



Copyright © 2018 NTT DATA INTRAMART CORPORATION

目次

- 改訂情報
- はじめに
 - 本書の目的
 - 前提条件
 - 対象読者
 - 用語解説
- 各種インストール・設定変更
- intra-mart Accel Platform 構成ファイルの作成
- Payara のインストール
 - Payara をインストールする
 - Payara の起動と停止
 - 管理コンソールにアクセス
- Payara の設定
 - インスタンス設定の追加
 - JVM の設定
 - Network の設定
 - Thread Pool の設定
 - インスタンスの追加
 - データベースの接続設定
 - war ファイルのデプロイ
 - インスタンスの起動と停止
 - Executor Service の設定
- テナント環境の構築
- 付録
 - Payara のサービス化
 - NIC が複数ある場合
 - Payara のクラスタリング
 - Apache Solr
- セットアップで困ったら . . .
- アップデート パッチの適用

改訂情報

変更年月日	変更内容
2018-08-01	初版
2018-12-01	第2版 下記を変更しました <ul style="list-style-type: none"> ▪ 「Oracle の Connection Pool の作成」の「Resource Type」の説明を変更
2019-04-01	第3版 下記を変更しました <ul style="list-style-type: none"> ▪ 「intra-mart Accel Platform 構成ファイルの作成」に Metro と OpenPortal WSRP の選択に関する注意事項を追加
2020-04-01	第4版 下記を変更しました <ul style="list-style-type: none"> ▪ 「はじめに」の前提条件を変更 ▪ 「JVM の設定」の「JVM Options の設定」の設定例を変更 ▪ 「JVM の設定」の「JVM Options の設定」に注意書きを追加
2022-06-01	第5版 下記を変更しました <ul style="list-style-type: none"> ▪ 「Payara をインストールする」に jakarta.mail.jar の差し替えに関する記載を追加
2022-12-01	第6版 下記を変更しました <ul style="list-style-type: none"> ▪ 「SQL Server の Connection Pool の作成」に Additional Properties の設定で <code>SendTimeAsDatetime</code> を指定する手順を追加
2023-04-01	第7版 下記を変更しました <ul style="list-style-type: none"> ▪ 「JVM の設定」の「JVM Options の設定」に、カスタマーサクセスライセンスを利用する場合に必要な JVM 引数を追加のコラムを追加
2023-10-01	第8版 下記を変更しました <ul style="list-style-type: none"> ▪ 「はじめに」を Payara Server 5.2022.5 向けに変更 ▪ 「Payara をインストールする」の「jakarta.mail.jar の差し替え」を削除 ▪ 「JVM の設定」の「JVM Options の設定」に <code>--add-opens</code> を指定する手順を追加 ▪ 「war ファイルのデプロイ」にデプロイに失敗した場合のコラムを追加

本書の目的

本書では Payara Server 5.2022.5 に intra-mart Accel Platform のセットアップを行う手順について説明します。



コラム

Payara についての詳しい情報は、 Payara の Web サイトでご確認ください。

<https://www.payara.fish/> (English)

また、 Payara Server のドキュメントも併せてご活用ください。

<https://docs.payara.fish/> (English)

前提条件

リリースノートに記載されているシステム要件を満たしている必要があります。

詳細は「[リリースノート](#)」を参照してください。

対象読者

以下の利用者を対象としています。

- Payara Server 5.2022.5 に intra-mart Accel Platform のセットアップを行われる方

用語解説

Payara のホームディレクトリを %PAYARA_HOME% と略します。

— intra-mart Accel Platform セットアップガイド (Payara編) 第8版 2023-10-01
各種インストール・設定変更

intra-mart Accel Platform のセットアップに必要なコンポーネントのインストールおよび設定を行います。

具体的な手順は「[intra-mart Accel Platform セットアップガイド](#)」の「[intra-mart Accel Platform を利用するためのミドルウェアのインストールと設定](#)」を参照してください。

intra-mart Accel Platform の設定およびwarファイルの出力を行います。

具体的な手順は「[intra-mart Accel Platform セットアップガイド](#)」の「[WARファイルの作成](#)」を参照してください。



コラム

Payara Server 5.2022.5 で intra-mart Accel Platform を動作させるには、ベースモジュールとして 2023 Autumn(Hollyhock) 以降を選択する必要があります。



注意

追加リソースとして、必ず [Payara 5用設定ファイル] を出力する必要があります。

① 利用するプラットフォーム用の追加リソース(設定ファイル等)を配置します。

- Resin 4.0.x用設定ファイル
- Payara 5用設定ファイル
- Weblogic 12c用設定ファイル
- SAStruts用設定ファイル (Resin)
- SAStruts用設定ファイル (Payara)
- SAStruts用設定ファイル (WebSphere)
- SAStruts用設定ファイル (Weblogic)

また、 SAStruts を利用する場合は、 [SAStruts用設定ファイル (Payara)] を出力する必要があります。

① 利用するプラットフォーム用の追加リソース(設定ファイル等)を配置します。

- Resin 4.0.x用設定ファイル
- Payara 5用設定ファイル
- Weblogic 12c用設定ファイル
- SAStruts用設定ファイル (Resin)
- SAStruts用設定ファイル (Payara)
- SAStruts用設定ファイル (WebSphere)
- SAStruts用設定ファイル (Weblogic)



注意

Juggling プロジェクトのビルド時には、サーバ製品として必ず [Payara 5] を選択する必要があります。



注意

Payara Server を利用する場合、下記のモジュールを選択しないでください。デプロイ時にエラーが発生します。

- ライブラリ > サードパーティ製ライブラリ > Metro
- ライブラリ > サードパーティ製ライブラリ > OpenPortal WSRP

Payara をインストールする

項目

- インストール
- JDBC ドライバの配置

インストール

1. 以下から Full version の Payara Server 5.2022.5 をダウンロードします。
<https://www.payara.fish/downloads> (English)
2. ダウンロードした <payara-5.xxx.zip> を任意のディレクトリに展開します。

JDBC ドライバの配置

1. JDBC ドライバをダウンロードします。
JDBC ドライバの入手先は以下を参照してください。
[JDBC ドライバ](#)
2. ダウンロードした、 JDBC ドライバを以下のディレクトリにコピーします。
 - Linux

```
%PAYARA_HOME%/glassfish/lib
```

- Windows

```
%PAYARA_HOME%\glassfish\lib
```

Payara の起動と停止

項目

- サーバの起動
- サーバの停止
- サーバの再起動
- サーバのログ

サーバの起動

1. 以下のコマンドを実行します。

```
%PAYARA_HOME%/glassfish/bin/asadmin start-domain
```

または

```
%PAYARA_HOME%/glassfish/bin/startserv
```

2. 起動が完了すると以下のメッセージが出力されます。

- `asadmin start-domain` で起動した場合

```
Command start-domain executed successfully.
```

- `startserv` で起動した場合

```
[#|2023-05-15T15:17:42.913+0900|情報|Payara 5.2022.5|javax.enterprise.web|_ThreadID=151;_ThreadName=Thread-13;_TimeMillis=1684131462913;_LevelValue=800;_MessageID=AS-WEB-GLUE-00172;|  
Loading application [_admingui] at [/]|#]
```

サーバの停止

- 以下のコマンドを実行します。

```
%PAYARA_HOME%/glassfish/bin/asadmin stop-domain
```

または

```
%PAYARA_HOME%/glassfish/bin/stopserv
```

- 停止が完了すると以下のメッセージが出力されます。

```
Command stop-domain executed successfully.
```

サーバの再起動

- 以下のコマンドを実行します。

```
%PAYARA_HOME%/glassfish/bin/asadmin restart-domain
```

- 再起動が完了すると以下のメッセージが出力されます。

```
Command restart-domain executed successfully.
```

サーバのログ

以下のファイルに出力されます。

```
%PAYARA_HOME%/glassfish/domains/domain1/logs/server.log
```

管理コンソールにアクセス

Payara をインストールしたホスト（ローカル）から管理コンソールにアクセスする場合は、特別な設定なしにブラウザからアクセスできます。

別のホストからアクセスする場合は、セキュア管理を有効にする必要があります。

項目

- ローカルから管理コンソールにアクセス
- リモートホストから管理コンソールにアクセス

ローカルから管理コンソールにアクセス

- サーバを起動します。起動方法は「[サーバの起動](#)」を参照してください。
- ブラウザで以下にアクセスします。

```
http://localhost:4848
```

