

InnoRules

Installation and Operation Guide

InnoRules v7.2



InnoRules Version 7.2 Copyright © 2007 INNORULES Corporation. All rights reserved.

InnoRules® and InnoProduct® are registered trademarks of INNORULES Corporation.

目次

[1. Quick Start 8](#_Toc40449504)

[1.1 インストール前の必要事項 8](#_Toc40449505)

[1.2 インストールウィザードの実行 9](#_Toc40449506)

[1.3 ルールシステム構成ウィザードの実行 13](#_Toc40449507)

[1.4 サービス構成ウィザードの実行 20](#_Toc40449508)

[1.5 InnoRulesサーバの起動 23](#_Toc40449509)

[1.6 ルールビルダーの接続 25](#_Toc40449510)

[1.7 InnoRulesサーバの停止 27](#_Toc40449511)

[2. Wizards 29](#_Toc40449512)

[2.1 インストールウィザード 29](#_Toc40449513)

[2.1.1 導入 30](#_Toc40449514)

[2.1.2 JDKホームディレクトリ 30](#_Toc40449515)

[2.1.3 インストールディレクトリ 31](#_Toc40449516)

[2.1.4 ログディレクトリ 32](#_Toc40449517)

[2.1.5 InnoRulesサーバ設定 32](#_Toc40449518)

[2.1.6 ライセンスファイル 33](#_Toc40449519)

[2.1.7 インストール概要 34](#_Toc40449520)

[2.2 設定クラスタウィザード 34](#_Toc40449521)

[2.2.1 導入 35](#_Toc40449522)

[2.2.2 インストールノード(初回) 35](#_Toc40449523)

[2.2.3 リモートノード設定のためのプロトコル 36](#_Toc40449524)

[2.2.4 プロトコル詳細 36](#_Toc40449525)

[2.2.5 ローカルインストールノードの登録 38](#_Toc40449526)

[2.2.6 リモートインストールノードの登録 38](#_Toc40449527)

[2.2.7 インストールノード 39](#_Toc40449528)

[2.2.8 UUIDファイル 41](#_Toc40449529)

[2.2.9 ロックファイル 41](#_Toc40449530)

[2.3 ルールシステム構成ウィザード 41](#_Toc40449531)

[2.3.1 導入 42](#_Toc40449532)

[2.3.2 設定対象の選択 42](#_Toc40449533)

[2.3.3 単位ルールシステム管理 43](#_Toc40449534)

[2.3.4 業務データベース一覧 53](#_Toc40449535)

[2.4 サービス構成ウィザード 57](#_Toc40449536)

[2.4.1 導入 58](#_Toc40449537)

[2.4.2 サービスタイプ選択 58](#_Toc40449538)

[2.4.3 管理サービス 59](#_Toc40449539)

[2.4.4 ルールビルダーサービス 62](#_Toc40449540)

[2.4.5 TCPコネクタサービス 67](#_Toc40449541)

[2.4.6 ルールWebサービス 73](#_Toc40449542)

[2.4.7 ルールRESTサービス 78](#_Toc40449543)

[2.5 ルールアプリケーション構成ウィザード 83](#_Toc40449544)

[2.5.1 導入 83](#_Toc40449545)

[2.5.2 ルールシステム選択 84](#_Toc40449546)

[2.5.3 ルールアプリケーション設定 84](#_Toc40449548)

[2.5.4 ルールアプリケーション設定の基本情報 85](#_Toc40449549)

[2.5.5 ルールサービスタイプ 87](#_Toc40449550)

[2.5.6 ルールデータベースの接続設定 88](#_Toc40449551)

[2.5.7 ルールデータベースに関するデータソース(データソース参照方式) 89](#_Toc40449552)

[2.5.8 ルールデータベースに関するデータソース(専用コネクションプール方式) 89](#_Toc40449553)

[2.5.9 業務データベース一覧 90](#_Toc40449554)

[2.5.10 業務データベース接続方式の選択 91](#_Toc40449555)

[2.5.11 業務データベースの接続情報(専用コネクションプール接続情報指定) 92](#_Toc40449556)

[2.5.12 業務データベースの接続情報(データソース参照方式) 93](#_Toc40449557)

[2.5.13 リモートルールサービスのタイプ設定 94](#_Toc40449558)

[2.5.14 ルールサービス同時利用ユーザ数の設定 94](#_Toc40449559)

[2.5.15 ルールアプリケーション設定要約および保存 95](#_Toc40449560)

[2.6 アドオンウィザード 95](#_Toc40449561)

[2.6.1 導入 96](#_Toc40449562)

[2.6.2 アドオンタイプの選択 96](#_Toc40449563)

[2.6.3 ユーザ定義関数リスト 97](#_Toc40449564)

[2.6.4 コールバックサービスリスト 98](#_Toc40449565)

[2.6.5 コールバックサービス登録 100](#_Toc40449566)

[2.7 ウィザード使用方法 100](#_Toc40449567)

[2.7.1 入力 101](#_Toc40449568)

[2.7.2 ウィザード使用シナリオ 103](#_Toc40449569)

[2.7.3 設定適用時点 104](#_Toc40449570)

[3. Rule System Concepts 105](#_Toc40449571)

[3.1 InnoRulesをインストールするとは？ 105](#_Toc40449572)

[3.2 設定クラスタ 105](#_Toc40449573)

[3.3 ルールシステム 105](#_Toc40449574)

[3.3.1 単位ルールシステム 105](#_Toc40449575)

[3.3.2 移管 106](#_Toc40449576)

[3.4 業務データベース 107](#_Toc40449577)

[3.5 ルールサービス 108](#_Toc40449578)

[3.6 ルールアプリケーション設定 109](#_Toc40449579)

[3.6.1 単位ルールシステムの選択 110](#_Toc40449580)

[3.6.2 複数のルールアプリケーション設定と設定名の指定 111](#_Toc40449581)

[3.6.3 ログレベル指定 111](#_Toc40449582)

[3.6.4 ルールサービスの形態指定 112](#_Toc40449583)

[3.6.5 ルールデータベースの接続設定(ローカルルールサービスを使用する場合のみ) 112](#_Toc40449584)

[3.6.6 同時利用ユーザ数 114](#_Toc40449585)

[3.6.7 TCPコネクタサーバグループ選択 - リモートルールサービスにのみ該当 116](#_Toc40449586)

[3.6.8 業務データベース接続方法の設定 116](#_Toc40449587)

[3.7 ルールアプリケーションの初期化 117](#_Toc40449588)

[3.7.1 ルールアプリケーションの初期化ルーチン 118](#_Toc40449589)

[3.7.2 ルールアプリケーションの初期化リスナー 121](#_Toc40449590)

[3.7.3 ルールアプリケーション初期化クラス 123](#_Toc40449591)

[3.8 InnoRulesサービス 126](#_Toc40449592)

[3.8.1 InnoRulesサーバ 126](#_Toc40449593)

[3.8.2 管理サービス 127](#_Toc40449594)

[3.8.3 ルールビルダーサービス 128](#_Toc40449595)

[3.8.4 TCPコネクタサービス 136](#_Toc40449596)

[3.8.5 ルールWebサービス 140](#_Toc40449597)

[3.8.6 ルールRESTサービス 142](#_Toc40449598)

[4. エラーコードと処置方法 144](#_Toc40449599)

[4.1 IRE-10XXX 144](#_Toc40449600)

[4.1.1 IRE-10000 144](#_Toc40449601)

[4.1.2 IRE-10001 144](#_Toc40449602)

[4.1.3 IRE-10002 145](#_Toc40449603)

[4.1.4 IRE-10003 145](#_Toc40449604)

[4.1.5 IRE-10004 145](#_Toc40449605)

[4.1.6 IRE-10006 146](#_Toc40449606)

[4.1.7 IRE-10007 146](#_Toc40449607)

[4.1.8 IRE-10008 146](#_Toc40449608)

[4.1.9 IRE-10009 147](#_Toc40449609)

[4.1.10 IRE-10011 147](#_Toc40449610)

[4.1.11 IRE-10012 148](#_Toc40449611)

[4.1.12 IRE-10013 148](#_Toc40449612)

[4.1.13 IRE-10015 149](#_Toc40449613)

[4.1.14 IRE-10016 149](#_Toc40449614)

[4.1.15 IRE-10017 149](#_Toc40449615)

[4.1.16 IRE-10018 149](#_Toc40449616)

[4.1.17 IRE-10019 150](#_Toc40449617)

[4.1.18 IRE-10020 150](#_Toc40449618)

[4.1.19 IRE-10021 150](#_Toc40449619)

[4.1.20 IRE-10022 151](#_Toc40449620)

[4.1.21 IRE-10023 151](#_Toc40449621)

[4.1.22 IRE-10024 151](#_Toc40449622)

[4.1.23 IRE-10025 152](#_Toc40449623)

[4.1.24 IRE-10026 152](#_Toc40449624)

[4.1.25 IRE-10027 152](#_Toc40449625)

[4.1.26 IRE-10028 153](#_Toc40449626)

[4.1.27 IRE-10029 153](#_Toc40449627)

[4.1.28 IRE-10030 153](#_Toc40449628)

[4.1.29 IRE-10031 153](#_Toc40449629)

[4.1.30 IRE-10032 154](#_Toc40449630)

[4.1.31 IRE-10033 154](#_Toc40449631)

[4.1.32 IRE-10034 154](#_Toc40449632)

[4.1.33 IRE-10035 155](#_Toc40449633)

[4.1.34 IRE-10036 155](#_Toc40449634)

[4.1.35 IRE-10037 155](#_Toc40449635)

[4.1.36 IRE-10038 156](#_Toc40449636)

[4.1.37 IRE-10039 156](#_Toc40449637)

[4.1.38 IRE-10040 156](#_Toc40449638)

[4.1.39 IRE-10041 156](#_Toc40449639)

[4.1.40 IRE-10042 157](#_Toc40449640)

[4.1.41 IRE-10043 157](#_Toc40449641)

[4.1.42 IRE-10044 158](#_Toc40449642)

[4.1.43 IRE-10045 158](#_Toc40449643)

[4.1.44 IRE-10046 158](#_Toc40449644)

[4.1.45 IRE-10047 159](#_Toc40449645)

[4.1.46 IRE-10048 159](#_Toc40449646)

[4.1.47 IRE-10049 159](#_Toc40449647)

[4.1.48 IRE-10050 160](#_Toc40449648)

[4.1.49 IRE-10051 160](#_Toc40449649)

[4.1.50 IRE-10052 160](#_Toc40449650)

[4.1.51 IRE-10500 161](#_Toc40449651)

[4.1.52 IRE-10501 161](#_Toc40449652)

[4.1.53 IRE-10502 161](#_Toc40449653)

[4.2 IRE-11XXX 161](#_Toc40449654)

[4.2.1 IRE-11000 162](#_Toc40449655)

[4.2.2 IRE-11001 162](#_Toc40449656)

[4.2.3 IRE-11002 162](#_Toc40449657)

[4.2.4 IRE-11003 163](#_Toc40449658)

[4.2.5 IRE-11004 163](#_Toc40449659)

[4.3 IRE-12XXX 163](#_Toc40449660)

[4.3.1 IRE-12100\* 163](#_Toc40449661)

[4.3.2 IRE-12101\* 164](#_Toc40449662)

[4.3.3 IRE-12102\* 164](#_Toc40449663)

[4.3.4 IRE-12103\* 164](#_Toc40449664)

[4.3.5 IRE-12104\* 164](#_Toc40449665)

[4.3.6 IRE-12105\* 165](#_Toc40449666)

[4.3.7 IRE-12106\* 165](#_Toc40449667)

[4.3.8 IRE-12107\* 165](#_Toc40449668)

[4.3.9 IRE-12108\* 165](#_Toc40449669)

[4.3.10 IRE-12109\* 166](#_Toc40449670)

[4.3.11 IRE-12110\* 166](#_Toc40449671)

[4.3.12 IRE-12111\* 166](#_Toc40449672)

[4.3.13 IRE-12112 167](#_Toc40449673)

[4.3.14 IRE-12113 167](#_Toc40449674)

[4.3.15 IRE-12200 168](#_Toc40449675)

[4.3.16 IRE-12300\* 168](#_Toc40449676)

[4.3.17 IRE-12301 168](#_Toc40449677)

[4.3.18 IRE-12400\* 169](#_Toc40449678)

[4.3.19 IRE-12401 169](#_Toc40449679)

[4.3.20 IRE-12402\* 169](#_Toc40449680)

[4.3.21 IRE-12997 170](#_Toc40449681)

[4.3.22 IRE-12998 170](#_Toc40449682)

図目次

[[図 1 IRBuilder.exeのアイコン] 25](#_Toc40084695)

[[図 2 ルールビルダーのログイン画面] 26](#_Toc40084696)

[[図 3 接続情報の入力画面] 26](#_Toc40084697)

[[図 4 初期ログイン情報の入力画面] 27](#_Toc40084698)

[[図 5 ログイン正常実行後画面] 27](#_Toc40084699)

[[図 6 移管の例] 106](#_Toc40084700)

[[図 7 移管の制約] 107](#_Toc40084701)

[[図 8 Business Database] 107](#_Toc40084702)

[[図 9 Remote Rule Server] 109](#_Toc40084703)

[[図 10 Referencing a Data Source from a Rule Web Application] 114](#_Toc40084704)

[[図 11 Rule Interface] 115](#_Toc40084705)

[[図 12 Examples of Rule Application Initializations] 117](#_Toc40084706)

[[図 13 An Example of Management Service Configuration] 128](#_Toc40084707)

[[図 14 Monitor Session] 129](#_Toc40084708)

[[図 15 An Example of Rule Builder Server Clustering] 131](#_Toc40084709)

[[図 16 Single Configuration Cluster] 132](#_Toc40084710)

[[図 17 Separated Configuration Cluster] 133](#_Toc40084711)

[[図 18 Multi-Step Rule System] 135](#_Toc40084712)

[[図 19 Another Multi-Step Rule System] 136](#_Toc40084713)

[[図 20 TCP Connector Service] 137](#_Toc40084714)

[[図 21 An Example of a Replication] 139](#_Toc40084715)

[[図 22 An Example of a Replication with Local First Option] 139](#_Toc40084716)

# Quick Start

この章では、InnoRulesをインストールして構成する方法を説明する。インストール先のシステム構成は、次のとおりである。

* ルールシステムは、開発と運用の2つの単位ルールシステムで構成される。
* ルールビルダーサービスを有効にする。

## インストール前の必要事項

InnoRules Quick Startを実行するためには、次の環境を用意する必要がある。

* InnoRules用OSアカウント

OSにログインしてインストールを実行し、InnoRulesサーバを実行するためのアカウントである。書き込み権限が付与されている必要がある。

* JDK 6.0以上
* 最小インストール空き容量30MB

InnoRulesがインストールされるファイルシステム上の必要な最小の空き容量である。

* InnoRulesインストールウィザード：innorules-inst.jar

InnoRulesをインストールする実行可能なjarファイルである。

* InnoRulesライセンスファイル
* データベースサーバおよびアカウント

データベースは、各単位ルールシステムのルールデータを保存するために使用され、単位システムごとに異なるデータベースアカウントが必要である。本マニュアルでは、開発、運用システムを構成するため2つのデータベースアカウントを設定している。なお、データベースアカウントには次の権限が必要である。

* データベース接続
* テーブル作成
* データ作成、削除および照会
* JDBCドライバ

設定ウィザードおよびInnoRulesサーバが上記のデータベースにアクセスするために必要である。

* ルールビルダー

ルール管理クライアントツールで、ルールビルダーサービスを呼び出す。

## インストールウィザードの実行

InnoRulesインストールウィザードは、コマンドラインベースのアプリケーションである。インストールするためには、コンソールにログインする必要がある。本例では、UNIXターミナルにおける実行を例に挙げるが、Windows環境でも実行方法は変わらない。インストールウィザードの実行前に、次の表のようにインストールするホストにインストールウィザードファイルとライセンスファイルをコピーする。

[表 1]

|  |
| --- |
| Innorules:~/install$ ls  innorules-inst.jar innorules.lic  Innorules:~/install$ |

InnoRulesインストールウィザードを次の表のように実行する。環境変数のPATHにJAVA\_HOMEが登録されていない場合は、Javaのフルパスを入力する。

[表 2]

|  |
| --- |
| Innorules:~/install$ java -jar innorules-inst.jar |

インストールウィザードを実行すると、次の表のように[導入]画面が出力される。インストールするInnoRulesのバージョンを確認し、事前確認事項を確認する。本例では、InnoRulesサーバを利用してルールビルダーサービスを有効にするため、サービスのHTTPサービスポートと制御ポートの割り当てが必要である。そのため、ポートの割り当てについてはシステム管理者に確認すること。準備完了後、エンターキーを押して次に進む。

[表 3]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRules 7.2.0のインストールへようこそ。インストールする前に以下の事前確認事項を確認してください。  - JDK6.0（またはそれ以上）  - InnoRulesインストールディレクトリは、少なくとも30MB以上の空き容量が必要です。  - （選択事項）HTTPサービスポートと制御ポートが割り当てられていなければなりません。  [エンターキーを押下してください。] |

インストールウィザードおよびInnoRulesサーバが使用するJDKの場所を指定する([表 4])。インストールウィザードは、デフォルト値としてJDK6.0(「1.6 ルールビルダーの接続」)のインストールディレクトリを表示する。デフォルト値として表示されたJDKをそのまま使用する場合はエンターキーを押し、他のJDKを使用する場合は直接パスを入力し、エンターキーを押す。

[表 4]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  JDKホームディレクトリ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  JDKがインストールされたディレクトリを指定してください。JDKバージョンは6.0（1.6）以上でなければなりません。(デフォルト値: /opt/java/sun-jdk-1.6.0-64)  >> |

InnoRulesをインストールするディレクトリを指定する([表 5])。デフォルトとしてユーザホームディレクトリのinnorules-homeが指定される。変更する場合は、インストール対象ディレクトリのフルパスを入力する。その際に、ウィザードを実行しているシステムアカウントへインストールディレクトリの書き込み権限が設定されている必要がある。

[表 5]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストールディレクトリ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesをどのディレクトリにインストールしますか？(デフォルト値: /home/innorules/innorules-home)  >> |

InnoRulesで提供するサービスから出力されるログファイルのパスを、次の表のように指定する。デフォルトパスは、InnoRulesサーバのデフォルトログディレクトリである。変更する場合は、ログファイルディレクトリのフルパスを入力する。ただし、InnoRulesサーバにログディレクトリへの書き込み権限が設定されている必要がある。InnoRulesサービスについては、「3.8 InnoRulesサービス」を参照のこと。

[表 6]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ログディレクトリ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ログファイルを出力するディレクトリを指定してください。InnoRulesサービスとルールランタイムが該当ディレクトリにログを記録します。(デフォルト値: /home/innorules/innorules-home/innorules-server/logs)  >> |

InnoRulesサーバで使用するサービスポートと制御ポートを、次の表のように指定する。これらのポートは、InnoRulesサーバが使用する。InnoRulesサーバを実行しない場合は任意の値を許容する。ビルダーサービスのようなInnoRulesサービスを使用する場合、システム管理者と協議してサービスポートと制御ポートを付与してもらう必要がある。

[表 7]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesサーバ設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesサーバは管理サービス、ルールビルダーサービス等のサービスを提供する独立サーバです。サーバはこのインストールノードで実行される可能性があるので、先にポートを設定することをお勧めします。また、制御ポートはInnoRulesサーバを停止するために使用します。  ポート番号を入力してください。  [必須] HTTPサービスポート(デフォルト値: 25802) >> |

InnoRulesサーバが使用するポートを指定する([表 8])。InnoRulesサーバは基本的に2つのポートを使用するが、1つはHTTPサービスのためのポートで、もう1つはサーバを制御するためのポートである。サービスポートと制御ポートを入力し、エンターキーを押して次に進む。

[表 8]

|  |
| --- |
| …  ポート番号を入力してください。  [必須] HTTPサービスポート(デフォルト値: 25802) >>  [必須] 制御ポート(デフォルト値: 25805) >> |

次は、製品のライセンスを入力する段階である([表 9])。インストールウィザードは、作業ディレクトリで使用できるライセンスファイルを検索して表示する。ライセンスファイルの設定は、本手順のようにインストール時に一覧から使用するライセンスファイルを指定をするか、インストール後にもライセンスファイルを所定のフォルダにコピーすることで設定することもできる。

[表 9]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ライセンスファイル  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  現作業ディレクトリから次のライセンスファイルを発見しました。どのライセンスファイルをインストールしますか？  (\*) 1. innorules.lic  Test (Expires: 2016-01-31)  2. インストール後、手作業でコピー  選択してください。 >> |

インストールウィザードの作業ディレクトリにライセンスファイルがない場合は、ライセンスファイルの一覧ではなく[表 10]のメッセージが出力される。この場合でも、インストール作業には支障はないが、インストール完了後に該当ディレクトリへライセンスファイルを手作業でコピー後にInnoRulesを使用する必要がある。

[表 10]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ライセンスファイル  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  現作業ディレクトリからライセンスファイルを発見できませんでした。インストール後に手作業でライセンスファイルをコピーする必要があります。  [エンターキーを押下してください。] |

InnoRulesの[インストール概要]画面である。次の表のように今まで設定した内容の概要が表示されるので確認する。確認後、問題なければYesまたはYを入力し、エンターキーを押して次に進む。

[表 11]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストール概要  JDK Home : /opt/java/default-jdk  Install Directory : /home/innorules/innorules-home  Service Port : 25802  Admin Port : 25805  License File : /home/innorules/install/innorules.lic  この設定でInnoRulesをインストールします。よろしいですか？ (Yes/No) >> Yes |

正常にインストールが完了すると、次の表のインストール成功メッセージが出力される。エンターキーを押してインストールを終了する。

[表 12]

|  |
| --- |
| [情報] Installation succeeded.[エンターキーを押下してください。] |

InnoRulesインストールの完了後、インストールウィザードで指定したInnoRulesインストールディレクトリの下位ディレクトリへ次の表のようにフォルダが作成されたことの確認ができる。

[表 13]

|  |
| --- |
| innorules:~/innorules-home$ ls  config innorules-server lib license manual-config script |

## ルールシステム構成ウィザードの実行

インストール完了後に、単位ルールシステムを構成する。単位ルールシステムを構成するために、ルールを保存するデータベースのアクセス権の設定が必要である。データベースにアクセスするために、「1.1 インストール前の必要事項」で準備したJDBCドライバをインストールディレクトリのlib/jdbc-driversにコピーする。例えば、ルールデータをOracleデータベースに保存する場合(かつJDK6を使用する場合)は、次の表のようにojdbc6.jarをlib/jdbc-driversにコピーする。

[表 14]

|  |
| --- |
| innorules@:~/innorules-home/lib/jdbc-drivers$ ls  ojdbc6.jar  innorules@:~/innorules-home/lib/jdbc-drivers$ |

次の表のように、ルールシステム構成ウィザードを実行する。Windowsでは、system-wiz.batを実行する。

[表 15]

|  |
| --- |
| innorules@:~/innorules-home/script$ ./system-wiz.sh |

次の表は、ルールシステム構成ウィザードの[導入]画面である。

[表 16]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム構成ウィザードへようこそ。このウィザードではルールシステムを構成できます。  [エンターキーを押下してください。] |

次の表のように、ルールシステムウィザードを利用して、単位ルールシステムへの設定、およびルールシステムからアクセスする業務データベースへの設定ができる。本例では、開発と運用の2つの単位ルールシステムを設定する。

[表 17]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象を選択してください。  1. 単位ルールシステム管理  単位ルールシステム設定を管理します。  2. 業務データベース管理  業務データベース設定を管理します。  選択してください。 >> 1 |

単位ルールシステム管理を選択すると、登録された単位ルールシステムの一覧が表示される。次の表の例では、登録された単位ルールシステムがないため、一覧は空である。‘a’を入力して追加を選択する。

[表 18]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  登録された単位ルールシステムのリストです。新規単位ルールシステムの追加および既存単位ルールシステムの修正ができます。  [項目]  -----------------  | ID | 名称 |  -----------------  [可能な作業]  a. 追加  作業を選択してください。スキップするにはエンターキーを押下してください。 >> a |

まず、追加対象の単位ルールシステムのIDと名前を入力する([表 19])。IDは、10文字以内の識別子でアルファベットの大小文字、数字またはアンダースコア(‘\_’)のみで構成する。例では、開発段階という意味で‘DEV’を入力する。名称は、任意の文字(60文字以内)とする。

[表 19]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステムの基本情報を入力してください。  [必須] ID >> DEV  [必須] 名称 >> development |

単位ルールシステムの基本情報の入力後、この単位ルールシステムのルールデータを保存するデータベースを設定する。次の表のように、ルールデータベースの名称と説明を入力する。

[表 20]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDBの基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールDBの基本情報を入力してください。情報は一度設定すると変更できません。  [必須] 名称 >> RULEDB  [必須] 説明(デフォルト値: ) >> |

次はデータベースのタイプを選択する段階である([表 21])。サポートするDBMS一覧がウィザードに登録されている。使用するデータベースがこの一覧にない場合は、技術サポートが必要である。

本例では、Oracleを選択する。

[表 21]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDBサーバタイプ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールDBとして使用するデータベースサーバのタイプを選択してください。  1. Oracle  Oracle database (with type 4 driver)  2. MSSQL  Microsoft SQL server  3. DB2  IBM DB2 with DB2 UDB JDBC universal driver  4. MySQL  MySQL database server  5. Postgre  Postgre database server  6. Hsqldb  Hsqldb database server  選択してください。 >> 1 |

データベースサーバの選択後、次の表のようにサーバに接続するための[ルールDBの接続情報]画面が表示される。なお、入力必須情報は、選択したデータベースにより異なる。

[表 22]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDBの接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールDB'DEV\_RULE\_DB'に接続するための情報を入力してください。  [必須] address >> your\_database\_ip  [必須] port(デフォルト値 : 1521) >> your\_database\_port  [必須] SID or service name(デフォルト値 : ORCL) >> your\_database\_sid  [必須] ID >> your\_database\_account  [必須] パスワード >> your\_database\_password |

[表 22]の項目の値をすべて入力すると、次の表のようにルールデータベース登録確認メッセージが出力される。入力したすべての値が正しいか確認し、正しい場合は、YesまたはYを入力する。

[表 23]

|  |
| --- |
| この情報を用いてルールDBを登録しますか？登録する場合、この接続情報を用いてDBにログイン後、テーブルの生成およびデータの修正ができます。(Yes/No) >> Yes |

[表 23]でYesを入力後、ウィザードは[表 22]の情報を利用してデータベースサーバに接続し、一連のテーブルを作成する。既にテーブルが作成されている場合は、テーブルおよびデータの整合性をチェックする。

現在の単位ルールシステムが開発システムであり、ルールデータベースの初回登録である場合、次の表のように基本日付形式を入力する。基本日付形式は日付形式を使用するルール文法で使用され、デフォルト値はyyyy-MM-ddである。この形式はJavaのSimpleDateFormat形式に従う。ここではエンターキーを押して次に進む。

[表 24]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステムで使用する基本日付形式を入力してください。基本日付形式は、日付形式を使用するルール文法で使用されます。この形式は、JavaのSimpleDateFormat形式に従います。 (デフォルト値: yyyy-MM-dd)  >> |

正常に登録された場合、[ルールDB管理]画面が表示され、単位ルールシステムDEVに登録されたルールデータベースの一覧が表示される([表 25])。作業を選択しない状態でエンターキーを押した場合、次に進む。

[表 25]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDB管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'DEV'に登録されたルールDBのリストです。新規ルールDBの登録および既存ルールDBの情報を修正できます。  [項目]  --------------------------------------------------------------------------  | 名称 | JDBC-URL | ID | 説明 |  --------------------------------------------------------------------------  1 | RULEDB | jdbc:oracle:thin:@ip:port: SID | your account | |  --------------------------------------------------------------------------  [可能な作業]  a. 追加 b. 情報更新 c. 上に移動 d． 下に移動  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

開発単位ルールシステムの登録が完了し、[単位ルールシステム管理]画面にDEVシステムが登録されたことが表示される([表 26])。

[表 26]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  登録された単位ルールシステムのリストです。新規単位ルールシステムの追加および既存単位ルールシステムの修正ができます。  [項目]  ------------------------  | ID | 名称 |  ------------------------  1 | DEV | Development |  ------------------------  [可能な作業]  a. 追加 b. 修正およびルールDB管理 c. 最終単位ルールシステム削除  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

DEV単位ルールシステムの登録と同様の方法で、運用単位ルールシステムを追加する。次の表は、PRODというIDにて単位ルールシステムを追加した例である。

[表 27]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  登録された単位ルールシステムのリストです。新規単位ルールシステムの追加および既存単位ルールシステムの修正ができます。  [項目]  -------------------------  | ID | 名称 |  -------------------------  1 | DEV | Development |  -------------------------  2 | PROD | Production |  -------------------------  [可能な作業]  a. 追加 b. 修正およびルールDB管理 c. 最終単位ルールシステム削除  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

## サービス構成ウィザードの実行

ここでは、サービス構成ウィザードを実行しルールビルダーサービスを構成する。サービス構成ウィザードに関する詳細については、「4. エラーコードと処置方法」を参照のこと。

次の表のように、サービス構成ウィザードを実行する。Windowsではservice-wiz.batを実行する。

[表 28]

|  |
| --- |
| innorules@:~/innorules-home/script$ ./service-wiz.sh |

サービス構成ウィザードを実行すると、次の表のように[導入]画面が出力される。エンターキーを押して次に進む。

[表 29]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesサービス構成ウィザードへようこそ。このウィザードではホストに様々なサービスを設定するサポートを行います。  [エンターキーを押下してください。] |

InnoRulesで提供されるサービスを構成できる[サービスタイプ選択]画面が、次の表のように出力される。ルールビルダーサービスを構成するために‘2’を入力し、エンターキーを押して次に進む。

[表 30]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定したいサービスのタイプを選択してください。終了する場合、エンターキーを押下してください。  1. 管理サービス  登録されたルールランタイムシステムとルールシステムコンポーネントの管理  ができます。  2. ルールビルダーサービス  ルールの作成および管理ができます。  3. TCPコネクタサービス  リモートアプリケーションがルールサービスを呼び出すことができます。  4. ルールWebサービス  ルールWebサービスを設定しWeb環境でルールを呼び出せます。  5. ルールRESTサービス  ルールRESTサービスを設定してWeb環境にてルールを呼び出します。  選択してください。 >> 2 |

[ビルダーサービス構成]画面が、次の表のように出力される。ビルダーサービスを有効にするためには‘d’を入力し、エンターキーを押して次に進む。

[表 31]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ビルダーサービス  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ビルダーサービスに対する作業を選択してください。  [グローバル設定]  移管専用 : 未設定  平均同時利用ユーザ数: 2  最大同時利用ユーザ数: 10  暗号化方式 : DES  テーブルルール条件式チェック: : 未設定  クラスタ接続タイムアウト(ms) : 5,000  クラスタレスポンスタイムアウト(ms) : 60,000  クラスタ文字セット : UTF-8  [項目]  --------------------------------------------------------------------  | 名称 | アドレス | サービスポート | モニターポート | ローカル |  --------------------------------------------------------------------  [可能な作業]  c. グローバル設定変更 d. ローカルインストールノードのビルダーサービスを有効化  作業を選択してください。スキップするにはエンターキーを押下してください。 >> d |

[モニターサーバ接続情報登録]画面が、次の表のように出力される。ビルダーサーバの機能であるモニターサーバが使用するアドレスとポートを選択する。モニターポートは、システム担当者またはネットワーク担当者に問い合わせること。デフォルト値は25810である。

[表 32]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ビルダーサービス -> モニターサーバの接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  モニターサーバの接続情報を入力してください。  [必須] 接続するアドレス  1. 192.168.0.88  >> 1  [必須] ポート(デフォルト値: 25810) >> |

ビルダーサービスの構成が正常に完了すると、次の表のようにビルダーサービス一覧に追加されたサーバが表示される。今回は作業を選択せずにエンターキーを押して、サービスウィザードを終了する。

[表 33]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ビルダーサービス  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ビルダーサービスに対する作業を選択してください。  [グローバル設定]  移管専用 : 未設定  平均同時利用ユーザ数: 2  最大同時利用ユーザ数: 10  暗号化方式 : DES  テーブルルール条件式チェック : 未設定  クラスタ接続タイムアウト(ms) : 5,000  クラスタレスポンスタイムアウト(ms) : 60,000  クラスタ文字セット : UTF-8  [項目]  ------------------------------------------------------------------------------------  | 名称 | アドレス | サービスポート | モニターポート | ローカル |  ------------------------------------------------------------------------------------  1 | 906946d6-8a78-4877-9930-f9d160254d46 | 192.168.0.88 | 25802 | 25810 | V |  ------------------------------------------------------------------------------------  [可能な作業]  a. 上に移動 b. 下に移動 c. グローバル設定変更 d. ローカルインストールノードのビルダーサービスを無効化 e. ビルダーサービス接続URLを確認  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

## InnoRulesサーバの起動

InnoRulesインストールが正常に完了した後、InnoRulesサーバを起動する。InnoRulesサーバに関する詳細については、「3.8.1 InnoRulesサーバ」を参照のこと。

InnoRulesサーバはインストールディレクトリのスクリプトにより、次の表のように次のコマンドで実行する。Windowsの場合は、start-server.batを実行する。

[表 34]

|  |
| --- |
| innorules@:~/innorules-home/script$ ./start-server.sh |

InnoRulesサーバが正常に起動すると、次の表のように出力される。

[表 35]

|  |
| --- |
| Using CATALINA\_BASE: /home/innorules/innorules-home/innorules-server/default  Using CATALINA\_HOME: /home/innorules/innorules-home/innorules-server/apache-tomcat-7  Using CATALINA\_TMPDIR: /home/innorules/innorules-home/innorules-server/default/temp  Using JRE\_HOME: /opt/java/default-jdk  Using CLASSPATH: :/home/innorules/innorules-home/lib/jdbc-drivers/\*:/home/innorules/innorules-home/innorules-server/apache-tomcat-7/bin/bootstrap.jar:/home/innorules/innorules-home/innorules-server/apache-tomcat-7/bin/tomcat-juli.jar  Tomcat started. |

InnoRulesサーバのプロセスは、次の表のように、コンソールから次のコマンドより確認できる。

[表 36]

|  |
| --- |
| innorules@siyang-ubuntu-pc:~$ ps -aef | grep catalina  innorules 8442 1 8 18:02 pts/1 00:00:04 /opt/java/default-jdk/bin/java -Djava.util.logging.config.file=/home/innorules/innorules-home/innorules-server/default/conf/logging.properties -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager -Dinnorules.home=/home/innorules/innorules-home -Dinnorules.log=/home/innorules/innorules-home/innorules-server/logs -Dinnorules.server.uuid=906946d6-8a78-4877-9930-f9d160254d46 -Dinnorules.server.port=25802 -Dinnorules.server.admin-port=25805 -Djava.endorsed.dirs=/home/innorules/innorules-home/innorules-server/apache-tomcat-7/endorsed -classpath :/home/innorules/innorules-home/lib/jdbc-drivers/\*:/home/innorules/innorules-home/innorules-server/apache-tomcat-7/bin/bootstrap.jar:/home/innorules/innorules-home/innorules-server/apache-tomcat-7/bin/tomcat-juli.jar -Dcatalina.base=/home/innorules/innorules-home/innorules-server/default -Dcatalina.home=/home/innorules/innorules-home/innorules-server/apache-tomcat-7 -Djava.io.tmpdir=/home/innorules/innorules-home/innorules-server/default/temp org.apache.catalina.startup.Bootstrap start  innorules@:~/innorules-home/script$ |

netstatコマンドを利用して、次の表のようにモニターサーバの使用ポートが確認できる。

[表 37]

|  |
| --- |
| innorules@:~/innorules-home/script$ netstat -a | grep 25810  tcp6 0 0 192.168.0.88%4163:25810 [::]:\* LISTEN  tcp6 0 0 192.168.0.88%4163:25810 192.168.0.88%4168:56791 ESTABLISHED  tcp6 0 0 192.168.0.88%4163:25810 192.168.0.88%4168:56792 ESTABLISHED  tcp6 0 0 192.168.0.88%4163:56792 192.168.0.88%4168:25810 ESTABLISHED  tcp6 0 0 192.168.0.88%4163:56791 192.168.0.88%4168:25810 ESTABLISHED  innorules@siyang-ubuntu-pc:~$ |

