(秒)(必須)	大グループ(必須) ※不要の場合は「X」を定義	優先順位(必須) ※不要の場合は「X」を定義	小グループ(必須) ※不要の場合は「X」を定義	優先順位(必須)	有効日	無効日
12	ルール1	^.*Error.*\$	^.*HDD.*\$	check_Error	ITA(ver1)	ITA_NAME=test_ita,SYMPHONY_CLASS_ID=1,OPERATION_ID=1	X	1	1	0	0	X	X	X	X	X		
13	ルール2	^.*Warning.*\$	^.*memory.*\$	check_Warning	なし	X	X	1	1	0	0	X	X	X	X	X		
14	ルール3	^.*info.*\$	^.*%.*\$	check_Info	なし	X	X	1	1	0	0	X	X	X	X	X		
18	「ルールを複数選択する場合は、12行目以降をコピー、挿入によって行を複数作成してください。」																	

①コメント部	空白可。説明文など、自由なテキスト記述に使用可能。
②条件部	ルールがマッチングする条件を作成する。
③アクション部	<ul style="list-style-type: none"><li>ルール名ごとにどのようなアクションを実行するか設定可能。</li><li>「アクション種別」に指定可能なのは「アクション設定」画面で登録したドライバのみ。</li><li>アクション種別ごとに「アクションパラメータ情報」の書き方が異なるため要注意。</li><li>アクションを実行してもよいか、事前承認メールを送る設定も可能。</li></ul>
④アクション条件部	空白可。ルールを適用する期間の始まりから終わりまでを設定することが可能。



#### POINT

値の記述方法はディシジョンテーブルファイルの「記述例」シートを参照ください。  
ディシジョンテーブルファイルの更新後、任意の名称にファイル名を変更することができます。

## 4.1 ディシジョンテーブルファイル作成 (3/3)

### アクションパラメータ情報の記入方法

- 以下の項目を入力する

No.	アクション パラメータ情報	設定値
1	ITA_NAME	OASEの「ITA Driver ver1」タブ>「名前」に登録している名称
2	SYMPHONY_CLASS_ID	ITAの「Symphony」メニュー群>「Symphony作業実行」メニュー>「Symphony[一覧]」サブメニューに記載のID
3	OPERATION_ID	ITAの「Symphony」メニュー群>「Symphony作業実行」メニュー>「オペレーション[一覧]」サブメニューに記載のID

アクション種別  
(必須) アクションパラメータ情報 (必須)  
※ダブルクオーテーションは使用不可

ITA(ver1) ITA\_NAME=test\_ita, SYMPHONY\_CLASS\_ID=1, OPERATION\_ID=1

アクション設定

ITA Driver ver1

操作 名前 バージョン プロトコル  
test\_ita 1.7.1 http 10.1

①

Exastro IT Automation Symphony

Menu

Symphony[フィルタ] Symphony[一覧]

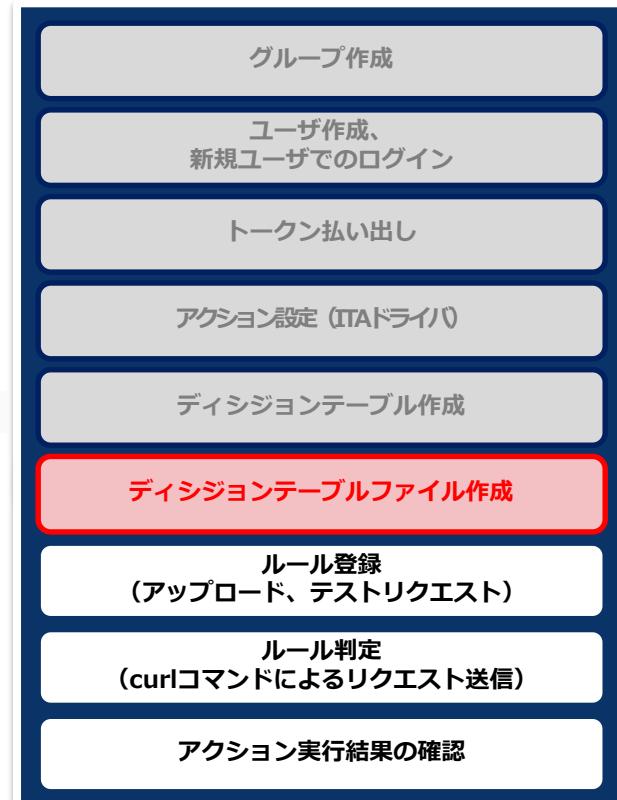
選択 SymphonyクラスID ⇄ Symphony名称 ⇄ 説明 ⇄ アクセス権  
workflow

フィルタ結果件数: 1 ②

オペレーション[フィルタ] オペレーション[一覧]