リモートホストから管理コンソールにアクセス

- サーバを起動します。起動方法は「[サーバの起動](#)」を参照してください。
- 以下のコマンドを実行して管理者のパスワードを設定します。

- Linux

```
%PAYARA_HOME%/glassfish/bin/asadmin change-admin-password
```

- Windows

```
%PAYARA_HOME%\glassfish\bin\asadmin.bat change-admin-password
```

上記のコマンドを実行すると、以下の入力を求められます。

Enter admin user name [default: admin]> 管理者のユーザ名を入力します。デフォルトは admin です。

Enter the admin password> 管理者のパスワードを入力します。デフォルトは 空 です。

Enter the new admin password> 管理者の新しいパスワードを入力します。

Enter the new admin password again> 管理者の新しいパスワードを再度入力します。

3. 以下のコマンドを実行して、セキュア管理を有効にします。

- Linux

```
%PAYARA_HOME%/glassfish/bin/asadmin enable-secure-admin
```

- Windows

```
%PAYARA_HOME%\glassfish\bin\asadmin.bat enable-secure-admin
```

上記のコマンドを実行すると、以下の入力を求められます。

Enter admin password for user “admin”> 管理者のパスワードを入力します。

4. サーバ再起動します。再起動方法は「[サーバの再起動](#)」を参照してください。

5. ブラウザで以下にアクセスします。

```
https://<%PAYARA_HOST%>:4848
```

Payara の設定を行います。

i コラム

インスタンスを作成することで独立した JVM で intra-mart Accel Platform を構築できます。
これにより、デプロイした別のアプリケーションの影響を受けずに intra-mart Accel Platform を稼働できます。

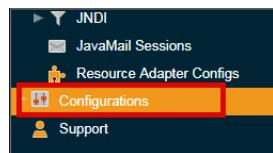
ここでは、インスタンスを作成する方法も含めて、 Payara の設定方法をご紹介します。

! 注意

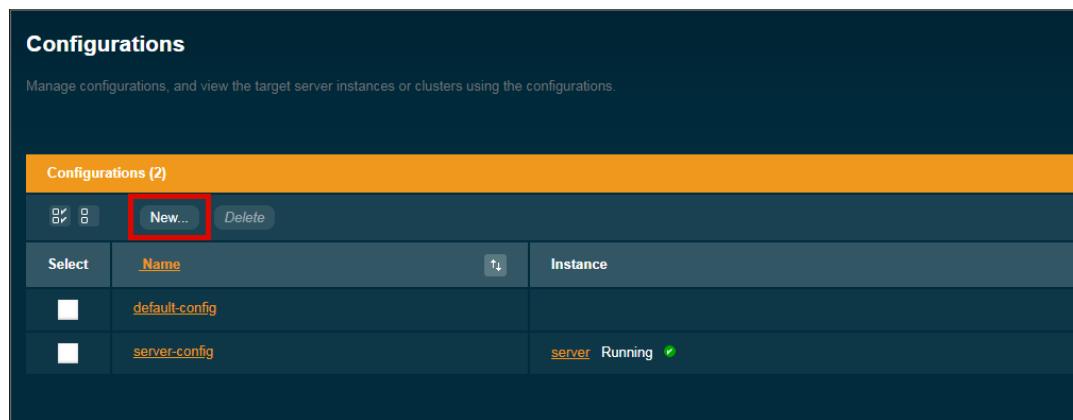
インスタンスを作成しない場合、 [Availability Service] は利用できません。
これにより、セッションレプリケーションなどを利用できません。
分散環境を構築する場合は、必ずインスタンスを作成してください。

インスタンス設定の追加

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Configurations] を選択します。



3. [New...] を選択します。



Configurations

Manage configurations, and view the target server instances or clusters using the configurations.

Configurations (2)		
Select	Name	Instance
<input type="checkbox"/>	default-config	
<input type="checkbox"/>	server-config	server Running ✓

4. [New Configuration] で以下を設定して [OK] を選択します。



New Configuration

Name: * imart-config

Copy configuration from: * default-config

OK Cancel

Name 設定の名前を入力します。例として、ここでは imart-config と入力します。

Copy configuration from コピー元の設定の名前を選択します。例として、ここでは default-config を選択します。

JVM の設定

項目

- JVM Options の設定
- Debug の設定

JVM Options の設定

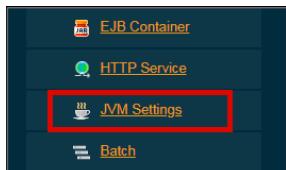
1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Configurations] を選択します。



3. 「[インスタンス設定の追加](#)」で追加した設定を選択します。例として、ここでは [imart-config] を選択します。

Select	Name	Instance
<input type="checkbox"/>	default-config	
<input type="checkbox"/>	imart-config	
<input type="checkbox"/>	server-config	server Running ✓

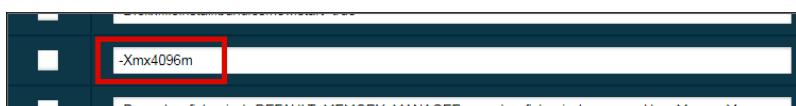
4. [JVM Settings] を選択します。



5. [JVM Options] を選択します。

Select	Value
<input type="checkbox"/>	-server

6. Java 最大ヒープサイズの設定を行います。デフォルトで設定されている値から以下のように変更します。



この例では、以下を設定しています。

-Xmx4096m

7. [Add JVM Option] をクリックして、`--add-opens` オプションを設定します。

--add-opens=java.base/sun.util.locale=ALL-UNNAMED
--add-opens=java.xml/com.sun.org.apache.xerces.internal.jaxp=ALL-UNNAMED
--add-opens=java.base/sun.util.calendar=ALL-UNNAMED
--add-opens=java.xml/com.sun.org.apache.xerces.internal.jaxp.datatype=ALL-UNNAMED
--add-opens=java.base/java.text=ALL-UNNAMED
--add-opens=java.desktop/java.awt.font=ALL-UNNAMED
--add-opens=java.base/java.math=ALL-UNNAMED
--add-opens=java.base/java.util.concurrent=ALL-UNNAMED

この例では、以下を設定しています。

```
--add-opens=java.base/sun.util.locale=ALL-UNNAMED
--add-opens=java.xml/com.sun.org.apache.xerces.internal.jaxp=ALL-UNNAMED
--add-opens=java.base/sun.util.calendar=ALL-UNNAMED
--add-opens=java.xml/com.sun.org.apache.xerces.internal.jaxp.datatype=ALL-UNNAMED
--add-opens=java.base/java.text=ALL-UNNAMED
--add-opens=java.desktop/java.awt.font=ALL-UNNAMED
--add-opens=java.base/java.math=ALL-UNNAMED
--add-opens=java.base/java.util.concurrent=ALL-UNNAMED
```

8. [Add JVM Option] をクリックして、その他の JVM Option を設定します。

Select	Min JVM Version	Max JVM Version	Value
<input type="checkbox"/>			-Xms4096m
<input type="checkbox"/>			-Dfile.encoding=UTF-8
<input type="checkbox"/>			-Duser.timezone=Asia/Tokyo
<input type="checkbox"/>			-XX:+UseG1GC
<input type="checkbox"/>			-XX:MaxGCPauseMillis=200
<input type="checkbox"/>			-XX:InitiatingHeapOccupancyPercent=30
<input type="checkbox"/>			-XX:+HeapDumpOnOutOfMemoryError
<input type="checkbox"/>			-Djava.io.tmpdir=/var/payara-tmp