InnoRulesサーバのログは、次の表に示すパスのログファイルから確認できる。

[表 38]

|  |
| --- |
| ~/innorules-home/innorules-server/logs/system/innorules-server.log |

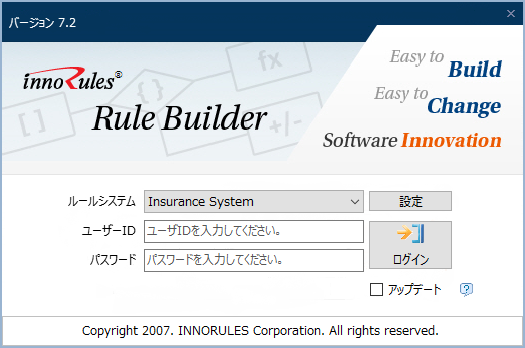
## ルールビルダーの接続

ここでは、ルール開発のためのクライアントツールであるルールビルダーを利用してルールビルダーサービスを実行する。ルールビルダーはWindows用アプリケーションであり、リモートでルールビルダーサービスを実行できる。クライアントPCにルールビルダーをインストールして、次の図のようにルールビルダーを実行する。



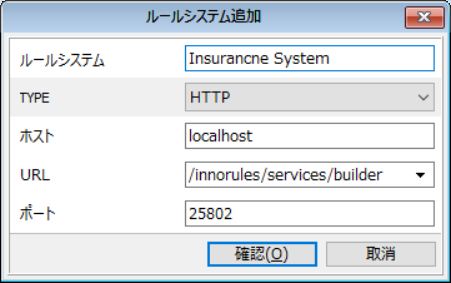
[図 1 IRBuilder.exeのアイコン]

次の図は、[ルールビルダーのログイン]画面である。



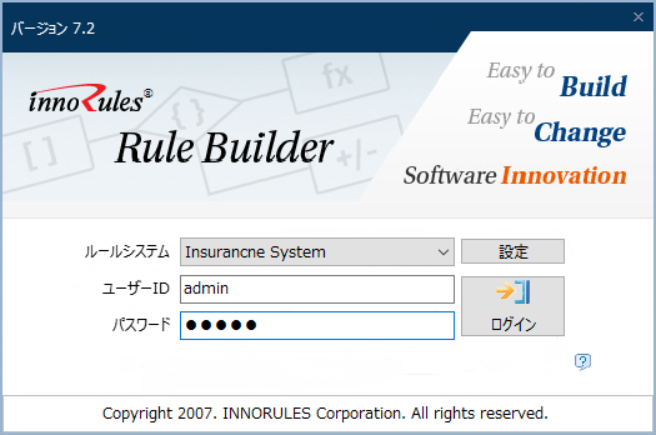
[図 2 ルールビルダーのログイン画面]

[設定]ボタンをクリックしルールビルダーサービスの接続情報を入力する([図 3 接続情報の入力画面])。ホストには、InnoRulesをインストールしたホストのアドレスを入力する。URLは/innorules/services/builderを設定する。ポートに、「1.2 インストールウィザードの実行」で設定したHTTPサービスポートを入力する。



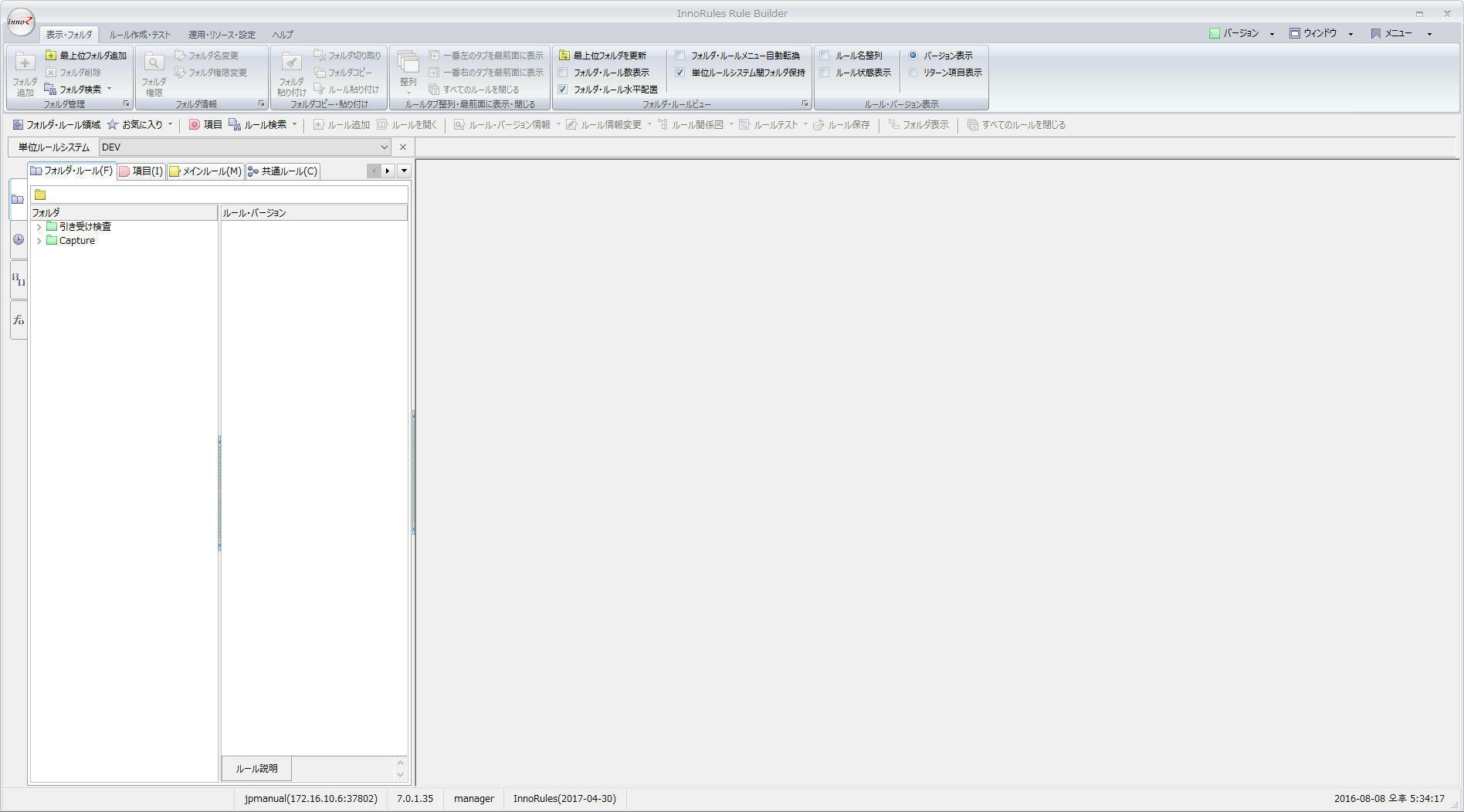
[図 3 接続情報の入力画面]

InnoRulesサーバ接続情報を入力後、次の図のように初回接続時に‘admin’ユーザで接続する。‘admin’ユーザの初期パスワードは、‘admin’である。InnoRulesサーバのインストール直後には‘admin’ユーザのみ存在する。



[図 4 初期ログイン情報の入力画面]

初回ログインに必要な設定が正常に完了した場合、次の図のようにルールビルダーがInnoRulesサーバに接続される。ルールビルダー使用方法については「InnoRules User‘s Guide of Rule Builder」を参照のこと。



[図 5 ログイン正常実行後画面]

## InnoRulesサーバの停止

InnoRulesサーバは、インストールディレクトリの下位のscriptディレクトリにて次の表のコマンドを実行し停止する。Windowsの場合は、stop-server.batを実行する。

[表 39]

|  |
| --- |
| innorules@:~/innorules-home/script$ ./stop-server.sh |

InnoRulesサーバが正常に停止された場合、次の表のように出力される。

[表 40]

|  |
| --- |
| Using CATALINA\_BASE: /home/innorules/innorules-home/innorules-server/default  Using CATALINA\_HOME: /home/innorules/innorules-home/innorules-server/apache-tomcat-7  Using CATALINA\_TMPDIR: /home/innorules/innorules-home/innorules-server/default/temp  Using JRE\_HOME: /opt/java/default-jdk  Using CLASSPATH: :/home/innorules/innorules-home/lib/jdbc-  drivers/\*:/home/innorules/innorules-home/innorules-server/apache-tomcat-7/bin/bootstrap.jar:/home/innorules/innorules-home/innorules-server/apache-tomcat-7/bin/tomcat-juli.jar |

# Wizards

InnoRulesでは、製品をインストールして構成を設定できるように、次のウィザードが提供される。

* インストールウィザード(Installation Wizard)
* 設定クラスタウィザード(Configuration Cluster Wizard)
* ルールシステム構成ウィザード(Rule System Configuration Wizard)
* サービス構成ウィザード(Service Configuration Wizard)
* ルールアプリケーション構成ウィザード(Rule Application Configuration Wizard)
* アドオンウィザード(Addon Wizard)

インストールウィザードはInnoRulesをインストールするために使用され、インストール完了後は他のウィザードを必要に応じて実行する。この章では各ウィザードの機能と使用方法について説明し、代表的なシナリオにてウィザードの実行手順を説明する。

## インストールウィザード

インストールウィザードは、InnoRulesを対象ホストにインストールする実行可能なjarファイルである。インストールディレクトリのディレクトリとファイルを、インストールノード(Installation Node)という。同一ホストでもパスの異なる複数のノードをインストールできる。

1つのインストールノードには、単位ルールシステム設定、InnoRulesサービスの各種設定、ルールアプリケーション設定が作成される場合がある。これらの設定は、互いに関連性および依存性がある。そのため、完全に独立した2つのルールシステムを1つのインストールノードに構成することはできない。独立したルールシステムを構成するには、InnoRulesを別のディレクトリにインストールしてそのノードに設定する必要がある。

それぞれのインストールノードは、独立したインストールディレクトリに位置する。したがって、インストールディレクトリを指定する際には、既に別のインストールノードが使用しているディレクトリまたはその下位ディレクトリを指定できない。また、それぞれのインストールノードには固有の識別子が付与されるため、ディレクトリのコピーなど適切な方法以外のインストールをしてはならない。

インストールウィザードは、次の表のコマンドを利用して実行する。

[表 41]

|  |
| --- |
| java -jar innorules-inst.jar |

### 導入

次の表は、インストールウィザードの導入段階である。この段階では、InnoRulesをインストールするのに必要なシステム要求事項を表示する。

[表 42]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRules 7.2.0のインストールへようこそ。インストールする前に以下の事前確認事項を確認してください。  - JDK6.0（またはそれ以上）  - InnoRulesインストールディレクトリは、少なくとも30MB以上の空き容量が必要です。  - （選択事項）HTTPサービスポートと制御ポートが割り当てられていなければなりません。  [エンターキーを押下してください。] |

### JDKホームディレクトリ

次の表は、JDKホームディレクトリを選択する段階を示している。InnoRulesは、JavaベースのアプリケーションでJDK 6.0以上を必要とする。この段階で、各種ウィザードおよびInnoRulesサーバが使用するJDKのパスを指定する。

[表 43]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  JDKホームディレクトリ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  JDKがインストールされたディレクトリを指定してください。JDKバージョンは6.0（1.6）以上でなければなりません。(デフォルト値: /opt/java/sun-jdk-1.6.0-64)  >> |

インストールウィザードは、インストールウィザードを実行したJAVA\_HOMEを利用してJDKがインストールされているディレクトリをデフォルト値として表示する。JDKの場所を確認できない場合は、デフォルト値が表示されない。

指定したディレクトリが誤ったJDKディレクトリである場合、またはインストールされたJDKが破損し適切な実行権限がない場合は、エラーメッセージが出力される。また、指定したJDKが6.0以上でない場合にもエラーメッセージが出力される。

次は、エラーメッセージの例である。

* パスが正しくない場合

[表 44]

|  |
| --- |
| [エラー] The path is not a directory.[エンターキーを押下してください。]  or  [エラー] There is no bin directory.[エンターキーを押下してください。] |

* JDKのバージョンが製品前提条件を満たさない場合

[表 45]

|  |
| --- |
| [エラー] JDK version is lower than 6.0.[エンターキーを押下してください。] |

### インストールディレクトリ

次の表は、インストールディレクトリを指定する段階である。インストールディレクトリは、次の条件を満たす必要がある。

* 既に存在するディレクトリの場合は、空のディレクトリであり、ウィザードを実行したOSのユーザに下位ディレクトリとファイルの作成権限が必要。
* ディレクトリが存在しない場合は、ウィザードを実行したOSのユーザにディレクトリの作成権限が必要。
* ディレクトリには、最小30MBの空き容量が必要である。この空き容量は、初回インストール時に必要な容量である。インストール後にも、InnoRulesサーバのログ、サードパーティライブラリ、JDBCドライバの追加により容量の増加が発生する。必要容量を算出する時は、この容量も考慮し算出する必要がある。

[表 46]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストールディレクトリ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesをどのディレクトリにインストールしますか? （デフォルト値：/home/user/innorules-home）  >> |

### ログディレクトリ

次の表は、ログディレクトリを指定する段階である。InnoRulesサーバとルールアプリケーションは、指定されたディレクトリ下位にディレクトリを作成してログを保存する。

[表 47]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ログディレクトリ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ログファイルを出力するディレクトリを指定してください。InnoRulesサービスとルールランタイムが下位ディレクトリにログを記録します。(デフォルト値: /home/user/innorules-home/innorules-server/logs)  >> |

デフォルト値は、インストールディレクトリのinnorules-server/logsディレクトリである。「2.1.3 インストールディレクトリ」で説明したように、インストールに必要な最小容量にはログファイルが考慮されていないため、既定のログディレクトリを使用する場合は、ログ容量を考慮する必要がある。

InnoRulesサーバのみでなく、ルールアプリケーションについてもこのディレクトリにログを記録するため適切に権限を付与する必要がある。システムに、別のログディレクトリが存在する場合、容量や権限管理の観点から、そのログディレクトリの利用を推奨する。

### InnoRulesサーバ設定

次の表は、InnoRulesサーバが使用するポートを設定する段階である。

[表 48]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesサーバ設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesサーバは管理サービス、ルールビルダーサービス等のサービスを提供する独立サーバです。サーバはこのインストールノードで実行される可能性があるので、先にポートを設定することをお勧めします。また、制御ポートはInnoRulesサーバを停止するために使用します。  ポート番号を入力してください。  [必須] HTTPサービスポート(デフォルト値: 25802) >>  [必須] 制御ポート(デフォルト値: 25805) >> |

InnoRulesサーバは、Apache TomcatサーバとInnoRules Webアプリケーションで構成される。サービスポートはApache TomcatのHTTPポートであり、制御ポートはApache Tomcatサーバの終了に使用されるポートである。サービス構成ウィザードの設定に応じてInnoRules Webアプリケーションはサービスを行うが、ルールビルダーサービスやルールWebサービスなどはHTTPを利用してサービスされる。

すべてのインストールノードでInnoRulesサーバを実行する必要はない。InnoRulesサーバを実行しない場合は、任意のポートを指定する。

InnoRulesサーバがインストールされた場合、各サーバには固有のUUIDが付与される。このUUIDは、設定クラスタ(「3.2 設定クラスタ」を参照のこと)内でサーバを識別するために使用される。設定クラスタについては「3.2 設定クラスタ」を参照のこと。

### ライセンスファイル

次の表は、ライセンスファイルを選択する段階である。ライセンスファイルは、製品を購入する際にインストールファイルと共に提供される。InnoRulesをインストールする時にライセンスファイルをインストールウィザードの実行パスに設定した場合、次の表のように一覧にライセンスが表示される。実行パスに2つ以上のライセンスファイルがある場合は、そのうちの1つが選択できる。

また、インストール時にライセンスファイルを指定せず、インストールが完了した後でインストールディレクトリのライセンスディレクトリに手作業でコピーすることもできる。

[表 49]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ライセンスファイル  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  現作業ディレクトリから次のライセンスファイルを発見しました。どのライセンスファイルをインストールしますか？  (\*) 1. innorules.lic  Test (Expires: 2016-01-31)  2. インストール後、手作業でコピー  選択してください。 >> |

### インストール概要

対の表は、入力した情報を確認してインストールを行う段階である。

[表 50]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストール概要  JDK Home : /opt/java/default-jdk  Install Directory : /home/user/innorules-home  Service Port : 25802  Admin Port : 25805  License File : /home/innorules/innorules.lic  この設定でInnoRulesをインストールします。よろしいですか？ (Yes/No) >> y  [情報] Installation succeeded.[エンターキーを押下してください。] |

入力した情報を確認後、YesまたはYを押すと、インストールが開始される。インストールが完了すると、「[情報]Installation succeeded」が表示される。

## 設定クラスタウィザード

インストールノードの構成設定は、インストールディレクトリのconfig/rule-systems.xmlに保存される。他ホストのインストールノード同士が設定を共有すると、ルールシステム設定に変更が発生した場合においても、各インストールノードに変更事項が反映される。これはシステム構成を容易にし、作業者の設定ミスを軽減する。設定クラスタの概念に関する詳細については、「3.2 設定クラスタ」を参照のこと。

設定クラスタリングは、複数のインストールノードを1つのグループとして管理し、それらの設定を同期させる機能である。設定クラスタに属するインストールノードの中の1つで設定を変更すると、残りのインストールノードでも同様に変更される。各インストールノードは、複数の設定クラスタに属することはできない。設定クラスタウィザードは、設定クラスタリングを構成するウィザードである。

設定クラスタは、必須構成ではない。ルールシステムが1つのインストールノードのみで構成されている場合、設定クラスタの構成は不要である。

ウィザードはscriptディレクトリで、次のコマンドにより実行する。

[表 51]

|  |
| --- |
| UNIX系列では :  ./cluster-wiz.sh  Windowsでは :  cluster-wiz.bat |

### 導入

次の表は、設定クラスタウィザードの導入の段階である。

[表 52]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRules設定クラスタウィザードへようこそ。このウィザードでは設定クラスタを管理できます。  [エンターキーを押下してください。] |

### インストールノード(初回)

次の表は、設定クラスタを構成するインストールノードの一覧を示している。ここでの設定クラスタは、ウィザードが実行されているインストールノードの属するクラスタを意味する。

[表 53]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストールノードリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  作業を選択してください。  [項目]  ---------------------------------------  | タグ | 使用可能 | ローカル | 同期化 |  ---------------------------------------  [可能な作業]  a. ノード追加 c. 同期化 e. 一覧情報更新  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

設定クラスタが構成されていない場合は、[a. ノード追加]、[c. 同期化]、[e. 一覧情報更新]メニューのみが表示される。

### リモートノード設定のためのプロトコル

[表 53]でノード追加を選択すると、リモートインストールノードのプロトコルが選択される([表 54])。設定クラスタリングはリモートインストールノード設定ファイルにリモートアクセス方法を記述することで設定する。リモートファイルアクセスのためのプロトコルとして、現在SFTPとFTPがサポートされている。

[表 54]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストールノードリスト -> リモートインストールノードのプロトコル  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  プロトコルを選択してください。  1. SFTP  SFTPを用いてリモート設定にアクセスします。ホストのアドレス、ポート、ID  、パスワードが設定されたフルパスが必要です。  2. FTP  FTPを用いてリモート設定にアクセスします。ホストのアドレス、ポート、ID  、パスワードが設定されたフルパスが必要です。  選択してください。 >> |

### プロトコル詳細

プロトコルの詳細は、[表 54]で選択したプロトコルにより追加で必要な情報を入力する画面である。

SFTPを選択した場合、次の表の画面が表示される。

[表 55]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストールノードリスト -> プロトコル詳細  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  SFTP プロトコルに関連する情報を入力してください。  id: User ID  \*password: Password  host: Remote Host Name or Address  port: SSH Port  remote-dir: Remote InnoRules Home Directory  [必須] id >> |

[表 55]の各項目の意味は次のとおりである。

* id : SFTP ID
* password : SFTPパスワード
* host : リモートホストのIP
* port : リモートホストのSFTPポート
* remote-dir : リモートホストのインストールパス

[表 54]でFTPを選択した場合、[表 56]の画面が表示される。

[表 56]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストールノードリスト -> プロトコル詳細  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  FTP プロトコルに関連する情報を入力してください。  id: User ID  \*password: Password  host: Remote Host Name or Address  port: FTP Port  remote-dir: Remote InnoRules Home Directory  [必須] id >> |

[表 56]の各項目の意味は、次のとおりである。

* id : FTP ID
* password : FTPパスワード
* host : リモートホストのIP
* port : リモートホストのFTPポート
* remote-dir : リモートホストのインストールパス

### ローカルインストールノードの登録

ウィザードを実行しているインストールノードを、ローカルインストールノード(Local Installation Node)という。[表 53]でインストールノードを追加する時には、ローカルインストールノードを最初に追加する。追加しない場合は、次の表のようなエラーが発生する。

[表 57]

|  |
| --- |
| [エラー] インストールノードの登録中にエラーが発生しました。: Cannot update the configuration.  Not all clusters are synchronized or there is no local config.…  [エンターキーを押下してください。] |

クラスタにローカルインストールノードを追加する際は、IPアドレスにはループバックアドレス(localhost、127.0.0.1)の使用を許可しない。IPアドレスは後に追加されるインストールノードからこのインストールノードの設定にリモートアクセスする際にも使用されるため、別のインストールノードが認識およびアクセス可能なIPアドレスを使用する必要がある。ループバックアドレスを入力した場合は、次の表のようなエラーが発生する。

[表 58]

|  |
| --- |
| [エラー] インストールノードの登録中にエラーが発生しました。: Loopback address is not allowed.  java.lang.Exception: Loopback address is not allowed.  [エンターキーを押下してください。] |

### リモートインストールノードの登録

設定クラスタにローカルインストールノードを登録すると、リモートのインストールノードをクラスタに追加できる。リモートインストールノードをクラスタに追加する方法は、ローカルインストールノードを追加する方法と同様である。

リモートのインストールノードが新規にインストールされてconfigディレクトリが空の場合、クラスタに追加される時にローカルインストールノードの設定がリモートにコピーされる。リモートインストールノードでインストール後に設定作業をした場合(rule-systems.xmlが存在する場合)、インストールノードを追加する際に[表 59]のような警告メッセージが表示される。

[表 59]

|  |
| --- |
| リモートインストールノードで既存の設定ファイルを発見しましたがローカルの設定と異なります。ローカルの設定を新しいリモートノードにコピーしますか？ (Yes/No) >> |

Yesを選択すると、リモート設定はローカルの設定で上書きされ、リモートインストールノードはクラスタに追加される。Noを選択すると、リモートの設定は変更されず、クラスタには追加されない。

### インストールノード

次の表は、ローカルインストールノードと別のインストールノードをクラスタに追加した際のインストールノードの様子を示している。

正常にローカル設定が追加されると、次の表のようなノードリストが表示される。

[表 60]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  インストールノードリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  作業を選択してください。  [項目]  -----------------------------------------------------------------------------------  | タグ | 使用可能 | ローカル | 同期化 |  -----------------------------------------------------------------------------------  1 | SFTP/user@192.168.0.100:22//home/user/innorules-home | V | V | V |  -----------------------------------------------------------------------------------  2 | SFTP/user@192.168.0.101:22//home/user/innorules-home-remote| V | | V |  -----------------------------------------------------------------------------------  [可能な作業]  a. ノード追加 b. ノード削除 c. 同期化 d. エラー詳細 e. 一覧情報更新  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

一覧の各フィールドの意味は、次のとおりである。

* タグ

リモートインストールノードにアクセスする方法を表す。文字列の最初にはプロトコルが位置し、その後にはプロトコルに依存する設定が続く。

* 使用可能

一覧のインストールノードへのアクセス可否を示す。インストールノードにリモートアクセスできない場合、設定クラスタウィザード以外のウィザードでの設定変更ができない。管理上の目的でリモートホストを停止させた状態でルールシステムの設定を変更する場合は、[b. ノード削除]メニューを利用して使用できないインストールノードをクラスタから削除した後、設定を変更する。その後、リモートホストが再び稼働した後に、クラスタにリモートインストールノードを追加する。

* ローカル

一覧のインストールノードがローカルインストールノードであるかを示す。ローカルインストールノードの使用可能フィールドと同期化フィールドは、常にチェックされている。

* 同期化

一覧のインストールノード設定とローカルインストールノードの設定が同一であるかを示す。同期化されていない場合、インストールノード追加を含む、すべての設定変更ができない。同期化されていない状態でインストールノードを追加する場合、[表 57]のエラーが発生する。

作業メニューは、次のとおりである。

* ノード追加

新しいインストールノードを設定クラスタに追加する。

* ノード削除

登録済みのインストールノードをクラスタから削除する。ローカルインストールノードは最後に削除する必要があり、他のインストールノードがクラスタに登録された状態でローカルインストールノードを削除した場合、エラーが発生する。

* 同期化

リモートインストールノードの設定がローカルインストールノードの設定と異なる場合、[c. 同期化]メニューを利用して同期化させることができる。同期化を実行した場合、ローカルインストールノードの設定をリモートインストールノードにコピーするため、[c. 同期化]メニューを実行するインストールノードにより結果が異なることがある。そのため、設定が最新のインストールノードで[c. 同期化]メニューを実行する必要がある。

* エラー詳細

リモートインストールノードが使用できない場合は、エラーの詳細を確認できる。

* 一覧情報更新

リモートインストールノードの使用可能状態および同期化状態を更新する。

### UUIDファイル

設定クラスタウィザードを含むすべての設定ウィザードは、設定クラスタノードのうち、ローカルインストールノードを確認するため、ウィザードが起動時にconfigディレクトリにUUIDファイルを作成する。UUIDファイルサイズは0であり、次のような形式のファイル名を持つ。

550e8400-e29b-41d4-a716-446655440000

ウィザード終了時に、UUIDファイルは削除されるが、正常に終了しない場合は削除されないこともある。ローカルホストでウィザードが未実行状態でconfigディレクトリにUUIDファイルが存在する場合は、削除しても問題はない。

### ロックファイル

リモート設定ファイルを占有してアクセスするために、設定クラスタウィザードを含むすべての設定ウィザードは起動時に、リモートのconfigディレクトリにロックファイルを作成してロックを獲得する。ロックファイルには、各インストールノードにおけるロックの獲得状況に関する情報が記録される。ウィザード起動時、対象ノードのうちロック獲得できないノードがある場合は、次の表の警告が発生する。

[表 61]

|  |
| --- |
| リモートインストールノードからロックファイルを発見しました。他ノードが設定作業中または以前の設定作業が異常終了した可能性があります。ロックファイルを削除して新規作成しますか？ (Yes/No) >> |

ロックファイルは、ウィザード終了時に削除される。ただし、ウィザードが異常終了した場合、ロックファイルが削除されない場合がある。[表 61]のエラーは他の設定ウィザードが実行中、または、以前実行されたウィザードが異常終了したことを表す。したがって、ウィザードを利用した設定作業中に[表 61]のエラーが発生した場合は、Yesを選択して既存のロックファイルを削除し、新しいロックファイルを作成してロックを獲得できる。ただし、他のウィザードの実行状態の確認が必要である。

## ルールシステム構成ウィザード

ルールシステム構成ウィザードはルールシステムを構成する単位ルールシステムを構成および管理し、ルールシステムがアクセスする業務データベースの接続情報を管理できるようにサポートするウィザードである。

ルールシステム構成ウィザードは、scriptディレクトリで次のコマンドにより実行する。

[表 62]

|  |
| --- |
| UNIX系列では :  ./system-wiz.sh  Windowsでは :  system-wiz.bat |

### 導入

次の表はルールシステム構成ウィザードの導入段階である。

[表 63]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム構成ウィザードへようこそ。このウィザードではルールシステムを構成できます。  [エンターキーを押下してください。] |

### 設定対象の選択

ルールシステム構成ウィザードを利用して、単位ルールシステムと業務データベースの管理ができる。単位ルールシステムは、単独でサービス可能なルールおよび保存するためのデータベース、実行するためのアプリケーション設定などで構成される。単位ルールシステムに関する詳細については、「3.3.1 単位ルールシステム」を参照のこと。

業務データベースは、ルールから参照できるルールシステム外部のデータベースである。業務データベース管理では、業務データベース一覧と各データベースの基本接続情報を管理する。同一目的の業務データベースであっても単位ルールシステムごと、また、アプリケーションに応じて接続情報を別々に管理できる。業務データベースに関する詳細については、「3.4 業務データベース」を参照のこと。

[表 64]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象を選択してください。  1. 単位ルールシステム管理  単位ルールシステム設定を管理します。  2. 業務データベース管理  業務データベース設定を管理します。  選択してください。 >> |

[表 64]で、単位ルールシステム管理を選択した場合、「2.3.3 単位ルールシステム管理」に移動し、業務データベース管理を選択した場合は、「2.3.4 業務データベース一覧」に移動する。

### 単位ルールシステム管理

次の表は、登録された単位ルールシステムを示す。各単位ルールシステムには、固有のIDと名称が付与されている。

[表 65]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  登録された単位ルールシステムのリストです。新規単位ルールシステムの追加および既存単位ルールシステムの修正ができます。  [項目]  -------------------------  | ID | 名称 |  -------------------------  1 | DEV | development |  -------------------------  2 | TEST | test |  -------------------------  [可能な作業]  a. 追加 b. 修正およびルールDB管理 c. 一番下の単位ルールシステム削除  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

次は、各作業メニューの機能である。

* 追加

単位ルールシステムを追加して基本ルールデータベースを登録する。選択すると、「2.3.3.1 単位ルールシステム基本情報」に移動する。

* 修正およびルールDB管理

単位ルールシステムの情報を修正し、単位ルールシステムに登録されたルールデータベースを管理する。選択すると、「2.3.3.2 単位ルールシステム基本情報の修正」に移動する。

* 一番下の単位ルールシステムの削除

一覧の最下段にある単位ルールシステムを削除する。詳細は、「2.3.3.3 一番下の単位ルールシステムの削除」を参照のこと。単位ルールシステムが1つも登録されていない場合は、このメニューは非表示となる。

登録された単位ルールシステムの順序は、ルールシステム全体において重要な意味を持つ。上記で登録した単位システムで作成されたルールの一部はすぐ下の単位システムに移管できる。詳細は、「3.3.2 移管」を参照のこと。

#### 単位ルールシステム基本情報

単位ルールシステムを追加するためには、その基本情報を入力する必要がある([表 66])。

[表 66]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステムの基本情報を入力してください。  [必須] ID >> PROD  [必須] 名称 >> Production |

単位ルールシステムの基本情報には、IDと名称がある。

* ID

1文字から10文字以内の、アルファベットの大小文字、数字、アンダースコア(‘\_’)のみで構成する。全ルールシステムの中でユニークな単位ルールシステムIDとする。ウィザードが単位ルールシステムIDの固有性はチェックするが、設定クラスタがルールシステムの一部のみを管理する場合はルールシステム内における固有性をチェックすることはできない。

* 名称

単位ルールシステムの名称は、単位ルールシステムの明確化が目的である。60文字以内で構成し、文字の制限はない。名称は全ルールシステム内でユニークになる必要がある。

入力した単位ルールシステムが、単位ルールシステムのリストの最終行に追加される。一度追加されると、その順序を変更することはできない。そのため、単位ルールシステムを追加する際は、全ルールシステムがどのような単位ルールシステムで構成するのか考慮する必要がある。

既に存在する単位ルールシステムより前に追加する場合は、追加位置の後続単位ルールシステムを削除した後、対象の単位ルールシステムを追加し、削除した単位ルールシステムを再度追加する。

IDと名称を入力してエンターキーを押すと、「2.3.3.5 ルールデータベース基本情報」の段階に移動する。

#### 単位ルールシステム基本情報の修正

次の表は、単位ルールシステムの基本情報を修正する段階を示す。

[表 67]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステムの基本情報を入力してください。  [必須] 名称(デフォルト値: development) >> |

単位ルールシステムは、名称のみ修正できる。IDは修正できない。既存の単位ルールシステム名称がデフォルト値として表示される。エンターキーを押すと、「2.3.3.4 ルールデータベース管理」に移動する。

#### 一番下の単位ルールシステムの削除

最後に登録された単位ルールシステムを次の表のように削除できる。

[表 68]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  登録された単位ルールシステムのリストです。新規単位ルールシステムの追加および既存単位ルールシステムの修正ができます。  [項目]  -------------------------  | ID | 名称 |  -------------------------  1 | DEV | Development |  -------------------------  2 | TEST | Test |  -------------------------  [可能な作業]  a. 追加 b. 修正およびルールDB管理 c. 一番下の単位ルールシステム削除  作業を選択してください。スキップするにはエンターキーを押下してください。 >> c  単位ルールシステム'TEST'を削除しますか？ 削除するとルールDBの内容は保存されますが関連する設定（ルールアプリケーション、TCPコネクタサーバなど）は復元できません。 (Yes/No) >> |

単位ルールシステム一覧のうち、最終行の単位ルールシステムのみ削除できる。一覧から単位ルールシステムが削除された場合でも、ルールデータベースのデータは削除されずに保持される。ただし、単位ルールシステムに関連する他の設定はすべて削除される。削除対象の設定は、次のとおりである。

* ルールアプリケーション設定
* TCPコネクタおよびWebサービスアダプタ設定
* 業務データベース基本接続設定

#### ルールデータベース管理

ルールデータベース管理は、単位ルールシステムに登録されたルールデータベースの一覧を次の表のように表示する段階である。

[表 69]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDB管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'DEV'に登録されたルールDBのリストです。新規ルールDBの登録および既存ルールDBの情報を修正できます。  [項目]  ---------------------------------------------------------------------------  | 名称 | JDBC-URL | ID | 説明 |  ---------------------------------------------------------------------------  1 | RULEDB1 | jdbc:oracle:thin:@onlinedb:1521:ORCL | innorules | |  ---------------------------------------------------------------------------  2 | RULEDB2 | jdbc:oracle:thin:@batchdb:1521:ORCL | innorules | |  ---------------------------------------------------------------------------  [可能な作業]  a. 追加 b. 情報更新 c. 削除 d. 上に移動 e. 下に移動  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

ルールデータベースは、ルールデータが保存されるストレージである。ルールデータベースに関する詳細については、「3.3.1.1 ルールリポジトリ」を参照のこと。

1つの単位システムには、複数のルールデータベースが登録できる。ルールデータベースの登録順序には意味がある。ルールビルダーサービス(「3.8.3 ルールビルダーサービス」)がルールデータを保存する際は、登録されたすべてのルールデータベースに同一のデータを保存する。ただし、ルールデータを照会する際は、一覧で最も上にある([表 69]ではRULEDB1)ルールデータベースからデータを読み込む。ルールデータベース一覧で最も上にあるルールデータベースを、マスタールールデータベースという。

ルールビルダーサーバはすべてのルールデータベースに同一のデータを保存するため、ルールデータの同期化を保証する。そのため、ユーザはルールデータベースの一部データを変更してはならない。変更した場合、想定しない問題が発生する可能性がある。

次は、各作業メニューの機能である。

* 追加

ルールデータベースを追加する。選択すると、「2.3.3.5 ルールデータベース基本情報」に移動する。

* 情報更新

ルールデータベースの接続情報を更新する。選択すると、「2.3.3.6 ルールデータベースのサーバタイプの選択」に移動する。

* 削除

ルールデータベースを削除する。単位システムには最低限1つのルールデータベースの登録が必要で、登録されたルールデータベースが1つの時はこのメニューが表示されない。ルールデータベースを削除した場合においても、データが削除されることはない。

* 上に移動

一覧ですぐ上のルールデータベースと優先順位を入れ換える。

* 下に移動

一覧ですぐ下のルールデータベースと優先順位を入れ換える。

#### ルールデータベース基本情報

追加するルールデータベースの基本情報を次の表のように入力する段階である。

[表 70]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDB管理 -> ルールDBの基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールDBの基本情報を入力してください。情報は一度設定すると変更できません。  [必須] 名称 >>  [必須] 説明(デフォルト値: ) >> |

ルールデータベースの名称は、1文字から20文字以内のアルファベットの大小文字、数字またはアンダースコア(‘\_’)で構成する。ルールデータベースの名称は、単位ルールシステム内でユニークとする。ルールデータベースの説明には、ルールデータベースに関する説明を入力する。この説明は、[表 69]の一覧に表示される。