選択 オペレーションID ⇄ オペレーション名 ⇄ 実施予定日 ⇄ 最終実行日  
operation

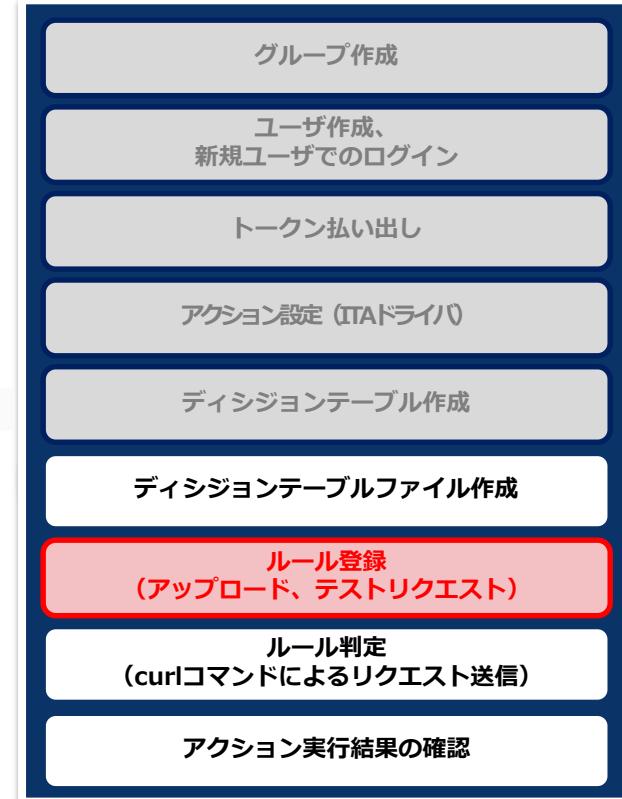
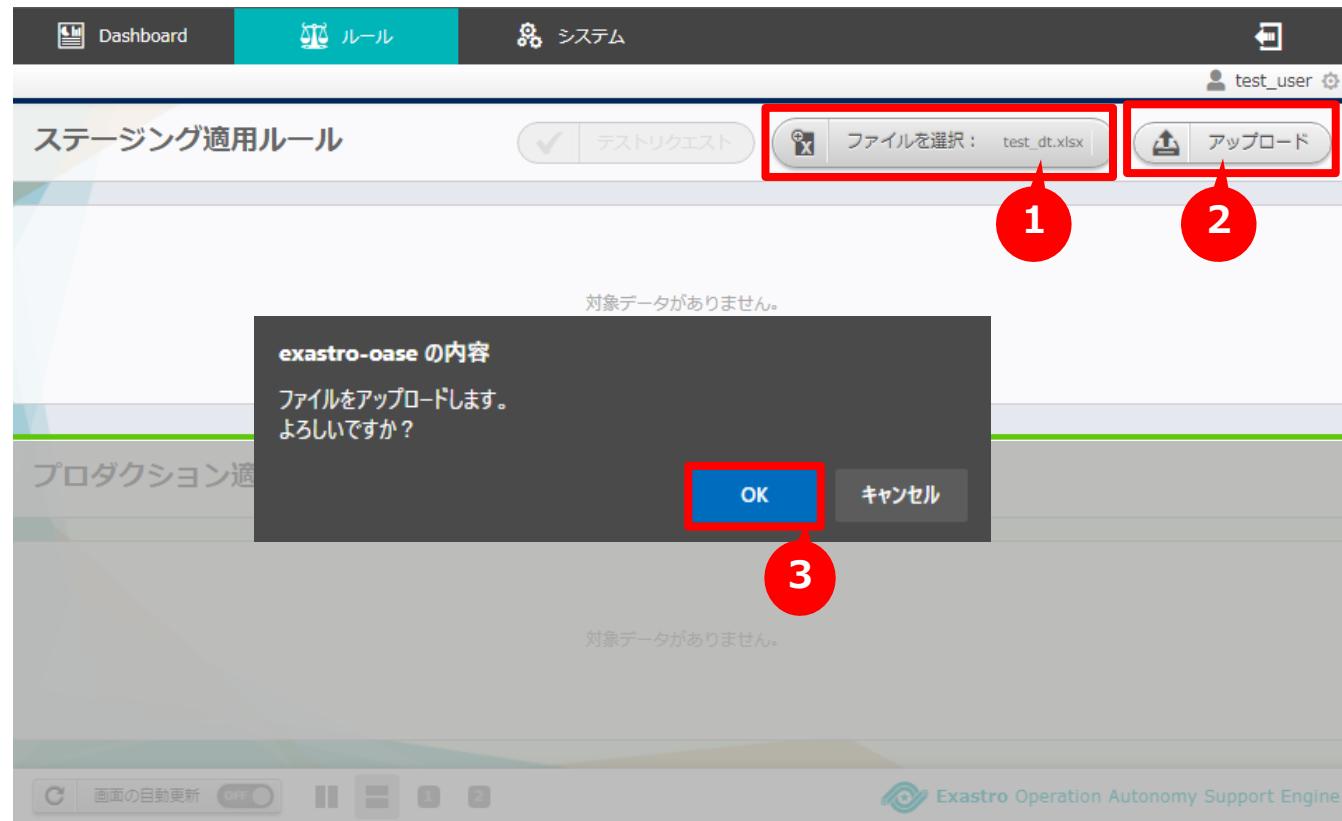
フィルタ結果件数: 1 ③



## 4.2 ルール登録（アップロード、テストリクエスト）(1/6)

### 【手順】テストリクエストしたいディシジョンテーブルファイルを選ぶ

- ① 「ルール」画面の「ファイルを選択」ボタンを押し作成したディシジョンテーブルファイルを選択
- ② 「アップロード」ボタンを押下
- ③ ダイアログの「OK」ボタンを押下