この例では、以下を設定しています。

```
-Xms4096m
-Dfile.encoding=UTF-8
-Duser.timezone=Asia/Tokyo
-XX:+UseG1GC
-XX:MaxGCPauseMillis=200
-XX:InitiatingHeapOccupancyPercent=30
-XX:+HeapDumpOnOutOfMemoryError
-Djava.io.tmpdir=/var/payara-tmp
--add-opens=java.base/sun.util.locale=ALL-UNNAMED
--add-opens=java.xml/com.sun.org.apache.xerces.internal.jaxp=ALL-UNNAMED
--add-opens=java.base/sun.util.calendar=ALL-UNNAMED
--add-opens=java.xml/com.sun.org.apache.xerces.internal.jaxp.datatype=ALL-UNNAMED
--add-opens=java.base/java.text=ALL-UNNAMED
--add-opens=java.desktop/java.awt.font=ALL-UNNAMED
--add-opens=java.base/java.math=ALL-UNNAMED
--add-opens=java.base/java.util.concurrent=ALL-UNNAMED
```



コラム

「-Djava.io.tmpdir」オプションを追加し、Payara Server が利用する作業ディレクトリを変更できます。このオプションが指定されていない場合、Payara Server が利用する作業ディレクトリは JVM のデフォルトの設定が利用されます。



コラム

カスタマーアクセスライセンスをご契約中の場合には、ライセンスポータルと通信してご契約内容の変更が自動反映されます。

Payara をインストールした環境からライセンスポータルへの通信にプロキシサーバを利用する場合は、次の JVM 引数の設定が必要です。

-Dhttps.proxyHost	プロキシサーバのホストURL
-Dhttps.proxyPort	プロキシサーバのポート番号
-Dhttps.proxyUser	プロキシサーバへの接続ユーザ
-Dhttps.proxyPassword	接続ユーザのパスワード

カスタマーアクセスライセンスについての詳細は、「[ライセンスの登録](#)」を参照してください。



注意

- 「-Djava.io.tmpdir」オプションにより指定されたディレクトリは事前に作成しておく必要があります。
- Payara Server 実行ユーザが読み込み、書き込みを行うことができる権限を設定しておく必要があります。
- Linux 系の環境では、このオプションが未指定の場合 /tmp が利用されます。
- cron 等の設定により定期的に/tmp 配下の内容が削除される設定が標準で組み込まれている場合があります。

9. [Save] を選択します。

The screenshot shows the Payara Server administration interface. The left sidebar lists configurations like JMS Resources, JNDI, JavaMail Sessions, Resource Adapter Configs, and imart-config. The imart-config section is expanded, showing sub-options like Admin Service, Data Grid, Request Tracing, Notification, HealthCheck, Availability Service, Connector Service, EJB Container, and HTTP Service. The central panel is titled 'JVM Options' and contains a table for setting JVM options. A row for '-server' is selected, with its 'Value' field set to '-Xbootclasspath/p:\${com.sun.aas.installRoot}/lib/grizzly-npn-bootstrap.jar'. The 'Save' button at the bottom right of the table is highlighted with a red box.

Debug の設定

Java のリモートデバッグを利用する場合、以下の手順を実施してください。

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Configurations] を選択します。



3. 「インスタンス設定の追加」で追加した設定を選択します。例として、ここでは [imart-config] を選択します。

Select	Name	Instance
<input type="checkbox"/>	default-config	
<input type="checkbox"/>	imart-config	
<input type="checkbox"/>	server-config	server Running ✓

4. [JVM Settings] を選択します。



5. [General] で [Debug] を ON にして [Save] を選択します。

Java Home: *	`\${com.sun.aas.javaRoot}`	Path to the directory in which the Java Development Kit is installed
Javac Options:	-g	Command line options to pass to the javac compiler
Debug:	<input checked="" type="checkbox"/> Enabled	Start server in debug mode ready for JPDA-based debugger
Debug Options:	-agentlib:jdwp=transport=dt_socket,server=y,suspend=n,address=5005	

Network の設定

- 項目
 - Network Listener の設定

Network Listener の設定

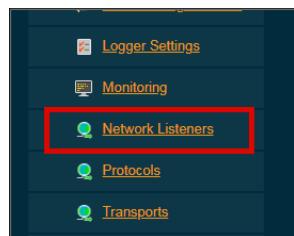
1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Configurations] を選択します。



3. 「インスタンス設定の追加」で追加した設定を選択します。例として、ここでは [imart-config] を選択します。

Select	Name	Instance
<input type="checkbox"/>	default-config	
<input type="checkbox"/>	imart-config	
<input type="checkbox"/>	server-config	server Running ✓

4. [Network Listeners] を選択します。



5. [http-listener-1] を選択します。

Select	Name	Port #	Protocol	Thread Pool	Enabled
<input type="checkbox"/>	admin-listener	<code> \${ASADMIN_LISTENER_PORT}</code>	sec-admin-listener	admin-thread-pool	true
<input type="checkbox"/>	http-listener-1	<code> \${HTTP_LISTENER_PORT}</code>	http-listener-1	http-thread-pool	true
<input type="checkbox"/>	http-listener-2	<code> \${HTTP_SSL_LISTENER_PORT}</code>	http-listener-2	http-thread-pool	true

6. [HTTP] を選択します。

HTTP

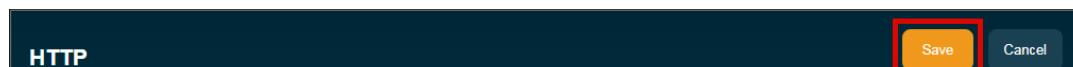
Modify HTTP settings for the protocol.

Load Defaults

Configuration Name: default-config

Protocol Name: http-listener-1

7. [Comet Support] と [Encoded Slash] を ON に設定して [Save] 選択します。



Modify HTTP settings for the protocol.

[Load Defaults](#)

Configuration Name: default-config

Protocol Name:	http-listener-1
Server Name:	
Alias name if server uses an alias. If a colon and port number are appended, that port will be used in URLs the server sends to the client.	
Default Virtual Server:	server
Use the Virtual Servers page to define a new virtual server	
Redirect Port:	
SSL port value for redirects	
Max Connections:	256
Maximum number of requests per connection in keep-alive mode.	
Timeout:	30 Seconds
Maximum time a connection can be deemed as idle and kept in the keep-alive state. A value of -1 will disable it.	
Upload Timeout:	<input checked="" type="checkbox"/> Enabled
Enable closing of connection for a servlet that reads bytes slowly after Connection Upload Timeout is reached	
Connection Upload Timeout:	300000 Milliseconds
Timeout for uploads if Upload Timeout is enabled. A value of -1 will disable it.	
Request Timeout:	900 Seconds
Time after which a request times out. A value of -1 will disable it.	
Send Buffer Size:	8192 Bytes
Size of the send buffer	
Header Buffer Length:	8192 Bytes
The size of the buffer used by the request processing threads to read the request data	
Max Post Size:	-1 Bytes
Maximum size of POST actions	
Max Form Post Size:	2097152 Bytes
Maximum size of a POST form	
Max Save Post Size	4096 Bytes
Maximum size of a POST which will be saved by the container during authentication.	
URI Encoding:	UTF-8
Character set used to decode the request URIs received	
HTTP/2:	<input checked="" type="checkbox"/> Enabled
Enable HTTP/2.	
Disable HTTP/2 Cipher Check:	<input type="checkbox"/> Enabled
Whether or not insecure cipher suites are allowed to establish TLS connections.	
HTTP/2 Max Concurrent Streams:	100
The number of concurrent streams allowed per HTTP/2 connection. The default is 100.	
HTTP/2 Initial Window Size:	65535 Bytes
The initial window size in bytes. The default is 64K - 1.	
HTTP/2 Max Frame Payload Size:	16777215 Bytes
The maximum size of the HTTP2 frame payload to be accepted. The default is 2 ²⁴ - 1.	
HTTP/2 Max Header List Size:	4096 Bytes
The maximum size, in bytes, of the header list.	
Compression:	off
Enable HTTP/1.1 GZIP compression to save server bandwidth	
Compressible Mime Types:	text/html;text/xml;text/plain
Comma-separated list of MIME types for which HTTP compression is used	
Compression Minimum Size:	2048 Bytes
Minimum size of a file when compression is applied	
No-Compression User Agents:	
Comma-separated list of regular expressions matching user agents of HTTP clients for which compression should not be used	

Restricted User Agent: List of restricted user agents on which HTTP compression is applied

Default Response Type: Specified as a semi-colon delimited string consisting of content-type, encoding, language, charset

Forced Response Type: The response type to be forced if the content served cannot be matched by any of the MIME mappings for extensions

Adapter: org.glassfish.grizzly.http.server.StaticHttpHandler
Class name of the static resources adapter

Comet Support: Enabled
Enable Comet support

DNS Lookup: Enabled
Enable Domain Name System (DNS) lookup

RCM Support: Enabled
Enable Resource Configuration Management (RCM) support

Trace: Enabled
Enable TRACE operation

Auth Pass Through: Enabled
Indicate that the network listener receives traffic from an SSL-terminating proxy server

Chunking: Enabled
Enable HTTP response chunking

XPowered By:
Include X-Powered-By: Servlet/3.0 in servlet-generated HTTP response headers

Server Header:
Include Server Header: Servlet/3.0 in servlet-generated HTTP response headers

XFrame Options:
Enabling this will set X-Frame-Options to "SAMEORIGIN" value in all HTTP response headers. If you have declared it in your application, it will take precedence.