ルールデータベースの追加後は、名称や説明の変更はできない。変更する場合は、ルールデータベースを削除後に再度追加する。

名称と説明を入力してエンターキーを押すと、「2.3.3.6 ルールデータベースのサーバタイプの選択」のサーバタイプの選択に移動する。

#### ルールデータベースのサーバタイプの選択

次の表は、ルールデータベースに使用したデータベースサーバを選択する段階である。

[表 71]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDB管理 -> ルールDBサーバタイプ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールDBとして使用するデータベースサーバのタイプを選択してください。  1. Oracle  Oracle database (with type 4 driver)  2. MSSQL  Microsoft SQL server  3. DB2  IBM DB2 with DB2 UDB JDBC universal driver  4. MySQL  MySQL database server  5. Postgre  Postgre database server  6. Hsqldb  Hsqldb database server  選択してください。 >> |

ルールデータベースに使用するデータベースサーバタイプを指定する。InnoRulesでは、様々なデータベースサーバへの接続方法とルールテーブルを作成するためのDDLを用意している。使用するデータベースサーバが一覧にない場合は、技術サポートが必要である。

InnoRulesは複数のデータベースへの接続方法とDDLを提供するが、JDBCドライバは提供しない。JDBCはデータベースサーバベンダーが提供する最新かつ稼動可能なJDBCドライバを必ず使用し、インストールウィザードを実行する前にこのドライバをlib/jdbc-driversに予めコピーする。各データベースを選択した時に必要なJDBCドライバクラスは、次のとおりである。

* Oracle

oracle.jdbc.driver.OracleDriver

* MSSQL

com.microsoft.sqlserver.jdbc.SQLServerDriver

* DB2

com.ibm.db2.jcc.DB2Driver

* MySQL

com.mysql.jdbc.Driver

* Postgre

org.postgresql.Driver

* Hsqldb

org.hsqldb.jdbc.JDBCDriver

ここに記述されたJDBCドライバ以外のドライバを使用する場合は、技術サポートが必要である。

データベースサーバを選択後に、データベースに接続情報を入力する「2.3.3.7 ルールデータベース接続情報」段階に移動する。

#### ルールデータベース接続情報

次の表は、追加するルールデータベースの接続情報を入力する段階である。

[表 72]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報 -> ルールDB管理 -> ルールDBの接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールDB'RULEDB3'に接続するための情報を入力してください。  [必須] address >>  [必須] port(デフォルト値: 1521) >>  [必須] SID or service name(デフォルト値: ORCL) >>  [必須] ID >>  [必須] パスワード >>  この情報を用いてルールDBを登録しますか？登録する場合、この接続情報を用いてDBにログイン後、テーブルの生成およびデータの修正ができます。 (Yes/No) >> |

データベースタイプごとにJDBC URLの形式が定義されており、ユーザは定義を確認する。データベースタイプによるJDBC URL形式は、lib/jdbc-drivers/database.xmlに定義されている。定義されているURL形式が使用するURLの形式と異なる場合は、技術サポートが必要である。

追加したルールデータベースは、登録されている単位ルールシステムおよびルールデータベースとの整合性を確保する必要がある。整合性を確保するために、ウィザードはルールデータベースにテーブルを作成する場合や、データを変更する場合もある。

* 現在の単位システムが一番目の単位システムであり、登録対象ルールデータベースが該当単位システムの最初に登録されたルールデータベースである場合、またはデータベースの中にテーブルが存在しない場合は、テーブルを新たに作成し初期データを入力する。テーブルが存在する場合は、テーブルを検証後、該当ルールデータベースにあるデータテーブルを使用する。
* 現在の単位システムが一番目の単位システムではなく、登録対象のルールデータベースが該当単位システムの最初に登録されたルールデータベースである場合、またはテーブルが存在しない場合は、テーブルを作成して初期データを入力する。テーブルが存在する場合は、テーブル検証および既存の単位ルールシステムのデータとの整合性をフラグ検査などでチェックする。異常がない場合は登録する。
* 登録対象のルールデータベースが該当単位システムの最初に登録されたルールデータベースではない場合またはテーブルが存在しない場合は、テーブルを新たに作成して最初のルールデータベースのデータをコピーする。テーブルが存在する場合は、テーブルを検証後に追加対象のデータベースと最初のルールデータベースのデータが同一の場合のみデータベースの追加を許可する。

追加完了後、「2.3.3.8 基本日付形式」の段階に移動する。

#### 基本日付形式

日付形式を使用するルール文法での基本日付形式を入力する段階である。この段階は開発系ルールシステムの一番目のルールデータベースを追加する場合のみ出力され、基本日付形式を指定すると以降追加するルールシステムは同一の日付形式を使用する。この日付形式は、JavaのSimpleDateFormat形式に従う。基本日付形式は設定後、変更不可である。

[表 73]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステムで使用する基本日付形式を入力してください。基本日付形式は、日付形式を使用するルール文法で使用されます。この形式は、JavaのSimpleDateFormat形式に従います。(デフォルト値: yyyy-MM-dd)  >> |

デフォルト値以外を入力する場合、次の表のように確認メッセージが出力される。

[表 74]

|  |
| --- |
| 日付形式が入力した後、変更が不可能であり、後に登録されているルールシステムに同じ日付形式で入力されます。この日付形式を登録しますか？: yyyyMMdd (Yes/No) >> |

#### 単位ルールシステム基本情報の処理

他のシステムで登録されたルールデータベースを使用して新規単位ルールシステムが追加された場合(「2.3.3.7 ルールデータベース接続情報」段階でテーブルを生成せずに、ルールデータをそのまま使用するように設定した場合)、ルールシステム構成ウィザードは該当ルールデータベースにある単位ルールシステムの基本情報をどのように処理するかを確認する。この段階は「2.3.3 単位ルールシステム管理」段階でユーザが入力した単位ルールシステムの基本情報と指定されたルールデータベース上のルールシステム基本情報(ID、名称、システムレベルなど)が異なる場合に実行される。つまり次のような場合である。

* ユーザが入力した単位ルールシステムのID、名称がルールデータベースに保存された値と異なる場合
* 選択した単位ルールシステムレベルとルールデータベース上の単位ルールシステムレベルが異なる場合

出力画面は次の表のとおりである。出力メッセージは指定中のルールデータベース上の単位ルールシステム情報を含む。

[表 75]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  登録するルールDBの中に既に他の単位ルールシステムで使用したシステム情報があります。システム情報をどのように処理しますか？  システムID: TEST, システム名称: test, システムレベル: 2, 基本日付形式: yyyy-MM-dd  1. ルールDBのシステム情報を使用  ルールDBにあるシステム情報を用いて単位ルールシステムを追加する。  2. 入力されたシステム情報を使用  入力されたシステム情報を用いてルールDBを更新し、単位ルールシステムを追加する。  選択してください。>> |

ユーザはメッセージに出力されたルールデータベースにあるシステム情報を使用するか、「2.3.3 単位ルールシステム管理」段階でユーザが入力したシステム情報にルールデータベースを更新して使用するかを選択して進行できる。単位システムが最初段階の単位システムではなく、ルールデータベース上のルールシステム基本情報が 単位システムより前段階の単位システムの場合、ルールシステム構成ウィザードは上記の1のオプションを表示せずに入力されたシステム情報を用いて単位ルールシステムを追加する。

ユーザが「入力されたシステム情報を使用」を選択した場合、次の表の確認メッセージを表示する。

[表 76]

|  |
| --- |
| システム情報を更新しますか？システム情報を除いた他のルールデータは変更されません。 (Yes/No) >> |

該当ルールデータベースを参照する他のインストールノードが存在し、ユーザが入力したシステム情報をルールデータベースに反映する場合、該当インストールノードとルールデータベースのシステム情報が異なり、該当インストールノードでシステム初期化エラーが発生する可能性がある。したがって、システム基本情報をどのように処理するかを選択する前に、該当ルールデータベースが他のインストールノードでの使用有無を確認する必要がある。

### 業務データベース一覧

ルールシステムからアクセスする業務データベースの一覧を次の表のように管理する段階である。

[表 77]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 業務データベースリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  業務データベースを管理します。作業を選択してください。  [項目]  ----------------------------------------------------------------------------------  | ID | 名称 | 説明 | 接続情報未登録 |  ----------------------------------------------------------------------------------  1 | 1 | APPDB | application database | V |  ----------------------------------------------------------------------------------  2 | 2 | | | |  ----------------------------------------------------------------------------------  [可能な作業]  a. 修正 b. 削除  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

一覧の各フィールドの意味は、次のとおりである。

* ID

業務データベースのIDである。業務データベースのIDはシリアル番号で付与される。任意で付与することはできない。

* 名称

業務データベースの名称である。名称が空になっている場合は、当該IDでは業務データベースが登録されていないことを意味する。名称が空になっているスロットは、常に1つ以上存在する。

* 説明

業務データベースの説明である。

* 接続情報未登録

業務データベースに接続する方法は、単位ルールシステムごとに異なる設定ができる。業務データベース接続方法が設定されていない単位ルールシステムが1つでもある場合、チェックマークが表示される。

次は、各作業メニューの機能である。

* 修正

業務データベースを追加および修正する。修正作業を選択して名称が空でないスロットのIDを選択するとその業務データベースが修正でき、名称が空のスロットのIDを選択すると業務データベースが追加できる。

* 削除

業務データベースについての接続設定を削除する。ただし、この業務データベースにアクセスするルールが存在する場合は削除されない。

#### 業務データベース基本情報

業務データベースの基本情報を設定する段階である。

[表 78]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 業務データベースリスト -> 業務データベースの基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  業務データベースの基本情報を入力してください。  [必須] 名称 >>  [必須] 説明 >> |

業務データベースの基本情報には名称と説明がある。名称は60文字以内とし、他の業務データベースの名称と重複してはならない。

基本情報を入力すると、「2.3.4.2 業務データベース基本接続情報」に移動する。

#### 業務データベース基本接続情報

この段階では、各単位システム別の業務データベースの接続情報を照会して管理できる([表 79])。

[表 79]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 業務データベースリスト -> 業務データベースの基本接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム別の業務データベースの基本接続情報です。修正するシステムを選択してください。  [項目]  --------------------------------------------------------------------------------------  | 単位ルールシステム | URL | ID | 使用ルール数 |  --------------------------------------------------------------------------------------  1 | DEV | jdbc:oracle:thin:@bizdev:1521:ORCL | APPDBU | 10 |  --------------------------------------------------------------------------------------  2 | TEST | jdbc:oracle:thin:@biztest:1521:ORCL | APPDBU | 8 |  --------------------------------------------------------------------------------------  [可能な作業]  a. 修正 b. 削除  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

この段階では、業務データベースについての各システムの基本接続情報を管理する。基本接続情報で多様なInnoRulesサービスを使用できる。(「2.5 ルールアプリケーション構成ウィザード」を参照のこと)

一覧の各フィールドの意味は、次のとおりである。

* 単位ルールシステム

接続情報を表示する単位ルールシステムの名称を表示する。

* URL

業務データベースに接続するためのJDBC URLである。

* ID

業務データベースに接続する際に使用されるデータベースユーザアカウントである。

* 使用ルール数

該当単位ルールシステム上で業務データベースを使用するルール数を表す。

次は、各作業メニューの機能である。

* 修正

単位ルールシステムについての基本接続情報を追加、修正する。

* 削除

単位ルールシステムについての基本接続情報を削除する。単位ルールシステムにこの接続情報を使用するルールが存在する場合においても、接続情報は削除できる。ただし、基本接続情報を使用するInnoRulesサービスなどではエラーが発生する。

#### 業務データベースサーバタイプの選択

業務データベースサーバのタイプを選択する段階である([表 80])。

[表 80]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 業務データベースリスト -> 業務データベースの基本接続情報 -> データベースサーバ選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'DEV'の業務データベース'1'のデータベースサーバタイプを選択してください。  1. Oracle  Oracle database (with type 4 driver)  2. MSSQL  Microsoft SQL server  3. DB2  IBM DB2 with DB2 UDB JDBC universal driver  4. MySQL  MySQL database server  5. Postgre  Postgre database server  6. Hsqldb  Hsqldb database server  選択してください。 >> |

使用するデータベースサーバのタイプが一覧にない場合は、技術サポートが必要である。データベースサーバを選択すると、「2.3.4.4 業務データベースの接続情報」段階に移動する。

#### 業務データベースの接続情報

業務データベースの接続情報を設定する段階である([表 81])。

[表 81]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 業務データベースリスト -> 業務データベースの基本接続情報 -> 業務データベースの接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'DEV'の業務データベース'1'に接続するための情報を入力してください。  [必須] address >>  [必須] port(デフォルト値: 1521) >>  [必須] SID or service name(デフォルト値: ORCL) >>  [必須] ユーザID >>  [必須] パスワード >> |

データベースごとにJDBC URL形式が定義されており、ユーザは定義を確認する。データベースサーバによるJDBC URL形式は、lib/jdbc-drivers/database.xmlに定義されている。使用するURLの形式が同一でない場合は、技術サポートが必要である。

入力完了後に接続テストが成功した場合に、設定変更されて「2.3.4.2 業務データベース基本接続情報」段階に戻る。

## サービス構成ウィザード

InnoRulesサーバは、ルールシステム管理に必要なサービスを提供する。サービス構成ウィザードは、ウィザードが実行されるローカルインストールノードにサービスを構成できるようにサポートする。サービス構成ウィザードはローカルインストールノードのサービスのみを構成するため、あるインストールノードにサービスを構成するにはそのインストールノードにあるサービス構成ウィザードを実行する必要がある。

サービス構成ウィザードを利用すると、次のInnoRulesサービスが構成できる。

* 管理サービス(「3.8.2 管理サービス」を参照のこと)
* ルールビルダーサービス(「3.8.3 ルールビルダーサービス」を参照のこと)
* TCPコネクタサービス(「3.6.4 ルールサービスの形態指定」を参照のこと)
* ルールWebサービス(「3.8.5 ルールWebサービス」を参のこと照)
* ルールRESTサービス(「3.8.6 ルールRESTサービス」を参照のこと)

サービス構成ウィザードはscriptディレクトリで次の表のコマンドで実行する。([表 82])

[表 82]

|  |
| --- |
| UNIX系列では :  ./service-wiz.sh  Windowsでは :  service-wiz.bat |

### 導入

次の表は、サービス構成ウィザードの導入段階である。

[表 83]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesサービス構成ウィザードへようこそ。このウィザードではホストに様々なサービスを設定するサポートを行います。  [エンターキーを押下してください。] |

### サービスタイプ選択

次の表は、管理するサービスのタイプを選択する段階である。1つのインストールノードに異なる2つ以上のサービスを有効にすることも可能である。

[表 84]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定したいサービスのタイプを選択してください。終了する場合、エンターキーを押下してください。  1. 管理サービス  登録されたルールランタイムシステムとルールシステムコンポーネントの管理  ができます。  2. ルールビルダーサービス  ルールの作成および管理ができます。  3. TCPコネクタサービス  リモートアプリケーションがルールサービスを呼び出すことができます。  4. ルールWebサービス  ルールWebサービスを設定しWeb環境でルールを呼び出せます。  5. ルールRESTサービス  ルールRESTサービスを設定してWeb環境にてルールを呼び出します。  選択してください。 >> |

選択したサービスによって「2.4.3 管理サービス」、「2.4.4 ルールビルダーサービス」、「2.4.5 TCPコネクタサービス」、「2.4.6 ルールWebサービス」、「2.4.7 ルールRESTサービス」に移動する。

### 管理サービス

管理サービスは、設定クラスタに属するルールアプリケーションの状態をモニタリングして管理するサービスである。

#### 管理サービス一覧

次の表は、設定クラスタに有効になっている管理サービスの一覧を表示する段階である。

[表 85]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> 管理サービス  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  管理サービスに関する作業を選択してください。  [項目]  ---------------------------------------------------------------------------------  | 名称 | バインドするアドレス | ポート | ローカル |  ---------------------------------------------------------------------------------  1 | d205593b-d43c-47c1-9304-39b8b6b649f0 | 10.10.10.1 | 25811 | |  ---------------------------------------------------------------------------------  [可能な作業]  a. 上に移動 b. 下に移動 c. ローカルインストールノードの管理サービスを有効化  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

1つの設定クラスタに複数の管理サービスを有効設定できる。有効化されたサービスは、管理サービスが有効化されたInnoRulesサーバが一覧表示される。管理サービスには優先順位が付与される。一覧上では上に位置するサービスの優先順位が高い。設定クラスタのInnoRulesサーバとルールアプリケーションは、まず優先順位の高い管理サービスに接続を試みる。管理サービスの優先順位に関する詳細については、「3.8.2 管理サービス」を参照のこと。

一覧の各フィールドの意味は、次のとおりである。

* 名称

管理サーバが有効化されたInnoRulesサーバの名称でインストールされる時に付与されたUUIDである。

* バインドするアドレスとポート

この管理サーバに接続するためのIPアドレスとポートである。

* ローカル

ローカルインストールノードのInnoRulesであるかを表す。チェックされている場合は、ローカルインストールノードで有効になっている管理サービスであることを意味する。

次は、各作業メニューの機能である。

* 上に移動

選択された管理サービスの優先順位を一段階高くする。

* 下に移動

選択された管理サービスの優先順位を一段階低くする。

* ローカルインストールノードの管理サービスを有効化

ローカルインストールノードで管理サービスを有効にする。このメニューは、ローカルインストールノードで管理サービスが無効時にのみ表示される。このメニューを実行すると、「2.4.3.2 管理サービス接続情報」に移動する。

* ローカルインストールノードの管理サービスを無効化

ローカルインストールノードで管理サービスを無効にする。このメニューは、ローカルインストールノードで管理サービスが有効時にのみ表示される。

#### 管理サービス接続情報

次の表は、管理サービスの接続情報を設定する段階である。

[表 86]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> 管理サービス -> 管理サービスの接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  管理サービスの接続情報を入力してください。  [必須] 接続するアドレス  1. 10.10.10.2  2. 10.10.10.102  3. 192.168.0.1  >> 3  [必須] ポート(デフォルト値: 25811) >> |

管理サービスを有効化するためには、クライアントの接続を待機するIPアドレスとポートを指定する必要がある。ウィザードは、ループバックアドレス以外のこのホストのIPアドレスをリストに表示する。管理サービスが使用するアドレスを選択するが、クライアントが識別・接続できるIPアドレスを指定する必要がある。

ポートのデフォルト値は25811であるが、システム管理者と協議して割り当ててもらうことを推奨する。

情報をすべて正常に入力すると管理サービスが有効になり、再び管理サービス一覧に戻る([表 87])。ローカル管理サービスが一番下に追加されたことを確認すること。

[表 87]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> 管理サービス  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  管理サービスに対する作業を選択してください。  [項目]  ----------------------------------------------------------------------------------  | 名称 | バインドするアドレス | ポート | ローカル |  ----------------------------------------------------------------------------------  1 | d205593b-d43c-47c1-9304-39b8b6b649f0 | 10.10.10.1 | 25811 | |  ----------------------------------------------------------------------------------  2 | 911e97bf-3a84-4a76-bf80-4ecd221cdc68 | 10.10.10.2 | 25811 | V |  ----------------------------------------------------------------------------------  [可能な作業]  a. 上に移動 b. 下に移動 c. ローカルインストールノードの管理サービスを無効化  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

### ルールビルダーサービス

ルールビルダーサービスは、ルール要素を作成および管理するサービスである。

#### ルールビルダーサービス一覧

次の表は、ルールビルダーサービス一覧を表示する段階である。

[表 88]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ビルダーサービス  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ビルダーサービスに対する作業を選択してください。  [グローバル設定]  移管専用 : 未設定  平均同時利用ユーザ数: 2  最大同時利用ユーザ数: 10  暗号化方式 : DES  テーブルルール条件式チェック : 未設定  クラスタ接続タイムアウト(ms) : 5,000  クラスタレスポンスタイムアウト(ms) : 60,000  クラスタ文字セット : UTF-8  [項目]  -----------------------------------------------------------------------------------  | 名称 | アドレス | サービスポート | モニターポート | ローカル |  ----------------------------------------------------------------------------------  1 | d205593b-d43c-47c1-9304-39b8b6b649f0 | 10.10.10.1 | 25802 | 25810 |  ----------------------------------------------------------------------------------  [可能な作業]  a. 上に移動 b. 下に移動 c. グローバル設定変更 d. ローカルインストールノードのビルダーサービスを有効化  e. ビルダーサービス接続URLを確認  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

グローバル設定は、この設定クラスタで有効化されるすべてのビルダーサービスに適用される設定である。グローバル設定パラメータに関する説明は、「2.4.4.2 ビルダーサービスのグローバル設定」を参照のこと。

1つの設定クラスタにルールビルダーサービスを複数有効化することもできる。有効化されたサービスは、[表 88]の一覧に表示される。このルールビルダーサーバには優先順位が付与され、動作中のルールビルダーサーバのうち最も 優先順位の高いものがサービスを提供する。これを、ルールビルダーサーバクラスタリングという。ルールビルダーサーバクラスタリングについては、「3.8.3.2 ビルダーサーバクラスタリング」を参照のこと。

一覧の各フィールドの意味は、次のとおりである。

* 名称

ルールビルダーサービスが有効化されたInnoRulesサーバの名称でインストールされる際に付与されたUUIDである。

* アドレス

このルールビルダーサービスに接続するためのIPアドレスである。

* サービスポート

ルールビルダーサービスを提供するポートである。ルールビルダーサービスはHTTPを利用して提供される。

* モニターポート

ルール変更に関する同期化をするためのモニターポートである。

* ローカル

ローカルインストールノードのInnoRulesであるかを表す。チェックされている場合は、ローカルインストールノードで有効になっている管理サービスであることを意味する。

次は、各作業メニューの機能である。

* 上に移動

選択されたルールビルダーサービスの優先順位を一段階高くする。

* 下に移動

選択されたルールビルダーサービスの優先順位を一段階低くする。

* グローバル設定変更

ルールビルダーサービスのグローバル設定を変更する。

* ローカルインストールノードのビルダーサービスを有効化

ローカルインストールノードでビルダーサービスを有効にする。このメニューは、ローカルインストールノードでルールビルダーサービスが無効化されている時にのみ表示される。

* ローカルインストールノードのビルダーサービスを無効化

ローカルインストールノードでビルダーサービスを無効にする。このメニューは、ローカルインストールノードでビルダーサービスが有効化されている時にのみ表示される。

* ビルダーサービス接続URLを確認

選択されたルールビルダーサービスの接続URLを確認する。このメニューは、登録したルールビルダーサービスが1つ以上ある時にのみ表示される。

#### ビルダーサービスのグローバル設定

次の表は、ビルダーサービスのグローバル設定をする段階を示している。

[表 89]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ビルダーサービス -> ビルダーサービスのグローバル設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ビルダーサービスのグローバル設定値を入力してください。  [必須] このクラスタのビルダーサーバを移管専用で設定しますか？  (\*) 1. 移管専用に設定しない  2. 移管専用に設定  >>  [必須] 平均同時利用ユーザ(デフォルト値: 2) >>  [必須] 最大同時利用ユーザ(デフォルト値: 10) >>  [必須] パスワード暗号化方式(デフォルト値: DES) >>  [必須] テーブルルール条件式チェック  (\*) 1.テーブルルールの条件式をチェックしない  2. テーブルルールの条件式をチェック  >>  [必須] クラスタ接続タイムアウト(ms)(デフォルト値: 5000) >>  [必須] クラスタレスポンスタイムアウト(ms)(デフォルト値: 60000) >>  [必須] クラスタ文字セット(デフォルト値: UTF-8) >> |

ビルダーサービスのグローバル設定パラメータは、次のとおりである。

* 移管専用

移管専用に設定された場合、この設定クラスタのビルダーサービスはルール保存機能が無効になって移管のみ可能である。移管専用については、「3.8.3.3 多段階ルールシステムと移管専用モード」を参照のこと。

* 平均同時利用ユーザ数と最大同時利用ユーザ数

同時にビルダーサービスを利用するユーザの数である。これは同時に実行されるルールビルダーの数ではなく、同時に呼び出されるビルダーサービスの数である。InnoRulesサーバは、これに基づいてローカルルールサーバの数など性能・リソースパラメータを調整する。

* パスワード暗号化方式

ユーザのパスワードをルールデータベースに保存する際に使用される暗号化方式である。サポートされる暗号化アルゴリズムにはDESとSHA256があり、デフォルト値はDESである。パスワード暗号化方式の変更はDESからSHA256への変更のみ可能であり、逆方向は不可能なため注意する。

* テーブルルール条件式チェック

テーブルールで使用する条件式の整合性をチェックする。整合性チェックの仕様は別添した「テーブルルール条件式チェック」文書を参照のこと。

* クラスタ接続タイムアウト

ビルダークラスタ内のビルダーサーバ間の接続タイムアウト時間である。単位はミリ秒(millisecond)でデフォルト値は5,000である。ビルダーサーバクラスタリングについては「3.8.3.2 ビルダーサーバクラスタリング」を参照のこと。

* クラスタレスポンスタイムアウト

ビルダークラスタ内のビルダーサーバ間の応答タイムアウト時間である。単位はミリ秒(millisecond)でデフォルト値は60,000である。 応答タイムアウトの値が小さすぎるとビルダーサーバのプロセスに問題が発生する場合があるため注意する必要がある。ビルダーサーバクラスタリングについては「3.8.3.2 ビルダーサーバクラスタリング」を参照のこと。

* クラスタ文字セット

ビルダークラスタ内のビルダーサーバ間の文字セットである。デフォルト値はUTF-8である。ビルダーサーバクラスタリングについては「3.8.3.2 ビルダーサーバクラスタリング」を参照のこと。

#### モニターサーバ接続情報

モニターサーバの接続情報を入力する段階である。

[表 90]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ビルダーサービス -> モニターサーバの接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  モニターサーバの接続情報を入力してください。  [必須] アドレス  1. 10.10.10.2  2. 10.10.10.102  3. 192.168.0.2  4. 127.0.0.1  >> 1  [必須] ポート(デフォルト値: 25810) >> |

モニターサーバは、ルールビルダーサーバの一部の機能でルールが変更される際にルールサーバが変更内容を同期化するサーバである。モニターサーバについては、「3.8.3.1 モニターサーバとモニターセッション」を参照のこと。

同一設定クラスタのルールサーバはこのモニターサーバに接続する必要があるため、ルールサーバが接続できるアドレスを選択する。

モニターサーバポートのデフォルト値は25810であるが、システム管理者と協議して設定するようにする。同一設定クラスタのルールサーバをこのモニターサーバに接続できるようにする場合は、ファイアウォール設定を行う。

ルールビルダーサービスはInnoRulesサーバのHTTP通信機能を利用するため、別途アドレスやポートを設定する必要はない。

モニターサーバの接続情報を設定すると、ルールビルダーサービスが追加されて「2.4.4.1 ルールビルダーサービス一覧」に移動する。

#### ルールビルダーサービス接続URL確認

ルールビルダーサービスを使用するための接続情報を確認する段階である。

[表 91]

|  |
| --- |
| ビルダサービス '619a14c0-56b3-4773-ac37-df7f45b939aa'を使用するためには次の情報をビルダクライアントに入力してください。  ただし、次の情報の変更（カスタマイズ等）がある場合には接続情報を管理者に確認してください。  ホスト : 192.168.0.147  URL : /innorules/services/builder  Port : 14701  [エンターを押してください。] |

有効化されたルールビルダーサービスの場合、ユーザは[表 91]に表示された接続情報をビルダークライアントに入力してルールビルダーサービスを使用できる。

ただし、接続情報に影響を与える内容(カスタマイズ、WebServerを利用した接続など)がある場合、実際の接続情報は管理者から確認する必要がある。

エンターキーを押すと「2.4.4.1 ルールビルダーサービス一覧」に移動する。

### TCPコネクタサービス

TCPコネクタサービスは、TCP/IPとInnoRules自身のプロトコルを利用してリモートでルールサービスを呼び出せるようにするサービスである。TCPコネクタサービスに関する詳細については、「3.8.4 TCPコネクタサービス」を参照のこと。

#### 単位ルールシステムの選択

TCPコネクタサービスを有効化するためには、先にルールのサービスを行う単位ルールシステムを選択する必要がある。次の表は、単位ルールシステムを選択する段階を示している。

[表 92]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> 単位ルールシステム選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  TCPコネクタサービスを使用して、ルールサービスしたい単位ルールシステムを選択してください。  1.DEV  Development  2.PROD  Production  選択してください。>> 2 |

ルールビルダーサービスが移管専用に設定されている場合、最初段階の単位ルールシステムは[表 92]の一覧に表示されず、選択することもできない。選択できる単位ルールシステムがない場合は、次の警告メッセージが表示される。

[表 93]

|  |
| --- |
| [警告] 使用可能な単位ルールシステムがありません。単位ルールシステムを先に登録してください。  [エンターキーを押下してください。] |

TCPコネクタサービスを有効にする単位ルールシステムを選択すると、「2.4.5.2 TCPコネクタサーバグループ一覧」に移動する。

#### TCPコネクタサーバグループ一覧

次の表は、設定クラスタに有効化されたPROD単位ルールシステムのTCPコネクタサーバグループの一覧を示している。

[表 94]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> TCPコネクタサーバグループリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'PROD'に登録されたTCPコネクタサーバグループのリストです。  [項目]  ------------------------------------------------------------------------------------  | 名称 | モード | サーバ数 | ローカル優先 | 平均同時処理量 | 最大同時処理量 | ローカル |  ------------------------------------------------------------------------------------  1 | online | load-balance | 1 | N | 100 | 100 | |  ------------------------------------------------------------------------------------  [可能な作業]  a. グループ管理 b. ローカルインストールノードのTCPコネクタサービスを有効化  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

TCPコネクタサーバは、負荷分散(Load Balancing)またはフェールオーバー(Fail Over)目的でグループ化される。このグループを、TCPコネクタサーバグループという。これに関する詳細については、「3.8.4.2 TCPコネクタサーバグループ」を参照のこと。

項目一覧の各フィールドの意味は、次のとおりである。

* 名称

TCPコネクタサーバグループの名称である。

* モード

TCPコネクタサーバグループの高可用性ポリシーである。load-balanceはこのグループが負荷分散を目的とし、fail-overはフェールオーバーを目的とすることを意味する。

* サーバ数

このグループに含まれるサーバ数を示す。

* ローカル優先

ローカル優先オプションが有効であるかを表す。

* 平均同時処理量および最大同時処理量

TCPサーバグループが処理する同時ルールサービスの想定処理量である。

* ローカル

ローカルインストールノードのTCPコネクタサーバがこのグループに属しているかを表す。

作業メニューは、次のとおりである。

* グループ管理(「2.4.5.6 TCPコネクタサーバ一覧」)

選択したTCPサーバグループの設定を変更する。

* ローカルインストールノードのTCPコネクタサービスを有効化(「2.4.5.3 TCPコネクタサービス接続情報」)

ローカルインストールノードでTCPコネクタサービスを有効にする。ローカルインストールノードでTCPコネクタサービスが無効時のみ表示される。

* ローカルインストールノードのTCPコネクタサービスを無効化

ローカルインストールノードでTCPコネクタサービスを無効にする。ローカルインストールノードでTCPコネクタサービスが有効時のみ表示される。

#### TCPコネクタサービス接続情報

TCPコネクタサービスを有効にするメニューを選択すると、次の表のようなTCPコネクタサービス接続情報の設定段階が表示される。

[表 95]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> TCPコネクタサーバグループリスト -> TCPコネクタサービスの接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  TCPコネクタサービスの接続情報を入力してください。  [必須] 接続するアドレス  1. 10.10.10.1  2. 10.10.10.2  3. 192.168.0.1  4. 127.0.0.1  >>  [必須] ポート >> 25800  [必須] 使用するルールDBを選択してください。  1. RULEDB1  2. RULEDB2  >> 1 |

接続するアドレスとポートは、リモートのルールアプリケーションがこのTCPコネクタサーバに接続時に使用されるアドレスとポートである。アドレスの一覧には、このホストのネットワークアドレスが表示される。このうち、リモートアプリケーションが接続可能なアドレスの中から1つを選択する。

ポートはシステム管理者と協議して付与し、この時ファイアウォール設定も同時に確認することを推奨する。

使用するルールデータベースにはこの単位システム(この例ではPROD)に登録されたすべてのルールデータベースが並び、そのうちの1つを選択する。TCPコネクタサーバは、ここで指定されたルールデータベースからルールデータを照会する。データベースの負荷の程度やシステムの条件などを考慮してルールデータベースを選択する。

接続情報を入力すると、「2.4.5.4 TCPサーバグループの選択」に移動する。

#### TCPサーバグループの選択

次の表は、有効TCPコネクタサーバの所属グループを指定する段階である。

[表 96]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> TCPコネクタサーバグループリスト -> TCPコネクタサーバグループの選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  TCPコネクタサーバに含むグループを選択してください。  1. online  2. (新規グループ生成)  選択してください。>> |

一覧にはこの単位システムで既に作成されているグループが表示され、これらのグループの中から1つを選択できる。なお、新しいグループを作成することもできる。既存のグループを選択するとTCPコネクタサーバが有効になり「2.4.5.2 TCPコネクタサーバグループ一覧」に戻り、新しいグループの作成を選択すると「2.4.5.5 TCPコネクタサーバグループの設定」に移動する。

#### TCPコネクタサーバグループの設定

次の表は、TCPコネクタサーバグループを設定する段階を示す。

[表 97]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> TCPコネクタサーバグループリスト -> TCPコネクタサーバグループの選択 -> TCPコネクタサーバグループの設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  システム'PROD'に追加するTCPコネクタサーバグループを設定してください。  [必須] グループ名称 >> batch  [必須] HA(High-Availability)モードを選択してください。  1. FAILOVER  2. LOADBALANCE  >> 2  [必須] ローカル優先を適用しますか？  1. はい  2. いいえ  >> 1  [必須] 平均同時処理量 >> 5  [必須] 最大同時処理量 >> 5 |

グループ名称は、この単位ルールシステム内のグループ識別に使用される。この名称は、単位ルールシステム内で ユニークな名称とする。グループ名称は、「2.4.5.4 TCPサーバグループの選択」でグループ追加を選択時のみ入力ができる。

HA(高可用性)モードは、フェールオーバーまたは負荷分散のどちらか1つを選択できる。高可用性モードについては、「3.8.4.2 TCPコネクタサーバグループ」を参照のこと。

ローカル優先モードは、このグループを使用するルールアプリケーションがサーバの優先順位に関係なく、アプリケーションと同一ホストで動作するサーバを優先して使用するようにするオプションである。このオプションに関する詳細については、「3.8.4.2 TCPコネクタサーバグループ」を参照のこと。

平均同時処理量と最大同時処理量は、このグループが処理する想定処理量である。想定処理量と高可用性モードによって、TCPコネクタサーバが占有するリソースの量が自動的に決定される。

すべての情報が入力されると、「2.4.5.6 TCPコネクタサーバ一覧」に移動する。

#### TCPコネクタサーバ一覧

次の表は、TCPコネクタサーバグループの設定とグループに属するサーバの一覧を表示する段階である。この段階は、「2.4.5.2TCPコネクタサーバグループ一覧」でグループ管理を選択する場合や「2.4.5.5 TCPコネクタサーバグループの設定」で新しいグループを追加した時に入ることができる。

[表 98]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> TCPコネクタサーバグループリスト -> TCPコネクタサーバグループの選択 -> TCPコネクタサーバグループの設定 -> TCPコネクタサーバリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'PROD'のTCPコネクタサーバグループ'batch'の設定とサーバのリストは次のとおりです。  モード: load-balance  ローカル優先: Y  平均同時処理量: 5  最大同時処理量: 5  [項目]  -----------------------------------------------------------------------------  | UUID | 接続するアドレス | ポート | ローカル |  -----------------------------------------------------------------------------  1 | 72631759-9cb6-4c33-a2ad-ba9f4d06a2cc | 10.10.10.2 | 25800 | |  -----------------------------------------------------------------------------  2 | 4451bb87-facf-4768-87ee-fa075333311e | 10.10.10.1 | 25811 | V |  -----------------------------------------------------------------------------  [可能な作業]  a. グループ設定の変更 b. 上に移動 c. 下に移動  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

[表 98]は、単位ルールシステムPRODに登録されたbatchというサーバグループを示している。[TCPコネクタサーバリスト]画面の上部には、グループの高可用性設定および同時処理量を表示する。

一覧には、このグループに属するTCPコネクタサーバを表示する。一覧の各フィールドの意味は、次のとおりである。

* UUID

TCPコネクタサーバが有効化されたInnoRulesサーバの名前でインストールされる時に付与されたUUIDである。

* 接続するアドレスとポート

このTCPコネクタサーバに接続するためのIPアドレスとポートである。

* ローカル

ローカルインストールノードのInnoRulesであるかを表す。チェックされている場合は、ローカルインストールノードで有効化された管理サービスであることを意味する。