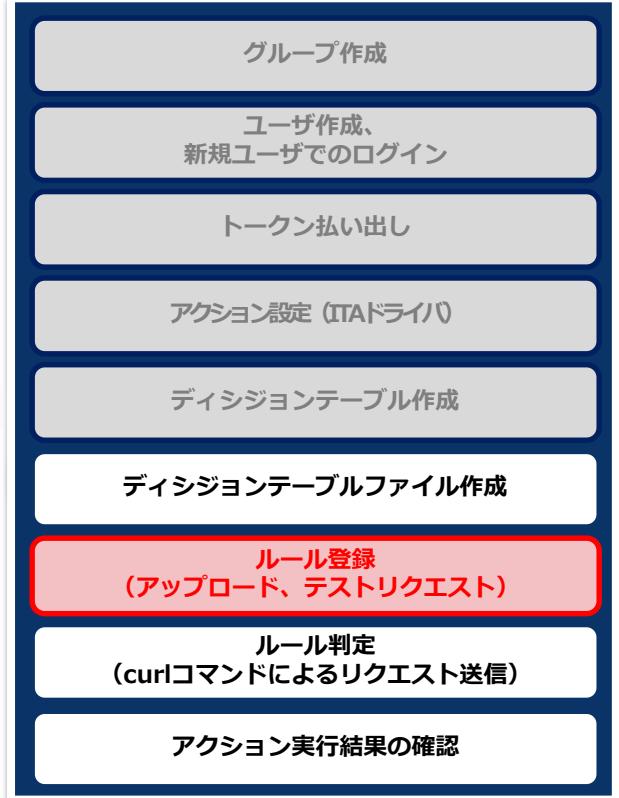


## 4.2 ルール登録（アップロード、テストリクエスト）(2/6)

### テストリクエスト対象の選択

- ① 「作業ステータス」欄が「ステージング適用完了」に遷移後「テストリクエスト」ボタンを押下
- ② 「ディシジョンテーブル」タブの「ディシジョンテーブル名選択欄にて、テストしたいディシジョンテーブル名を選択
- ③ 「テストリクエスト設定へ」ボタンを押下

The screenshot shows the 'Rules' tab of the Exastro interface. A red box highlights the 'Test Request' button in the top right corner of the main panel. Another red box highlights the 'Deployment Status' column, which shows 'Staging Applied Completed' for the selected row. A third red box highlights the 'Deployment Status' dropdown menu in the sub-panel, which also shows 'Staging Applied Completed'. Red circles numbered 1, 2, and 3 point to these respective elements.



### POINT

作業ステータスは5秒間隔で自動的に更新されます。作業ステータスの遷移については<[利用手順マニュアル - ルール画面編- \(1\)ルール画面\(ステージング\)](#)>を参照ください。

## 4.2 ルール登録（アップロード、テストリクエスト）(3/6)

### テストで値を入れて実行する

- ① 「設定」タブ内にて「一括テスト」タブを選択
- ② 「一括テスト用Excelファイルのダウンロード」ボタンを押下
- ③ 作成したルールに合致する値をファイルに記述

1

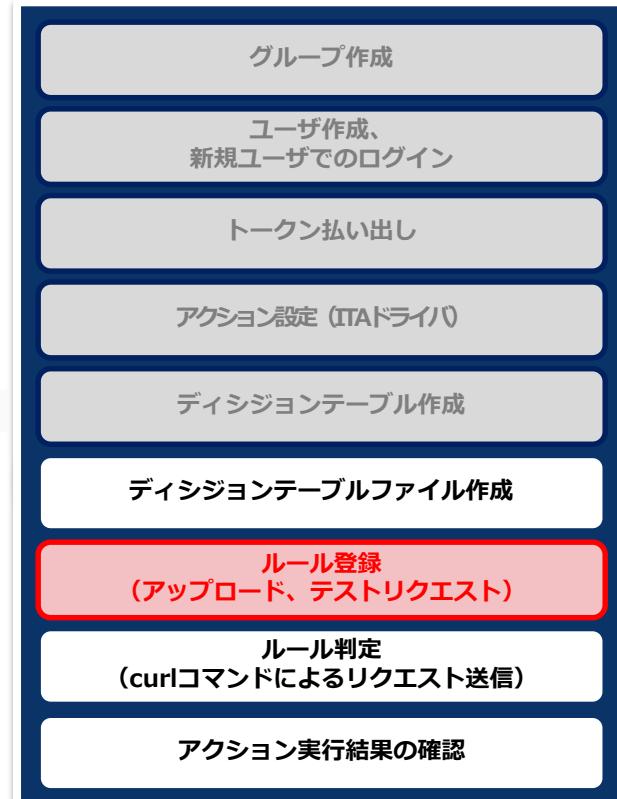
2

3

リクエスト名	イベント発生日時	アラートレベル	対象
ルール1用	2019-5-17 1:20:30	Error:	HDD usage 80% over
ルール2用	2019-5-17 1:20:30	Warning:	memory usage 80% over
ルール3用	2019-5-17 1:20:30	[info]	HDD usage 20% over