Encoded Slash: Enabled
Allow encoded slash in URIs

Websockets Support: Enabled

Scheme Mapping HTTP Header Name used for identifying the originating protocol of an HTTP request

Remote User Mapping HTTP Header Name used for identifying the originating user of an HTTP request

Thread Pool の設定

項目

- Thread Pool の設定

Thread Pool の設定

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Configurations] を選択します。

User: admin Domain: domain1 Server: localhost

Home About... Help Enable Admin Recorder

Payara Server Console - Common Tasks

Payara News

- [Get Payara Support](#)
- [Payara News](#)

Administration

- [Change Administrator Password](#)
- [List Password Aliases](#)

Deployment

- [List Deployed Applications](#)
- [Deploy an Application](#)

Monitoring

- [Monitoring Data](#)

Payara Documentation

- [Payara Documentation](#)

Resources

3. 「インスタンス設定の追加」で追加した設定を選択します。例として、ここでは [imart-config] を選択します。

User: admin Domain: domain1 Server: localhost

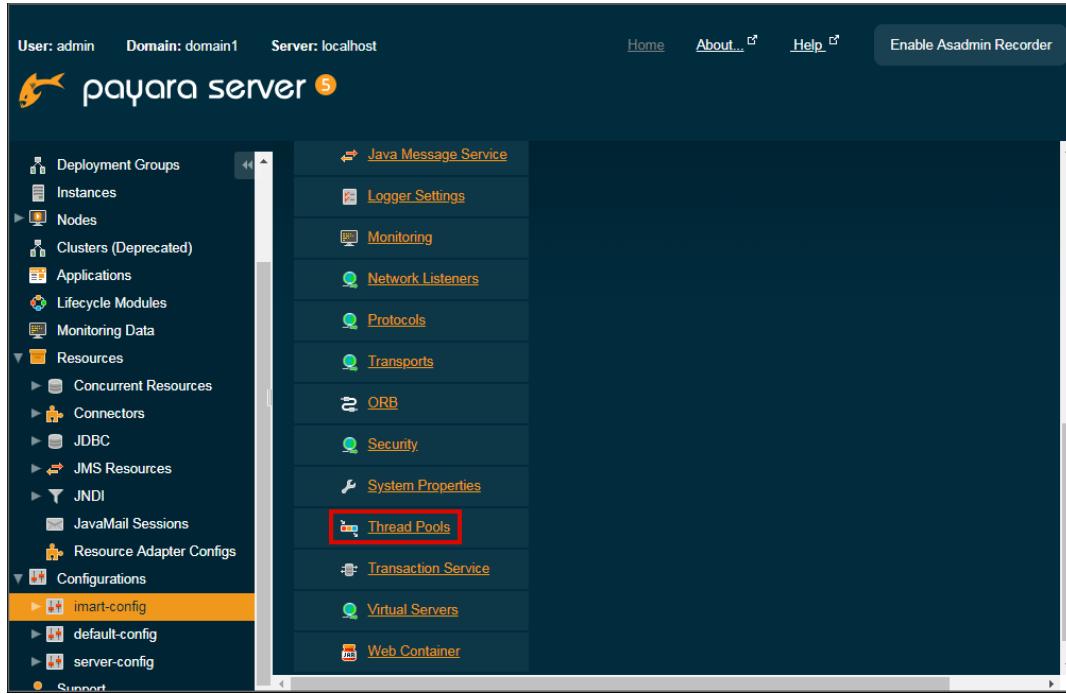
Home About... Help Enable Admin Recorder

Configurations

Manage configurations, and view the target server instances or clusters using the configurations.

Select	Name	Instance
<input type="checkbox"/>	default-config	
<input type="checkbox"/>	imart-config	
<input type="checkbox"/>	server-config	server Running ✓

4. [Thread Pools] を選択します。



5. [http-thread-pool] を選択します。

The screenshot shows the 'imart-config' configuration page within the Payara Admin Console. The left sidebar shows the same navigation menu as the previous screenshot, with 'imart-config' selected. The main content area is titled 'THREAD POOLS' and contains the following text: 'Use thread pools to limit a service to a specific number of concurrent threads.' Below this is a table titled 'Thread Pools (3)'. The table has columns: Select, Thread Pool ID, Max Thread Pool Size, Min Thread Pool Size, Max Queue Size, and Idle Timeout. There are three rows: 1. Thread Pool ID: admin-thread-pool, Max Thread Pool Size: 15, Min Thread Pool Size: 1, Max Queue Size: 256, Idle Timeout: 900. 2. Thread Pool ID: http-thread-pool, Max Thread Pool Size: 5, Min Thread Pool Size: 5, Max Queue Size: 4096, Idle Timeout: 900. 3. Thread Pool ID: thread-pool-1, Max Thread Pool Size: 200, Min Thread Pool Size: 2, Max Queue Size: 4096, Idle Timeout: 900. The 'http-thread-pool' row is highlighted with a red box.

Select	Thread Pool ID	Max Thread Pool Size	Min Thread Pool Size	Max Queue Size	Idle Timeout
	admin-thread-pool	15	1	256	900
	http-thread-pool	5	5	4096	900
	thread-pool-1	200	2	4096	900

6. [Max Thread Pool Size] に 256 を設定して [Save] 選択します。

User: admin Domain: domain1 Server: localhost

Edit Thread Pool

Modify an existing thread pool.

Configuration Name: imart-config

Name: http-thread-pool

Class Name: org.glassfish.grizzly.threadpool.GrizzlyExecutorService

Max Queue Size: 4096

Max Thread Pool Size: 256

Save

インスタンスの追加

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Instances] を選択します。



3. [New...] を選択します。

Payara Server Instances

Create and manage Payara server instances. Running instances cannot be deleted.

Server Instances (0)

New... Delete Start Stop -- More Actions --

Select	Name	LB Weight	Configuration	Node	Status
No items found.					

Save

4. [New Payara Server Instance] で以下のように入力して [OK] を選択します。

New Payara Server Instance

Create a Payara Server instance.

Instance Name: * imart-instance

Node: localhost-domain1

Configuration: imart-config

OK Cancel

Make a Copy of the Selected Configuration
Reference the Selected Configuration

Instance Name インスタンスの名前を入力します。例として、ここでは imart-instance と入力します。

Node	ノードを選択します。例として、ここでは localhost-domain1 を選択します。
Configuration	「インスタンス設定の追加」で追加した設定を選択します。例として、ここでは imart-config を選択します。 また、「インスタンス設定の追加」で追加した設定をこのインスタンスの設定として利用するため、[Reference the Selected Configuration] を選択します。

データベースの接続設定

データベースの接続設定を行います。

Connection Pool の作成

使用するデータベースに応じて設定を行ってください。

PostgreSQL の Connection Pool の作成

PostgreSQL を使用するための設定を行います。

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Resources] - [JDBC] - [JDBC Connection Pools] を選択します。

Select	Pool Name	Resource Type	Classname
	DerbyPool	javax.sql.DataSource	org.apache.derby.jdbc.ClientDataSource
	H2Pool	javax.sql.DataSource	org.h2.jdbcx.JdbcDataSource
	TimerPool	javax.sql.XADatasource	org.apache.derby.jdbc.EmbeddedXADatasource

3. [New...] を選択します。

User: admin Domain: domain1 Server: localhost

Home About Help Enable Admin Recorder

JDBC Connection Pools

To store, organize, and retrieve data, most applications use relational databases. Java EE applications access relational databases via the JDBC API. Before an application can access a database, it must get a connection.