作業メニューは、次のとおりである。

* グループ設定変更

TCPコネクタサーバグループの設定を変更する。このメニューを選択すると、「2.4.5.5 TCPコネクタサーバグループの設定」に移動する。

* 上に移動・下に移動

選択されたTCPコネクタサーバの順序を上または下へ移動する。順序は、高可用性モードでは重視して使用される。TCPコネクタの順序が高可用性モードでどのような意味を持つのかについては、「3.8.4.2 TCPコネクタサーバグループ」を参照のこと。

### ルールWebサービス

ルールWebサービスはSOAP/HTTPを利用してルールサービスを呼び出せるようにするサービスである。ルールWebサービスに関する詳細については、「3.8.5 ルールWebサービス」を参照のこと。

#### ルールWebサービス

次の表は、登録された単位ルールシステムのリストと各単位システムでルールWebサービスが有効になっているかを確認する段階である。

[表 99]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールWebサービス  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールWebサービスに対する作業を選択してください。  [項目]  ----------------------------------------------------------------------  | システムID | システム名称 | エイリアス | 有効有無 | ローカル |  ----------------------------------------------------------------------  1 | DEV | 開発系 | N/A | | |  ----------------------------------------------------------------------  2 | TEST | テスト系 | N/A | | |  ----------------------------------------------------------------------  [可能な作業]  a. ルールWebサービスURLの照会 b. エイリアス変更 c. ローカルインストールノードのルールWebサービスを有効化  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

リストの各フィールドの意味は次のとおりである。

* システムIDとシステム名称

登録された単位ルールシステムのIDとシステム名称である。各単位ルールシステム別にルールWebサービスの有効・無効の指定ができる。

* エイリアス

Webサービスを利用してルールを呼び出す時、どの単位システムのルールを呼び出すかを示すエイリアスである。エイリアスを指定すると、システムIDの代わりにエイリアスがURLに挿入される。

* 有効有無

ローカルインストールノードを含めて設定クラスタ内でこの単位ルールシステムのためのルールWebサービスの有効有無を表示する。

* ローカル

ローカルインストールノードにこの単位ルールシステムのためのルールWebサービスが有効かどうかを表示する。

作業メニューは次のとおりである。

* ルールWebサービスURLの照会

単位ルールシステムのルールWebサービスを呼び出せるURLのリストを表示する。複数のInnoRulesサーバが、ある単位ルールシステムのルールをWebサービス化できるため、URLは複数になる。ルールWebサービスを呼び出すアプリケーションにこのURLを提供できる。

* エイリアス変更

エイリアスを変更する。このメニューを選択すると、「2.4.6.2 ルールWebサービスシステムのエイリアスを変更」に移動する。

* ローカルインストールノードのルールWebサービスを有効化

ローカルインストールノードのInnoRulesサーバでルールWebサービスを有効・無効にする。このメニューを選択すると、「2.4.6.3 ルールWebサービスを有効にする」に移動する。

#### ルールWebサービスシステムのエイリアスを変更

1つのInnoRulesサーバが複数の単位ルールシステムのルールをWebサービス化できるため、URLに単位ルールシステムを指定できなければならない。単位ルールシステムのIDを使用する場合、URLにシステムIDが表示されるため、これを避けるために単位システム別に名前を付与できる。この名前をルールWebサービスシステムエイリアスという。ルールWebサービスURLは次の形式を従う。

http://[IP]:[PORT]/innorules/services/ws/[ALIAS]/[SERVICENAME]

ここで、IPとPORTはInnoRulesサーバのアドレスとサービスポートであり、ALIASはルールWebサービスシステムエイリアスである。SERVICENAMEはWebサービスするルールに与えられるサービス名である。

InnoRulesサーバが1つの単位ルールシステムのみにWebサービスの提供や、主にサービスする単位システムがある場合、この単位システムにエイリアスを付与しない場合もある。エイリアスが付与されていないルールWebサービスを基本ルールWebサービスという。ただし、設定クラスタには1つの単位システムのみを基本ルールWebサービスで指定できる。

エイリアスは10文字以内のアルファベットの大小文字、数字またはアンダースコア(‘\_’)のみで構成される。

次の表は、エイリアスを指定する段階である。

[表 100]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールWebサービス -> ルールWebサービスシステムのエイリアス変更  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'DEV'のエイリアスを変更します。エイリアスを指定しない場合、  この単位システムが基本ルールWebサービスとしてサービスされます。  基本ルールWebサービスは単位システム1つのみ指定できます。(デフォルト値: DEV)  >> |

デフォルト値としては既に指定されたエイリアスが設定される。エイリアスは他の単位システムのエイリアスと重複できない。この単位ルールシステムを基本ルールWebサービスでサービスする場合、^eを入力して設定する。設定終了後「2.4.6.1 ルールWebサービス」に移動する。

#### ルールWebサービスを有効にする

1つのInnoRulesサーバは複数の単位ルールシステムに対してルールWebサービスができる。次の表はWebサービスを有効にする単位ルールシステムを選択する段階である。

[表 101]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールWebサービス -> ルールWebサービス有効化  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールWebサービスを有効化するシステムを選択してください。未選択のシステムはルールWebサービスが無効化されます。  何も選択しない場合、ローカルインストールノードのすべてのルールWebサービスが無効化されます。  (\*) 1. DEV  開発系  2. TEST  テスト系  複数選択してください。選択を終了するには'0'を入力してください。 >> |

サービスを有効にする単位システムの番号を一度に1件ずつ入力する。選択完了後に0を入力する。単位システムの番号を入力せずに0を入力すると、すべてのルールWebサービスが無効化する。有効化された単位システムを変更しない場合は何も入力せずにエンターキーを押す。現在、有効になった単位ルールシステムは番号前に(\*)表示が付く。

有効にする単位システムがある場合は「2.4.6.4 ルールWebサービス設定」に移動、そうではない場合は「2.4.6.1 ルールWebサービス」に移動する。

#### ルールWebサービス設定

次の表はルールWebサービスURLで表示できるIPアドレスを選択する段階である。このIPは「2.4.6.1 ルールWebサービス」を生成することに使用される。

[表 102]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールWebサービス -> ルールWebサービス有効化 -> ルールWebサービスの設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールWebサービスの情報を入力してください。  [必須] ルールWebサービスに接続するアドレスを選択してください。  1. 172.16.25.1  2. 172.16.15.1  3. 192.168.0.72  4. 127.0.0.1  >> |

ルールWebサービスはInnoRulesのHTTPサービスを利用するため、InnoRulesサーバがバインドしたすべてのIPアドレスを利用してアクセスができる。InnoRulesサーバは[表 102]のリストの全IPアドレスをバインドして実行されるため、IPアドレス選択がWebサービスの動作に影響を及ぼすことはない。ただし、選択されたアドレスがURLに反映され、当該URLがアプリケーション開発者に伝達されるため、Webサービスを呼び出すアプリケーションからアクセスできるIPアドレスを選択する。

IPアドレスを選択すると「2.4.6.5 システムエイリアス設定」に移動する。

#### システムエイリアス設定

次の表は、「2.4.6.3 ルールWebサービスを有効にする」で選択した単位システムに対するルールWebサービスのシステムエイリアスを設定する段階である。

[表 103]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールWebサービス -> ルールWebサービス有効化 -> ルールWebサービスシステムのエイリアス設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  選択されたルールWebサービスシステムのエイリアスを設定してください。  [必須] DEV(デフォルト値: DEV) >> ^e  [必須] TEST(デフォルト値: TEST) >> |

エイリアスは単位ルールシステム別に設定する。エイリアスを設定する方法は「2.4.6.2 ルールWebサービスシステムのエイリアスを変更」と同様である。[表 103]の例のDEVシステムは^eを入力して基本ルールWebサービスとして登録した。エイリアス設定が完了すると「2.4.6.1 ルールWebサービス」に移動する。

### ルールRESTサービス

ルールRESTサービスはRESTを利用してルールサービスを呼び出すサービスである。ルールRESTサービスについては「3.8.6 ルールRESTサービス」を参照のこと。

#### ルールRESTサービス

次の表は登録した単位ルールシステムのリストと各単位システムのルールRESTサービス活性化有無を表示する。

[表 104]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールRESTサービス  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールRESTサービスに対する作業を選択しください。  [項目]  --------------------------------------------------------------------  | システムID | システム名 | エイリアス | 活性化有効 | ローカル |  --------------------------------------------------------------------  1 | DEV | 開発用 | N/A | | |  --------------------------------------------------------------------  [可能な作業]  a. ルールRESTサービスのURL参照 b. エイリアス変更 c. ローカルインストールノードにルールRESTサービスを活性化  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

リストの各フィールドの意味は次のとおりである。

* システムIDとシステム名称

登録された単位ルールシステムのIDとシステム名称である。各単位ルールシステム別にルールRESTサービスの有効・無効の指定ができる。

* エイリアス

ルールRESTサービスを利用してルールを呼び出す時、どの単位システムのルールを呼び出すかを指定するためにエイリアスがURLに挿入される。

* 活性化有効

ローカルインストールノードを含めて設定クラスタ内でこの単位ルールシステムのためのルールRESTサービスが有効かどうかを表示する。

* ローカル

ローカルインストールノードにこの単位ルールシステムのためのルールRESTサービスが有効かどうかを表示する。

作業メニューは次のとおりである。

* ルールRESTサービスURLの照会

単位ルールシステムのルールRESTサービスを呼び出せるURLのリストを表示する。ルールRESTサービスを呼び出すアプリケーションにこのURLを提供できる。

* エイリアス変更

エイリアスを変更する。このメニューを選択すると、「2.4.7.2 ルールRESTサービスシステムのエイリアス変更」に移動する。

* ローカルインストールノードのルールWebサービスを有効化

ローカルインストールノードのInnoRulesサーバでルールRESTサービスを有効・無効にする。このメニューを選択すると、「2.4.7.3 ルールRESTサービス活性化」に移動する。

#### ルールRESTサービスシステムのエイリアス変更

1つのInnoRulesサーバが複数の単位ルールシステムのルールをRESTサービス化できるため、URLに単位ルールシステムを指定できなければならない。単位ルールシステムのIDを使用する場合、URLにシステムIDが表示されるため、これを避けるために単位システム別に名前を付与できる。この名前をルールRESTサービスシステムエイリアスという。ルールRESTサービスURLは次の形式を従う。

http://[IP]:[PORT]/innorules/services/rest/[ALIAS]/[RULECODE]/[DATE]

ここで、IPとPORTはInnoRulesサーバのアドレスとサービスポートであり、ALIASはルールRESTサービスシステムエイリアスである。RULECODEは呼び出す対象ルールのコードであり、DATEはルール実行基準日である。DATEは省略できる。

エイリアスは10文字以内のアルファベットの大小文字、数字またはアンダースコア(‘\_’)のみで構成される。

次は、エイリアスを指定する段階である。

[表 105]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールRESTサービス -> ルールRESTサービスシステムエイリアス変更  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム 'DEV'のエイリアスを変更します。(デフォルト値: DEV)  >> |

デフォルト値としては単位システムに指定したルールエイリアスが設定される。エイリアスは他の単位システムと重複できない。設定が完了すると「2.4.7.1 ルールRESTサービス」に移動する

#### ルールRESTサービス活性化

1つのInnoRulesサーバが複数の単位ルールシステムのルールをRESTサービス化できる。

次はRESTサービスを活用する単位ルールシステムを選択する段階である。

[表 106]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールRESTサービス -> ルールRESTサービスを活性化  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールRESTサービスを活性化するシステムを選択してください。ここで選択されていない単位ルールシステムについては、ルールRESTサービスが無効になります。 n何も選択しない場合は、ローカルインストールノードのすべてのルールRESTサービスが無効になります。  (\*) 1. DEV  開発環境  複数選択してください。選択を終了するには'0'を入力してください。 >> |

サービスを有効にする単位システムの番号を一度に1件ずつ入力する。選択完了後に0を入力する。RESTサービスを無効化するためには単位システムの番号を入力せずに0を入力する。有効化された単位システムを変更しない場合は何も入力せずにエンターキーを押す。現在、有効になった単位ルールシステムは番号前に(\*)表示が付く。

有効にする単位システムがある場合は「2.4.7.4 ルールRESTサービス設定」に移動、そうではない場合は「2.4.7.1 ルールRESTサービス」に移動する。

#### ルールRESTサービス設定

ルールRESTサービスURLで表示できるIPアドレスを選択する段階である。このIPは「2.4.7.1 ルールRESTサービス」のURLを生成することに使用される。

[表 107]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールRESTサービス -> ルールRESTサービスを活性化 -> ルールRESTサービスの設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールRESTサービスの情報を入力してください。  [必須] ルールRESTサービスの接続アドレスを選択してください。  1. 172.17.0.1  2. 192.168.122.1  3. 192.168.0.147  4. 127.0.0.1  >> |

ルールRESTサービスはInnoRulesのHTTPサービスを利用するため、InnoRulesサーバがバインドしたすべてのIPアドレスを利用してアクセスができる。InnoRulesサーバはリストの全IPアドレスをバインドして実行されるため、IPアドレス選択がRESTサービスの動作に影響を及ぼすことはない。ただし、選択されたアドレスがURLに反映され、当該URLがアプリケーション開発者に伝達されるため、Webサービスを呼び出すアプリケーションからアクセスできるIPアドレスを選択する。

IPアドレスを選択すると「2.4.7.5 システムエイリアス設定」に移動する。

#### システムエイリアス設定

次の表は「2.4.7.3 ルールRESTサービス活性化」で選択した単位システムに対するルールRESTサービスのシステムエイリアスを設定する段階である。

[表 108]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  サービスタイプ選択 -> ルールRESTサービス -> ルールRESTサービスを活性化 -> ルールRESTサービスシステムエイリアス設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  選択したルールRESTサービスのエイリアスを設定してください。  [必須] DEV(デフォルト値: DEV) >> |

エイリアスは単位ルールシステム別に設定する。エイリアスを設定する方法は「2.4.7.2 ルールRESTサービスシステムのエイリアス変更」と同様である。表の例のDEVシステムは^eを入力して基本ルールRESTサービスとして登録した。エイリアス設定が完了すると「2.4.7.1 ルールRESTサービス」に移動する。

## ルールアプリケーション構成ウィザード

ルールアプリケーションは、ルールサービスを呼び出すアプリケーションである。アプリケーションがルールサーバにルールサービスを要求する。ルールサーバはTCPコネクタサーバのように離れた場所にある場合や、アプリケーション内部にあることもある。どのような場合でも、ルールサービスを呼び出すためにはアプリケーションにルールサービスを呼び出せるリソースを生成する必要がある。これらのリソースには、TCPコネクタサーバに接続できるコネクションプールまたはローカルルールサーバおよびルールデータベースコネクションプールなどが含まれるケースがある。

ルールアプリケーション構成ウィザードは、アプリケーションのルールサービス呼び出し方式、および内部リソースの構成を設定できるウィザードである。

ルールアプリケーションウィザードは、ルールアプリケーションを構成するウィザードである。scriptディレクトリでは、次のコマンドにより実行する。

[表 109]

|  |
| --- |
| UNIX系列では :  ./runtime-wiz.sh  Windowsでは :  runtime-wiz.bat |

### 導入

次の表は、ルールアプリケーション構成ウィザードの導入段階である。

[表 110]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesルールアプリケーション設定ウィザードへようこそ。このウィザードではルールアプリケーションの設定を管理できます。  [エンターキーを押下してください。] |

### ルールシステム選択

アプリケーションがどのような単位システムのルールサービスを呼び出すのかを選択する段階である。1つのアプリケーションは、1つの単位システムのルールのみ呼び出すことができる。

[表 111]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステムを選択してください。  1. DEV  Development  2. PROD  Production  選択してください。 >> 2 |

### ルールアプリケーション設定

次の表は、選択された単位ルールシステムに登録されたルールアプリケーション設定の一覧を示す。

[表 112]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステム'PROD'のルールアプリケーションを設定します。作業を選択してください。  [項目]  --------------------------------------------------------------  | 設定名 | ルールサービスタイプ | DB | ログレベル | 説明 |  --------------------------------------------------------------  1 | PROD-Online | local | RULEDB | INFO | Rule Service for online App |  ------------------------------------------------------------------------------------  [可能な作業]  a. 追加 b. 削除 c. 修正 d. 詳細情報  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

1つのルールアプリケーション設定を複数のアプリケーションが使用できる。同一設定を利用するアプリケーションは、同一方式でルールサービスを呼び出す。

一覧の各フィールドの意味は、次のとおりである。

* 設定名

ルールアプリケーション設定の名前である。アプリケーションが設定を指定する際に使用される。設定名称は、単位ルールシステムで固有のものとする。

* ルールサービスタイプ

ローカルルールサーバを利用するのか、リモートルールサーバを利用するのかを表す。

* DB

ルールサービスタイプがローカルルールサーバである場合、サーバが使用するルールデータベースの名称である。

* ログレベル

ルールアプリケーションのログレベルである。

* 説明

この設定についての説明である。

作業メニューは、次のとおりである。

* 追加

ルールアプリケーション設定を追加する(「2.5.4 ルールアプリケーション設定の基本情報」)。

* 削除

ルールアプリケーション設定を削除する。この設定を使用するアプリケーションがある場合エラーが発生するため、この設定の使用状況を確認する必要がある。

* 修正

ルールアプリケーション設定を変更する(「2.5.4 ルールアプリケーション設定の基本情報」)。

* 詳細情報

ルールアプリケーション設定を要約して表示する。

### ルールアプリケーション設定の基本情報

次の表は、ルールアプリケーション設定の基本情報を入力する段階である。

[表 113]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールアプリケーション設定の基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールアプリケーション設定の基本情報を入力してください。  [必須] ルールアプリケーション設定名 >> online-webapp  [オプション] ルールアプリケーション設定の説明 >> Rule application configuration for an online web application  [必須] ログレベル  1. ERROR  2. WARN  (\*) 3. INFO  4. DEBUG  5. TRACE  >> 4  [オプション] ログタグを指定してください。タグを指定するためにルールアプリケーションのシステムプロパティ${system-property}を使用できます。 >> ${container.name} |

入力の必要な基本情報は、次のとおりである。

* ルールアプリケーション設定名
* ルールアプリケーション設定に関する説明
* ログレベル

ルールアプリケーション全体のログレベルを意味するものではなく、ルールサービスの呼び出しと関係する部分のログを意味する。ログレベル別に記録されるログは、次のとおりである。

* ERROR

ルールシステムに関連する重大な障害

* WARN

ルールサービスの呼び出しがエラーを返した場合。入力値のエラーなどビジネスロジックレベルのエラーによるルールサービス呼び出し失敗

* INFO

ルールシステムの主な作業ログ。ルールサーバの初期化や終了など

* DEBUG

ルールサービス呼び出しや業務データベースクエリなどに所要した時間など

* TRACE

ルールシステムレベルにおける実行パス追跡のためのログ。ログレベルはERRORが最も高くTRACEが最も低い。ログのレベルが低いほど記録されるログの量が多くなり、実行性能は低下される。

* ログタグ

単位ルールシステムでルールアプリケーション設定が複数になることもあるため、それぞれのログファイルを分離する必要がある。ログタグを指定すると基本ログディレクトリの下位にディレクトリを作成し(ディレクトリの名前はログタグと同一)、その下位にルールアプリケーションのログを記録する。

ログタグに${システムプロパティ}形式の文字列を含めることができる。その場合、この文字列を指定したシステムプロパティの値に変換してログタグとして使用する。この方法は、1つのルールアプリケーション設定を複数のルールアプリケーションが利用する時にログディレクトリを分離するために使用できる。[表 113]の例では、ルールアプリケーションのcontainer.nameシステムプロパティの値がログタグとして使用される。

ルールアプリケーション設定の基本情報を入力すると、「2.5.5 ルールサービスタイプ」に移動する。

### ルールサービスタイプ

次の表は、ルールアプリケーションがどのようなルールサーバを使用するのかを指定する段階である。

[表 114]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールサービスタイプを選択してください。  1. ローカルルールサービス  ルールアプリケーションのJVMの中のルールサーバを実行します。Javaのメソッド  呼び出しを使用して、ルールサービスを呼び出します。  2. リモートルールサービス  リモートのTCPコネクタサーバにルールサービスを要求します。サービス呼び出し  にはTCP/IP通信を使用します。  選択してください。 >> 1 |

ローカルルールサービスを選択する場合、ルールアプリケーション内でローカルルールサーバが実行され、このサーバでサービスを呼び出すことになる。同一JVM内にルールサーバが位置するため、Javaメソッド呼び出しから直接サービスが呼び出されるため性能が向上する。ただし、ローカルルールサーバがアプリケーションJVMのリソースを使用するため、リソース使用量が増えることになる。ローカルルールサービスを選択すると、「2.5.6 ルールデータベースの接続設定」に移動する。

リモートルールサービスを選択する場合、TCPコネクタサーバにルールサービスの呼び出しをすることになり、アプリケーションはTCPコネクタサーバへのTCP/IP接続のみを管理する。起動時間が短く、使用するリソースの量も少ないメリットがある一方、TCP/IP通信による速度の低下が発生することもある。リモートルールサービスを選択すると、「2.5.13 リモートルールサービスのタイプ設定」に移動する。

### ルールデータベースの接続設定

次の表は、ローカルルールサーバが使用するルールデータベースのコネクションプールを指定する段階を示す。

[表 115]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> ルールDB接続設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ローカルルールサーバがルールDBへ接続するために使用する方式を指定してください。  1. データソース参照方式  ローカルルールサーバに既に存在するデータソースからルールDB接続を取得します。  2. 専用コネクションプール方式  ローカルルールサーバが専用で使用するコネクションプールデータソースをロ  ーカルで生成します。  選択してください。 >> |

多くの場合は、ポリシーによりデータベースコネクションプールのようなリソースはアプリケーションフレームワークが管理し、アプリケーションはこの参照のみ可能である。代表的な例として、Webアプリケーション環境でWASがコネクションプールを管理し、Webアプリケーションがこれを参照のみ可能である。

データソース参照方式は、ローカルルールサーバがJNDIを利用してフレームワークが提供するデータソースを参照し、ルールデータベースで利用する方式である。データソース参照方式を利用するためには、フレームワーク担当者と協議して予めフレームワークにルールデータベースに関するデータソースを作成する必要がある。この方式を選択すると、「2.5.7 ルールデータベースに関するデータソース(データソース参照方式)」に移動する。

この方式を選択すると、「2.5.8 ルールデータベースに関するデータソース(専用コネクションプール方式)」に移動する。

### ルールデータベースに関するデータソース(データソース参照方式)

次の表は、データソース参照名を選択する段階である。

[表 116]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> ルールDBに対するデータソース  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールサーバが参照するデータソースの名称を入力してください。  >> RULEDB |

フレームワークに登録されたルールデータベースに関するデータソース名称を入力する。データソースの名称が有効であるかウィザードでは確認できない。誤った名称を入力した場合、ルールアプリケーションが初期化される際にエラーが発生する。したがって、名称を入力する際には注意が必要である。

Webアプリケーションの場合、web.xmlのresource-refに登録されたリソースを参照するためにはリソース参照名の前に“java:comp/env/”を付ける。これについては、Oracle社が公開しているサーブレット仕様書を参照のこと。

データソース名の入力が完了すると、「2.5.9 業務データベース一覧」に移動する。

### ルールデータベースに関するデータソース(専用コネクションプール方式)

次の表は、ルールデータベースに関するコネクションプール作成のための接続情報を選択する段階を示す。

[表 117]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> ルールDBに対するデータソース  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  どのルールDBに対するコネクションプールを生成しますか？  1. RULEDB1  First Production Rule DB  2. RULEDB2  Second Production Rule DB  選択してください。>> 2 |

[表 117]の単位ルールシステムには2つのルールデータベース、RULEDB1とRULEDB2が登録されている。ルールサーバは選択されたデータベースの接続情報を利用してルールアプリケーション内にルールデータベースに関するコネクションプールを作成し、これを利用してルールデータにアクセスする。

ルールデータベースを選択すると、「2.5.9 業務データベース一覧」に移動する。

### 業務データベース一覧

システム構成ウィザードでルールシステムがアクセスする業務データベースについて定義し、基本接続情報も指定した。ルールアプリケーションは、この接続情報を利用できない場合がある。アプリケーションが呼び出すルールサービスがすべての業務データベースにアクセスするわけでもない。

この段階では、ルールアプリケーションからアクセスする業務データベースおよびアクセス方式を指定する。

[表 118]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> 業務データベースリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  接続方式を設定する業務データベースを選択してください。  [項目]  -----------------------------------------------------------------  | ID | 名称 | 接続方式 | 説明 |  -----------------------------------------------------------------  1 | 1 | CUSTOMER | 設定しない | Customer Database |  -----------------------------------------------------------------  2 | 2 | PFDB | 設定しない | Product Factory Database |  -----------------------------------------------------------------  [可能な作業]  a. 編集  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

[表 118]は、業務データベースのリストを表している。業務データベースは、システム構成ウィザードで登録したものである。一覧の各フィールドの意味は、次のとおりである。

* ID、名称、説明

システム構成ウィザードで登録した業務データベースのID、名称、説明である。

* 接続方式

業務データベースへのアクセス方法を示す。接続方式の種類は「2.5.10 業務データベース接続方式の選択」に記載されており、各方式に関する詳細は「3.6.8 業務データベース接続方法の設定」を参照のこと。デフォルトは、未設定で表示されている。

編集を選択して編集する業務データベースのIDを入力すると、「2.5.10 業務データベース接続方式の選択」に移動する。

作業を選択せずにエンターキーを押した場合、「2.5.14 ルールサービス同時利用ユーザ数の設定」に移動する。

### 業務データベース接続方式の選択

次の表は、業務データベースへの接続方式を選択する段階を示す。

[表 119]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> 業務データベースリスト -> 業務データベース接続方式選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールサーバが業務データベースに接続する方式を選択してください。  (\*) 1. 設定しない  この設定を使用するルールアプリケーションは、この業務データベースを使用  しません。  2. 専用コネクションプール（基本接続情報）  システムウィザードで登録した基本接続情報を用いて業務データベースの専用  コネクションプールを生成します。  3. 専用コネクションプール（接続情報指定）  データベース接続情報を指定して業務データベースの専用コネクションプール  を生成します。  4. データソース参照  既存の生成されたデータソースを用いて業務データベースにアクセスします。  選択してください。 >> |

次は各接続方式に関する説明である。

* 設定しない

業務データベースへの接続方式を設定しない。ルールアプリケーションが呼び出すルールサービスがこの業務データベースにアクセスしない場合に選択する。未設定時に、この業務データベースにアクセスするルールサービスを呼び出すと、ルールサービス呼び出しエラーが発生する。

この方式を指定した場合、「2.5.9 業務データベース一覧」に戻る。

* 専用コネクションプール(基本接続情報)

ルールアプリケーションの中にコネクションプールを作成してそれを利用する。システム構成ウィザードに登録した該当単位ルールシステムでこの業務データベースの基本接続情報を接続情報として利用する。

この方式を指定した場合、追加的な情報に関わらず「2.5.9 業務データベース一覧」に戻る。

* 専用コネクションプール(接続情報指定)

ルールアプリケーションの中にコネクションプールを作成してそれを利用する。接続情報は別途指定する。業務データベースがミラーリングされており、ルールアプリケーションがミラーリングされているデータベースにアクセスする場合に、この方式が利用できる。この方式を指定した場合、「2.5.11 業務データベースの接続情報(専用コネクションプール接続情報指定)」に移動する。

* データソース参照

フレームワークで管理されるデータソースを利用して業務データベースにアクセスする場合、この方式を利用する。この方式を指定した場合、「2.5.12 業務データベースの接続情報(データソース参照方式)」に移動する。

### 業務データベースの接続情報(専用コネクションプール接続情報指定)

次の表は、専用コネクションプール(接続情報指定)方式を選択した時、接続情報を指定する段階を示す。

[表 120]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> 業務データベースリスト -> 業務データベース接続方式選択 -> 業務データベース接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  業務データベースの接続情報を入力してください。  [必須] JDBCドライバクラス >> oracle.jdbc.driver.OracleDriver  [必須] JDBC URL >> jdbc:oracle:thin:@batdb:1521:orcl  [必須] データベースID >> customer  [必須] データベースパスワード >> |

ここで使用されるJDBCドライバライブラリはInnoRulesインストールディレクトリではなく、ルールアプリケーションのクラスパスに位置する必要がある。

接続情報指定が完了すると、「2.5.9 業務データベース一覧」に戻る。

### 業務データベースの接続情報(データソース参照方式)

次の表は、データソース参照方式を選択した際のデータソースの名前を指定する段階を示す。

[表 121]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> 業務データベースリスト -> 業務データベース接続方式選択 -> 業務データベース接続情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  業務データベース連結を提供するデータソースの情報を入力してください。  [必須] データソース名 >> DATASOURCE  [オプション] InitialFactoryクラス >>  [オプション] ProviderURL >> |

ルールアプリケーションがWebアプリケーションである場合、resource-refに登録されたリソースを使用する際にはデータソース名の前にjava:comp/env/をつける。またはアプリケーションフレームワークのネーミングシステムに特定のInitialFactoryやProviderURLが必要な場合は、指定できる。

接続情報指定が完了すると、「2.5.9 業務データベース一覧」に戻る。

### リモートルールサービスのタイプ設定

リモートルールサービスを選択した後、次はルールアプリケーションが使用するTCPコネクタサーバグループを選択する。

[表 122]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> TCPコネクタサーバグループ選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  リモートルールサービスの呼び出しに使用するTCPコネクタサーバグループを選択してください。  (\*) 1. online  選択してください。 >> |

ルールアプリケーションは、TCPコネクタサーバグループに指定されたポリシーに従い、そのグループに属するTCPコネクタサーバにルールサービスを要求する。

### ルールサービス同時利用ユーザ数の設定

次の表は、アプリケーションが同時に呼び出すルールサービスの数を指定する段階である。

[表 123]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールサービスのタイプ -> 同時利用ユーザ数の設定  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールサービスの同時利用ユーザ数を設定してください。  [必須] 最大同時利用ユーザ数 >>  [必須] 平均同時利用ユーザ数 >> |

このユーザ数に基づいて、ルールサービスを呼び出すためのリソースが割り当てられる。このようなリソースにはデータベースコネクション(ローカルルールサービス)や、TCP/IPセッション(リモートルールサービス)などがある。

### ルールアプリケーション設定要約および保存

ルールアプリケーションの情報をすべて入力後に、次の表のような要約情報を確認して設定を構成する。設定を構成しないと作業内容が削除される。

[表 124]

|  |
| --- |
| [ルールアプリケーション設定の要約]  \* 設定の名称: DEVPART  \* 説明: dev  \* ログ  レベル: TRACE  タグ: test/  \* 同時利用ユーザ  最大: 10  平均: 10  \* ルールサービスのタイプ: ローカルルールサービス  ルールDB接続方式: 専用コネクションプール方式（RULESDB\_DEV）  業務データベースリスト  メタDB: 専用コネクションプール（基本接続情報）  業務DB: 専用コネクションプール（基本接続情報）  \*モニターサーバ接続情報: Use default information  \*ルール呼出ログ情報: Disabled  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  この設定でルールアプリケーションを構成します。よろしいですか？  1. Yes  (\*) 2. No  選択してください。>> |

## アドオンウィザード

InnoRulesは固有の機能以外に、ユーザが追加可能な複数のインタフェースを提供する。これはアドオンで、次のような機能がある。

* ユーザ定義関数
* コールバックサービス

ユーザ定義関数とコールバックサービスに対する詳細は「InnoRules Application Programming Interface Guide」と「InnoRules User's Guide of Rule Builder」を参照のこと。

アドオンウィザードは上記のアドオンをルールシステムに容易に登録できるようにする。アドオンウィザードは次のコマンドで実行する。

[表 125]

|  |
| --- |
| UNIX系列では :  ./addon-wiz.sh  Windowsでは :  addon-wiz.bat |

アドオンを設定するためには、アドオンウィザード実行の前に、ユーザが作成したjarファイル形式のアドオンライブラリをインストールディレクトリのlib/addonsにコピーする。アドオンウィザードはこのディレクトリのライブラリを読み込んで使用可能なアドオンリストを抽出する。

### 導入

次の表は、アドオンウィザードの導入段階である。

[表 126]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  導入  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  InnoRulesアドオンウィザードへようこそ。このウィザードではルールアプリケーションに様々な機能を設定できます。  [エンターキーを押下してください。] |

### アドオンタイプの選択

次の表は、設定するアドオンの類型を選択する段階である。

[表 127]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  アドオンタイプ選択  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  構成するアドオンのタイプを選択してください。終了はエンターキーを押下してください。  1. ユーザ定義関数  ユーザ定義関数を構成します。  2. コールバックサービス  コールバックサービスを構成します。  選択してください。 >> |

ユーザ定義関数を選択すると「2.6.3 ユーザ定義関数リスト」に移動する。コールバックサービスを選択すると「2.6.4 コールバックサービスリスト」に移動する。

### ユーザ定義関数リスト

次の表は登録済み、または、登録可能なユーザ定義関数のリストを示す。

[表 128]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  アドオンタイプ選択 -> ユーザ定義関数のリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  次は使用可能なユーザ定義関数のリストです。  [項目]  ----------------------------------------------------------------------------------  | 関数名 | クラス | 登録有無 | エラー |  ----------------------------------------------------------------------------------  1 | CONVERTCOLTOROW | com.innorules.udf.ConvertColToRow | V | |  ----------------------------------------------------------------------------------  2 | EXECRULEPARAMS | com.innorules.udf.FuncEXECRULEPARAMS| V | |  ----------------------------------------------------------------------------------  3 | RID | com.innorules.udf.FuncRID | | |  ----------------------------------------------------------------------------------  [可能な作業]  a. 登録 b. 登録解除 c. すべて登録 d. 一覧情報更新  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

ウィザードは既存の登録されたユーザ定義関数をリストの上段に見せる。登録有無フィールドの「V」は、関数が既に登録されているユーザ定義関数であることを示す。登録された関数に加えて、ウィザードはlib/addonsディレクトリのjarファイル中のユーザ定義関数インタフェース(com.innoexpert.innorulesj.suite.udf.UserDefinedFunction)を実装したクラスを検索してリストに表示する。既に登録された関数のクラスはこのjarファイルの中にある。ない場合は、エラーフィールドに「クラスが見つからない」という警告メッセージが表示される。また、既に登録した関数名と重複してはいけない。重複した場合は、エラーフィールドに「同一名前のユーザ定義関数が既に登録されている」という警告メッセージが表示される。

各フィールドの意味は、次のとおりである。

* 関数名

ユーザ定義関数名である。ユーザ定義関数名はUserDefinedFunction.getNameメソッドを呼び出して取得する。関数名は登録後にも変更できるが(ユーザ定義関数コードを修正してgetNameで別の名前に変換できるようにする)、その場合には既に登録された他の関数と関数名が重複する可能性がある。この場合、上記のように警告メッセージが出力される。

* クラス

ユーザ定義関数クラス名である。

* 登録有無

このユーザ定義関数をルールシステムで使用できるように登録した場合、「V」が表示される。

* エラー

登録されたユーザ定義関数に問題がある時、この警告を解消しない場合はランタイムでエラーが起こる。

作業メニューは次のとおりである。

* 登録

登録されていないユーザ定義関数を登録する。既に登録されたユーザ定義関数に登録する関数名と同一名称の関数がある場合は登録できない。

* 登録解除

既に登録されたユーザ定義関数に対して登録を解除する。

* すべて登録

登録できるユーザ定義関数をすべて登録する。既に登録されたユーザ定義関数に同じ名前の関数がある場合、該当関数を除いて登録する。

* 一覧情報更新

ユーザ定義関数の使用可能な状態をリフレッシュする。ウィザードはlib/addonsディレクトリのライブラリを 再度分析の上、リストを更新して、既に登録されたユーザ定義関数に問題があるかを検査する。

### コールバックサービスリスト

次の表は、登録済み、または登録可能なコールバックサービスファクトリのリストを示す。

[表 129]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  アドオンタイプ選択 -> コールバックサービスのリスト  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  次は使用可能なコールバックサービスファクトリのリストです。  [項目]  ---------------------------------------------------------------------------------------  | クラス | サービス名称 | 登録可否 | エラー |  ---------------------------------------------------------------------------------------  1 | com.innorules.callbacks.EAIDispatcherFactory | EAI | V | |  ---------------------------------------------------------------------------------------  2 | com.innorules.callbacks.WSInvokerFactory | | | |  ---------------------------------------------------------------------------------------  [可能な作業]  a. 登録 b. 登録解除 c. 一覧情報更新  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