↑リクエストを増やす場合は、4行目以降をコピー、挿入によって行を増やしてください

Values example +



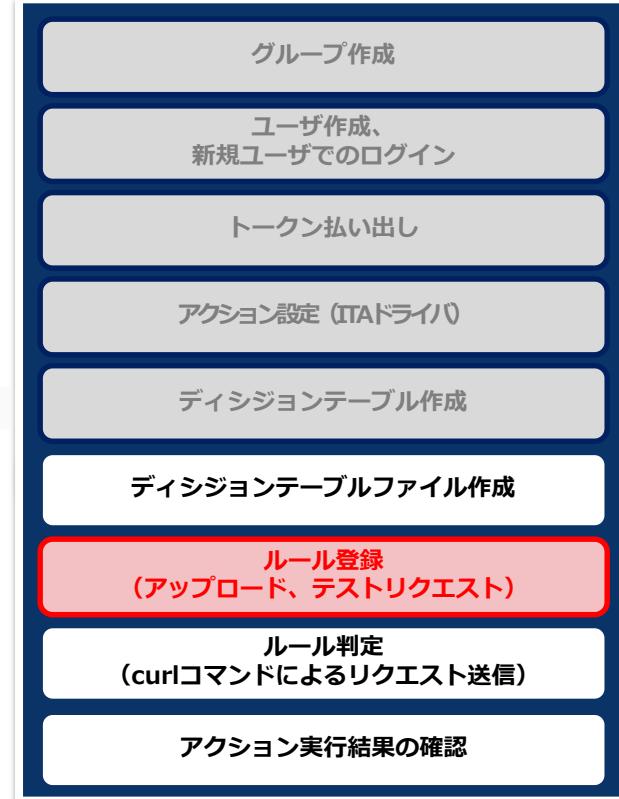
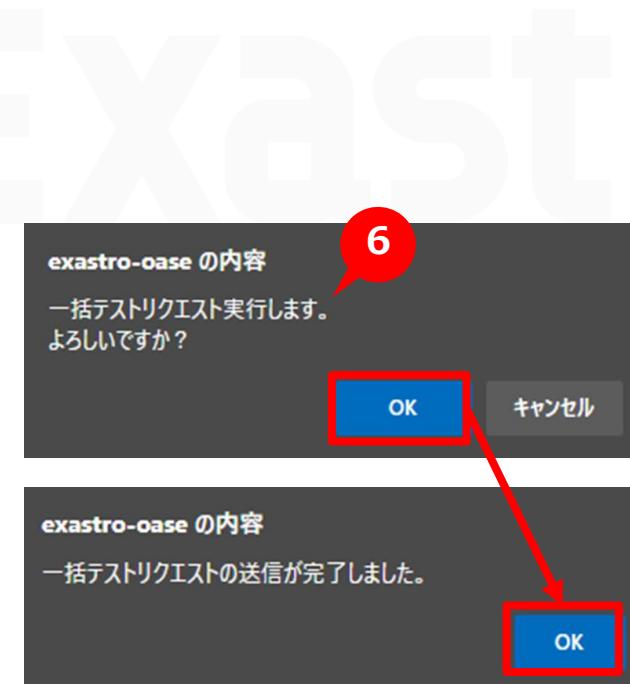
### POINT

作成したディシジョンテーブルファイルの「条件部」に合致する値か否かテストします。

## 4.2 ルール登録（アップロード、テストリクエスト）(4/6)

### テストで値を入れて実行する

- ④ 「ファイルを選択する」ボタンを押し下し、更新した一括テスト用Excelファイルを選択
- ⑤ 「実行」ボタンを押下
- ⑥ ダイアログの「OK」ボタンを押下