Pools (3)				
	Pool Name	Resource Type	Classname	Details
<input type="checkbox"/>	DerbyPool	javax.sql.DataSource	org.apache.derby.jdbc.ClientDataSource	
<input type="checkbox"/>	H2Pool	javax.sql.DataSource	org.h2.jdbcx.JdbcDataSource	
<input type="checkbox"/>	TimerPool	javax.sql.XADataSource	org.apache.derby.jdbc.EmbeddedXADataSource	

4. [New JDBC Connection Pool (Step 1 of 2)] で以下のように入力して [Next] を選択します。

New JDBC Connection Pool (Step 1 of 2)

Identify the general settings for the connection pool.

* Indicates required field

General Settings

Pool Name: *	PgPool
Resource Type:	javax.sql.XADataSource
Database Driver Vendor:	Postgresql
Introspect:	<input type="checkbox"/> Enabled

Next Cancel

Pool Name コネクションプールの名前を入力します。例として、ここでは PgPool と入力します。

Resource Type リソースタイプを選択します。 javax.sql.XADataSource を選択します。

Database Driver Vendor JDBC ドライバのベンダを選択します。 Postgresql を選択します。

5. [New JDBC Connection Pool (Step 2 of 2)] の [Additional Properties] で以下を変更します。

All connections use same isolation level, requires Transaction Isolation

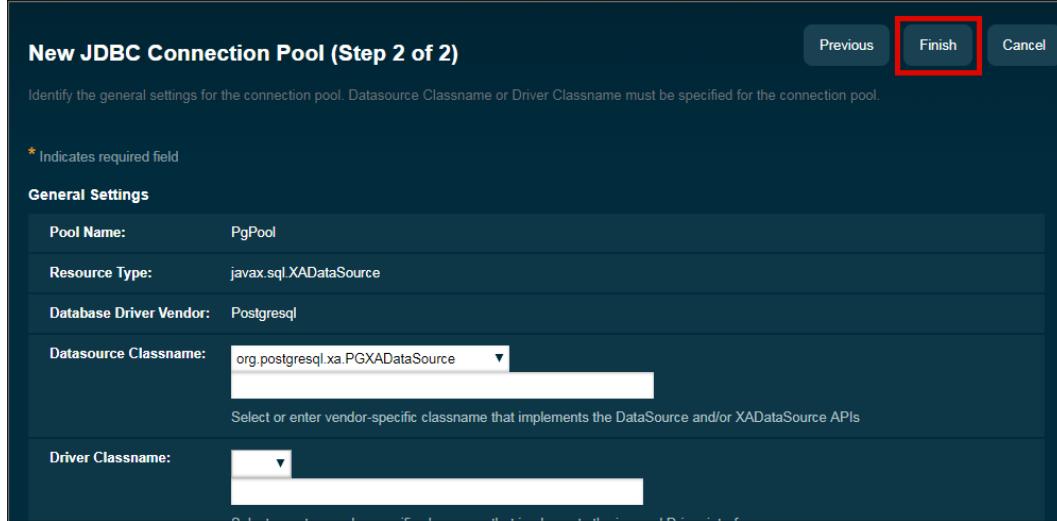
Additional Properties (58)

Select	Name	Value	Description
<input type="checkbox"/>	Url	jdbc:postgresql://localhost/iap_db	
<input type="checkbox"/>	User	imart	
<input type="checkbox"/>	Password	imart	
<input type="checkbox"/>	PreparedStatementCacheQueries	0	
<input type="checkbox"/>	PreparedStatementCacheSizeMiB	0	

Url データベースURLを設定します。例として、ここでは jdbc:postgresql://localhost/iap_db を設定しています。

User	データベースユーザを設定します。例として、ここでは imart を設定しています。
Password	データベースユーザのパスワードを設定します。例として、ここでは imart を設定しています。
PreparedStatementCacheQueries	プリペアドステートメントのキャッシュ数を設定します。例として、ここでは 0 を設定しています。
PreparedStatementCacheSizeMiB	プリペアドステートメントのキャッシュサイズを設定します。例として、ここでは 0 を設定しています。

6. [New JDBC Connection Pool (Step 2 of 2)] で [Finish] を選択します。



Oracle の Connection Pool の作成

Oracle を使用するための設定を行います。

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Resources] - [JDBC] - [JDBC Connection Pools] を選択します。

The screenshot shows the Payara Server management console. The left sidebar has a tree view with nodes like Deployment Groups, Instances, Nodes, Clusters (Deprecated), Applications, Lifecycle Modules, Monitoring Data, Resources (selected), Concurrent Resources, Connectors, JDBC (selected), JDBC Resources, JDBC Connection Pools (selected), JMS Resources, JNDI, JavaMail Sessions, Resource Adapter Configs, Configurations, imart-config, and default-config. The main content area is titled 'JDBC Connection Pools' and contains a table with the following data:

Select	Pool Name	Resource Type	Classname	De
<input type="checkbox"/>	DerbyPool	javax.sql.DataSource	org.apache.derby.jdbc.ClientDataSource	
<input type="checkbox"/>	H2Pool	javax.sql.DataSource	org.h2.jdbcx.JdbcDataSource	
<input type="checkbox"/>	TimerPool	javax.sql.XADataSource	org.apache.derby.jdbc.EmbeddedXADataSource	

3. [New...] を選択します。

User: admin Domain: domain1 Server: localhost

Home About Help Enable Admin Recorder

JDBC Connection Pools

To store, organize, and retrieve data, most applications use relational databases. Java EE applications access relational databases via the JDBC API. Before an application can access a database, it must get a connection.

Pools (3)	New...	Delete		
Select	Pool Name	Resource Type	Classname	Details
<input type="checkbox"/>	DerbyPool	javax.sql.DataSource	org.apache.derby.jdbc.ClientDataSource	
<input type="checkbox"/>	H2Pool	javax.sql.DataSource	org.h2.jdbcx.JdbcDataSource	
<input type="checkbox"/>	TimerPool	javax.sql.XADatasource	org.apache.derby.jdbc.EmbeddedXADataSource	

4. [New JDBC Connection Pool (Step 1 of 2)] で以下のように入力して [Next] を選択します。

New JDBC Connection Pool (Step 1 of 2)

Identify the general settings for the connection pool.

* Indicates required field

General Settings

Pool Name:	OraclePool
Resource Type:	javax.sql.XADatasource
Database Driver Vendor:	Oracle
Introspect:	<input checked="" type="checkbox"/> Enabled

Next Cancel

Pool Name コネクションプールの名前を入力します。例として、ここでは OraclePool と入力します。

Resource Type リソースタイプを選択します。 javax.sql.XADatasource を選択します。

Database Driver Vendor JDBC ドライバのベンダを選択します。 Oracle を選択します。

5. [New JDBC Connection Pool (Step 2 of 2)] の [Additional Properties] で以下を変更します。

Additional Properties (19)

Add Property Delete Properties

Select	Name	Value	Description
<input type="checkbox"/>	URL	jdbc:oracle:thin:@localhost:1521:orcl	
<input type="checkbox"/>	User	IMART	
<input type="checkbox"/>	Password	imart	