ウィザードは既に登録されたコールバックハンドラファクトリをリストの上段に表示する。登録可否フィールドの「V」は、ファクトリがコールバックサービスとして登録されていることを示す。既に登録されたコールバックハンドラファクトリに加えて、ウィザードはlib/addonsディレクトリのjarファイル中のコールバックハンドラファクトリインタフェース(com.innoexpert.rulesclient.CallbackHandlerFactory)を実装したクラスを検索してリストに表示する。既に登録されたファクトリのクラスはこのjarファイルの中に存在する。存在しない場合、エラーフィールドに「クラスが見つからない」という警告メッセージが表示される。

各フィールドの意味は、次のとおりである。

* クラス

コールバックハンドラファクトリクラス名である。

* サービス名称

コールバックハンドラファクトリのコールバックサービスへの登録状況を表示する。コールバックサービスとして登録されていないファクトリは空欄で表示される。複数のファクトリが同一コールバックサービスとして登録できない。

* 登録可否

コールバックハンドラファクトリがコールバックサービスとして登録されている場合は「V」が表示される。

* エラー

既に登録されているファクトリクラスがlib/addons/から見つからない場合、警告メッセージが表示される。この警告を解消しない場合はランタイムでエラーが起こる。

作業メニューは、次のとおりである。

* 登録

未登録のコールバックハンドラファクトリをコールバックサービスに登録する。このメニューを選択すると、「2.6.5 コールバックサービス登録」に移動する。

* 登録解除

コールバックサービスとして登録されたコールバックハンドラファクトリを登録解除する。

* 一覧情報更新

コールバックハンドラファクトリの使用状態をリフレッシュする。ウィザードはlib/addonsディレクトリのライブラリを再度分析の上、リストを更新して、既に登録されたコールバックサービスファクトリの問題有無を検査する。

### コールバックサービス登録

次の表は、コールバックハンドラファクトリをコールバックサービスとして登録する段階である。

[表 130]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  アドオンタイプ選択 -> コールバックサービスのリスト -> コールバックサービスの登録  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  コールバックハンドラファクトリcom.innorules.callbacks.WSInvokerFactoryに対応するサービス名称を入力してください。  >> |

コールバックサービス名は64字以内の英文字・数字・アンダースコア(‘\_’)で構成される。また、既に登録されているサービス名を使用することはできない。

## ウィザード使用方法

ここでは、ウィザードを使用において必要かつ簡単な方法について説明する。

### 入力

#### エンターの用法

入力しなくてもいいフィールドがある場合、ラベルの前に[選択]というタグがつく。[選択]フィールドで値を入力せずにエンターキーを押すと、未設定と見なされる。

[表 131]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールシステム選択 -> ルールアプリケーション設定 -> ルールアプリケーション設定の基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  ルールアプリケーション設定の基本情報を入力してください。  [必須] ルールアプリケーション設定名 >> batch  [オプション] ルールアプリケーション設定の説明 >> |

[表 131]の「ルールアプリケーション設定に関する説明」は、選択フィールドである。ここで値を入力せずにエンターキーを押した場合、未設定と見なされる。

既に設定された内容の修正およびルールシステムが推奨する値がある場合は、デフォルト値が出力される。

[表 132]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理 -> 単位ルールシステム基本情報  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  単位ルールシステムの基本情報を入力してください。  [必須] 名称(デフォルト値: Development) >> |

括弧の中に「デフォルト値：」というラベルがあるため、デフォルト値が存在する。デフォルト値が存在する場合、値を入力せずにエンターキーを押すことで、デフォルト値が設定できる。[表 132]の例で、名称の欄でエンターキーを押した場合、名称に「Development」が指定される。

リストを照会する画面で作業を選択すると、その画面がサブメニューに移動する。その画面で作業を選択せずにエンターキーを押すと、次に進む。

[表 133]

|  |
| --- |
| \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  設定対象選択 -> 単位ルールシステム管理  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  登録された単位ルールシステムのリストです。新規単位ルールシステムの追加および既存単位ルールシステムの修正ができます。  [項目]  -------------------------  | ID | 名称 |  -------------------------  1 | DEV | Development |  -------------------------  2 | PROD | Production |  -------------------------  [可能な作業]  a. 追加 b. 修正およびルールDB管理 c. 一番下の単位ルールシステム削除  作業を選択してください。スキップするにはエンターキーを押下してください。 >> |

[表 133]の単位ルールシステム管理には「追加」、「修正およびルールDB管理」などのサブメニューがある。これを選択せずにエンターキーを押した場合、次に進む。

#### 特殊文字

値を入力する欄で、次の特殊文字が利用できる。特殊文字を利用する場合、他の文字との同時使用や文字の前後に空白を入力しないことに注意する。

* ^b

「戻る」をクリックしたと見なす。直前に実行した段階に移動する。

* ^e

空白を入力したと見なす。

* ^q

ウィザードを終了する。Ctrl-Cを利用して終了する場合、ウィザードが正常終了されていないため、ロックファイルやUUIDファイルが残ることがある。

#### 確認メッセージに対する応答

[表 134]

|  |
| --- |
| 設定を終了しますか？ (Yes/No) >> |

[表 134]のように「Yes」または「No」の確認メッセージで、「Yes」または「No」すべてを入力せずに、大小文字区分なく「Yes」の一部分のみを入力してもよい。例えば、次はすべて「Yes」と見なされる。

* Yes
* yes
* y

### ウィザード使用シナリオ

インストールウィザードを利用してイントールが完了した場合、必要に応じて順序に関係なく設定ウィザードが使用できる。次は、頻繁に発生するシナリオと、使用するウィザードを順番とおりに記述した例である。

* 開発システムの構成：ルール開発者がルールビルダーを利用したルール開発ができるように依頼された場合、または開発システムにルールシステムのインストールを依頼された場合
* インストールウィザードを利用して開発システムにInnoRulesをインストールする。
* ルールシステム構成ウィザードを利用して、開発単位システムを設定する。
* サービス構成ウィザードを利用してルールビルダーサービスを有効にし、InnoRulesサーバを実行する。
* アプリケーションの構成：開発システムに業務アプリケーションがインストールされた後、業務アプリケーションがルールサービスを呼び出せるように依頼された場合
* ルールアプリケーション構成ウィザードを利用してルールアプリケーションを設定し、アプリケーションにルール初期化ができるように構成
* 業務データベースの追加：ルール開発者が業務データベースにアクセスするルールサービスを開発できるように要求された場合
* ルールシステム構成ウィザードを利用して、業務データベースを登録
* ルールアプリケーション構成ウィザードを利用して、アプリケーションが業務データベースに接続する方法を登録
* 運用システムの構成：運用システムがインストールされた後、運用システムにルールシステムのインストールを依頼された場合
* インストールウィザードを利用して、運用システムにInnoRulesをインストールする。
* 設定クラスタウィザードを利用して、運用システムにインストールされたInnoRulesをクラスタとして構成する。
* ルールシステム構成ウィザードを利用して、運用系クラスタに開発単位システム、運用単位システムを登録した後に、業務データベースを登録する。
* サービス構成ウィザードを利用して、ルールビルダーサーバを移管専用として有効にする。
* ルールアプリケーション構成ウィザードを利用して、運用アプリケーションがルールサービスを呼び出せるようにする。

### 設定適用時点

ウィザードを利用して設定を変更した場合、設定内容は設定クラスタ内のすべてのインストールノードのconfig/rule-systems.xmlに保存される。ただし、それがクラスタ内で実行されるすべてのInnoRulesサービスやルールアプリケーションに反映されるわけではない。InnoRulesサービスはInnoRulesサーバを再起動する際に反映され、ルールアプリケーション設定の変更もルールアプリケーションの再起動の際に反映される。

# Rule System Concepts

本章では、InnoRulesのインストールと運用において必要ないくつかの概念について説明する。

## InnoRulesをインストールするとは？

InnoRulesの**インストール(installation)**とは、InnoRulesに関連する実行スクリプト、プログラムライブラリ、設定ファイルなどを既に定義されたディレクトリ構造でファイルシステムに位置づけることを意味する。インストールによって作成されたディレクトリとファイルを、**インストールノード(Installation Node)**という。1つのホストにディレクトリが異なるようにインストールすると、複数のインストールノードを作成できる。ただし、それぞれのインストールノードはそれぞれライセンスを必要とする。それぞれのインストールノードの設定を変更するためには、そのノードに含まれているウィザードスクリプトを実行する。

## 設定クラスタ

2つ以上のインストールノードの設定を同期化して管理できる。設定が同期化されたインストールノードの集合を、**設定クラスタ**という。設定クラスタのノードのいずれかで設定を変更すると、その変更内容は他のすべてのノードでも同様に変更され同期化される。

設定クラスタにインストールノードを追加するためには、そのノードの設定ファイルにリモートでアクセスし修正する方法を指定しなければならない。InnoRulesでは、リモートファイルアクセスプロトコルとして、SFTPとFTPがサポートされている。任意のインストールノードから他のインストールノードにアクセスできるようにする必要があるため、localhostや127.0.0.1のようなループバック(loopback)アドレスはインストールノードのアドレスとして指定できない。

設定クラスタにインストールノードを追加するためには、**クラスタ構成ウィザード(Cluster Configuration Wizard)**を利用する。任意のノードでウィザードを利用して設定クラスタの構成を変更するためには、そのノードがまず設定クラスタに追加されている必要がある。

インストール後に、設定クラスタを必ず構成する必要があるわけではない。インストールノードが1つのみの場合は、設定クラスタを構成する必要はない。

## ルールシステム

**ルールシステム(Rule System)**とは、ライフサイクル(ルールの作成、照会、保存、実行)を管理するために有機的に動作するソフトウェアコンポーネントとリソースのことをいう。

### 単位ルールシステム

アプリケーションは、運用環境に展開されるまで開発やテストなどの段階を経る。ルールも開発から運用に至るまでの複数の段階を経る場合がある。ルールはアプリケーションと密接な関係を持つため、一般的にルールの開発段階とアプリケーションの開発段階は1：1で対応する。ルールシステムでアプリケーション開発段階に対応する概念が、単位ルールシステムである。

**単位ルールシステム(Unit Rule System)**とは、同期化された一連のルールリポジトリと、これらを共有するルールサーバおよびルールビルダーサーバの集合を意味する。ここで同期化(synchronization)とは、ルールリポジトリの内容がすべて同様に管理されるという意味である。

一般的に単位ルールシステムの名称は、対応するアプリケーション開発段階の名称とする。例えば、アプリケーションの開発段階が開発、テスト、運用で構成されたシステムのルールシステムは、それに対応する3つの単位ルールシステムで構成される。この単位ルールシステムは対応するアプリケーション開発段階の名称にちなんで、開発ルールシステム、テストルールシステム、運用ルールシステムという名称の付与を推奨する。

#### ルールリポジトリ

ルールや項目などルールの内容を構成する要素を、**ルール要素(Rule Element)**という。また、ルール要素が保存される場所を、**ルールリポジトリ(Rule Repository)**という。ルール要素はルールビルダーサーバによって照会・修正される。

通常、データベースがルールリポジトリとして使用される。**ルールデータベース(Rule DB)**とは、ルールリポジトリの用途で作成された独立したスキーマが付与された一連のテーブルのことをいう。データベースサーバとしては、SQL-99とJDBC3.0に準拠するすべてのデータベースサーバが使用できる。

1つの単位ルールシステムには、複数のルールデータベースが設定できる。ルール編集などルール要素を変更した場合、ルールビルダーサーバはすべてのルールデータベースをリアルタイムで同様の変更を実施し同期化を行う。

### 移管

アプリケーションシステムと同様に、ルールシステムでも開発ルールシステムで開発が完了したルール要素をテストルールシステムへコピーや、テストルールシステムでテストが完了したルール要素を運用ルールシステムにコピーできる。このように、単位ルールシステムリポジトリのルール要素の一部を他の単位ルールシステムのリポジトリにコピーすることを、**移管(Transition)**という。



[図 6 移管の例]

[図 6 移管の例]で、Unit Rule System1のルール要素はUnit Rule System2に移管され(transition 1)、Unit Rule System2のルール要素はUnit Rule System3に移管される(transition 2)。移管において、コピーするルール要素が位置する単位ルールシステムを**コピー元ルールシステム(Source Rule System)**といい、ルール要素がコピーされる単位ルールシステムを**コピー先ルールシステム(Target Rule System)**という。コピー元ルールシステムとコピー先ルールシステムをそれぞれ略して、コピー元システムとコピー先システムともいう。

移管には、次のような制約がある。

* 1つの単位ルールシステムは、2つ以上の単位ルールシステムのコピー元システムにはなれない。また、1つの単位ルールシステムは、2つ以上の単位ルールシステムのコピー先システムにもなれない。(次の図のa))
* 1つの単位ルールシステムから移管されたルール要素は、自ルールへ移管することはできない。すなわち、ループ形式の移管は不可能である。(次の図のb))

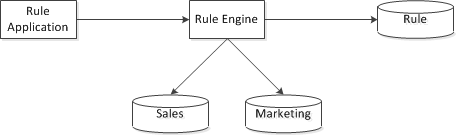


[図 7 移管の制約]

移管された要素の変更履歴はコピー先システム上に記録され、ルール要素の変更履歴とその内容をコピー先システムで確認できる。

## 業務データベース

ビジネスルールを処理する中で、業務データを照会する場合がある。ルールデータベースと対照的な概念で、業務データが保存されているデータベースを**業務データベース(Business Database)**または**業務DB**という。ルールシステムは業務データベースにアクセスしてクエリを実行できるが、主にルールエンジンがデータベースルールまたはデータルールを実行する際に照会される。



[図 8 Business Database]

[図 8 Business Database]は、業務データベースにアクセスする単位ルールシステムの例である。[図 8 Business Database]のルールエンジンは、それぞれ異なる目的で3つのデータベースにアクセスする。\*1 Ruleと表記されているデータベースはルールデータベースで、単位システムのルールが保存されており、ルールエンジンがルールデータを読み込むために照会する。SalesとMarketingで表記されているデータベースにはそれぞれ営業データとマーケティングデータが保存されており、それぞれに対応する業務アプリケーションがこれらのデータを管理する。ルールアプリケーションがルールサービスを呼び出すと、ルールエンジンは(必要ならば)ルールデータベースからルールデータを読み込んでルールを実行し、この過程でDBルールまたはデータルールが実行された場合、これらの業務データベースにアクセスする。

ルールシステムからアクセスできる業務データベースは、ルールシステム構成ウィザード(System Wizard)を利用して登録できる。業務データベースを登録する際、SalesやMarketingのように業務データベースを識別できる名称と物理的な接続情報を登録する。データベースのルールまたはデータモデルがどの業務データベースにアクセスするかは、ルールビルダーを利用してそれぞれのルールまたはデータモデルに指定する。ただし、ルールビルダーには物理的な接続情報は表示されず、業務データベースの名称のみが表示される。

同一の業務データベースでも、ルールアプリケーションごとに異なるデータベースサーバにアクセスする必要がある。例えば、Salesデータの照会のために、オンラインアプリケーションとバッチアプリケーションがそれぞれ異なるサーバに接続する必要がある。業務データベースに関する情報は、ルールアプリケーションの設定でオーバーライド(override)できる(「3.6.8 業務データベース接続方法の設定」の設定参照)。そうでない場合は、システムウィザードを利用して登録した接続情報を基に使用することになる。

1. 一般的に、各データベースの接続情報は異なると考えられる。

## ルールサービス

**ルールサービス(Rule Service)**とは、アプリケーションからの入力によって指定されたルールを判定し、その結果をアプリケーションに返すサービスのことをいう。ルールサービスはルールサーバが提供し、ルールサービスを呼び出すアプリケーションを**ルールアプリケーション(Rule Application)**という。

ルールサービスはルールアプリケーションとルールサーバとの通信方法によって、ローカルルールサービスとリモートルールサービスに区分される。**ローカルルールサービス(Local Rule Service)**はルールサーバがアプリケーションと同一JVMで実行され、通信レイヤを介さずにJavaメソッドのみを利用して呼び出されるルールサービスである。ルールサーバがJavaで作成されているため、Javaアプリケーションのみがローカルルールサービスを利用できる。一方、**リモートルールサービス(Remote Rule Service)**はサービスの呼び出しのためにネットワーク層を利用するルールサービスである。ネットワーク層を利用するため、ルールアプリケーションはJavaで作成されたアプリケーションに限定されない。

リモートでルールをサービスするためには、ネットワークを介したルールサービス要求を受け入れルールエンジンに伝達し、ルール実行結果をアプリケーションに伝達するサーバが必要となる。そのようなサーバには、次のようなものがある。

* TCPコネクタサーバ(次の図のa))
* Webサービスアダプタ(次の図のb))



[図 9 Remote Rule Server]

**TCPコネクタサーバ(TCP Connector Server)**は、TCP/IPとInnoRules固有のメッセージプロトコルを利用してリモートルールサービスを提供する。TCPコネクタサーバを利用してルールサービスを呼び出すためには、InnoRules Rule APIを使用する。APIが使用可能な場合、異なるプログラミング言語で作成されたルールアプリケーションもTCPコネクタサーバを利用できる。ルールAPIがサポートするプログラミング言語については、「InnoRules Application Programming Interface Guide」を参照のこと。

**Webサービスアダプタ(Web Service Adapter)**は、SOAP/HTTPを利用してルールサービスを呼び出すリモートルールサービスである。Webサービスアダプタは公開されたプロトコルのSOAPを使用するため、InnoRules Rule APIが必要なく、プログラミング言語の制約もない。

TCPコネクタサーバとWebサービスアダプタは、InnoRulesサーバで有効にできる(「3.8.4.1 サービス有効化」を参照のこと)。

## ルールアプリケーション設定

ルールシステムは、多様なサービスを提供する。ルールサーバはローカルルールサービスを提供し、ルールビルダーサーバはルールビルダーサービス、TCPコネクタサーバはリモートルールサービスを提供する。これらのサーバはJavaで作成されており、JVM上で動作する。このサーバが正常に動作するため、またサーバが提供するサービスを呼び出すためには、様々なコンポーネント(component)がJVM上に構成される必要がある。次は、代表的な構成要素である。\*2

* ロギング
* データソース：ルールデータベースおよび業務データベースについて
* ルールリポジトリキャッシュ
* ルールインタフェースクラスタ：ローカルまたはリモートルールサーバでのインタフェース管理

このようなコンポーネントは、他のコンポーネントまたはサーバと依存関係を持つ。そのため、任意のサーバを実行するためには、サーバが必要とするコンポーネント、およびそれらコンポーネントが必要とする別のコンポーネントがJVMに構成されている必要がある。

ルールシステムサービスを提供するために、またサービスを呼び出すためにJVMにコンポーネントとサーバを構成したものを**ルールランタイム(Rule Runtime)**といい、ルールランタイムを構成する過程を**ルールランタイム初期化(Rule Runtime Initialization)**という。ルールランタイムにコンポーネントとサーバをどのように構成するのかを指定したものを、**ルールランタイム設定(Rule Runtime Configuration)**という。

ルールアプリケーションはルールサービスを呼び出すため、ルールサービスを呼び出すためのルールランタイムがルールアプリケーションのJVMに構成される必要がある。ルールサービスを呼び出せるようにルールランタイムを初期化することを、特に**ルールアプリケーションの初期化(Rule Application Initialization)**という。ルールアプリケーションはルールサービスを呼び出す前に(一般的にアプリケーションが始まる時に)まずルールアプリケーションを初期化し、ルールサービスの呼び出しがすべて終了後(一般的にアプリケーションが終了する直前)、ルールランタイムを終了する。ルールアプリケーションをどのように初期化するかの設定を、**ルールアプリケーション設定(Rule Application Configuration)**という。

ルールアプリケーション設定には、**ルールアプリケーション設定ウィザード(Rule Application Configuration Wizard)**を利用する。ルールサービスを呼び出すすべてのアプリケーションはルールアプリケーションであるが、ウィザードはJavaルールアプリケーションのみをサポートするため、本書ではルールアプリケーションをJavaルールアプリケーションに限定する。他の言語環境におけるルールアプリケーションの設定については、「InnoRules Application Programming Interface Guide」を参照のこと。

1. このコンポーネントの中のいくつかの詳細は、文書の範囲を外れるため省略する。

### 単位ルールシステムの選択

一般的なルールアプリケーションは、自分の配置された段階の単位システムのルールサービスのみを呼び出す。例えば、開発段階の業務アプリケーションは開発ルールシステムのルールのみ、運用段階の業務アプリケーションは運用ルールシステムのルールのみを呼び出す。

ルールアプリケーションの設定は、使用する単位ルールシステムのIDの選択から開始される。ウィザード以降の段階では、選択された単位ルールシステムのリソースのみが指定できる。このようなリソースには、ルールデータベースやTCPコネクタサーバグループなどがある。

### 複数のルールアプリケーション設定と設定名の指定

1つの単位ルールシステムに複数のルールアプリケーション設定を作成できる。それぞれの設定は、名称で識別される。その名称を、**ルールアプリケーション設定名(Rule Application Configuration Name)**という。この名前は、単位ルールシステムの中で一意である。

複数のルールアプリケーションが1つのルールアプリケーションの設定を利用して初期化できる。例えば、Webアプリケーションとして作成された業務アプリケーションがあり、このアプリケーションはクラスタを構成する多数のWASコンテナに配置(deploy)される時、各コンテナへの配置時に同一の設定が適用され、ルールアプリケーション設定名も含まれる。その場合、各コンテナに配置されるアプリケーションは同一のルールアプリケーション設定により初期化される。

同種のアプリケーションでなくても、ルールアプリケーション設定が共有できる。例えば、バッチアプリケーションとオンラインアプリケーションがあり、これらが同一構成のルールランタイムを利用してルールサービスを利用する場合、1つのルールアプリケーション設定を利用して初期化できる。

### ログレベル指定

ログにはルールアプリケーションの初期化や、ルールサービスが呼び出される間に発生したイベントが記録される。イベントは重要度によってレベルに分けられている。レベルとそれに該当するイベントは、次のとおりである。

* ERROR：最も高いレベル

ルールサービスが正常に呼び出されない可能性のあるレベルの障害

* WARN

ルールサービスは呼び出されたが、ルール実行中にエラーが発生する。ルールサービス呼び出しの有無とは関係のない業務レベルのエラー

* INFO

ルールアプリケーションの初期化および終了などに関する主な情報

* DEBUG

ルールの実行時間や処理にかかる時間など、追跡に関するイベント

* TRACE：最も低いレベル

ルールサービスの入力項目と結果

ルールアプリケーション設定時は、ログとして残すイベントのレベルであるログレベルが指定できる。指定されたログレベルと同一または指定したレベルよりも高いレベルのイベントのみがログに残る。

また、ログはファイルに保存される。ログファイルは製品をインストールする際に、指定したディレクトリの下位に次のようなパスに作成される。

[単位ルールシステム名]/[ルールアプリケーション設定名]/innorules.log

ログファイルのデフォルト名は、innorules.logである。1つのルールアプリケーションの設定を利用して複数のルールアプリケーションが実行可能であり、ルールアプリケーション別にログファイルを区分するため**ログタグ(Log Tag)**を指定できる。ログタグを指定すると、次のパスにログファイルが作成される。

[単位ルールシステム名]/[ルールアプリケーション設定名]/[ログタグ]innorules.log

ログタグとしてシステムプロパティとパス区切り文字が使用できる。次はその使用例である。

* ${container.name}

同一ルールアプリケーション設定を利用するWebアプリケーションを複数のコンテナに配置する際、コンテナ名を利用してファイルを分割できる。各コンテナのcontainer.nameシステムプロパティにコンテナ名が保存されている場合、ログタグを上記のように設定すると、各ルールアプリケーションのログファイル名の前にコンテナ名-(ハイフン)がつく。

* ${container.name}/または${container.name}${file.separator}

上記の例で、コンテナ別にルールアプリケーションログを他のサブディレクトリに保存するために、ログタグにパス区切り文字を使用できる。パス区切り文字は、文字またはfile.separatorのようなシステムプロパティが使用できる。

ログファイルにはルールアプリケーションで発生したイベントのみが記録される。ルールアプリケーションがローカルルールサービスを利用するように設定した場合、ルールサーバがルールアプリケーションのJVM内で実行されるため、ルールサーバのログがログファイルに記録される。ただし、リモートルールサービスを利用するように設定した場合、ルールサーバはリモートで実行されるため、ルールサーバのログはリモートルールサーバのログファイルに記録される。

### ルールサービスの形態指定

ルールアプリケーションがローカルルールサービス、または、リモートルールサービスを使用するのかを指定する必要がある。ウィザード以降の段階で、サービス形態に依存する追加情報を入力する。ローカルルールサービスを選択する場合は、ルールデータベースを接続するための設定を行う。リモートルールサービスを選択する場合は、呼び出すTCPコネクタサーバに関する情報を入力する。

### ルールデータベースの接続設定(ローカルルールサービスを使用する場合のみ)

ルールアプリケーションがローカルルールサービスを使用するように設定すると、アプリケーションのJVM内にルールサーバが実行されるように構成される。ルールサーバはルールデータを照会するため、ルールデータベースに接続できる方法に関する追加設定が必要になる。

ルールサーバが使用するデータソースを指定して、ルールデータベース接続方法を設定する。データソースとして、ネーミングシステムに既に登録されているデータソースを参照することもでき、ルールアプリケーション専用のコネクションプールも作成可能である。前者の方式を**データソース参照方式(Data Source Referencing Type)**といい、後者の方式を**専用コネクションプール方式(Private Connection Pool Type)**という。

**データソース参照方式**を使用する場合、データソースのJNDI名を指定する。ルールサービスはデータソースを検索するためにJNDI APIを使用する。この方式を使用する代表的なアプリケーション形態はWebアプリケーションである。Webアプリケーションでなくても、ネーミングシステムを使用可能な場合はこの方式を指定できる。ただし、設定方法においてWebアプリケーションとわずかに差異がある。次は、Webアプリケーション環境での設定方法と意味をまず説明した後、Webアプリケーション以外の場合について説明する。

WebアプリケーションはWebアプリケーションサーバ(WAS)で動作する。一般的にデータソースはWASによって作成・管理され、Webアプリケーションはこれを参照して使用する。WebアプリケーションがWASのデータソースを参照するためには、Webアプリケーション配置記述子(Web Application Deployment Descriptor、web.xml)に参照名を記述する必要がある。\*3 この参照名をWebアプリケーションで参照するにはその前にプレフィックスjava:/comp/env/をつける必要があるため、ルールアプリケーションの設定には参照名にプレフィックスをつけて指定する必要がある。

例えば、WASにルールデータベースについてのデータソースが登録されており、リソースの名前はRULEDB\_POOLだとする。ルールアプリケーションのweb.xmlにはルールデータベースのリソース参照の記述が必要であり、次の表のようにその名称がRULEDBに指定されているとする。

[表 135 Part of Rule Web Application Deployment Descriptor]

|  |
| --- |
| <web-app>  …  <resource-ref>  …  <res-ref-name>RULEDB</res-ref-name>  <res-type>javax.sql.DataSource</res-type>  …  </resource-ref>  …  </web-app> |

ルールアプリケーションがルールデータベースに対するデータソースとしてRULEDB\_POOLというデータソースを使用するためには、次の2点を満たす必要がある。

* ルールアプリケーションの設定においてルールデータベース接続方式をデータソース参照方式に設定し、参照名をjava:comp/env/RULEDBにする。これは、ルールアプリケーション管理者の役割である。
* Webアプリケーションを配置(deploy)する際に、リソース参照RULEDBがWASのデータソースRULEDB\_POOLを参照するようにマッピングする。Webアプリケーションの配置は、WAS管理者の役割である。

この方式でルールWebアプリケーションが配置されたものを図で表すと、次の図のようになる。



[図 10 Referencing a Data Source from a Rule Web Application]

Webアプリケーションではない一般的なJavaアプリケーションも、基本的には同一方法を使用する。ルールアプリケーションの設定の際に、データソース参照名としてアプリケーション実行環境に登録されたデータソースのJNDI名を入力する。更に入力可能なパラメータとして、イニシャルコンテキストファクトリ(Initial Context Factory)とプロバイダURL(Provider URL)がある。\*4

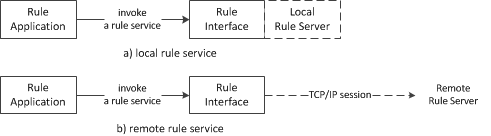
**専用コネクションプール方式**は、ルールアプリケーションが初期化する際にルールデータベース専用のコネクションプールを作成する方式である。コネクションプールのデータベース接続情報として、単位ルールシステムに登録されたルールデータベース接続情報(「3.3.1.1 ルールリポジトリ」を参照のこと)のうちの1つを選択できる。単位ルールシステムにルールデータベースの接続情報を登録する方法については、「2.3 ルールシステム構成ウィザード」を参照のこと。コネクションプールで管理するコネクション数は、ルールサービスの同時利用ユーザ数に合わせて適切に設定される。ルールサービス同時利用ユーザ数については、「3.6.6 同時利用ユーザ数」で説明する。コネクションプールは、ルールアプリケーションの終了時に消滅する。

専用コネクションプール方式は、Webアプリケーションを含むすべての形態のルールアプリケーションに適用できる。ただし、アプリケーションにルールデータベースへのコネクションを管理する他のデータソースがある場合は不要なリソースを消費するため、専用コネクションプール方式を選択する際はこの点を考慮する。

1. Webアプリケーション配置記述子およびリソース参照については、Oracle社が公開しているサーブレット仕様書を参照のこと。この節では、ユーザがWebアプリケーションのインストールに関して十分な知識を持っていることを前提とする。
2. 詳細は、Oracle社が公開しているJNDI関連資料を参照のこと。

### 同時利用ユーザ数

ルールアプリケーションは、同時に複数のルールサービスを個別に呼び出すことが可能である。マルチスレッドアプリケーションがこのような形態でルールサービスを呼び出す。それぞれのサービスの呼び出しは、他のサービスの呼び出しに対して一定量のリソースを占有するようになる。占有するリソースは、ルールサービスの形態によって異なる。ローカルルールサービスの場合、ルールサービスが呼び出されている間に一定量のJavaヒープ(heap)を占有する。リモートルールサービスの場合は、リモートルールサーバへのTCP/IPセッション1つを占有する。ルールサービスの形態によってそれぞれ異なるリソースを共通のインタフェースとして抽象化したものが、**ルールインタフェース(Rule Interface)**である。アプリケーション開発者はルールサービスの形態に関係なく、ルールインタフェースのみを利用してルールサービス呼び出しコードを作成する。それを図で表すと次の図のようになる。



[図 11 Rule Interface]

[図 11 Rule Interface]からわかるように、ルールアプリケーションはルールインタフェースのみを利用してルールサービスを呼び出し、サービスの形態に依存的なリソースはルールアプリケーションに対して隠蔽される。

ルールサービスを呼び出すたびにルールインタフェースおよび関連するリソースを作成した場合、オーバーヘッドが発生する。例えば、リモートルールサービスを呼び出すたびにルールインタフェースを作成すると、毎回新しいTCP/IPセッションを生成し性能が低下する。**ルールインタフェースクラスタ(Rule Interface Cluster)**は、一度作成されたルールインタフェースを破棄せずに再使用できるようにしてくれるルールランタイムのコンポーネントである。ルールインタフェースクラスタは、次のような機能を実行する。

* ルールアプリケーションがルールインタフェースを要求する際、再使用のために保存されているルールインタフェースがある場合はそれを提供する。
* ルールアプリケーションがルールインタフェースを要求する際、再使用のために保存されているルールインタフェースがない場合は、新しいルールインタフェースを作成して提供する。
* ルールアプリケーションがルールインタフェースの使用を終了しクラスタに返した場合、使用したルールインタフェースを再使用するために保管するかまたは破棄する。

ルールアプリケーションは複数のスレッドを利用して同時にルールサービスを呼び出せるため、ルールインタフェースクラスタは多数のルールインタフェースを保管できる。ただし、保管されたルールインタフェースの数が平均的な同時ルールサービスの呼び出し数より多い場合、余分なルールインタフェースは長期間使用されずにリソースのみを占有してしまう。リモートルールサービスの場合を例にすると、クラスタに10件のルールインタフェースが保管されている場合、10つのTCP/IPセッションが存在する。ただし、平均的な同時ルールサービスの要求数が5件程度なら、使用されない5つのTCP/IPセッションは不要なリソースを消費することになる。

ルールアプリケーションが同時に呼び出すルールサービスの数を、**同時利用ユーザ数(The Number of Concurrent Users)**という。同時利用ユーザ数を指定することで、ルールインタフェースクラスタのリソース使用の限界と性能を調節できる。ルールインタフェースにおいて、次を指定できる。

* 最大同時利用ユーザ数(Maximum Number of Concurrent Users)
* 平均同時利用ユーザ数(Average Number of Concurrent Users)

最大同時利用ユーザ数は、ルールインタフェースクラスタが占有するリソースの最大値に影響を与える。非常に多くのルールサービスが同時に要求された場合においても、ルールインタフェースは最大同時利用ユーザ数に指定された数以上作成されることはない。それ以上のルールサービスの要求は、他のルールサービスが処理されるまで待機することになる。例えば、リモートルールサービスに対して最大同時利用ユーザ数が10に指定された場合、同時に20件のルールサービスの要求があった場合、TCP/IPセッションは10件のみ作成され、10件のルールサービスのみが同時に処理される。残り10件の要求は待機となり、他のルールサービスの処理が完了したアイドル状態のTCP/IPセッションが生じた時に処理される。

平均同時利用ユーザ数は、再使用のために保管するルールインタフェースの数を指定することで、平均的なリソース使用量と性能に影響を与える。ルールアプリケーションがルールインタフェースの使用を終了して返す際、現在使用中のルールインタフェースと保管中のルールインタフェースの数の合計が平均同時利用ユーザ数より多い場合、ルールインタフェースは消滅し、そうでない場合は再使用のためにクラスタに保管される。ルールアプリケーションによって使用されずにクラスタに保管されているルールインタフェースを、**アイドルルールインタフェース(Idle Rule Interface)**という。平均同時利用ユーザ数が平均的な負荷に比べて少なく設定された場合、頻繁にルールインタフェースが作成されるため性能が低下する。一方、平均同時利用ユーザ数の設定が多すぎる場合は、使用されないアイドルインタフェースにより不要なリソースを消費する。

### TCPコネクタサーバグループ選択 - リモートルールサービスにのみ該当

リモートルールサービスを選択した場合、サービスを呼び出すTCPコネクタサーバを指定する必要がある。TCPコネクタサーバは使用目的別にグループ化されているが、これをTCPコネクタサーバグループといい、ルールアプリケーション設定時はTCPコネクタサーバグループを指定することになる。TCPサーバグループに関する詳細は、「3.8.4.2 TCPコネクタサーバグループ」を参照のこと。

### 業務データベース接続方法の設定

業務データベースのリストと接続情報は、システムウィザードを利用して管理すると説明した。ルールアプリケーションは、この接続情報を利用して業務データベースの接続を管理するコネクションプールを作成することもできる。ただし、ルールアプリケーションが登録された接続情報を利用できない場合がある。代表的な例として、

1. ルールアプリケーション別に接続する業務データベースが異なる場合がある。例えば、アプリケーション環境がオンラインとバッチに区分されており、使用するデータベースサーバが分離されている場合、各アプリケーションには業務データベースに対する接続情報が異なるように指定する必要がある。
2. アプリケーションまたはアプリケーションフレームワークが業務データベースに対するコネクションプールを直接管理する場合、ルールアプリケーションは別途コネクションプールを作成せず、既に存在するコネクションプールを参照する。代表的な例がWebアプリケーション環境で、ルールアプリケーションはWASに作成された業務データベースに対するデータソースを参照する。

ルールアプリケーションは、次のいずれかの方法で業務データベースに対し接続するように設定できる。

* 専用コネクションプール方式

ルールアプリケーションは業務データベースに対するコネクションプールデータソースを作成し、そこからデータベースへの接続情報を取得する。データベース接続情報はシステムウィザードに登録された接続情報が基本的に使用されるが、直接接続情報を指定することもできる。上記の例外1)の場合、専用コネクションプール方式を利用して直接接続情報を入力することで解決できる。