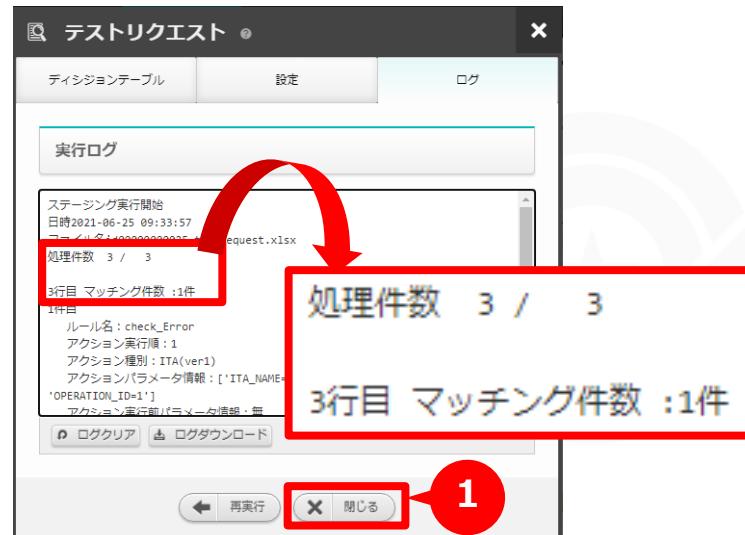


## 4.2 ルール登録（アップロード、テストリクエスト）(5/6)

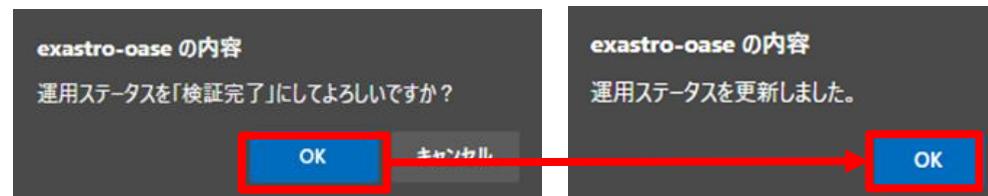
### ルールが有効か確認する

- 「ログ」タブの「実行ログ」欄にてログを確認

  - 「閉じる」ボタンを押下
  - ダイアログの「OK」ボタンを押下



- 正常に処理されルールがマッチングした場合
  - ダイアログの「OK」ボタンを押下



### POINT

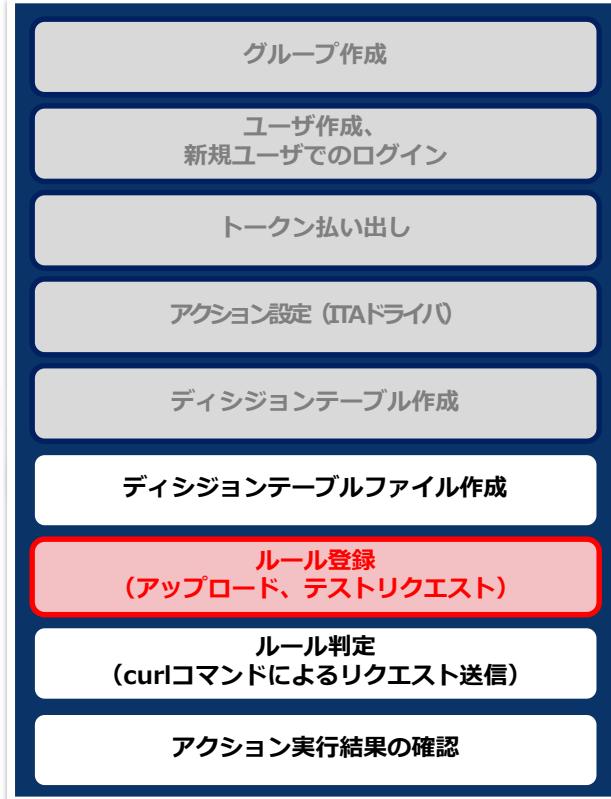
前述の<[4.1 ディシジョンテーブルファイル作成](#)>で作成したルールに合致する場合、「実行ログ」欄に「○行目 マッチング件数：○件」と表示されます。ルールがマッチングすると「運用ステータス」が次のステータスに移ります。

## 4.2 ルール登録（アップロード、テストリクエスト）(6/6)

### 検証完了したルールを本番環境で使用できるようにする

- ルールを本番環境で使用できるようにするために「ステージング適用ルール」から「プロダクション適用ルール」に適用させる

- 「ステージング適用ルール」の「運用ステータス」欄が「検証完了」に遷移していることを確認
- 「操作」欄の「適用ボタン」を押下
- ダイアログの「OK」ボタンを押下



### POINT

作業ステータスは5秒間隔で自動的に更新されます。作業ステータスの遷移については<[利用手順マニュアル -ルール画面編- \(2\)ルール画面\(プロダクション\)](#)>を参照ください。

## 4.3 ルール判定 (curlコマンドによるリクエスト送信) (1/2)

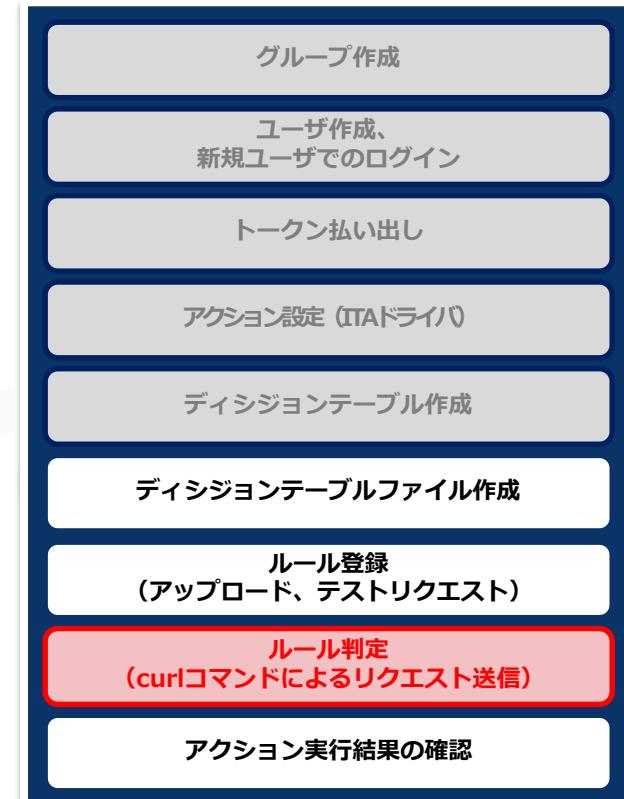
### メッセージを投入しプロダクション適用ルールとマッチングさせる

- ターミナルを開き、以下のコマンドをルールに合わせ書き換えたうえで実行する

```
curl -X POST -k "https://<①ホスト名>/oase_web/event/event/eventsrequest" \
      -H "accept: application/json" \
      -H "Authorization: Bearer <⑥トークン>" \
      -d '{"decisiontable": "<②ディシジョンテーブル名>","requesttype": "<③リクエスト種別>","eventdatetime": "<④イベント発生日時>","eventinfo": ["<⑤イベント情報>"]}'
```

① ホスト名	有効なホスト名、IPアドレスを入力
② ディシジョンテーブル名	プロダクション適用済みのディシジョンテーブル名を入力
③ リクエスト種別	投入先「1:プロダクション」の「1」を入力 例) "requesttype": "1"
④ イベント発生日時	「yyyy/mm/dd hh:mm:ss」形式で日付を入力 例) "eventdatetime": "2020/01/01 01:01:01"
⑤ イベント情報	リスト形式で指定　例) ["2","あああ"]
⑥ トークン	「トークン払い出し」画面で払い出したトークンを入力