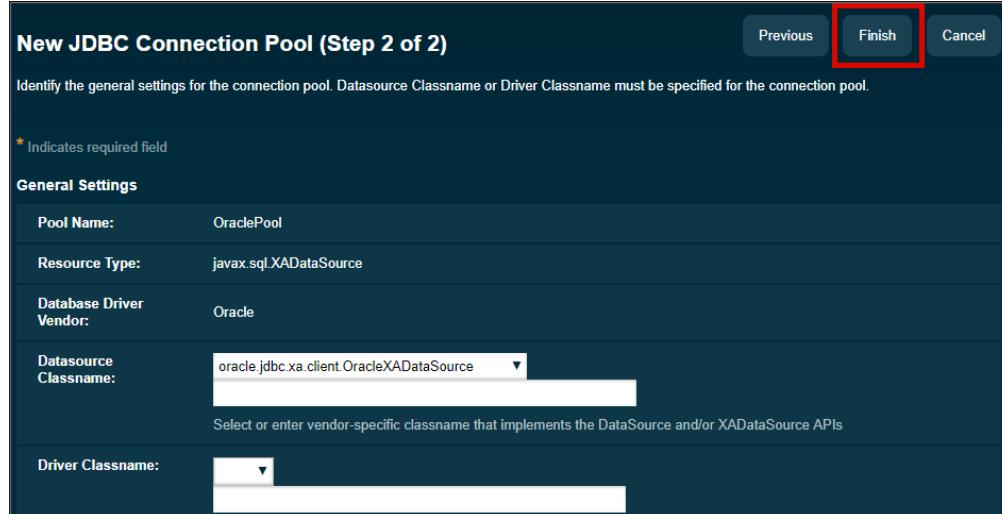
URL データベースURLを設定します。例として、ここでは `jdbc:oracle:thin:@localhost:1521:orcl` を設定しています。

User データベースユーザを設定します。例として、ここでは IMART を設定しています。

Password

データベースユーザのパスワードを設定します。例として、ここでは imart を設定しています。

6. [New JDBC Connection Pool (Step 2 of 2)] で [Finish] を選択します。



New JDBC Connection Pool (Step 2 of 2)

Identify the general settings for the connection pool. Datasource Classname or Driver Classname must be specified for the connection pool.

* Indicates required field

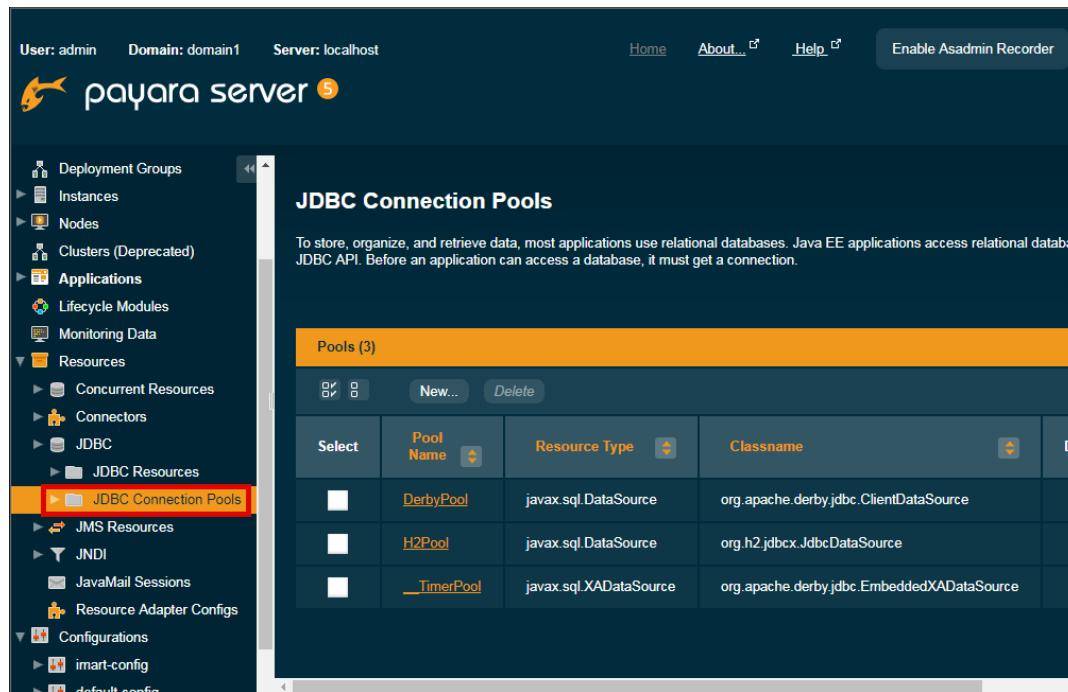
General Settings

Pool Name:	OraclePool
Resource Type:	javax.sql.XADataSource
Database Driver Vendor:	Oracle
Datasource Classname:	oracle.jdbc.xa.client.OracleXADataSource
Select or enter vendor-specific classname that implements the DataSource and/or XADataSource APIs	
Driver Classname:	oracle.jdbc.xa.client.OracleXADataSource

SQL Server の Connection Pool の作成

SQL Server を使用するための設定を行います。

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Resources] - [JDBC] - [JDBC Connection Pools] を選択します。



User: admin Domain: domain1 Server: localhost

JDBC Connection Pools

To store, organize, and retrieve data, most applications use relational databases. Java EE applications access relational databases via the JDBC API. Before an application can access a database, it must get a connection.

Pools (3)

Select	Pool Name	Resource Type	Classname	De
	DerbyPool	javax.sql.DataSource	org.apache.derby.jdbc.ClientDataSource	
	H2Pool	javax.sql.DataSource	org.h2.jdbcx.JdbcDataSource	
	TimerPool	javax.sql.XADataSource	org.apache.derby.jdbc.EmbeddedXADataSource	

3. [New...] を選択します。

User: admin Domain: domain1 Server: localhost

JDBC Connection Pools

To store, organize, and retrieve data, most applications use relational databases. Java EE applications access relational databases via the JDBC API. Before an application can access a database, it must get a connection.

Pools (3)

Select	Pool Name	Resource Type	Classname
<input type="checkbox"/>	DerbyPool	javax.sql.DataSource	org.apache.derby.jdbc.ClientDataSource
<input type="checkbox"/>	H2Pool	javax.sql.DataSource	org.h2.jdbcx.JdbcDataSource
<input type="checkbox"/>	TimerPool	javax.sql.XADatasource	org.apache.derby.jdbc.EmbeddedXADataSource

4. [New JDBC Connection Pool (Step 1 of 2)] で以下のように入力して [Next] を選択します。

User: admin Domain: domain1 Server: localhost

New JDBC Connection Pool (Step 1 of 2)

Identify the general settings for the connection pool.

* Indicates required field

General Settings

Pool Name: *	SqllserverPool
Resource Type:	javax.sql.XADatasource
Database Driver Vendor:	MicrosoftSqlServer

Pool Name

コネクションプールの名前を入力します。例として、ここでは SqllserverPool と入力します。

Resource Type

リソースタイプを選択します。 javax.sql.XADatasource を選択します。

Database Driver Vendor

JDBC ドライバのベンダを選択します。 MicrosoftSqlServer を選択します。

**注意**

SQL Server on Linux では分散トランザクションを利用できません。そのため、javax.sql.XADatasource ではなく javax.sql.DataSource を利用してください。

**注意**

SQL Server on Linux では分散トランザクションを利用できません。そのため、バーチャルテナント機能は利用できません。



注意

SQL Server (Windows) で分散トランザクション(javax.sql.XADataSource)を利用するには、SQL Server に対して以下の手順を事前に実施する必要があります。

<https://docs.microsoft.com/ja-jp/sql/connect/jdbc/understanding-xa-transactions>

5. [New JDBC Connection Pool (Step 2 of 2)] の [Additional Properties] で以下を変更します。

Name	Value
DatabaseName	iap_db
User	imart
URL	jdbc:sqlserver://localhost:1433
Password	imart
SelectMethod	cursor
SendTimeAsDatetime	false

URL

データベースURLを設定します。例として、ここでは `jdbc:sqlserver://localhost:1433` を設定しています。

User

データベースユーザを設定します。例として、ここでは imart を設定しています。

Password

データベースユーザのパスワードを設定します。例として、ここでは imart を設定しています。

DatabaseName

データベース名を設定します。例として、ここでは iap_db を設定しています。

SelectMethod

参照クエリ実行時にカーソルの利用可否を設定します。 cursor を設定します。

SendTimeAsDatetime

`java.sql.Time` を `datetime` 型としてサーバに送信するか否かを設定します。 false を設定します。

6. [New JDBC Connection Pool (Step 2 of 2)] で [Finish] を選択します。

User: admin Domain: domain1 Server: localhost

New JDBC Connection Pool (Step 2 of 2)

Identify the general settings for the connection pool. Datasource Classname or Driver Classname must be specified for the connection pool.