* データソース参照方式

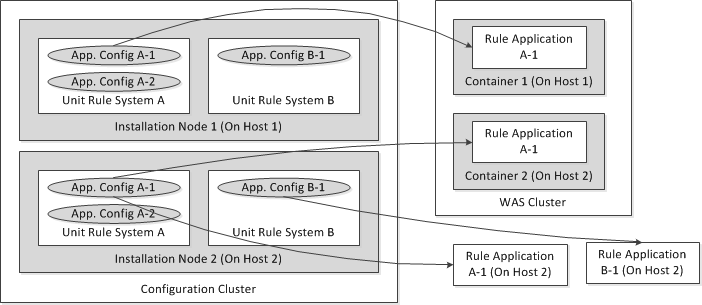
JNDIを利用してネーミングシステムからデータソースを検索し、データベース接続情報を取得する。「3.6.5 ルールデータベースの接続設定(ローカルルールサービスを使用する場合のみ)」のルールデータベースへのデータソース参照方式のような方法でデータソース名を指定し、オプションとしてイニシャルコンテキストファクトリとプロバイダURLが指定できる。

ルールアプリケーションがWebアプリケーションの場合、Webアプリケーション配置記述子には業務データベースに対するリソース参照名が指定され、その名称を外部データベースのデータソース名で指定する必要がある。上記の例外2)の場合、データソース参照方式を使用することで解決できる。

ルールアプリケーション設定の際、システムウィザードで登録したすべての業務データベースに対する接続方法を指定する必要はない。アプリケーションが呼び出すルールがどの業務データベースにアクセスするのか把握している場合は、該当データベースにのみ接続方法を指定すればよい。そうでない場合は、すべての業務データベースに対する接続方法を指定することを推奨する。(ルール実行中に接続方法が指定されていない業務データベースにアクセスした場合、接続方法が指定されていないというエラーが発生するため。)

## ルールアプリケーションの初期化

アプリケーションがルールサービスを使用するためには、ルールアプリケーションの初期化が必要になる。ルールアプリケーションの初期化はルールアプリケーション設定を利用してルールランタイムを構成するプロセスであり、アプリケーションが使用するルールアプリケーション設定名を引数にしてルールアプリケーションの初期化ルーチン(「3.7.1 ルールアプリケーションの初期化ルーチン」)を呼び出す。初期化ルーチンがルールシステム設定ファイルとインストールディレクトリの複数のファイルにアクセスするため、ルールアプリケーションはこのファイルにアクセス可能で、かつ読み取り権限を持っている必要がある。1つのルールアプリケーションの設定によって、複数のアプリケーションを初期化できる。



[図 12 Examples of Rule Application Initializations]

[図 12 Examples of Rule Application Initializations]は、ルールアプリケーションの設定と、それを利用して初期化されたアプリケーションの例である。図の左は、InnoRulesがどこにインストールされどのように設定されているかを表している。2つのインストールノードがあり、これらはそれぞれホスト1とホスト2にインストールされている。この2つのインストールノードは設定クラスタを構成しており、同一の設定が保存されている。この設定クラスタには単位ルールシステムAと単位ルールシステムBが構成されており、単位ルールシステムAにはルールアプリケーション設定としてA-1とA-2が存在し、単位ルールシステムBにはルールアプリケーション設定B-1が存在する。

[図 12 Examples of Rule Application Initializations]の右上は、WASクラスタを表している。2つのWebアプリケーションコンテナがWASクラスタを構成している。コンテナ1はホスト1にインストールされているため、同一ホストのファイルシステムのインストールノード1の設定にのみアクセスでき、コンテナ2は同様の事由でインストールノード2の設定にのみアクセスできる。Webアプリケーションはクラスタを構成しているため、すべての設定が同一である必要がある。各Webアプリケーションがアクセスするルールシステム設定ファイルはそれぞれ異なるホストにあるが、設定クラスタによってその内容が同期化されるため、ルールWebアプリケーションは同一設定で初期化される。[図 12 Examples of Rule Application Initializations]は、WASクラスタに配置されている2つのWebアプリケーションが「A-1」というルールアプリケーション設定によって初期化されていることを表している。ここで、コンテナ1のアプリケーションはインストールノード1の設定ファイルに、コンテナ2のアプリケーションはインストールノード2の設定ファイルにアクセスしている。

[図 12 Examples of Rule Application Initializations]の右下は、ホスト2で動作する他のルールアプリケーションを表している。このアプリケーションは一般的なJavaアプリケーションであり、Webアプリケーションではない。左側のアプリケーションは、他のWebアプリケーションが使用している「A-1」設定により初期化されている。このように1つのルールアプリケーション設定を、他のアプリケーションが共有して使用できる。右側のアプリケーションは、単位システムBの「B-1」アプリケーション設定を使用している。このように1つのホストでそれぞれ異なる単位システムの、それぞれ異なる設定を使用するルールアプリケーションを実行できる。

### ルールアプリケーションの初期化ルーチン

**ルールアプリケーションの初期化ルーチン(Rule Application Initialization Routine)**とは、ルールアプリケーション設定を読み込み、ルールアプリケーション環境にルールランタイムを構成または、構成されているルールタイムを削除するJavaコードのことをいう。このコードはinnorules-core.jarライブラリに含まれる。このライブラリはINNORULES\_INSTALLATION/libディレクトリに位置する。

初期化ルーチンに対するJava言語のレベルに関する説明は、「InnoRules Application Programming Interface Guide」を参照のこと。本節では、初期化とクリーンアップメソッドに伝達されるパラメータについて説明する。

#### 初期化メソッド

初期化メソッドはルールアプリケーションにルールランタイムを構成するメソッドであり、次の情報をパラメータとして受け取る。

* InnoRulesインストールディレクトリ(必須)
* 単位ルールシステム名(必須)
* ルールアプリケーション設定名(必須)
* ライブラリテンプレート名(選択)

InnoRulesインストールディレクトリは、InnoRulesがインストールされたディレクトリのことである。1つのホストにInnoRulesが複数ノードインストールされている場合は、使用するルールアプリケーション設定のあるインストールノードのディレクトリを指定する。

インストールディレクトリとして、ネットワークファイルシステムのパスを指定することもできる。InnoRulesインストールディレクトリには設定ファイルやライブラリのみでなくライセンス情報もあり、このライセンスには許可されているIPアドレスなどネットワーク関連の制約事項が含まれている場合もある。そのため、インストールノードのネットワークURLやルールアプリケーションのネットワークURLが異なる場合、適切なライセンスでないと正常に実行されないこともある。

単位ルールシステム名は、アプリケーションが属する単位ルールシステムの名称である。選択された単位ルールシステムのルールがサービスされる。

ルールアプリケーション設定名は、アプリケーションに適用するルールアプリケーション設定の名称である。この名称は、選択された単位ルールシステムに存在する設定の名称でなければならない。

ライブラリテンプレートは、ルールランタイムを構成する時に参照するライブラリのリストファイルである。このテンプレートはINNORULES\_INSTALLATION/libにあり、拡張子はpropertiesでJava propertiesファイルの形式である。ライブラリテンプレートの名称は、テンプレートファイル名からpropertiesを除いた残りの名前である。InnoRulesがインストールされる時、基本的に2つのテンプレートファイル、rule-service.propertiesとinnorules-service.propertiesが作成される。

ルールアプリケーションのクラスパスには、innorules-api.jarとinnorules-core.jarのみを登録する。このライブラリには、ルールサービスAPIとルールアプリケーションの初期化ルーチンなどルールアプリケーションの作成において必要なクラスとインタフェースがすべて含まれている。ただし、ルールランタイムが実行されるためにはより多くのクラスとサブパーティライブラリを必要とする。ライブラリテンプレートファイルは、ルールランタイムが使用するライブラリを記述したファイルである。この追加的なクラスは、初期化ルーチンが作成した専用クラスローダによってローディングされる。

rule-service.propertiesは、ルールサービスのための基本ライブラリテンプレートである。このファイルには、次のライブラリが並べられている。

[表 136]

|  |
| --- |
| commons-logging-1.1.2.jar=  commons-codec-1.10.jar=  log4j-1.2.17.jar=  axiom-api-1.2.15.jar=  axiom-impl-1.2.15.jar=  netty-3.10.4.jar=  innoutils.jar=com.innoexpert.utility.Release  innorulesj.jar=com.innoexpert.innorulesj.engine.Release  configuration-wizard.jar=com.innorules.configurator.initializer.ConfigurationConverter |

log4j-1.2.17.jarやinnoutils.jarのように、プロパティキーにはライブラリのパスが指定される。パスとして、INNORULES\_INSTALLATION/libに対する相対パスの指定や絶対パスを指定できる。初期化ルーチンは、テンプレートに指定されたファイルをソースにするクラスローダを作成する。アプリケーションのクラスローダが作成されたクラスローダの上位になる。初期化ルーチンは、作成されたクラスローダを利用してルールランタイムを初期化する。

プロパティの値は、ライブラリが正常にロードされたかどうかを確認するためのテストクラスである。初期化ルーチンは、クラスローダを作成した後、テストクラスが正常に読み込まれるか確認する。正常に読み込まれていない場合は、例外を投げる。また、テストクラスは省略できる。

rule-service.propertiesのライブラリ構成がルールサービスに必要な基本的な構成ではあるが、場合によっては異なる構成を使用することもある。次のような場合が、それに該当する。

* アプリケーションのクラスパスに同一ライブラリが既に存在する場合

アプリケーションのクラスパスにlog4jのようなサードパーティライブラリが登録される場合がある。ライブラリの重複を防止するために、重複するライブラリをテンプレートから削除する場合もある。

* アドオンライブラリ

ルールランタイムにはユーザ定義関数やビルダーサービスモジュールなど、多様なアドオンが追加できる。アドオンによってライブラリを追加する必要がある場合は、テンプレートにライブラリを追加する。

上記のような事由によってライブラリの構成が変更される場合でも、rule-service.propertiesを直接修正するより、新しいテンプレートファイルを作成して変更されたライブラリ構成を記述したほうがよい。初期化メソッドにライブラリテンプレート名を指定し、ルールアプリケーションがどのようなライブラリ構成を使用するかを指定できる。指定していない場合は、rule-service.propertiesが使用される。

#### クリーンアップメソッド

ルールアプリケーションは、クリーンアップメソッドを呼び出してルールランタイムが占有しているリソースを解放できる。クリーンアップ後は、それ以上ルールサービスを呼び出せない。クリーンアップ後も、再び初期化メソッドを呼び出してルールランタイムを構成することもできる。ただし、アプリケーションが始まる時に初期化し、アプリケーションが終了する前にクリーンアップすることを推奨する。

#### 使用可能なアプリケーションの形態

初期化ルーチンは、すべての形態のJavaアプリケーションで使用できる。初期化を呼び出してクリーンアップを呼び出す時点と呼び出すコードは、アプリケーションに依存する。

public static void main( String[ ] args )から作成するアプリケーションの場合、mainメソッドの開始時に初期化し、終了する直前にクリーンアップを推奨する。

Webアプリケーションの場合、サーブレットコンテキストリスナー(servlet context listener)が使用できる。従来のWebアプリケーションに登録されているリスナーを利用する場合は、contextInitializedメソッドに初期化ルーチンを呼び出すコードを、contextDestroyedメソッドにクリーンアップルーチンを呼び出すコードを挿入する。ルールアプリケーションの初期化の用途で新たにリスナーを作成する場合は、次のプロセスのとおりにする。

* サーブレットコンテキストリスナークラスを作成し、contextInitializedメソッドで初期化ルーチンを、contextDestroyedメソッドでクリーンアップルーチンを呼び出すように作成する。
* 作成されたクラスをWebアプリケーションのクラスパスに位置づける。
* Webアプリケーション配置記述子(Web Application Deployment Descriptor)にリスナーを登録する。

再配布などの方法で変更内容を適用する。

アプリケーションの形態は非常に多様であるため、ここですべてのケースを言及することはできない。また、同形態のアプリケーションだとしても、初期化ルーチンを呼び出す方法が1つのみではない。そのため、初期化ルーチンの呼び出し位置と呼び出しコードの位置はアプリケーションの形態と目的に合わせて開発者が適切に選択する。

### ルールアプリケーションの初期化リスナー

ルールアプリケーションがWebアプリケーションである場合、ビルトイン初期化リスナーを利用して初期化ルーチンを呼び出すことができる。初期化リスナークラスの名前は、次のとおりである。

com.innorules.rrt.RuleAppInitializerListener

このリスナーはjavax.servlet.ServletContextListenerを実装しており、contextInitializedメソッドでは初期化ルーチンを、contextDestroyedメソッドではクリーンアップルーチンを呼び出すようになっている。Webアプリケーションが開始および終了する時にリスナーが呼び出されるようにweb.xmlに設定することで、ルールアプリケーションが初期化できる。このリスナーはinnorules-core.jarに含まれている。

初期化に必要な情報は、コンテキストパラメータに指定する。パラメータは、次のとおりである。

* innorules.home

(必須)InnoRulesインストールディレクトリ

* innorules.system

(必須)単位ルールシステム

* innorules.rule-application-config

(必須)ルールアプリケーション設定名

* innorules.library-template

(選択)ライブラリテンプレート名 指定していない場合、「rule-service」が設定される。

Webアプリケーションのクラスパスには、innorules-api.jarとinnorules-core.jarを追加する。このライブラリは、INNORULES\_INSTALLATION/libにある。このファイルをコピーしてWebアプリケーションのクラスパスにコピーしてもよいが、極力このディレクトリのライブラリを直接参照したほうがアップグレードやパッチのような保守作業が容易となる。このライブラリをコピーして使用するしかない場合は、保守作業の対象から漏れないよう適切な管理方法を講じる必要がある。

次は、web.xmlにルールアプリケーションの初期化リスナーを設定した例である。

[表 137]

|  |
| --- |
| 1 <?xml version="1.0" encoding="ISO-8859-1"?>  2 <!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN" "http://java.sun.com/dtd/web-app\_2\_3.dtd">  3 <web-app>  4 <display-name>Sample Rule Web Application</display-name>  5  ...  6 <context-param>  7 <param-name>innorules.home</param-name>  8 <param-value>**/opt/innorules-home**</param-value>  9 </context-param>  10 <context-param>  11 <param-name>innorules.system</param-name>  12 <param-value>**PRODUCTION**</param-value>  13 </context-param>  14 <context-param>  15 <param-name>innorules.rule-application-config</param-name>  16 <param-value>**online**</param-value>  17 </context-param>  18 <context-param>  19 <param-name>innorules.library-template</param-name>  20 <param-value>**rule-service**</param-value>  21 </context-param>  ...  22 <listener>  23<listener-class>com.innorules.rrt.RuleAppInitializerListener</listener-class>  24 </listener>  ...  25 </web-app> |

上記の例にて、太字で表記されている部分はユーザの環境に合わせて変更する必要のある部分である。[6~21行目]は、コンテキストパラメータを設定する部分である。[6~9行目]では、InnoRulesインストールディレクトリを/opt/innorules-homeに設定している。[10~13行目]では、このアプリケーションが属する単位ルールシステムのIDを指定している。例では、PRODUCTION単位ルールシステムのサービスを利用するように設定している。/opt/innorules-homeのインストールノードには、PRODUCTION単位ルールシステムを設定する必要がある。[14~17行目]では、このアプリケーションがonlineというルールアプリケーションの設定を使用するように設定している。PRODUCTION単位ルールシステムには、onlineというルールアプリケーションの設定が必要である。[18~21行目]では、ライブラリテンプレートとしてrule-serviceを指定している。このアプリケーションは、rule-service.propertiesファイルに指定されたJavaライブラリを利用して初期化する。

アプリケーションに他のコンテキストリスナーがあり、これらのルールアプリケーションの初期化リスナーの実行順序を調整する場合、[22~24行目]の位置を変更して順序を調整する。Oracle社が公開しているサーブレット仕様3.0(Servlet Specification 3.0)以上では、web.xmlに記述された順序どおりにコンテキストリスナーが呼び出される。

次の表は、ルールアプリケーションの初期化リスナーであるRuleAppInitializerListenerクラスを簡単に表している。ルールアプリケーションの初期化ルーチンは、contextInitializedメソッドで呼び出すと説明したが、設定または運用環境の問題によって初期化に失敗した場合、リスナーはonErrorメソッドを呼び出す。

[表 138]

|  |
| --- |
| package com.innorules.rrt;  ...  public class RuleAppInitializerListener implements ServletContextListener  {  public void contextInitialized( ServletContextEvent event );  public void contextDestroyed( ServletContextEvent event );  **protected void onError( Exception e );**  } |

onErrorメソッドは、基本的に標準エラー出力に例外情報を出力する。初期化エラーが発生した時に他の処理をする場合は、RuleAppInitializerListnerを継承して新しいリスナーを作成し、onErrorメソッドをオーバーライドして処理できる。web.xmlには、新たに作成されたリスナークラスを登録する。

### ルールアプリケーション初期化クラス

ルールアプリケーションがWebアプリケーション初期化リスナーを使用できない場合、基本提供されるルールアプリケーション初期化クラスを使用してルールアプリケーションを初期化できる。初期化クラスの名前は次のとおりである。

com.innorules.rrt.Initializer

次の表はこのクラスを使用するルールアプリケーションを初期化してクリーンアップまで呼び出す例である。

[表 139]

|  |
| --- |
| 1 public static void main( String[] args )  2 {  3 Initializer init = null;  4 try  5 {  6 Properties props = new Properties();  7 props.put( "innorules.home", **"/opt/innorules-home**" );  8 props.put( "innorules.system", "**DEV**" );  9 props.put( "innorules.rule-application-config", "**batch**" );  10 props.put( "innorules.library-template", "**rule-service**" );  11 init = new Initializer( props );  12 init.initialize( null );  13  14 // rule call process  15 }  16 catch( Exception e )  17 {  18 e.printStackTrace();  19 }  20 finally  21 {  22 if( init != null && init.hasInitialized() )  23 init.cleanup();  24 }  25 } |

上記の例の[11行目]のInitializerクラスのコンストラクタでjava.util.Propertiesクラスをパラメータとして使用している。[7~10行目]ではルールランタイムシステムを初期化するために必要なパラメータをProperties クラスに設定する。この時、使用されるパラメータのキー値は「3.7.2 ルールアプリケーションの初期化リスナー」で説明した4つのパラメータと同一な値を使用する。

次の表はルールアプリケーション初期化クラスであるInitializerクラスを簡単に示している。

[表 140]

|  |
| --- |
| package com.innorules.rrt;  public class Initializer  {  public Initializer( Properties props ) throws Exception;  public void initialize( Object appctx ) throws Exception;  public void cleanup();  public boolean hasInitialized();  } |

Initializerクラスはルールアプリケーション内に1つのオブジェクトのみ存在する必要がある。ルールアプリケーション開発者はアプリケーションが始まる時点で初期化し、アプリケーションが終了する時点でクリーンアップを呼び出す必要がある。InnoRulesではルールアプリケーション開発者が上記のコードを簡単に使用できるようにInitializerHelperクラスを提供している。

次はInitializerHelperクラスを使用してルールアプリケーションを初期化し、クリーンアップを呼び出す例である。

[表 141]

|  |
| --- |
| 1 public static void main( String[] args )  2 {  3 try  4 {  5 Properties props = new Properties();  6 props.put( "innorules.home", "/opt/innorules-home" );  7 props.put( "innorules.system", "DEV" );  8 props.put( "innorules.rule-application-config", "batch" );  9 props.put( "innorules.library-template", "rule-service" );  10 InitializerHelper.initialize( props );  11  12 // rule call process  13 }  14 catch( Exception e )  15 {  16 e.printStackTrace();  17 }  18  19 InitializerHelper.cleanup();  20 } |

上記の例はInitializerクラスと似ているがInitializerHelperクラスではInitializerクラスをSingleton方式で管理している。開発者はInitializerHelperクラスを使用することで、ルールアプリケーションの初期化重複を防いでサービスを提供できる。

## InnoRulesサービス

本節では、InnoRulesの様々なサービスと、そのサービスを提供するInnoRulesサーバについて説明する。

### InnoRulesサーバ

InnoRulesサーバは、ルールシステムに関連する様々なサービスを提供するサーバである。InnoRulesサーバは、Apache TomcatとInnoRulesWebアプリケーションで構成されている。サーバは製品をインストールする際に自動でインストールされ、インストール段階でHTTPサービスポートと制御ポートを設定する。サーバはINNORULES\_INSTALLATION/innorules-serverにインストールされる。このディレクトリには、次のような下位ディレクトリがある。

* apache-tomcat-X

Apache Tomcatのバイナリがあるディレクトリ。

* default

InnoRulesWebアプリケーションが含まれたApache Tomcat既定のインスタンス。このディレクトリのwebapps/innorulesにInnoRulesWebアプリケーションがインストールされている。

* logs

既定のログディレクトリ。製品をインストールする際、ログディレクトリを他の場所に指定しなかった場合はこのディレクトリ配下のsystemディレクトリに、指定した場合はそのディレクトリ配下のsystemディレクトリにサーバのログが記録される。

サーバのインストール自体が、そのサーバが実行されるということを意味するわけではない。サーバは別のスクリプトを利用して実行する必要があり、そのインストールノードにサーバを実行する必要がない場合は実行しなくてもよい。サーバはINNORULES\_INSTALLATION/scriptディレクトリのstart-server.shとstop-server.sh(またはstart-server.

batとstop-server.bat)スクリプトを利用して開始または終了させる。

InnoRulesサーバでは、次のサービスが提供される。

* 管理サービス
* ルールビルダーサービス
* TCPコネクタサービス
* ルールWebサービス

サービスウィザード(Service Wizard)を利用して、上記のサービスを個別に有効または無効にできる。サーバは有効化されたサービスのみを提供する。

システムには多数のInnoRulesインストールノードを設定でき、そのためサーバが複数存在することもある。製品がインストールされる時には、サーバにはUUIDという固有名称が付与される。この名称を利用して各サーバが識別できる。

### 管理サービス

管理サービス(Management Service)は、設定クラスタに属するルールアプリケーションのルールランタイムの状態をモニタリングして管理するサービスである。このサービスは1つのIPアドレスとポート\*5を占有して接続し、接続されたランタイムを管理する。

管理サービスが使用するIPアドレスとポートは、管理サービスを有効にする時に設定する。管理サービスを有効にするホストにネットワークアダプタが複数ある場合は、そのアドレスのうちの1つを選択する。そのアドレスは、設定クラスタに属するすべてのルールランタイムにアクセスできるアドレスである必要がある。

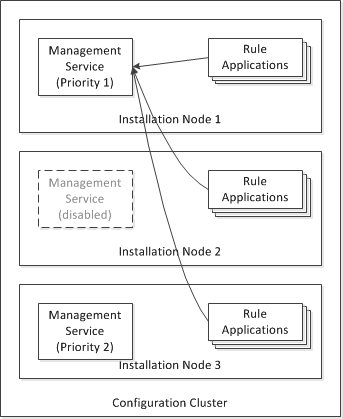
1つのインストールノードには管理サービスを1つのみ有効にできるが、設定クラスタには多数のインストールノードがあり得るため、有効化された管理サービスが2つ以上の場合もある。この管理サービスには優先順位が付与される。ルールランタイムは使用できる最も優先順位の高い管理サービスに接続する。

管理サービスの有効化および無効化、優先順位の変更にはサービスウィザードを使用する。ただし、サービスウィザードはウィザードが実行されたインストールノードの管理サービスに限り、有効化・無効化・優先順位を変更できる。この作業を行う方法については、サービスウィザードを参照のこと。

ルールアプリケーションは、初期化段階に最も優先順位の高い管理サービスへの接続を試みる。接続が失敗した場合、次の順位の管理サービスに接続を試みる。この過程で、どの管理サービスへも接続できない場合においても、初期化は失敗することなく、管理サービスに接続していない状態で初期化が行われる。最も順位の高い管理サービスに接続できなかった場合、周期的に現在接続中の管理サービスより高い順位の管理サービスへの接続を試みる。接続できた場合、現在の管理サービスとの接続を切断し(接続されている状態のみ)、新たに接続されたサーバと通信する。

ルールアプリケーションの動作中に管理サービスとの接続が切断された場合でも(サーバやネットワークなどの障害、またはサーバの停止などにより)、周期的に最も高い順位のサーバから低い順位のサーバへ再接続を試みる。

次の図は、設定クラスタに管理サービスが有効化された状態を表している。設定クラスタは、インストールノード1、インストールノード2、インストールノード3の3つのノードで構成されている。管理サービスはノード1と3のみが有効になっており、優先順位はノード1、ノード3の順になる。次の図のように、ルールアプリケーションは最も優先順位の高いノード1の管理サービスに接続する。ノード1の管理サービスが終了したら、ルールアプリケーションはノード3の管理サービスに接続し、周期的にノード1のサーバに接続を試みる。



[図 13 An Example of Management Service Configuration]

管理サービスに接続されたルールアプリケーションの状態は、管理Webページに接続して確認できる。管理Webページは、今後提供される予定である。

1. InnoRulesサーバのHTTPサーバのバインディングアドレスおよびポートとは別である。

### ルールビルダーサービス

ルールビルダーサービス(Rule Builder Service)は、ルール要素を作成し、管理するサービスである。ルール要素(Rule Element)とは、ルール、項目、ルールWebサービスなどルールシステムで管理するデータ要素のことをいう。ルールビルダーサービスの主な機能は、次のとおりである。

* ルール要素の作成、編集および削除
* ルール要素の移管
* 作成されたルールのテスト

ビルダーサービスを呼び出す代表的なアプリケーションとして、ルールビルダーがある。ルールビルダーは、ルールの開発と管理ためのクライアントツールである。ルールビルダーに関する詳細は、「InnoRules User’s Guide of Rule Builder」を参照のこと。また、ルールビルダーサービスを呼び出せるJava APIであるルールビルダーサービスAPI(Rule Builder Service API)が提供される。Javaアプリケーションは、ルールビルダーサービスAPIを利用してサービスを呼び出すことができる。このAPIについては、「InnoRules Application Programming Interface Guide」を参照のこと。

管理サービスと同様に、ルールビルダーサービスもサービスウィザードを利用してインストールノード別に有効にできる。サービスの有効化には、サービスウィザードを利用する。インストールノードにビルダーサービスが有効化されると、そのノードのInnoRulesサーバに**ルールビルダーサーバ(Rule Builder Server)**が有効化され、ルールビルダーサービスが提供される。サービスが有効化された場合、次のURLにアクセスしてサービスを呼び出すことができる。このURLを、ルールビルダーサービスURLという。

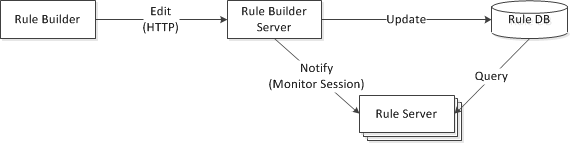
http://IP-ADDRESS:PORT/innorules/builder

ここで、IP-ADDRESSとPORTはそれぞれInnoRulesサーバのアドレスとHTTPサービスポートである。

ルールビルダーサービスを呼び出すためには、ルールビルダーサービスURLを指定する必要がある。ルールビルダーを利用する際は、ログインウィンドウのサーバ接続情報にURLを設定する。ルールビルダーサービスAPIを利用する時は、ログインメソッドにそのURLを伝達する。サービスURLには専用クライアントツールやAPIを利用してアクセスする必要があり、一般的なWebブラウジングでアクセスしてはならない。

#### モニターサーバとモニターセッション

ルールデータは、ルールビルダーのサーバによって更新される。



[図 14 Monitor Session]

ルール開発者がルールビルダーを利用して編集作業を行った場合、HTTPを通じてルールビルダーサーバにサービス要求が送信される。編集作業にはルールの作成、修正、削除のみでなく、ルール要素を変更するすべての作業が含まれる。ルールビルダーサーバは、サービス要求を分析して関連するルールデータベースのテーブルデータを適切に更新する。

ただし、ルールデータベースにアクセスするのはルールビルダーサーバのみではない。ルールサーバもルールデータを照会する。ルールサーバはTCPコネクタサーバのようにリモートルールサービスを提供するサーバである場合もあり、ルールアプリケーションがを含むローカルルールサーバである場合もある。ルールサーバは必要な時点でその都度ルールデータの照会も行うが、場合によってはデータ変更時にリアルタイムで照会したほうがよい場合もある。ルールデータの変更リストをリアルタイムで受け渡すためにルールビルダーサーバとルールサーバの間にTCP/IPセッションを確立しており、これを**モニターセッション(Monitor Session)**という。

ルールサーバはルールの実行性能を向上させるために、ルールデータの一部分をメモリにキャッシュする。そのため、ルールデータが更新された場合、メモリのキャッシュも同様に更新される必要がある。そうでない場合、ルールサーバは変更されたデータでサービスできなくなる。

ルールサーバは、起動時にルールビルダーサーバへモニター接続を試みる。ルールビルダーサーバはモニター接続を受け入れ、接続されたセッションを管理する機能を持っている。この機能を**モニターサーバ(Monitor Server)**という。モニターサーバは、この接続専用でTCP/IPアドレスとポートを開いて待機する。このアドレスとポートは、ルールビルダーサービスを有効にする時に指定される。

ルールビルダーサーバはルールデータを更新してコミットした後、接続されているすべてのモニターセッションに変更されたリストを通知する。通知されたルールサーバはルールデータベースから変更内訳を照会し、必要に応じてキャッシュを更新する。ルールサーバは変更を通知した後、ルールサーバがその内訳をすべて更新するまでルールデータベースを変更しない。

#### ビルダーサーバクラスタリング

1つの設定クラスタに複数のルールビルダーサービスを有効化できる。有効化されたビルダーサーバの間には優先順位が付与される。優先順位はサービスウィザードを利用して変更できる。

クラスタ内で複数のビルダーサーバが有効化されているとしても、利用可能なビルダーサーバのうち最も優先順位の高いビルダーサーバのみがサービスを提供する。「利用可能な」というのは、サービスを提供できる状態であることを意味する。最も優先順位が高いとしても、障害またはサーバの停止などによってサービスを提供できないビルダーサーバは順位から除外される。設定クラスタの中の「利用可能な最優先順位のビルダーサーバ」を、**マスタービルダーサーバ(Master Builder Server)**という。それに比べ、サービスを提供できる状態ではあるがマスタービルダーサーバより優先順位の低いビルダーサーバを、**スレーブビルダーサーバ(Slave Builder Server)**という。

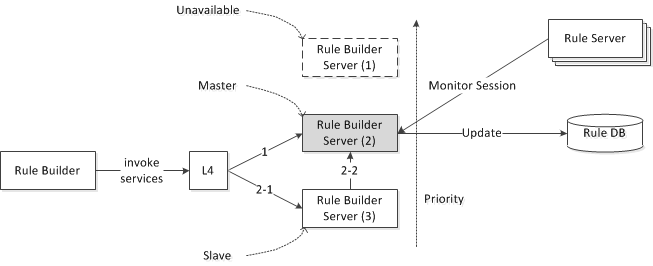
マスタービルダーサーバに入る、つまりマスタービルダーサーバのサービスURLに格納されるビルダーサービス要求は、このビルダーサーバが処理する。ただし、スレーブビルダーサーバは自身に格納されるサービス要求をマスタービルダーサーバのサービスURLに再伝送(redirect)する。マスタービルダーサーバでのみモニターサーバが有効化される。クラスタ内のすべてのルールサーバは、マスタービルダーサーバのモニターサーバにモニターセッションを接続する。

多数のビルダーサーバがサービス要求を再伝送してマスタービルダーサーバが処理する構成を、**ビルダーサーバクラスタリング(Builder Server Clustering)**という。ビルダーサーバクラスタリングの目的は、1つのビルダーサーバのみがルールデータベースを変更できるようにすることで、ルールサーバがルールデータの変更に対して整合性を維持できるようにすることである。

マスタービルダーサーバの停止や障害が発生してサービスを提供できない状態になった場合、その次の順位のスレーブビルダーサーバがマスターに切り替わる。新しいマスタービルダーサーバは、モニターサーバを有効にする。既存のマスタービルダーサーバの障害によってモニターセッションが切断された場合、ルールサーバは新しいマスタービルダーサーバにモニターセッションを接続する。

マスタービルダーサーバより優先順位の高いビルダーサーバがサービスを再開すると、再開されたサーバがマスタービルダーサーバになり、既存のマスタービルダーサーバはスレーブに切替わりモニターサーバが中止される。モニターサーバは新しいマスタービルダーサーバで有効となり、ルールサーバはそのサーバにモニターセッションを接続する。

次の図は、ルールビルダーサーバクラスタリングの例を表している。



[図 15 An Example of Rule Builder Server Clustering]

この設定クラスタには、3つのルールビルダーサーバが有効となっている。優先順位は、Rule Builder Server1、Rule Builder Server2、Rule Builder Server3の順である。Rule Builder Server1は優先順位が最も高いが、停止状態である。そのため、現在の状態でマスタールールビルダーサーバはRule Builder Server2である。Rule Builder Server2ではモニターサーバが実行され、すべてのルールサーバはRule Builder Server2とモニターセッションを接続する。

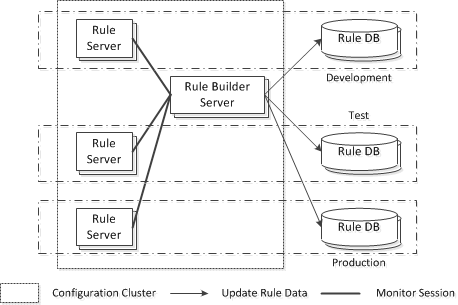
ルールビルダーは特定のビルダーサーバのサービスURLに直接サービス要求する場合もあるが、通常このような多重化構成ではL4スイッチがビルダーサーバの直前に位置し、ルールビルダーはL4の仮想URLにサービスを要求する。L4はポリシーに従って、Rule Builder Server2またはRule Builder Server3に要求を再転送する。

Rule Builder Server2に要求があった場合(矢印1)、その要求はRule Builder Server2で処理される。Rule Builder Server2はルールデータベースをアップデートして、Rule Serverにその内訳を通知する。

Rule Builder Server3に要求があった場合(矢印2-1)、スレーブであるRule Builder Server3は要求を再びRule Builder Server2に再転送する(矢印2-2)。Rule Builder Server2は要求を処理した後、Rule Builder Server3に結果を返却し、その結果は再びRule Builderに返却される。

#### 多段階ルールシステムと移管専用モード

1つの設定クラスタには、1つのマスタービルダーサーバのみが存在する。設定クラスタに属するすべてのルールサーバは、マスタービルダーサーバにモニターセッションを確立する。次の図は、そのような事例を表している。



[図 16 Single Configuration Cluster]

[図 16 Single Configuration Cluster]は、3つのアプリケーション段階を持つシステムを表している。アプリケーションの段階は、開発(Development)、テスト(Test)、運用(Production)になっている。ルールシステムは、次のようにインストールおよび設定されている。

* 各段階は、複数の物理的なサーバで構成されることもある。開発段階には複数の開発サーバがあり、これはテスト・運用段階においても同様である。このサーバにはそれぞれInnoRulesがインストールされている。すべてのインストールノードは、1つの設定クラスタを構成している。
* アプリケーションの段階に対応するように、単位ルールシステムの開発系、テスト系、運用系が設定されている。ルールデータベースは、単位ルールシステムに対応する段階のサーバ上にある。例えば、開発系のルールデータベースは開発サーバ上に存在する。
* 各単位ルールシステムには、ルールアプリケーション設定が登録されている。ルールアプリケーションは、この設定によって各段階の物理的なサーバで実行される。
* 1つ以上のインストールノードにルールビルダーサーバが有効となっている。ルールビルダーサーバは3つの単位ルールシステムのルールデータベースすべてに接続し、アップデートする。
* ルールアプリケーションにインベッドされたルールサーバまたはTCPコネクタサーバのようなリモートルールサーバは、マスタービルダーサーバにモニター接続をしている。

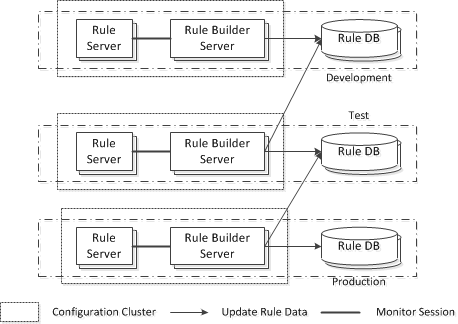
この構成は単にインストールして設定できる構成だが、システム運用ポリシーによって制約がある場合がある。