※curlコマンドの使用例は後述の[A 付録 サンプル1](#)を参照



### POINT

HTTPSリクエストの詳細については  
[<RestAPI機能 利用マニュアル>](#)を参  
照してください。

## 4.3 ルール判定 (curlコマンドによるリクエスト送信) (2/2)

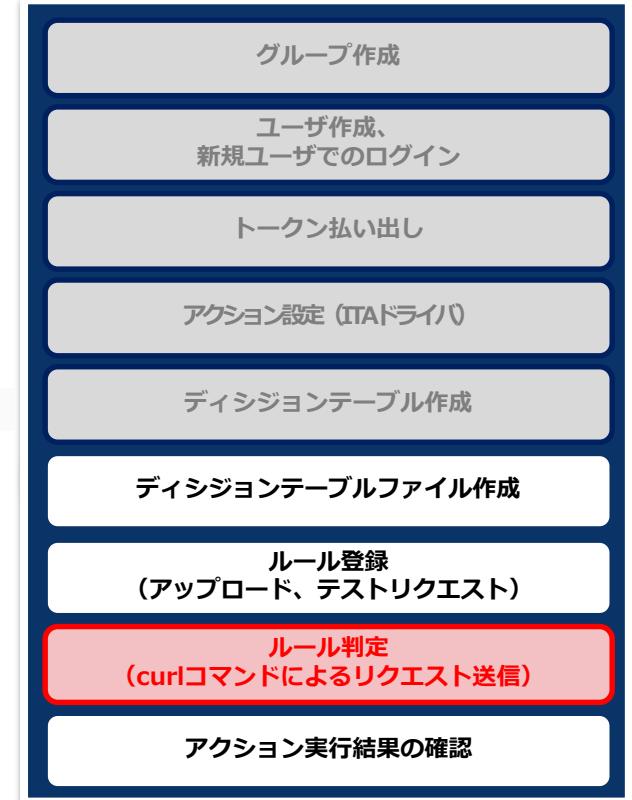
### 「リクエスト履歴」画面を確認

- curlコマンドでパラメータを指定しリクエスト送信した履歴が追加される
- 「リクエスト履歴」画面で表示する項目は「■」ボタン押下により変更可能

ルールマッチ状況	リクエスト種別	ディシジョンテーブル名	リクエスト受信日時	イベント情報
✓	プロダクション環境	test_dt	2021年6月25日9:42	{"EVENT_INFO":["Error:", "HDD usage 80% over"]}
?	ステージング環境	test_dt	2021年6月25日9:33	{"EVENT_INFO":["[info]", " HDD usage 20% over"]}
✓	ステージング環境	test_dt	2021年6月25日9:33	memory usage 80% over"]}]
✓	ステージング環境	test_dt	2021年6月25日9:33	D usage 80% over"]}]
✓	ステージング環境	test_dt	2021年6月25日9:33	usage 80% over","Error: HDD usage 80% over"]}]

件数 5 表示する件数 50 / 1 + ▲ ▼ ■

ルールマッチ状況  
リクエスト種別  
ディシジョンテーブル名  
リクエスト受信日時  
イベント情報  
イベント発生日時  
イベントシリアルNo.



## 4.4 アクション実行結果の確認(1/3)

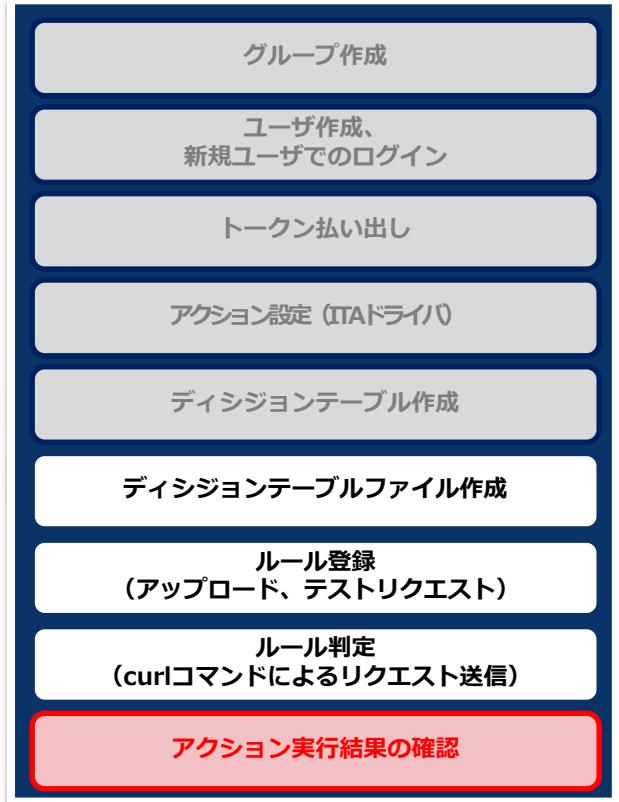
### アクション実行

- ルールマッチした場合、事前設定で設定したとおりアクションが実行され「アクション履歴」画面で結果の確認が可能
- 前述した<[4.1 ディシジョンテーブルファイル作成](#)>のアクション部にて設定した内容でアクションが実行される

### アクション履歴

- ルールマッチングし実行されたルールが「アクション履歴」画面に表示されていることを確認する

アクション履歴						
状態	操作	ディシジョンテーブル名	ルール名	アクション種別	最終実行日時	最終実行者
✓		test_dt	check_Error	ITA(ver1)	2021年 06月 25日 10:49	test_user
!		test_dt	check_Error	ITA(ver1)	2021年 06月 25日 09:45	test_user