* Indicates required field

General Settings

Pool Name:	SqlserverPool
Resource Type:	javax.sql.XADataSource
Database Driver Vendor:	MicrosoftSqlServer
Datasource Classname:	com.microsoft.sqlserver.jdbc.SQLServerXADataSource
Driver Classname:	

JDBC Resource の作成

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Resources] - [JDBC] - [JDBC Resources] を選択します。

3. [New...] を選択します。

JDBC Resources

JDBC resources provide applications with a means to connect to a database.

Resources (3)

Select	JNDI Name	Logical JNDI Name	Enabled	Connection Pool	Description
<input type="checkbox"/>	jdbc/_TimerPool		✓	_TimerPool	
<input type="checkbox"/>	jdbc/default	java:comp/DefaultDataSource	✓	H2Pool	

4. [New JDBC Resource] で以下のように入力して [OK] を選択します。

JNDI Name: OK Cancel

Specify a unique JNDI name that identifies the JDBC resource you want to create. The name must contain only alphanumeric, underscore, dash, or dot characters.

Pool Name: PgPool Use the [JDBC Connection Pools](#) page to create new pools

Description:

Status: Enabled

Additional Properties (0)

Add Property Delete Properties

Select	Name	Value	Description
No items found.			

Targets

Available Targets:

- server

Selected Targets:

- imart-instance

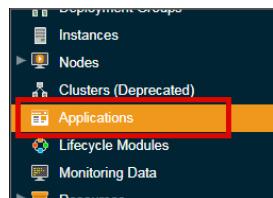
JNDI Name JNDI 名を設定します。 jdbc/default を設定してください。

PoolName 「[Connection Pool の作成](#)」で作成したコネクションプールを設定します。

Targets 「[インスタンスの追加](#)」で作成したインスタンスを [Selected Targets] に設定します。

war ファイルのデプロイ

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Applications] を選択します。



3. [Deploy...] を選択します。

Applications

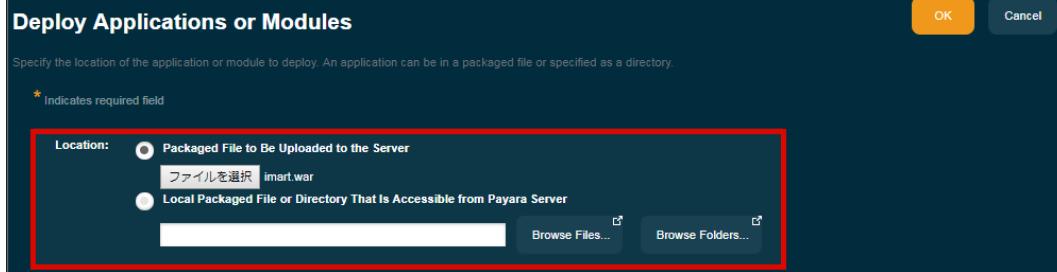
Applications can be enterprise or web applications, or various kinds of modules. Restart an application or module by clicking on the reload link, this action will apply only to the targets that the application or module is enabled on.

Deployed Applications (0)

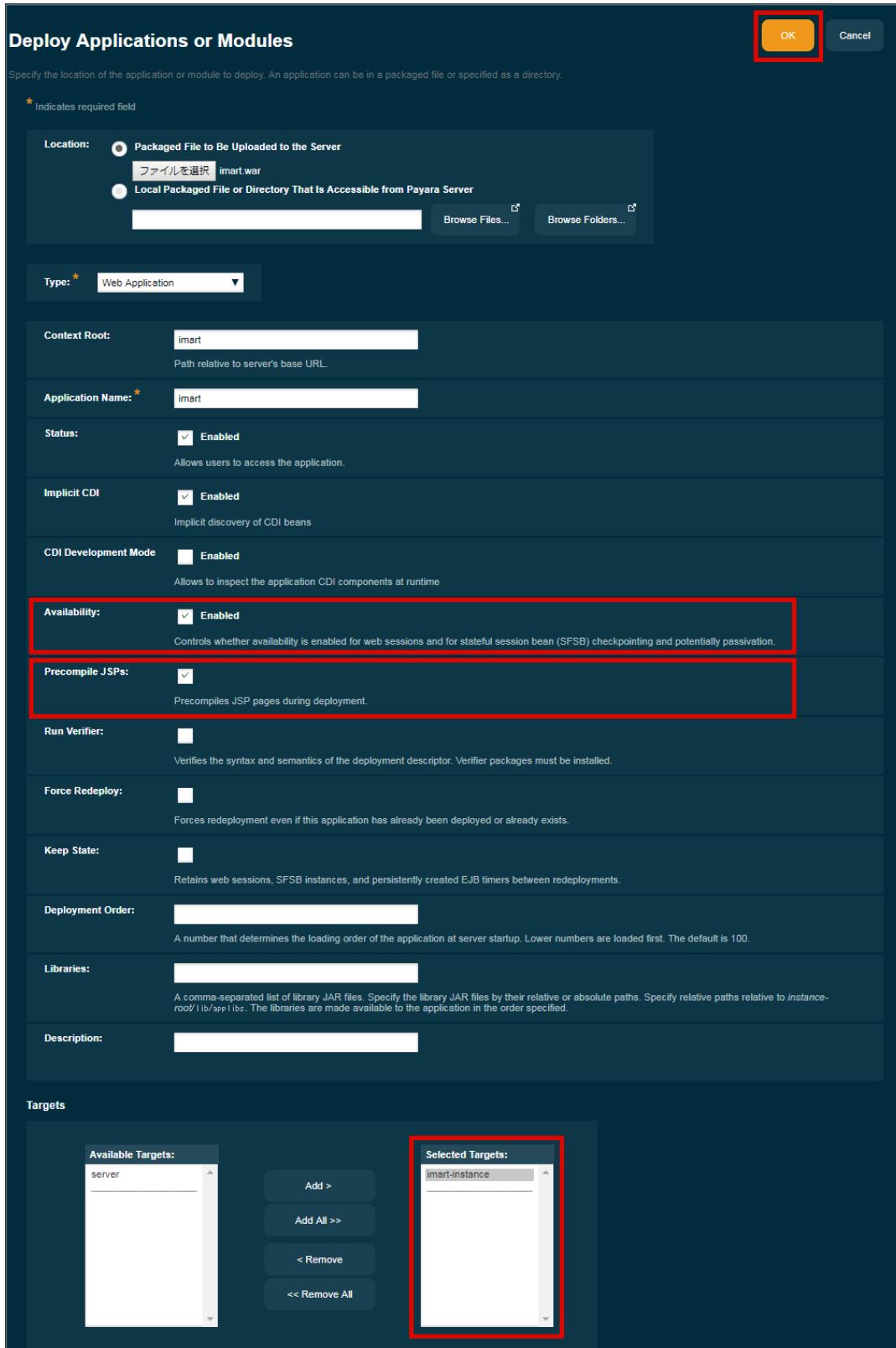
Deploy... Undeploy Enable Disable Filter: ▾

Select	Name	Deployment Order	Time Taken To Deploy (Milliseconds)	Date/Time When Application was deployed	Enabled	Engines	Action
No items found.							