第1に、モニターセッションに関連する制約である。ルールサーバは、モニターセッションのためにマスタービルダーサーバにTCP/IPコネクションを生成する。マスタービルダーサーバが開発サーバで実行されるのであれば、テストと運用のルールサーバは開発サーバにTCP/IPセッションを確立する必要がある。一方、マスタービルダーサーバが運用サーバで実行される場合、開発とテストのルールサーバは運用サーバにTCP/IPコネクションを生成する。一般的に段階を超えるTCP/IPコネクションには制約が多く、下位段階から上位段階への接続の場合はさらに制約が多くなる(例：テスト-運用)。

第2に、データベース接続に関連する制約である。ビルダーサーバは、すべての段階のルールデータベースに接続してアップデートする必要がある。例えば、運用サーバで実行されるマスタービルダーサーバは開発とテストのルールデータベースに接続し、アップデートする。モニター接続と同様に、段階を超えるデータベース接続への制約が起こる。開発サーバから運用データベースへの接続の場合はさらに制約が多くなり、それによって開発サーバにビルダーサーバを運用することは難しくなる。

第3に、設定同期化に関連する制約である。ウィザードを利用して設定を変更する時、ウィザードはSSHやFTPを利用して他のインストールノードの設定を同様に変更して同期化する。ただし、段階を超えるSSHまたはFTP接続に制約があると、設定同期化が不可能な場合がある。

上のエラーを解決するために、各段階に設定クラスタを分離することも考えられる。この場合の設定は、各段階別にのみ同期化される。次の図は、その構成を示す。



[図 17 Separated Configuration Cluster]

[図 16 Single Configuration Cluster]との差異は、段階別に設定クラスタを分離したという点である。この構成の特徴は、次のとおりである。

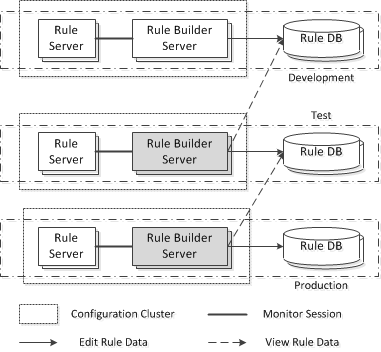
* 各段階の設定クラスタは、その段階のサーバにインストールされたノードでのみ構成されている。
* 設定クラスタには1つまたは連続する単位ルールシステム2つが登録されている。
* 単位ルールシステムは移管方向を基準に、開発系、テスト系、運用系の順である。
* 開発設定クラスタには、単位ルールシステムとして開発系が登録されている。
* テストおよび運用設定クラスタには、単位ルールシステムとしてそれぞれ開発系とテスト系、テスト系と運用系が登録されている。
* 設定クラスタには、1つの単位ルールシステムが共通的に登録されている。
* 開発クラスタとテストクラスタには、開発系が共通的に登録されている。
* テストクラスタと運用クラスタには、テスト系が共通的に登録されている。
* それぞれの設定クラスタには、ルールビルダーサーバが別途実行される。
* 単位ルールシステムが1つのみ登録されたクラスタのルールビルダーサーバは、そのシステムのルールを編集する(ex. 開発クラスタ)。
* 単位ルールシステムが2つ以上登録されたクラスタのルールビルダーサーバは前の単位ルールシステムのルールを編集し、後ろの単位ルールシステムに移管を行う(テストおよび運用クラスタ)。
* ビルダーサーバの名前はアプリケーション段階の名称に従う。開発段階のビルダーサーバは開発ビルダーサーバ、テスト段階と運用段階のビルダーサーバはそれぞれテストビルダーサーバと運用ビルダーサーバという。
* すべてのルールサーバは、自分と同一段階にあるルールビルダーサーバにモニター接続をする。開発段階のルールサーバは開発ビルダーサーバ、テスト段階のルールサーバはテストビルダーサーバ、運用段階のルールサーバは運用ビルダーサーバにそれぞれモニター接続をする。

この構成では、ルールサーバが段階を超えるモニター接続を結ぶ問題は発生しない。また、設定クラスタリングは段階内のインストールノードの間のみで行われるため問題はない。ルールビルダーサーバが移管のために連続する2つの段階のデータベースにアクセスする必要があるということが制約事項ではあるが、自分より下位段階のデータベースにアクセスするため制約も少なくなる。

ただし、この構成でもモニターに関連する問題は生じることもある。開発ビルダーサーバとテストビルダーサーバが開発系のルールを修正できる。開発段階のルールサーバは開発ビルダーサーバにモニター接続を結んでいるため、テストビルダーサーバを利用してルールを保存すると、このサーバに変更内容が通知されない。モニターに関する問題が発生しないようにするためには、上位段階で運用されるルールビルダーサーバが下位段階のルールを変更できないようにする必要がある。

ビルダーサーバが連続する2つ以上の単位ルールシステムを管理する際に、最初段階のシステムのルールを修正できないように設定することを**移管専用モード(Transition-Only Mode)**という。移管専用モードに設定されているビルダーサーバは、そのビルダーサーバが管理する単位ルールシステムのうち最初段階のルールを照会することはできるが、編集はできない。反面、移管専用モードに設定されていないルールビルダーサーバは、最初段階に対して照会および編集が可能である。移管専用モードの設定とは関係なく、2番目以降の単位ルールシステムに対して編集できないが、移管はできる。

移管専用モードは、サービスウィザードを利用して設定する。移管専用モードを利用して[図 17 Separated Configuration Cluster]の問題点を解決したものが、次の図である。



[図 18 Multi-Step Rule System]

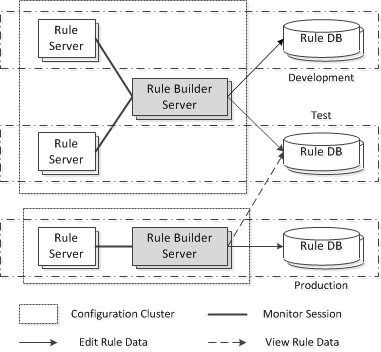
[図 18 Multi-Step Rule System]の中で、グレーで表示されたテストビルダーサーバと運用ビルダーサーバが移管専用モードに設定されている。このビルダーサーバは点線矢印で表示されているように、前の段階に位置するコピー元システムのみが照会できる。すべてのビルダーサーバは編集や移管時は自分が属する段階のルールデータのみを修正し、そのルールデータを照会するすべてのルールサーバはこのビルダーサーバにモニター接続をするため、モニターに関する問題が発生しない。

ビルダーサーバが前の段階のルールデータを変更しないというのが、前の段階のルールデータベースの内容を全く変更しないという意味ではない。ルールデータベースにはルール要素のみでなく、履歴と移管関連フラグも管理される。ルールビルダーサーバは移管の際にシステムのルールデータを変更するのみでなく、コピー元システムの履歴データおよびフラグも変更する。そのため、2つの設定クラスタに共有されている単位ルールシステムのデータが2つ以上のルールデータベースに保存される場合は、ルールデータベースは2つのクラスタに同様に登録される必要がある。[図 18 Multi-Step Rule System]の例の場合、開発クラスタとテストクラスタに単位ルールシステム開発系が登録されている。開発クラスタに開発系のルールデータベースとしてRULEDB1とRULEDB2が登録されている場合、テスト系に登録された開発系にもRULEDB1とRULEDB2がすべて登録されている必要がある。段階別にクラスタが分離された構成において、ルールシステム管理者はこの点に特に留意する必要がある。

多段階ルールシステムを構成する際は、次の規則を遵守する。

* 設定クラスタは段階単位で構成する。設定同期化のためのネットワーク接続が可能な場合は、2つの段階以上を1つの設定クラスタとして構成できる。(規則1)
* 隣接した設定クラスタは1つの単位ルールシステムを共有する。(規則2)
* 各クラスタにおいて、少なくとも1つ以上のインストールノードでビルダーサーバを有効にする。(規則3)
* ビルダーサーバの移管専用モードは最初段階のクラスタでは無効化、それ以外の段階では有効化される。(規則4)

ネットワークポリシーが開発段階とテスト段階の設定同期化を許可する場合、上記の規定を適用して次の図のような構成も可能である。



[図 19 Another Multi-Step Rule System]

開発サーバとテストサーバのインストールノードが1つの設定クラスタを構成しており、運用サーバのインストールノードが1つの設定クラスタになっている(規則1)。単位ルールシステムのテストは、開発-テスト設定クラスタと運用設定クラスタにすべて登録されている(規則2)。開発-テストクラスタと運用クラスタにはルールビルダーサーバが有効化されている(規則3)。開発-テストクラスタのビルダーサーバは開発サーバまたはテストサーバでの有効化が可能で、どちらが共有に設定されても問題はない。開発-テストクラスタは最初段階の設定クラスタであるため、ビルダーサーバの移管専用モードが無効化されており、運用クラスタには移管専用モードが有効化されている(規則4)。

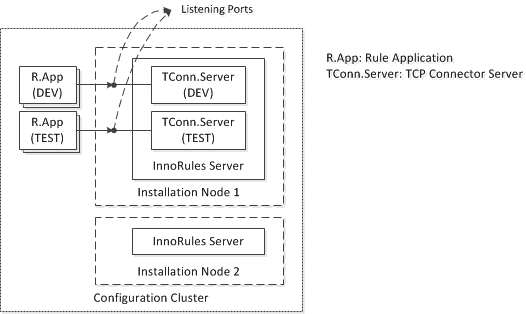
### TCPコネクタサービス

#### サービス有効化

**TCPコネクタサービス(TCP Connector Service)**は、TCP/IPとInnoRules自体のプロトコルを使用するリモートルールサービスである。TCPコネクタサービスは、サービスウィザードを利用してインストールノード別に有効化できる。TCPコネクタサービスを有効にすると、InnoRulesサーバでTCPコネクタサーバが実行されてサービスを提供する。

TCPコネクタサーバは、独立したアドレスとポートを使用する。TCPコネクタサーバを有効にする際には、アドレスとポートが指定できる。このアドレスは、サービスを呼び出すルールアプリケーションがアクセスできるネットワークのアドレスである。

TCPコネクタサービスは、単位ルールシステムに依存的である。有効にするためには、このサービスがどのような単位システムのルールをサービスするのかを先に選択する必要がある。複数の単位ルールシステムが登録されている場合、単位ルールシステム別にサービスを有効にし、アドレスとポートもそれぞれ付与する。



[図 20 TCP Connector Service]

[図 20 TCP Connector Service]は、2つのインストールノードで構成されている設定クラスタを表している。インストールノードの名前は、インストールノード1とインストールノード2である。それぞれのインストールノードではInnoRulesサーバが実行されている。TCPコネクタサーバはインストールノード1でのみ有効になっており、そのためこのノードのInnoRulesサーバでTCPコネクタサーバが実行されている。

TCPコネクタサーバは、単位ルールシステムDEVとTESTに対してそれぞれ有効となっている。そのため、2つのサーバがそれぞれアドレスとポートを占有して実行されている。開発段階のアプリケーションはDEVのルールアプリケーション設定を使用するため、「3.6 ルールアプリケーション設定」で記述したように、DEV TCPコネクタサーバに接続してルールサービスを要求するようになる。同様の方式で、テスト段階のアプリケーションはTEST TCPコネクタサーバにルールサービスを要求するようになる。

#### TCPコネクタサーバグループ

様々な業務または目的のルールアプリケーションがTCPコネクタサービスを呼び出す。そのため、業務別または目的別に異なるTCPコネクタサーバを使ってリソースの使用を分離する必要がある。同一目的を持つ1つ以上のTCPコネクタサーバのグループを、**TCPコネクタサーバグループ(TCP Connector Server Group)**という。

サービスウィザードを利用して、TCPコネクタサーバクループとTCPコネクタサーバを登録できる。TCPコネクタサーバを登録する際に、グループを指定する必要があるが、基本的にデフォルトグループが作成されている。登録されたTCPコネクタサーバグループは削除できるが、デフォルトグループは削除できない。

1つのグループに複数のTCPコネクタサーバが登録できる。このサーバはグループ内での順番が付与されている。この順番はウィザードを利用して変更できる。

ルールアプリケーションのサービス要求を処理するサーバを指定するために、ルールアプリケーション設定にTCPコネクタサーバグループを指定する。ルールアプリケーション設定へのサーバグループ設定については、「3.6.7 TCPコネクタサーバグループ選択 - リモートルールサービスにのみ該当」を参照のこと。アプリケーションがサービス要求をするグループ内サーバ、および要求順序を指定するために、次の高可用性方式が指定できる。

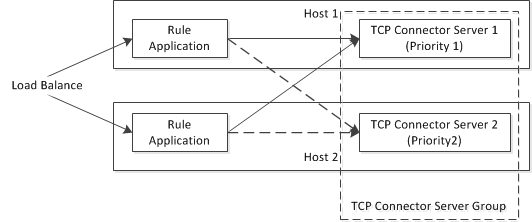
* フェールオーバー(Fail Over)
* 負荷分散(Load Balance)

**フェールオーバー(Fail Over)方式**はサーバにグループ内の優先順位を定義し、アプリケーションが使用できる最優先順位のルールサーバにのみルールサービス要求を送るようにする方式である。優先順位は数字によって付与され、数字が小さいサーバの優先順位が高い。アプリケーションは1つのサーバにのみルールサービスを要求し、そのサーバをこのアプリケーションの**アクティブサーバ(Active Server)**という。アクティブサーバの中断や障害が発生して使用できなくなった場合、その次の優先順位のルールサーバがアクティブサーバになる。

**負荷分散(Load Balance)方式**は、アプリケーションがすべてのサーバに均等にサービス要求を送信する方式である。アプリケーションは、サーバのグループ内での順番にサービス要求を送信する。例えば、最初の要求は1番サーバに、次の要求は2番サーバに送信する方式である。最後の順番のサーバに要求を送った後、その次の要求は再び1番サーバに送信される。中断や障害が発生したサーバは、順番から除外される。

冗長化(Replication)または高可用(High Availability)構成のために、フェールオーバーまたは負荷分散方式が使用できる。

多くの二重化構成において、ルールアプリケーションとTCPコネクタサーバはペアで構成される。次の図は、そのような二重化構成の例を表している。アプリケーションとTCPコネクタサーバは、同じホストに構成されている。アプリケーションが使用するルールサーバの優先順位は絶対的ではなく、アプリケーションと対をなすルールサーバが優先され、そのサーバが使用できない場合は他のホストのルールサーバを使用するようになっている。

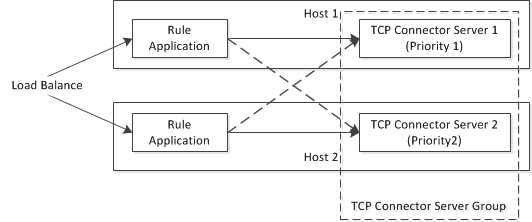


[図 21 An Example of a Replication]

この例では、二重化で動作する2つのホストを表している。各ホストにはルールアプリケーションとTCPコネクタサーバが動作している。TCPコネクタサーバはTCPコネクタサーバグループで構成されており、アプリケーションはこのグループにルールサービスを要求する。外部からの要求はL4スイッチなどを利用して各アプリケーションに分散されるため、2つのアプリケーションのルールサービス呼び出しの頻度は同程度である。そのため、TCPコネクタサーバへのルールサービス要求を再び分散させる必要がないため、負荷分散方式に指定する必要はない。ただし、1つのTCPコネクタサーバに障害が発生した場合でもルールは正常にサービスされる必要があるため、TCPコネクタサーバグループをフェールオーバー方式に設定する。

TCPコネクタサーバには順番が付与されており、その順番によりTCPコネクタサーバの優先順位はTCPコネクタサーバ1、TCPコネクタサーバ2の順となる。そのため、サーバ1が正常に動作している場合は、どんなルールアプリケーションであれサービス要求はサーバ1に送信される。ルールアプリケーションとTCPコネクタサーバの間の実線矢印がそれを表している。サーバ1に障害が発生してサーバ2がアクティブサーバになると、すべてのルールサービス要求はサーバ2に送信される。ルールアプリケーションとTCPコネクタサーバの間の点線矢印がそれを表している。

このように構成される場合、すべてのルールサービスの負荷が一方のホストに集中される問題が発生する。この問題は、ルールアプリケーションがTCPコネクタサーバの優先順位に関係なく、自ホストで実行されるサーバに優先的にサービスを要求することで解決する。次の図が、そのような構成を表している。



[図 22 An Example of a Replication with Local First Option]

[図 22 An Example of a Replication with Local First Option]で、ルールアプリケーションは優先的に自分が実行しているホストのTCPコネクタサーバにサービスを要求する。ホスト1のアプリケーションはサーバ1に、ホスト2のアプリケーションはサーバ2にサービスを要求する。ルールアプリケーションからサーバへの実線矢印がそれを表している。外部からルールアプリケーションへの負荷が分散し、アプリケーションは各自のTCPコネクタサーバにサービスを要求するため、結果的にTCPコネクタサーバへの負荷も均等に分散する。

サーバ1が停止した場合、ホスト1のアプリケーションはサーバ2にサービスを要求する。一方、サーバ2が停止すると、ホスト2のアプリケーションがサーバ1にサービスを要求する。

このように、アプリケーションTCPコネクタサーバグループに指定された優先順位とは関係なく、同一ホストサーバを優先的に使用するようにするオプションを**ローカル優先(Local First)**オプションという。このオプションは、TCPコネクタサーバグループに指定される。このオプションを有効にした場合、指定された高可用方式によってサーバの優先順位が再調整される。

フェールオーバー方式では

* TCPコネクタサーバグループにルールアプリケーションと同一ホストで動作するサーバがあると、そのサーバのうち優先順位が高いサーバが優先される。
* 選択されたTCPコネクタサーバグループにルールアプリケーションと同一ホストで動作するサーバがない場合は、グループに登録された順番に従う。

負荷分散方式では

* TCPコネクタサーバグループにルールアプリケーションと同一ホストで動作するサーバがある場合、そのサーバのうち優先順位が高いサーバを優先して、そのサーバ以降の順番のサーバがその次となる。選択されたサーバがグループで最優先順位でない場合は、グループで最初に選択されたサーバの前の順番までのサーバがその次になる。
* 選択されたTCPコネクタサーバグループにルールアプリケーションと同一ホストで動作するサーバがない場合は、グループに登録された順番に従う。

### ルールWebサービス

ルールWebサービスは、ルールサービスのSOAP/HTTP呼び出しを可能にするサービスである。ルールWebサービスを利用してルールサービスを呼び出すためには次の事項が必要である。

* ルールWebサービスを有効にする

InnoRulesサーバでルールWebサービスを有効にしなければならない。2つ以上のインストールノードでルールWebサービスを有効にすることもできる。ルールWebサービスの有効化については「2.4.6 ルールWebサービス」を参照のこと。

* ルールをWebサービスとして登録

作成されたすべてのルールがWebサービスされることではない。Webサービスで使用するルールはルールビルダーでWebサービスとして登録しなければならない。この過程では項目とルールリターン名をWebサービスの入出力変数にマッピングする過程も含まれる。

#### WebサービスURL

HTTPを利用してSOAPリクエストができるすべてのアプリケーションはルールWebサービスが呼び出すことができる。InnoRulesではルールWebサービスを呼び出すためのAPIを別途提供しない。

SOAP/HTTPリクエストのためには、ルールWebサービスに対するエンドポイントURLを指定する必要がある。URLは次のように構成する。

http://[IP]:[PORT]/innorules/services/ws/[ALIAS]/[SERVICE]

URLを構成する要素の意味は次のとおりである。

* [IP]

InnoRulesサーバのアドレス

* [PORT]

InnoRulesサーバのHTTPサービスポート

* innorules

InnoRulesサービスWebアプリケーションのコンテキストパス

* services/ws

InnoRulesサービス中ルールWebサービスであることを表す部分

* [ALIAS]

どの単位ルールシステムのルールを呼び出すかを指定する。各単位ルールシステムでは識別のためのエイリアスが与えられる場合もある。これをルールWebサービスシステムエイリアスという。エイリアスが与えられた場合、「ALIAS」にエイリアスを指定する。特定単位ルールシステムに対しては基本ルールWebサービスとして登録できる。基本ルールWebサービスにはエイリアスが与えられていないため、この場合、ALIASとその後ろに付く「/」は省略しなければならない。

* [SERVICE]

ルールをWebサービスとして登録する時に付与されるサービス名を指定する。この値により呼び出しルールは指定される。

#### ルールWebサービスシステムエイリアス

1つのInnoRulesサーバが複数の単位ルールシステムのルールをWebサービス化できるため、URLでは単位システムを区分できる識別子が付与される必要がある。URLは外部に提供される特性があるため、単位ルールシステムID・名称を使用する場合にその情報が外部へ流出する危険がある。単位ルールシステムの情報をURLで隠蔽するためにエイリアスを付与するが、これをルールWebサービスシステムエイリアスという。このエイリアスは「3.8.5.1WebサービスURL」で説明したようにルールWebサービスURLを構成する1つの要素である。

場合によっては、InnoRulesサーバは単一の単位ルールシステムに対してルールWebサービスを提供することがある。複数の単位ルールシステムにサービスをしても、その中の1つがマスターになる場合がある。この場合、1つの単位システムに対してエイリアスを付与しないことがある。エイリアスが与えられないシステムのWebサービスURLには「3.8.5.1WebサービスURL」で説明したようにエイリアスを指定しない。

#### 高可用性設定

複数のInnoRulesサーバでルールWebサービスを有効にして、ロードバランスやフェールオーバーでHA設定を行う場合がある。InnoRulesでは高可用のために別の機能をサポートしないため、L4スイッチなど一般的なWebアプリケーションアーキテクチャのHA設定と同一方法を利用する。

設定クラスタ内で利用可能なルールWebサービスとそのURLに対して照会するためには、「2.4.6.1 ルールWebサービス」の[a. ルールWebサービスURLの照会]メニューを使用する。

### ルールRESTサービス

ルールRESTサービスはHTTP基盤のRESTを利用し、ルールサービスの呼び出しを可能にするサービスである。ルールRESTサービスを利用してルールサービスを呼び出すためには次の事項が必要である。

* ルールRESTサービスを有効にする

InnoRulesサーバでルールRESTサービスを有効にしなければならない。2つ以上のインストールノードでルールRESTサービスを有効にすることもできる。ルールRESTサービスの有効化については「2.4.7 ルールRESTサービス」を参照のこと。

有効になった単位ルールシステムに登録されたすべてのルールはルールRESTサービスの対象になる。

#### ルールRESTサービス呼び出し規約

HTTPクライアント系列のすべてのアプリケーションはルールRESTサービスを呼び出すことができる。

ただし、次の規約を守らなければならない。

* HTTP Header

HTTPメッセージのHeaderに次の値を含める。

* content-type：application/json (必須)
* HTTP METHOD：POST (POST以外の方式はサポートしない)
* リクエストデータタイプ：JSON (JSON 規約を守るTextのみサポートする)

#### ルールRESTサービスURL

RESTリクエストのためには、ルールRESTサービスに対するエンドポイントURLを指定する必要がある。URLは次のように構成する。

http://[IP]:[PORT]/innorules/services/rest/[ALIAS]/[RULECODE]/[DATE]

URLを構成する要素の意味は次のとおりである。

* [IP]

InnoRulesサーバのアドレス

* [PORT]

InnoRulesサーバのHTTPサービスポート

* innorules

InnoRulesサービスWeb Applicationのコンテキストパス

* services/rest

InnoRulesサービスの中でrestサービスを示す部分

* [ALIAS]

呼び出す単位システムを指定する。各単位システムには識別のためルールRESTサービスシステムエイリアスが付与される。

* [RULECODE]

サービス対象ルールコード、ルール呼出方式(JSON入力時のcodeType)に合わせて指定する。

* [DATE]

ルール実行基準日。省略が可能で、省略した場合実行した日の日付が実行基準日として使用される。

#### ルールRESTサービスシステムエイリアス

1つのInnoRulesサーバが複数の単位ルールシステムのルールをRESTサービス化できるため、URLでは単位システムを区分できる識別子が付与される必要がある。URLは外部に提供される特性があるため、単位ルールシステムID・名称を使用する場合にその情報が外部へ流出する危険がある。単位ルールシステムの情報をURLで隠蔽するためにエイリアスを付与するが、これをルールRESTサービスシステムエイリアスという。このエイリアスは「3.8.6.1 ルールRESTサービス呼び出し規約」で説明したようにルールRESTサービスURLを構成する1つの要素である。

#### 高可用設定

複数のInnoRulesサーバでルールRESTサービスを有効にして、ロードバランスやフェールオーバーでHA設定を行う場合がある。InnoRulesでは高可用のために別の機能をサポートしないため、L4スイッチなど一般的なWebアプリケーションアーキテクチャのHA設定と同一方法を利用する。

設定クラスタ内で利用可能なルールRESTサービスとそのURLに対して照会するためには、「2.4.7.1 ルールRESTサービス」の[a. ルールRESTサービスのURL参照]メニューを使用する。

# エラーコードと処置方法

ルールシステムに障害\*6が発生すると、ルールシステムはJavaの例外を投げるか、ログにエラー事項を記録する。例外やログには障害の現象、原因を把握できるエラーコードが付与される。このエラーコードは障害を処理する際に使用する。

この章ではエラーコード別に、その原因と対処方法を説明する。本文書に記述されていないエラーコードについては技術サポートが必要である。

1. 障害の種類は多様である。ルールシステム自体の障害である可能性もあり、また、サービスを呼び出すアプリケーションで誤ったパラメータを引き渡した場合もある。障害レベルもリカバリ不可能なものからリカバリ可能で無視できるものまで多様である。障害が影響を及ぼす範囲もルールシステム全体、特定のルールランタイム、または特定サービス呼び出し時のみのように様々である。

## IRE-10XXX

ルールサービスを呼び出す途中に起こるエラーである。エラーは、ログに記録され、ルールサービスを呼び出したアプリケーションにも伝えられる。ルールサービスを呼び出し中に、エラーが発生すると、アプリケーションは適切な例外処理をする必要がある。

エラーの原因を理解するためにはルールサービスAPIまたはルール作成方法に対する理解が必要である。これについては「InnoRules Application Programming Interface Guide」と「InnoRules User's Guide of Rule Builder」を参照すること。

### IRE-10000

* エラーメッセージ

項目が入力されていません。

* 原因

ルール実行に必要な項目がアプリケーションから入力されていない。その項目のID、名称とエイリアスがあるとエイリアスがメッセージに含まれる。次の場合、項目が入力されていないと判定する。

* I/Oオブジェクトの方式でルールを呼び出す場合、RuleReqオブジェクトに項目値が含まれていない。
* I/Oアダプタ方式でルールを呼び出す場合、ItemProviderのgetAsXXXメソッドがnullを返却する。
* 処置方法

ルールサービスを呼び出す際、不足した項目値を入力するようにアプリケーションを修正する。

### IRE-10001

* エラーメッセージ

定義されていないルールです。

* 原因

アプリケーションがルールを呼び出す時、ルールのコードとして有効ではないID、名称、またはエイリアスを伝える。「有効ではない」の意味はそのIDを持つルール(または、その名称のルール・エイリアスの持つルール)が定義されていないことを示す。エラーになるID(または、名称・エイリアス)がメッセージに含まれる。

* 処置方法

アプリケーション開発者はルール開発者から呼び出すルールのコードを確認して、アプリケーションを修正する。

### IRE-10002

* エラーメッセージ

定義されていない項目です。

* 原因

アプリケーションがルールを呼び出す時、項目のコードとして有効ではないID、名称、またはエイリアスを伝える。「有効ではない」の意味はそのIDを持つ項目(または、その名称の項目・エイリアスの項目)が定義されていないことを示す。メッセージに、エラーになる項目(または、名称、エイリアス)のIDが含まれる。

* 処置方法

アプリケーション開発者はルール開発者から呼び出す項目のコードを確認して、アプリケーションを修正する。

### IRE-10003

* エラーメッセージ

入力項目値が定義されたデータ型と異なります。

* 原因

アプリケーションがI/Oオブジェクト方式でルールを呼び出す時、I/Oオブジェクトに設定した値のデータ型がルールシステムに登録されているデータ型と異なる場合に発生する。例えば、ルールシステムに項目が文字型で登録されているがアプリケーションはその項目値で数値を設定した場合である。

* 処置方法

アプリケーション開発者はルール開発者からエラーになる項目のデータ型を確認して、アプリケーションを修正する。

### IRE-10004

* エラーメッセージ

項目値の誤り

* 原因

ルールアプリケーションがI/Oアダプタ方式でルールを呼び出す時に起こる。ItemProviderが項目値を配列で返却する場合は

* 数値型項目は getAsNumberでdouble[ ]
* 文字型項目は getAsStringでString[ ]

を返却する必要があり、配列の長さは1以上である。特にString[]の各要素はnullを許可しない。配列の長さが0、またはString[]の要素中にnullが含まれている場合にこのエラーが起こる。メッセージに問題になる項目のIDが含まれる。

* 処置方法

アプリケーション開発者はItemProviderが指定された項目値で長さ0の配列を返却するか、nullを含めたString[ ]を返却するかを確認して、必要であればアプリケーションを修正する。

ルールサーバはルールサービスを実行中に、ArrayIndexOutOfBoundsExceptionまたはNullPointerExceptionが発生した場合、入力された項目の有効性を確認して、上記のエラー事項があればこのエラーを発生させる。誤った項目値がArrayOutOfBoundsExceptionまたはNullPointerExceptionを直接的な原因でない場合も存在するが、因果関係があるため項目値の有効性を優先して確認する。項目値が有効であるにも関わらず、このエラーが継続していた場合、技術サポートが必要である。

### IRE-10006

* エラーメッセージ

内部関数の引数に誤りがあります。

* 原因

ルール実行中に、許容されていない引数が内部関数に渡された時に発生する。内部関数の引数のデータ型式はルールを保存する時にチェックされるが、値の有効性・配列の長さなどはランタイムでチェックされる。エラーメッセージに関数名と引数不正の事由が含まれる。

* 処置方法

ルール開発者は許容されていない引数が渡された事由を確認する。その原因が、ルール作成にある場合はルールを修正して、アプリケーションからの入力値にある場合はアプリケーション開発者がアプリケーションを修正する。

### IRE-10007

* エラーメッセージ

ユーザ定義関数がエラーを返しました。

* 原因

ルール実行中に、ユーザ定義関数から例外を投げた時に発生する。関数のメッセージにエラーを返却したユーザ定義関数名が含まれる。

* 処置方法

ユーザ定義関数の開発者にエラーメッセージを伝えて、エラーの事由を確認した後、適切な処置を行う。

### IRE-10008

* エラーメッセージ

ユーザ定義エラーが発生しました。

* 原因

ルールを開発する時、特定条件を満たす場合に、ERROR関数が呼び出されるようにしてルールの実行を防ぐことができる。ERROR関数にはメッセージとユーザ定義エラーコードを伝えられる。エラーメッセージにはそのメッセージとユーザ定義エラーコード(既に与えた場合)が含まれる。

* 処置方法

このエラーは業務段階のエラーであるため、ルール開発者とアプリケーション開発者が協議して処理する。ルール開発者が例外処理のためにこのエラーを発生させた場合、アプリケーション開発者はアプリケーション正常実行のために、このエラーコードに対する適切な処理を行う必要がある。

### IRE-10009

* エラーメッセージ

ルール呼び出し基準日付に適用できるバージョンがありません。

* 原因

ルールアプリケーションはルールを呼び出す時にルール実行の基準日付が通知される。通知されない場合はルールサービス呼び出し日付がその基準日付になる。ルールサービス実行中に、メインルール、またはサブルールに基準日付で適用可能なバージョンがない場合にこのエラーが発生する。このエラーの意味を業務的に解析すると「与えられた日付に適用できる規則がない」である。

エラーメッセージに問題が発生したルール名とアプリケーションから通知された基準日付が含まれる。

処置方法

メインルールからエラーが起きた場合には、メインルールの開始時点前の日付でルールサービスが呼び出されないようにアプリケーションを修正する。また、業務的に可能であれば、メインルールの開始時点を変更する。サブルールからエラーが発生した場合には、下位ルールの開始時点が実行された上位ルールバージョンの開始時点以降にならないよう、ルールを再構成する。例えば、次の場合を想定する。

* ルール実行の基準日付は2015/05/01である。
* 上位ルールには開始時点が2015/01/01のバージョンのみある。
* 下位ルールには開始時点が2015/06/01のバージョンのみある。

この場合、上位ルールは基準日付で適用できないバージョンがあるため正常に実行される。ただし、下位ルールは適用できるバージョンがないため、下位ルールが呼び出される時にこのエラーが発生する。これを解決するためには、上位ルールを2015/01/01に開始するバージョンと2015/06/01に開始するバージョンを分離し、前者では下位ルールを参照しないようにする。

### IRE-10011

* エラーメッセージ

ルール結果をI/Oアダプタに設定中にエラーが発生しました。

* 原因

ルールアプリケーションがI/Oアダプタ方式でルールを呼び出す場合、ルールサーバはルールが正常に実行された場合、その結果をResultConsumerのsetメソッドを利用してI/Oアダプタに設定する。この際、setメソッド内で例外が投げられて、このエラーが発生する。メッセージに例外のメッセージが含まれる。

* 処置方法

ResultConsumer開発者にこのエラーメッセージに含まれたメッセージを伝えて、その原因によって処置する。

### IRE-10012

* エラーメッセージ

0除算エラー

または

^ 演算の最初の被演算子には0を設定できません。

* 原因

エラーメッセージによって原因が異なる。「0除算エラー」は次の場合に発生する。

* 割り算で除数が数字である場合、その値が0である時
* 割り算で除数が多重値である場合、多重値の中に0がある時
* 割り算で除数の値が0である時

「^ 演算の最初の被演算子には0を設定できません」は^演算の基数(最初の被演算子)が0の時に発生する。

* 処置方法

被演算子に0を与えないよう処置する。

### IRE-10013

* エラーメッセージ

ブランチルールの分岐条件結果と一致するブランチノードがありません。

* 原因

ブランチルールは分岐条件を実行してその結果と一致する名前のブランチルールを実行するルールである。分岐条件を実行したが指定した名前のルールがない場合、またはルールはあるがそのルールがブランチノードで設定されていない場合、このエラーが発生する。ただし、デフォルト値がある場合にはエラーが発生せずデフォルト値が実行される。メッセージには分岐条件の結果が含まれる。

* 処置方法

ルール開発者は次の順番でチェックする。

* 分岐条件を満たすかを確認する。
* 分岐条件に使用される項目の値が有効の値であるかを確認する。
* 新しいブランチノードの追加や、既存のルールをブランチノードで設定するかどうかを確認する。
* 業務的にデフォルト値を設定するかどうかを確認する。

### IRE-10015

* エラーメッセージ

文法エラーによりルールを実行できません。

* 原因

ルールサービス実行中に実行するメインルール、またはその下位ルールに文法エラーがあって実行できない場合発生する。メッセージにエラーがあるルールの情報が含まれる。

* 処置方法

ルール開発者にエラーが発生したルールの情報を伝えて文法エラーを修正する。

### IRE-10016

* エラーメッセージ

項目値にnullが含まれています。

* 原因

ユーザ定義関数からルールを呼び出す時に、引き渡された項目がnullである場合に発生する。

* 処置方法

ユーザ定義関数の開発者はルールを呼び出す時に項目値にnullを渡さないように関数を修正する。

ユーザ定義関数のexecuteメソッドにはcom.innoexpert.innorulesj.suite.udf.Engineオブジェクトが渡されて、このオブジェクトのexecuteRule、executeRuleByName、executeRuleAliasメソッドを利用して下位ルールが呼び出せる。この際、追加的な項目を渡すことができるが、この項目値はnullを許可しない。ユーザ定義関数に対する詳細は「InnoRules Application Programming Interface Guide」を参照のこと。

### IRE-10017

* エラーメッセージ

定義されていないルールです。

* 原因

ユーザ定義関数から呼び出すルールが、定義されていない場合に発生する。メッセージにユーザ定義関数が要求したルールのID、ルール名およびルールのエイリアスが含まれる。

* 処置方法

ユーザ定義関数の開発者はルール内部を確認して関数を修正する。

### IRE-10018

* エラーメッセージ

定義されていない項目です。

* 原因

ユーザ定義関数からルールを呼び出す時に渡した項目が、定義されていない項目である場合に発生する。メッセージにユーザ定義関数が渡した項目のID、項目名および項目のエイリアスが含まれる。