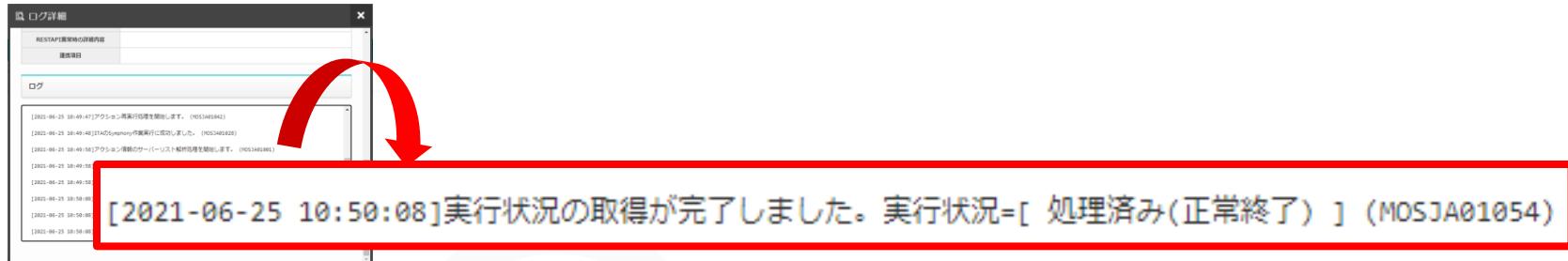
### POINT

アイコンの詳細については<[利用手順マニュアル -アクション履歴編-](#)>を参照してください。

## 4.4 アクション実行結果の確認(2/3)

### ITAで処理が実行されたことを確認する

- OASEのログ詳細より、正常に処理が完了したことを確認



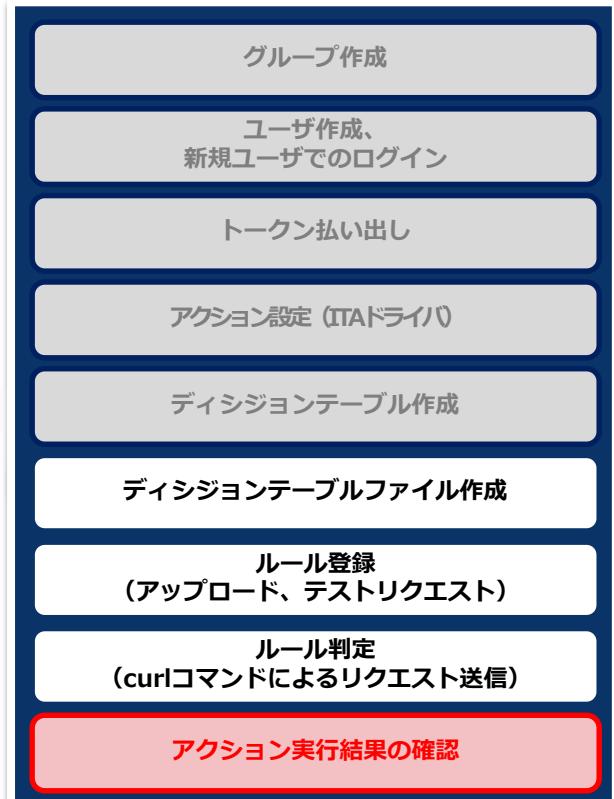
- ITAの「Symphony作業一覧」より、「ステータス：正常終了」となっていることを確認



- 設定した内容で新規にディレクトリが作成されていることを確認

The screenshot shows two terminal sessions. The first session shows the creation of two new directories: './testdirectory' and './oase\_testdirectory'. The second session shows the results of a 'find' command, where these newly created directories are listed under the current directory. A red arrow points from the 'Actions' section of the previous slide to the newly created directory names in the terminal output.

```
tmp]# pwd  
/tmp  
[ tmp]# find ./ -name "*directory" -type d  
. ./testdirectory  
./oase_testdirectory  
[ tmp]#  
tmp]# find ./ -name "*directory" -type d  
./testdirectory  
./oase_testdirectory  
[ tmp]#
```



## 4.4 アクション実行結果の確認(3/3)

### 未知事象の場合

- 既知事象として定義できていないものがリクエスト送信された場合、「[OASE]未知事象通知」というタイトルのメールが届く



The screenshot shows the 'Rule History' section of the OASE interface. A red box highlights the third row of the table, which has a question mark icon in the 'Match Status' column. Below the table, a red box contains the text 'Rule Match Status: Rule Not Found'.

ルールマッチ状況	リクエスト種別	ディシジョンテーブル
✓	プロダクション環境	test_dt
✓	プロダクション環境	test_dt
?	ステージング環境	test_dt
✓	ステージング環境	test_dt
✓	ステージング環境	test_dt
✓	ステージング環境	test_dt

ルールマッチ状況：ルール未検出

2021/06/25 (金) 12:40  
noreply@example.com  
[OASE]未知事象通知  
本メールは、『Operation Autonomy Support Engine』より  
自動送信しています。