* 処置方法

ユーザ定義関数の開発者は項目コードを確認して関数を修正する。

### IRE-10019

* エラーメッセージ

項目のデータ型に誤りがあります。

* 原因

ユーザ定義関数からルールを呼び出す時に渡した項目のデータ型が、ルールシステムに登録されたデータ型とは異なる場合に発生する。メッセージにその項目のID、項目名および項目エイリアスが含まれる。

* 処置方法

ユーザ定義関数の開発者は項目のデータ型を確認して関数を修正する。

### IRE-10020

* エラーメッセージ

再帰ルールの呼び出しが発生しました。

* 原因

1つのルールは実行中に他のルールを呼び出すことができる。この呼び出し関係で、自分が呼び出したルール、またはそのルールの下位から呼び出されたルールが、自分をまた呼び出すことを再帰呼び出しという。このエラーは再帰呼び出しが発生した際に発生する。メッセージには再帰呼び出しされたルールのIDが含まれる。

InnoRulesはルールを保存する時に、呼び出し可能な関係を確認して、再帰呼び出しが発生する可能性がある場合にはエラーを発生させる。ただし、EXEC関数、またはEXECRULE関数を利用する場合やユーザ定義関数で動的にルールを呼び出す場合には、ランタイムにより呼び出しルールが決まるため、ルール保存時に再帰呼び出しを検出できない。

* 処置方法

ルール開発者はEXEC関数、EXECRULE関数、またはユーザ定義関数が再帰呼び出しを発生させるかを確認して、再帰呼び出しが発生しないように修正する。このような処置を行ったにも関わらず、エラーが発生し続けた場合、技術サポートが必要である。

### IRE-10021

* エラーメッセージ

定義されていないコード体系です。

* 原因

ユーザ定義関数でcom.innoexpert.innorulesj.suite.udf.Engine.executeメソッドに渡されたコード体系が有効ではない場合に発生する。ユーザ定義関数から渡されたコード体系の値がメッセージに含まれる。

* 処置方法

ユーザ定義関数の開発者はコード体系の値を確認して関数を修正する。サポートするコード体系は 「InnoRules Application Programming Interface Guide」を参照のこと。

### IRE-10022

* エラーメッセージ

文法に合わない表現式です。

* 原因

ユーザ定義関数の中でcom.innoexpert.innorulesj.suite.udf.Engine.executeメソッドによって実行するルール表現式に文法エラーがある場合に発生する。メッセージに文法エラーメッセージが含まれる。

* 処置方法

ユーザ定義関数の開発者は文法エラーメッセージを確認して、文法に合わない表現式が伝達されないように修正する。

### IRE-10023

* エラーメッセージ

項目値の入力が重複しました。

* 原因

ルールアプリケーションがI/Oオブジェクト方式でルールを呼び出す時に、RuleReqオブジェクトに渡された項目の中でコードが重複された場合に発生する。

* 処置方法

ルールアプリケーションの開発者はRuleReqオブジェクトに同一コードの項目が一回のみ保存されるようにアプリケーションを修正する。

### IRE-10024

* エラーメッセージ

アイテムプロバイダが許容していないオブジェクト型を返しました。

* 原因

ルールアプリケーションがI/Oアダプタ方式でルールを呼び出す時に使用したItemProviderが、許容されないオブジェクト型を返却した時にこのエラーが発生する。項目のデータ型によってItemProvierはgetXXXメソッドで次のJavaオブジェクトを返却する。

* 数値型項目

getAsNumberメソッドはjava.lang.Doubleまたはdouble[ ]を返却する。

* 文字型項目

getAsStringメソッドはjava.lang.Stringまたはjava.lang.String[ ]を返却する。

エラーメッセージには実際に返却されたオブジェクトのデータ型が含まれる。

* 処置方法

ルールアプリケーションの開発者は許容された型のオブジェクトのみを返却するようにItemProviderを修正する。

### IRE-10025

* エラーメッセージ

項目値のクエリ中にアイテムプロバイダが例外をスローしました。

* 原因

ルールアプリケーションがI/Oアダプタ形式でルールを呼び出す時に発生する。ルールサーバがルールサービスを実行する途中に項目値をクエリするためにItemProviderのgetXXXメソッドを呼び出したが、このメソッドから例外が投げられた時にこのエラーが発生する。例外のメッセージがエラーメッセージに含まれる。

* 処置方法

ItemProviderの開発者に例外メッセージを伝えて処置方法を問い合わせする。

### IRE-10026

* エラーメッセージ

要求されるデータ型で、EXEC関数実行結果を変換できません。

* 原因

EXEC、EXECRULE関数はテーブルルールの条件セル、結果セル、DBルール、またはブランチルールなどのデフォルト値のセルで使用できる。このセルは特定のデータ型を持つ。例えば、テーブルルールの条件セルは論理値を持つ。テーブルルールの結果セルまたはデフォルト値セルはルールデータ型と互換する値を持つ。EXEC、EXECRULE関数の結果がこの条件を満たさない時このエラーが発生する。セルが要求するデータ型と関数結果のデータ型がメッセージに出力される。

* 処置方法

ルール開発者はセルが要求するデータ型を返却するようにEXECまたはEXECRULE関数の引数を修正する。

### IRE-10027

* エラーメッセージ

ユーザ定義関数で実行エラーが発生しました。

* 原因

ルールサーバがユーザ定義関数の実行コードを呼び出して、そのコードからJava例外を投げた時にこのエラーが発生する。ユーザ定義関数のエラーメッセージがこのメッセージに含まれる。

* 処置方法

ユーザ定義関数の開発者にエラーメッセージを伝えて、発生した原因を把握して適切に処置する。

### IRE-10028

* エラーメッセージ

ユーザ定義関数の実行結果が要求されるデータ型ではありません。

* 原因

ユーザ定義関数はcheckメソッドとexecuteメソッドを持っている。checkメソッドはユーザ定義関数を含んでいるルール表現式が保存される時に、関数の引数の有効性を検証して関数のリターン形式を返却する関数である。executeメソッドは引数の値を受けて結果を返却する実行メソッドである。executeメソッドの結果はcheckメソッドで返却したリターン形式であり、そうではない場合にこのエラーが発生する。

* 処置方法

ユーザ定義関数の開発者はexecuteメソッドの結果がcheckで返却した形式と一致するようにユーザ定義関数を修正する。

### IRE-10029

* エラーメッセージ

行数が0の場合、項目に割り当てられません。

* 原因

項目を新しく割り当ててルールを呼び出す文法({ルール:[項目]=値})で「値」が多重値である場合に行数は少なくとも1行以上でなければならない。

* 処置方法

ルール開発者は項目に割り当てる値の行が1行以上になるようにルールを修正する。

### IRE-10030

* エラーメッセージ

結合する2つの値の行数が一致しません。

* 原因

次のセルが原因である。

* テーブルルールで同一行の結果セル
* デフォルト値のセル
* フロールールで1つの結果ノードセル

で、2つ以上のセルに多重値が使用された場合に、多重値の行数は同一であるべきである。そうではない場合にこのエラーが発生する。メッセージに多重値の行数が含まれる。

* 処置方法

ルール開発者は上記の多重値の行数が同一になるようにルールを修正する。

### IRE-10031

* エラーメッセージ

コールバックサービスに対するハンドラが登録されていません。

* 原因

ルールサービスの実行中にコールバックルールを実行するように構成したが、コールバックルールで指定した名称のコールバックサービスがルールアプリケーション環境に登録されていない時に発生する。メッセージにコールバックサービス名が含まれる。

* 処置方法

ルールシステム管理者はルールアプリケーション環境にコールバックサービスを登録する。

### IRE-10032

* エラーメッセージ

コールバックハンドラの生成中、またはハンドラがサービスを実行中にエラーが発生しました。

* 原因

ルールサーバはコールバックサービスを処理するためにCallbackHandlerFactory.createCallbackHandlerを利用してCallbackHandlerを生成する。生成されたCallbackHandlerのdoCallbackを呼び出してサービスを実行する。createCallbackHandlerまたはdoCallbackで例外が発生するとこのエラーが発生する。例外のメッセージがエラーメッセージに含まれる。

* 処置方法

コールバックハンドラの開発者にエラーメッセージを伝えて必要な処置を行う。

### IRE-10033

* エラーメッセージ

出力コールバックパラメータの行数に誤りがあります。

* 原因

コールバックハンドラがサービスの処理結果を2つ以上のコールバックパラメータに保存する時には各パラメータの行数が一致するように値を設定する。そうではない場合にこのエラーが発生する。エラーメッセージに問題になるコールバック出力パラメータ名が含まれる。

* 処置方法

コールバックサービスの開発者は出力パラメータ値の行数がすべて一致するようにコールバックハンドラを修正する。詳細内容は「InnoRules Application Programming Interface Guide」を参照のこと。

### IRE-10034

* エラーメッセージ

コールバック出力パラメータにnull値が含まれています。

* 原因

コールバックハンドラが文字型コールバック出力パラメータにnull値を保存した場合にこのエラーが発生する。エラーメッセージに問題になるパラメータ名が含まれる。

* 処置方法

コールバックハンドラの開発者は文字型コールバック出力パラメータにnull値を保存しないようにコールバックハンドラを修正する。

### IRE-10035

* エラーメッセージ

ループルール項目値の行数が一致しません。

* 原因

2つ以上のループ項目に多重値が設定される場合にこの多重値の行数が一致しないとこのエラーが発生する。問題になるループ項目のIDがメッセージに含まれる。

* 処置方法

ルール開発者はループ項目に設定される多重値の行数が一致するようにルールを修正する。

### IRE-10036

* エラーメッセージ

ルールサーバのトレーシングが有効になっていません。

* 原因

トレース機能が無効化されているルールサーバにルールサービスを要求する時に実行パスを作成するようにした場合にこのエラーが発生する。

* 処置方法

このルールサーバでトレース機能を有効化するか、トレース機能が有効化されているルールサーバを利用して実行パスを作成する。

ルールサーバのトレース機能の有効化または無効化をウィザードでできない場合は技術サポートが必要である。デフォルトとしてトレース機能は有効化されている。

### IRE-10037

* エラーメッセージ

ルールの結果がありません。

* 原因

単一値を持つべきルールを実行したが、結果がない場合は次のケースを含む。

* テーブルにすべての条件を満たす行が1つもない場合
* DBルールでクエリの結果が0行で、デフォルト値も設定されていない場合
* ディシジョンテーブル、またはデータルールの場合に条件を満たす行がない場合

各ルール形式別に結果がない場合は「InnoRules User’s Guide of Rule Builder」を参照のこと。

* 処置方法

ルール開発者は、単一値を持つべきルールが結果を必ず持つようにルールを修正する。例外条件がある場合に、ERROR関数などを利用して業務的に意味があるエラーメッセージを作成する。

### IRE-10038

* エラーメッセージ

バインドする値の行数が0です。

* 原因

DBルールでバインド構文(<$$ 表現式 $$>)の表現式で使用する値の行数が0の場合に発生する。

* 処置方法

ルール開発者はバインド構文内で多重値が使用された場合に行数が0にならないようにルールを修正する。

### IRE-10039

* エラーメッセージ

条件に一致した結果が1件以上あります。

* 原因

ディシジョンテーブル、またはデータルールで条件を満たす行が2つ以上ある場合にこのエラーが発生する。ルールが単一か多重かは関係ない。

* 処置方法

ルール開発者は、条件を満たす行が1つのみになるようにルールまたはデータを修正する。多重ルールの場合「すべてを満たす条件実行」を選択できる。どの方法を選択するかは業務的な観点で決める。

### IRE-10040

* エラーメッセージ

データルールを実行中に、データベースエラーが発生しました。

* 原因

データルールを実行するためにデータベースにクエリを発行したが、JDBC段階でエラーが発生した場合にこのエラーが発生する。メッセージにJDBCメッセージが含まれる。

* 処置方法

JDBCメッセージによって処置する。データベースサーバで問題が発生した場合、まずDBAに問い合わせる。

### IRE-10041

* エラーメッセージ

データルールクエリの結果をルールシステムに変換できません。

* 原因

データルールクエリを実行して、その結果の各カラム値を対応するルールリターンのデータ型への変換ができない場合にこのエラーが発生する。次はルールリターンの型によって許容されるカラムのデータ型である。ここで、カラムの型ということはカラムのSQLタイプではなく、ResultSet.getObject()を実行した時に返す形式である。

* 数値型

java.lang.Numberの下位クラスの場合、またはオブジェクトのtoString()の値がDouble.valueOf()を利用してDoubleに変換できるすべてのクラス

* 文字型

toString()を利用して文字列に変換できるすべてのクラス

* 論理型

java.lang.Booleanクラス、またはオブジェクトのtoString()の値がBoolean.valueOf()を利用してBooleanに変換できるすべてのクラス

* 処置方法

ルール開発者はルールのリターン形式をデータベースの形式と互換できるように変更する。

### IRE-10042

* エラーメッセージ

登録されていないルールインタフェースクラスタです。

* 原因

ルールアプリケーションがクラスタ管理者にClusterManager.get( clusterName )またはClusterManager.getInterface( clusterName )を利用してルールインタフェースクラスタを要求したが、そのルールインタフェースクラスタが登録されていない場合にこのエラーが発生する。クラスタ名がメッセージに含まれる。

* 処置方法

クラスタが登録されていない事由は次のとおりである。

* ルールアプリケーションが誤ったクラスタ名を渡した場合、ルールアプリケーションの開発者はルールシステム管理者に問い合わせてクラスタ名を修正する。設定されていない場合はルールシステム管理者が設定に追加する。
* 設定には追加されているが、初期化段階でルールインタフェースクラスタを生成中にエラーが発生した場合、ルールシステム管理者はログを確認して、ルールインタフェースが生成されていない原因を確認し処置する。

### IRE-10043

* エラーメッセージ

DBルールクエリのセレクトアイテム数とルールカラム数が異なります。

* 原因

DBルールクエリのセレクトアイテム数は、そのルールのリターン数と同じでなければならない。そうではない場合にこのエラーが発生する。ルールIDとクエリのセレクトアイテム数、ルールのリターン数がメッセージに含まれる。

* 処置方法

ルール開発者はクエリ、またはルール情報を修正してセレクトアイテム数とルールのリターン数を一致させる。

### IRE-10044

* エラーメッセージ

DBルールをアップデータで使用するためには、データベースのコネクションが自動コミット(auto commit)に設定されていなければなりません。

* 原因

UPDATE文で構成されているDBルールを実行するためには、DBルールの実行に使用されたデータベースコネクションの自動コミットオプションが設定されている必要がある。そうではない場合にこのエラーが発生する。DBルールがSELECT文で構成されている場合は該当しない。

* 処置方法

ルールシステム管理者はDBルールが使用しているデータソースの自動コミットオプションを有効にする。データソースの管理主体によっては、WASのようなアプリケーションフレームワーク担当者の協力が必要な場合もある。

### IRE-10045

* エラーメッセージ

1つの数値型カラムで構成されたDBルールのみアップデートで使用できます。

* 原因

UPDATE文で構成されているDBルールは、必ず数値型リターン1つのみで構成される必要がある。そうではない場合にこのエラーが発生する。ただし、ルールが単一か多重かは関係ない。

* 処置方法

ルール開発者は数値型のリターン1つのみを返すようにルールの情報を修正する。

### IRE-10046

* エラーメッセージ

DBルールの実行中にデータベースエラーが発生しました。

* 原因

DBルールの実行のためにSQLを実行したが、データベースからエラーが発生した場合である。エラーの原因はSQLシンタックス、データ欠陥、通信上の問題、データソースレベルの問題など様々である。正確な原因はメッセージに含まれたデータベースエラーを利用してトレースする必要がある。

* 処置方法

データベースのエラーメッセージを確認して原因を探して処置する。

### IRE-10047

* エラーメッセージ

DBルールの実行中にメモリ不足が発生しました。

* 原因

DBルールを実行する途中でOutOfMemoryErrorが発生した場合である。メモリ不足の原因に対するDBルールを断定することはできない。JVMのメモリがあまりにも少なく設定されていた場合や、他のアプリケーションでメモリリークなどが発生して、使用可能メモリ領域が不足した状態でDBルールが実行されて、このエラーが発生した可能性もある。このエラーメッセージに含まれたDBルールの照会件数が過度に多い場合はDBルールを原因として推定する。

* 処置方法

メッセージに含まれたDBルールの照会件数が非常に多い場合はDBルールのクエリに問題がないのか確認する。正常なクエリでありながら、照会件数が多い場合はルール構成を修正することで照会件数を減らす。DBルールの照会件数が少なくてもこのエラーが発生した場合、システム管理者に問い合わせてヒープ分析によるメモリ不足の原因を探す。

### IRE-10048

* エラーメッセージ

ルールインタフェースのクローズ時のエラーが疑われます。

* 原因

ルールアプリケーションがルールインタフェースを使った後にcloseメソッドを呼び出さずに、参照を切断した場合に発生する。ルールアプリケーションはクラスタ管理者(またはルールインタフェースクラスタ)からルールインタフェースを使用した後、必ずcloseを呼び出してインタフェースを返さなければならない。メッセージにこのインタフェースを借りたスレッドID、スレッド名、使用した総時間が含まれる。

* 処置方法

ルールシステム管理者はスレッド情報と借用時間を利用して、ルールインタフェースを返さないアプリケーションを検索して、このアプリケーション開発者はルールインタフェースを返すようにアプリケーションを修正する。

### IRE-10049

* エラーメッセージ

ルールインタフェースクラスタに使用可能なプールがありません。

* 原因

ルールインタフェースクラスタには1つ以上のルールインタフェースプールがあり、アプリケーションがルールインタフェースを要求する時にHA(高可用性)ポリシーより、その中の1つのプールからルールインタフェースを提供する。すべてのインタフェースプール(またはこのプールと連携されたルールサーバ)が障害・シャットダウンされて、使用できるプールがない場合にこのエラーが発生する。

* 処置方法

ルールシステム管理者はルールインタフェースプールがルールサーバを使用できない事由を調査して、それをリカバリする。リカバリができない場合はこのインタフェースクラスタを使用するルールアプリケーションはルールサービスを呼び出せなくなるため、場合によってはアプリケーションを中止して、他の正常なアプリケーションによる代替サービスができるように処置する。

### IRE-10050

* エラーメッセージ

ルールインタフェースクラスタに使用可能なインタフェースがありません。

* 原因

ルールインタフェースクラスタの使用可能な全インタフェースプールのルールインタフェースが使用中である状態で、アプリケーションがルールインタフェースを要求した場合に発生する。また、アプリケーションの負荷がルールインタフェースクラスタで設定した負荷を超過している場合にも発生する。

ルールインタフェースクラスタのwhen-exhaust-actionがblockに設定された場合には、このエラーが発生せずに、使用可能なルールインタフェースが生成されるまでインタフェースを要求したアプリケーションがブロックされる。ウィザードを利用した場合、このオプションはデフォルトでblockに設定されており、このエラーが発生することはない。ルールインタフェースを返さないルールアプリケーションが短時間で数回実行される場合、IRE-10048エラーよりこのエラーが先に起こる。以降、GCが発生する時にIRE-10048エラーが発生する。

* 処置方法

同時に実行するルールアプリケーション数を減らすか、ルールインタフェースプールのボリュームを増やすか、when-exhaust-actionをblockに設定するかが考えられる。特定アプリケーションが短時間で繰り返して実行された後にこのエラーが発生し、そのあとIRE-10048が発生した場合、インタフェースリークが想定される。

### IRE-10051

* エラーメッセージ

業務データベース接続中にエラーが発生しました。

* 原因

DBルール、データルールなどを実行するために指定された業務データベースへの接続を試みたがJDBCエラーが発生した場合である。エラーの正確な原因はJDBCエラーメッセージを確認する。

* 処置方法

JDBCメッセージを確認し、確認結果に伴う処置をする。

### IRE-10052

* エラーメッセージ

単一値を持つべきDBルールのクエリ結果の行数が1件ではありません。

* 原因

単一で設定されたDBルールのクエリを実行したが、クエリ結果行数が1ではない場合に発生する。

* 処置方法

ルール開発者はクエリ、データなどを確認してルール、データ変更を行う。

### IRE-10500

* エラーメッセージ

SOAPメッセージに誤りがあります。

* 原因

SOAP/HTTPを利用してルールWebサービスを呼び出す時に発生する可能性がある。SOAPメッセージがルールのWSDLを従わない場合に発生する。

* 処置方法

アプリケーション開発者は最新のWSDLを確認してSOAPメッセージを修正する。

### IRE-10501

* エラーメッセージ

サポートしていないSOAPバージョンです。

* 原因

SOAPメッセージのバージョンがサポートされていないバージョンである場合に発生する。SOAP1.1・1.2のみをサポートする。

* 処置方法

アプリケーション開発者はSOAPメッセージのバージョンを変更する。

### IRE-10502

* エラーメッセージ

登録されていないシステムエイリアスです。

* 原因

SOAP/HTTPのURLに指定されている単位システムのエイリアスが登録されていない場合に発生する。1つのWebサービスアダプタで複数の単位システムのルールをサービスする時に、どの単位システムのルールサービスを呼び出すかを区分するためにURLに単位システムと1：1でマッピングされたエイリアスが指定される。このエイリアスが識別できない場合である。

* 処置方法

SOAP/HTTPのURLを確認して修正する。

## IRE-11XXX

ルールサーバがルールリポジトリにアクセスする途中で発生するエラーである。ルールサーバがルールリポジトリにアクセスする場合は次のとおりである。

* ルールサーバが初期化される時
* ルールサービスを実行する時
* ルールビルダーサーバがルールデータを変更してルールリポジトリキャッシュを更新する時

### IRE-11000

* エラーメッセージ

ルールDBからデータ読み込み中に、データベースエラーが発生しました。

* 原因

ルールサーバがルールDBにルールデータをクエリする途中でJDBC段階のエラーが発生した場合である。メッセージにJDBCエラーメッセージが含まれる。正確な原因はJDBCエラーメッセージを確認する。メッセージには、どのルールデータにアクセスする途中でエラーが発生したかに対する情報が含まれる。

* 処置方法

JDBCエラーメッセージを確認して処置する。

### IRE-11001

* エラーメッセージ

JDBCローダからデータベースコネクションを取得できませんでした。

* 原因

ルールDBにアクセスするためにデータベース接続を取得する途中でエラーが発生した場合である。メッセージにJDBCエラーメッセージが含まれる。正確な原因はJDBCエラーメッセージを確認する。

* 処置方法

JDBCエラーメッセージを確認して処置する。

### IRE-11002

* エラーメッセージ

JDBCローダがネーミングシステムからデータソースが見つかりませんでした。

* 原因

JDBCローダに設定した名前のデータソースをネーミングシステムから見つけられなかった場合に発生する。データソースを生成する途中で問題があった可能性もある。ウィザードを使わない場合、JNDI設定とJDBCローダのデータソース名の不一致も想定される。メッセージにJNDIエラーメッセージが含まれる。

* 処置方法

JNDIエラーメッセージとルールサーバのログファイルを確認して処置する。データソースが生成される途中でエラーが発生した場合、その原因を探して処置する。JNDI設定とJDBCローダの設定が合わない場合には設定を修正する。ただし、ウィザードを使用した場合は該当しない。

### IRE-11003

* エラーメッセージ

ルールDBにシステム情報がありません。

* 原因

ルールDBにはそのDBのデータがどの単位ルールシステムに属するのかに関する情報が保存されている。ルールDBが壊れて関連情報がない場合にこのエラーが発生する。

* 処置方法

ルールDBが壊れた可能性があるため技術サポートが必要である。

### IRE-11004

* エラーメッセージ

タイムスタンプが一致しません。

* 原因

ルールビルダーサーバはルールデータが変更されると、変更されたルール要素のIDとタイムスタンプをルールサーバに伝送する。タイムスタンプにはルールデータの変更時点が記録されており、DBのルールデータとルールビルダーのルールデータ、ルールサーバでキャッシングしているデータの一致を保障することに使用されている。

ルールサーバがルールデータを更新した後に、ルールDBのタイムスタンプとルールビルダーサーバが伝送したタイムスタンプが一致しない場合に、このエラーが発生する。ルールサーバのルールデータキャッシングには変更された内容が反映されない。

誤った構成設定で、ルールビルダーサーバとルールサーバがアクセスするルールDBが異なる場合に発生する。

* 処置方法

構成設定を確認して、ルールビルダーサーバとルールサーバが同一ルールDBにアクセスするかを確認して、そうでない場合に構成を変更する。通常、ウィザードを使用した場合にはこのような問題は発生しないため、ウィザードを使用してこのエラーが発生した場合は技術サポートが必要である。

## IRE-12XXX

ルールランタイムの設定を誤った時や、初期化途中で起こるエラーである。エラーコードに付け加えて事由コードが含まれる場合もある。事由コードは詳細なエラーを示す。コードに「\*」表示がついたエラーは、ウィザードを利用した場合には発生しないエラーである。発生した場合、まずそのエラー以前に発生した他のエラーに対する処置を行い、その後もエラーが継続した場合は、技術サポートが必要である。

### IRE-12100\*

* エラーメッセージ

JDBCローダでデータソース名の設定が不足しています。

* 原因

JDBCローダの設定にデータソース名が漏れている。

* 処置方法

JDBCローダの設定にdsnameプロパティを利用してデータソース名を設定する。

### IRE-12101\*

* エラーメッセージ

TCPモニター設定に誤りがあります。

* 事由コード
* 00： アクセス文字列が指定されていない。
* 01： タイムアウト設定が正しい数値ではない。
* 02： 再接続間隔が正しい数値ではない。
* 03： モニターサーバアドレスが分からない。
* 04： モニターサーバポート設定が正しい数値ではない。

### IRE-12102\*

* エラーメッセージ

業務データベースに対するデータソースがネーミングシステムから見つかりませんでした。

* 事由コード
* 00： 業務データベースに対するデータソースがネーミングシステムに登録されていない。

### IRE-12103\*

* エラーメッセージ

ルールリポジトリのキャッシュ設定に誤りがあります。

* 事由コード
* 00： 同一名称のルールリポジトリキャッシュが既に登録されている。
* 01： ルールリポジトリキャッシュが指定されていない。
* 02： ルールリポジトリオブジェクトが生成できない。
* 03： ルールローダが指定されていない。
* 04： ルールローダオブジェクトが生成できない。

### IRE-12104\*

* エラーメッセージ

業務データベースの設定が不足しています。

* 事由コード
* 00： 業務データベースに対するデータソースが設定されていない。

### IRE-12105\*

* エラーメッセージ

ユーザ定義関数の設定に誤りがあります。

* 事由コード
* 00： 同一名称のユーザ定義関数が既に登録されている場合である。

### IRE-12106\*

* エラーメッセージ

ルールスイートの設定に誤りがあります。

* 事由コード
* 00： この形式のルールに対するルールスイートは登録されていない。
* 01： この形式のルールに対するルールスイートが既に登録されている。
* 02： ルールスイートクラス名が指定されていない。
* 03： ルールスイートクラスが出来ないか、生成できない。
* 04： この形式のルールに対するルールスイートローダが登録されていない。
* 05： ルールスイートローダクラス名が指定されていない。
* 06： ルールスイートローダクラスがロードできないか、生成できない。

### IRE-12107\*

* エラーメッセージ

ルールインタフェースクラスタの設定に誤りがあります。

* 事由コード
* 00： ルールインタフェースプール設定がない。
* 01： コールバッククラスパスが有効ではない。
* 02： コールバックハンドラクラスがロードできない。
* 03： コールバックハンドラファクトリが生成できない。
* 04： コールバック設定ファイルを読む込むことが出来ない。
* 05： 設定XMLエラーである。
* 06： 既に同一名称のルールインタフェースクラスタが登録されている。
* 07： 設定にルールインタフェースクラスタ名が不足している。

### IRE-12108\*

* エラーメッセージ

ルールインタフェースプールの設定に誤りがあります。(一般)

* 事由コード
* 00： ルールインタフェースファクトリクラスが見つからない。
* 01： ルールインタフェースファクトリオブジェクトが生成できない。
* 02： 設定値が正しい数値ではない。
* 03： 最大活性インタフェース数は0より大きくなければならない。
* 04： ルールインタフェースを生成中、エラーが発生した。

### IRE-12109\*

* エラーメッセージ

ルールインタフェースプールの設定に誤りがあります。(ローカルルールサーバ)

* 事由コード
* 00： ルールリポジトリキャッシュ名が漏れている。
* 01： ルールリポジトリキャッシュが登録されていない。
* 02： DBルールタイムアウト設定が正しくない。
* 99： その他のエラー。メッセージによって処置する。

### IRE-12110\*

ウィザードを利用した場合でも、事由コード08は起こる。TCPコネクタサーバが搭載されているInnoRulesサーバが中止になった場合、このサーバを含むTCPコネクタサーバグループを使用するルールアプリケーションからこのエラーが発生する。このエラーが発生してもTCPコネクタサーバグループの高可用性ポリシーにより他のTCPコネクタサーバが使用できる場合、ルールサービス呼び出しは問題ない。

* エラーメッセージ

ルールインタフェースプールの設定に誤りがあります。(TCP)

* 事由コード
* 00： ポートが指定されていない。
* 01： ポート値が正しくない、正数ではない。
* 02： 連結タイムアウト値が正しくない、0または正数でなければならない。
* 03： タイムアウト値が正しくない、0または正数でなければならない。
* 04： 通信バッファーのサイズが正しくない。最小1024バイトである。
* 05： 設定値が正しい数値ではない。
* 06： リモートルールサーバにログイン途中、サーバとの連結が切断された。
* 07： リモートルールサーバにログイン途中IOExceptionが発生した。
* 08： リモートルールサーバへの連結が失敗した。

### IRE-12111\*

* エラーメッセージ

ルールインタフェースプールの設定に誤りがあります。(スペクトル)

* 事由コード
* 00： スペクトル名が指定されていない。
* 01： 同一名称のスペクトルインタフェースファクトリが既に登録されている。
* 02： 指定した名前のスペクトルインタフェースファクトリがスペクトル管理者に登録されていない。スペクトルをスペクトル管理者から除去できない。
* 03： 指定したスペクトルは既にモニタリング中である。
* 04： 指定したスペクトルはモニタリング中ではない。
* 05： スペクトル分析に使用する代理インタフェースファクトリが指定されていない。
* 06： スペクトル分析に使用する代理インタフェースファクトリが見つからない。

### IRE-12112

* エラーメッセージ

ルール変更モニタリング中にエラーが発生しました。

* 事由コード
* 00： モニターサーバに連結できない。活性ルールビルダーが搭載されたInnoRulesサーバが中止されている場合、また、ルールサーバがこのモニターサーバに連結を試みた場合に起こる。ルールサーバは他のビルダーサーバにモニター連結を再び試みて、動作中である他のルールビルダーサーバに連結される。ルールシステム管理者はログを利用して再連結を試みた後にモニター連結ができたかを確認する。
* 01： モニターサーバが連結を拒否した。モニター連結が成立したルールビルダーサーバが終了するとこのエラーが発生する可能性がある。ルールシステム管理者はログを確認して再び連結を試みたか、連結が成功したかを確認する。
* 02： I/Oエラーが発生、技術サポートが必要である。
* 03： モニターイベントハンドラ作業中、例外をなげる。技術サポートが必要である。
* 99： 予期せぬエラーである。技術サポートが必要である。

### IRE-12113

* エラーメッセージ

アプリケーションが、使用中のルールリポジトリキャッシュを終了しようとしています。

* 原因

ルールサービス実行途中でルールサーバを終了した場合、このメッセージが表示される場合がある。ルールサーバが動作しているアプリケーションフレームワークが、アプリケーションが動作している途中でもルールランタイムを終了すると発生する。

* 処置方法

アプリケーションフレームワーク段階でアプリケーションが実行中であった場合、ルールランタイムを終了しないように修正する。

### IRE-12200

* エラーメッセージ

ライセンスエラー

* 事由コード

ライセンスに関わるエラーである。

* 00： 設定ファイルにライセンス情報がない場合である。ウィザードを利用する場合には発生しない。
* 01： 指定されたライセンスURLがない位置の指定または権限がない場合である。ライセンスファイルがあるか、ルールアプリケーションまたはInnoRulesサーバがライセンスファイルを読み込む権限があるかを確認する。
* 02： ネーミングシステムにライセンスをバインドする途中に発生したエラーである。詳細メッセージを添付して技術サポートが必要である。
* 03： ライセンスが既にネーミングシステムにバインドされている。詳細メッセージを添付して技術サポートが必要である。
* 04： 正しい形式のライセンスファイルでないか、ファイルが破損した。破損していないライセンスファイルに入れ替える。
* 05： ライセンス期間が満了された。InnoRules購入元に問い合わせする。
* 06： システムプロセッサ数がライセンス制限を越える。システムプロセッサの数を減らすか、InnoRules購入元に問い合わせする。
* 07： 登録されていない単位システムである。InnoRules購入元に問い合わせする。
* 08： ルール数がライセンスの制約を超過した。InnoRules購入元に問い合わせする。
* 09： ライセンス情報がネーミングシステムから見つからない。詳細メッセージを添付して技術サポートが必要である。
* 10： ユーザアカウント数がライセンスの制約を超過した。InnoRules購入元に問い合わせする。
* 99： その他のライセンスエラーは詳細メッセージを添付して技術サポートが必要である。

### IRE-12300\*

* エラーメッセージ

TCPコネクタサーバの設定に問題があります。

* 事由コード
* 00： サーバ名が設定されていない。
* 01： 同一サーバ名が既に登録されている。

### IRE-12301

* エラーメッセージ

TCPコネクタサーバの始動中にエラーが発生しました。

* 原因

TCPコネクタサーバ始動中にエラーが発生しサーバ実行が失敗した場合に発生する。サービスポートが既に使用されているか、MBeanサーバに登録中にJMX例外が発生した可能性もある。原因は追加メッセージを参照のこと。

* 処置方法

エラーメッセージに含まれた追加メッセージを確認して処置する。

### IRE-12400\*

* エラーメッセージ

Webサービスアダプタの設定に誤りがあります。

* 事由コード
* 01： 指定されたWSDLプロバイダクラスが見つからない。
* 02： WSDLプロバイダクラスを生成する途中でエラーが発生した。
* 03： 単位システムのIDが指定されていない。
* 04： エイリアスが重複、または基本単位システムを2つ以上指定する。
* 05： ルールDB設定が未設定、または誤っている。
* 06： 指定されたデータソースが見つからないか、ネーミングシステムで検索中にエラーが発生した。
* 07： ルールインタフェース設定が未設定、または誤っている。
* 08： ルールインタフェース設定が未設定、または誤っている。
* 09： Webサービスアダプタ名が指定されていない。
* 10： モニター設定が未設定、または誤っている。
* 11： 重複した単位システムである。

### IRE-12401

* エラーメッセージ

Webサービス情報のロード中にDBエラーが発生しました。

* 原因

ルールDBからWebサービスデータを読み込む途中にJDBCエラーが発生する。Webサービスアダプタが初期化される時、またはルールビルダーサーバがWebサービスデータを変更して同期化する時にルールDBにアクセスしてWebサービスデータを照会する。JDBCエラーメッセージがエラーメッセージに含まれる。

* 処置方法

JDBCエラーメッセージを確認して処置する。

### IRE-12402\*

* エラーメッセージ

同期化が失敗しました。復旧できませんでした。

* 原因

IRE-11004と同様である。ただし、ルールサーバではなくWebサービスアダプタから発生する。

* 処置方法

IRE-11004を参照のこと。

### IRE-12997

* エラーメッセージ

MBeanサーバにMBeanを登録または解除中に、エラーが発生しました。

* 原因

ルールシステムの各要素は管理のためにMBeanサーバに登録する。この際、問題が生じた場合、このエラーが発生する。詳細情報がエラーメッセージに含まれる。

* 処置方法

エラーメッセージを確認して処置する。

### IRE-12998

* エラーメッセージ

割り込みが発生しました。

* 原因

ルールランタイムにはバックグラウンドで動くスレッドがあって同期化のために多様なロックを使用する。ルールランタイムは待機状態にあるスレッドを起動するためにThread.interrupt()を呼び出すが、この時にこのエラーが発生する。ルールランタイムがこのメソッドを呼び出す場合は次のとおりである。

* ルールランタイムが終了する時、すなわち、ルールアプリケーションまたはInnoRulesサーバが終了する時
* ビルダーサーバクラスタリングからマスタービルダーサーバがスレーブに切替する時

Thread.interrupt()はアプリケーションフレームワークが終了する時にフレームワークによって呼び出されることもある。

* 処置方法

上記で列挙した場合に、このエラーは問題にならないため無視してもいい。ただし、システム運用中にエラーが発生する場合は技術サポートが必要である。