=====

設定されたルールにマッチしないイベント情報が OASE に投入されました。  
未知事象となりますのでご確認ください。

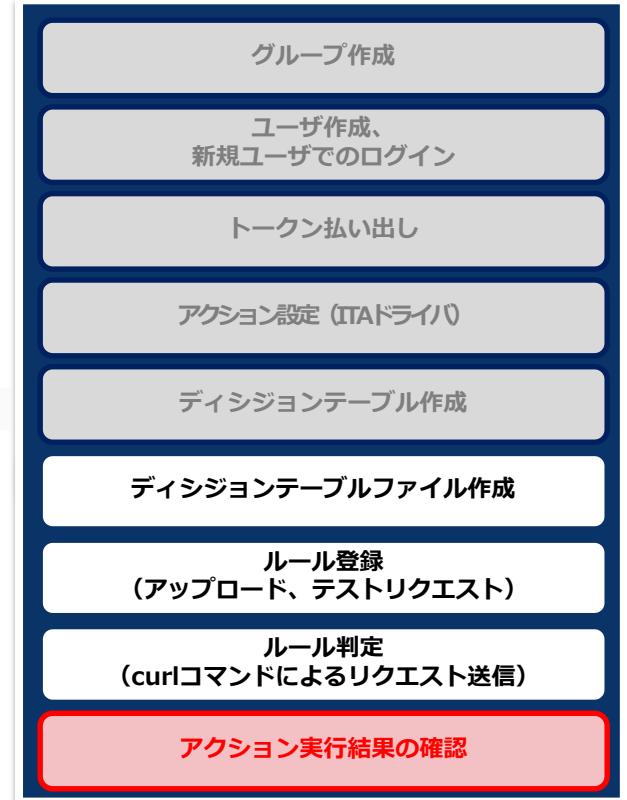
[ディシジョンテーブル名]  
test\_dt

[イベント発生日時]  
2021-04-26 02:20:00+00:00

[リクエスト受信日時]  
2021-06-25 03:39:48.301611+00:00

[イベント情報]  
{"EVENT\_INFO": ["message:", "SKIP"]}

[トレース ID]  
TOS\_20210625033948269850\_0000000144



## A 付録



# サンプル1(1/4)

## サンプル値を入力しOASEを実行する

- 「Error:」および「HDD」というメッセージが投入された場合、ITAが実行され「oase\_testdirectory」という名称のディレクトリが作成されるようとする
- 既知事象ではないメッセージ（「message:」および「SKIP」）を受け取った場合は「[OASE]未知事象通知」というタイトルのメールで通知するようとする

### 【事前設定】

#### ① 「トークン払い出し」画面

- curlコマンドによるリクエスト送信時に必要なトークンを用意する

トークン名 practice\_token

(他、必要情報を登録)

#### ② 「アクション設定」画面

- 「ITA Driver ver1」を用意する

名前 test\_ita

バージョン 1.7.1

(他、必要情報を登録)

#### ③ 「ディシジョンテーブル」画面

- 「アラートレベル」「対象」が合致条件となるディシジョンテーブルを作成する

ディシジョンテーブル名 test\_dt

権限の設定 (test\_group) 全て「更新可能」

条件名	条件式
アラートレベル	正規表現に一致する
対象	正規表現に一致する

未知事象通知 メールで通知する

(他、必要情報を登録)

### POINT

本書内<[3.3 トークンの払い出し](#)>、<[3.4 アクション設定\(ITAドライバ\)](#)>、<[3.5 ディシジョンテーブル作成](#)>の範囲です。

# サンプル1(2/4)

## 【作業実行】

### ④ 「ディシジョンテーブル」ファイル

- ・ディシジョンテーブルファイルをリネームして保存する

ファイル名 test\_dt.xlsx

- ・「アラートレベル：“Error:”」「対象：“HDD”」がヒットするルールを作成する

ルール説明	アラートレベル	対象	ルール名	アクション種別	アクションパラメータ情報
ルール1	^.*Error.*\$	^.*HDD.*\$	check_Error	ITA(ver1)	ITA_NAME=test_ita,SYMPHONY_CLASS_ID=1,OPERATION_ID=1
ルール2	^.*Warning.*\$	^.*memory.*\$	check_Warning	なし	X
ルール3	^.*info.*\$	^.*%.*\$	check_Info	なし	X

(他、ディシジョンテーブルファイルの「シート：記述例」を参考に必要情報を登録)

### ⑤ 「ルール（ステージング適用ルール）」画面

- ・作成したディシジョンテーブルファイルをアップロードする

ファイルを選択

test\_dt.xlsx

## POINT

本書内<[4.1 ディシジョンテーブルファイル作成](#)>、<[4.2 ルール登録（アップロード、テストリクエスト）](#)>の範囲です。

# サンプル1(3/4)

## ⑥「テストリクエスト」画面

- アップロードしたディシジョンテーブルファイルに複数のリクエストが来た場合、どのルールがマッチングするかテストする

ディシジョンテーブル名選択

test\_dt

### 一括テストリクエスト

リクエスト名	イベント発生日時	アラートレベル	対象
ルール1用	2019-5-17 1:20:30	Error:	HDD usage 80% over
ルール2用	2019-5-17 1:20:30	Warning:	memory usage 80% over
ルール3用	2019-5-17 1:20:30	[info]	HDD usage 20% over

ファイルを選択

id000000000xx\_testrequest.xlsx

## POINT

本書内 <4.2 ルール登録（アップロード、テストリクエスト）> の範囲です。

# サンプル1(4/4)

## ⑦ターミナル操作 (Linuxサーバ)

- 用意したルールに対しcurlコマンドでリクエストを投げる **※赤字箇所はご変更ください。**

```
curl -X POST -k "https://<Hostname>/oase_web/event/event/eventsrequest" -H "accept: application/json" ¥  
-d '{"decisiontable":"test_dt","requesttype":"1","eventdatetime":"2020/01/01  
01:01:01","eventinfo":["Error:","HDD usage 80% over"]}' -H "Authorization: Bearer <Access_Token>"
```

## ⑧未知事象通知

下記情報のメールが  
届いたことを確認する

件名	[OASE]未知事象通知
本文	[ディシジョンテーブル名] [イベント発生日時] [リクエスト受信日時] [イベント情報] [トレースID]



## POINT

本書内<[4.3 ルール判定 \(curlコマンドによるリクエスト送信\)](#)>および<[4.4 アクション実行結果の確認](#)>の範囲です。



**Exastro** 