4. [Location] に war ファイルを選択します。



5. 以下のように入力して [OK] を選択します。



Availability

[Enabled] を ON にします。

Precompile JSPs ON にします。

Targets 「[インスタンスの追加](#)」で作成したインスタンスを [Selected Targets] に設定します。

6. war がデプロイされました。

Select	Name	Deployment Order	Time Taken To Deploy (Milliseconds)	Date/Time When Application was deployed	Status	Engines	Action
<input type="checkbox"/>	imart	100	327	2018/07/09 17:04:35	Enabled on 1 of 1 Target(s)	web	Details

コラム

デプロイ完了後に %PAYARA_HOME%/glassfish/domains/domain1/logs/server.log に以下のログが出力されます。

```
| [2023-05-15T13:26:15.520+0900] [Payara 5.2022.5] [情報] [] [javax.enterprise.system.core] [tid: _ThreadID=203 _ThreadName=admin-thread-pool::admin-listener(3)] [timeMillis: 1684124775520] [levelValue: 800] [[
| imart was successfully deployed in 55,448 milliseconds.]]
```

コラム

デプロイが失敗した場合には、管理コンソールの Request Timeout に大きな値を設定することで回避できる場合があります。 Request Timeout は、左ペインのメニューから [Configurations] - [server-config] - [Network Config] - [Network Listeners] - [admin-listener] を選択し、HTTP タブで変更できます。

インスタンスの起動と停止

項目

- インスタンスの起動
- インスタンスの停止
- インスタンスの起動確認
- インスタンスのログ

インスタンスの起動と停止方法について記述します。



コラム

インスタンスの停止と起動によるアプリケーションの再起動を行った場合、以下の問題により非同期スレッドが実行されない可能性があります。

その場合、Payara Server 自体を再起動してください。

<https://github.com/payara/Payara/pull/2453>

インスタンスの起動

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Instances] を選択します。

Select	Name	LB Weight	Configuration	Node	Status
<input type="checkbox"/>	imart-instance	100	imart-config	localhost-domain1	St

3. 「[インスタンスの追加](#)」で追加したインスタンスを選択します。例として、ここでは [imart-instance] を選択します。

Server Instances (1)

Select	Name	LB Weight	Configuration	Node	Status
<input type="checkbox"/>	imart-instance	100	imart-config	localhost-domain1	St

4. [Start] を選択します。

General

Name: imart-instance
Status: Stopped ●
Restart Reason:
Uptime: (ms)
Deployment Group:
JVM: JVM Report
Node: localhost-domain1
Configuration: imart-config

5. 起動が完了すると Status に Running と表示されます。

User: admin Domain: domain1 Server: localhost

payara server 5

General Applications Resources Properties Monitoring Batch JMS Physical Destinations

General Information

Name: imart-instance
Status: **Running**

Restart Reason:

Uptime: 16599 (ms)

Deployment Group:

JVM: [JVM Report](#)

Node: localhost-domain1

Configuration: imart-config

インスタンスの停止

1. Payara の管理コンソールにアクセスします。アクセス方法は「[管理コンソールにアクセス](#)」を参照してください。
2. 左ペインのメニューから [Instances] を選択します。

User: admin Domain: domain1 Server: localhost

payara server 5

Common Tasks Domain Data Grid server (Admin Server) Deployment Groups Instances Nodes Clusters (Deprecated) Applications Lifecycle Modules Monitoring Data Resources Concurrent Resources Connectors JDBC JMS Resources JNDI JavaMail Sessions Resource Adapter Configs

Payara Server Instances

Create and manage Payara server instances. Running instances cannot be deleted.

Server Instances (1)					
	New...	Delete	Start	Stop	-- More Actions --
Select	Name	LB Weight	Configuration	Node	Status
<input type="checkbox"/>	imart-instance	100	imart-config	localhost-domain1	St

3. 「[インスタンスの追加](#)」で追加したインスタンスを選択します。例として、ここでは [imart-instance] を選択します。

The screenshot shows the Payara Server Instances management interface. On the left, a sidebar lists various server components like Domain, Deployment Groups, Instances, and Resources. The 'Instances' section is selected and highlighted in orange. In the main panel, a table titled 'Server Instances (1)' displays one row for 'imart-instance'. The table columns include Select, Name, LB Weight, Configuration, Node, and Status. The 'Name' column contains 'imart-instance', which is highlighted with a red box.

Select	Name	LB Weight	Configuration	Node	Status
<input type="checkbox"/>	imart-instance	100	imart-config	localhost-domain1	St

4. [Stop] を選択します。

The screenshot shows the 'General' tab of the configuration for 'imart-instance'. The 'Instances' item in the sidebar is selected. The main panel displays general information about the instance, including its name, status (Running), and deployment details. A red box highlights the 'Stop' button in the top navigation bar of the configuration panel.

Name:	imart-instance
Status:	Running
Restart Reason:	
Uptime:	506227 (ms)
Deployment Group:	
JVM:	JVM Report
Node:	localhost-domain1
Configuration:	imart-config

5. 停止が完了すると Status に Stopped と表示されます。

User: admin Domain: domain1 Server: localhost

payara server 5

General Applications Resources Properties Monitoring Batch JMS Physical Destinations

General Information

Name:	imart-instance
Status:	Stopped ●
Restart Reason:	
Uptime:	(ms)
Deployment Group:	
JVM:	JVM Report
Node:	localhost-domain1
Configuration:	imart-config

インスタンスの起動確認

- 左ペインのメニューから [Instances] を選択します。

User: admin Domain: domain1 Server: localhost

payara server 5

Common Tasks Domain Data Grid server (Admin Server) Deployment Groups Instances Nodes Clusters (Deprecated) Applications Lifecycle Modules Monitoring Data Resources Concurrent Resources Connectors JDBC JMS Resources JNDI JavaMail Sessions Resource Adapter Configs

Payara Server Instances

Create and manage Payara server instances. Running instances cannot be deleted.

Server Instances (1)					
Select	Name	LB Weight	Configuration	Node	Status
<input type="checkbox"/>	imart-instance	100	imart-config	localhost-domain1	St

- [imart-instance] を選択します。

The screenshot shows the Payara Server Instances management interface. On the left, a navigation sidebar lists categories like Common Tasks, Domain, Deployment Groups, Instances (which is selected), Nodes, Resources, and Applications. The main panel displays a table titled 'Server Instances (1)' with one row. The row contains columns for Select, Name (with 'imart-instance' highlighted in a red box), LB Weight (100), Configuration (imart-config), Node (localhost-domain1), and Status (Start). Buttons for New..., Delete, Start, Stop, and More Actions are at the top of the table.

3. [Properties] を選択します。

The screenshot shows the Properties tab for the 'imart-instance' server instance. The left sidebar remains the same. The main panel has tabs for General, Applications, Resources, and Properties (which is selected and highlighted in a red box). Below these tabs, there are two sub-tabs: System Properties and Instance Properties. The System Properties section shows the Instance Name: imart-instance. The Instance Properties section is currently empty. A 'Save' button is located in the top right corner of the properties area.

4. 以下のプロパティを確認します。

HTTP_LISTENER_PORT imart-instance が起動するポートです。

JAVA_DEBUGGER_PORT imart-instance の Java リモートデバッグのポートです。リモートデバッグを利用する場合はこのポートに接続してください。

Additional Properties (11)			
	Add Property	Delete Properties	
Select	Instance Variable Name	Current Value	Override Value
<input type="checkbox"/>	ASADMIN_LISTENER_PORT	24848	
<input type="checkbox"/>	HTTP_LISTENER_PORT	28080	
<input type="checkbox"/>	HTTP_SSL_LISTENER_PORT	28181	
<input type="checkbox"/>	IIOP_LISTENER_PORT	23700	
<input type="checkbox"/>	IIOP_SSL_LISTENER_PORT	23820	
<input type="checkbox"/>	IIOP_SSL_MUTUALAUTH_PORT	23920	
<input type="checkbox"/>	JAVA_DEBUGGER_PORT	29009	
<input type="checkbox"/>	JMS_PROVIDER_PORT	27676	

5. localhost:28080 ポートにアクセスします。

以下のような画面が表示されることを確認してください。

To replace this page, overwrite the file index.html in the document root folder of this server. The document root folder for this server is the docroot subdirectory of this server's domain directory.

To manage a server on the **local host** with the **default administration port**, [go to the Administration Console](#).

Payara Server Documentation

For more information about Payara Server, documentation and additional resources see the [Payara documentation](#)

Welcome to the Payara Community

As a member of the continuously growing and evolving open source community, you can benefit from some great support offered by other Payara Server users, and get involved yourself by posting feedback on the latest features and enhancements - or by making suggestions for future releases! And if you want to take it one step further, you can also become an official Payara Contributor. If you'd like to be a part of Payara Server's development, or simply get some help and advice from the community, [check out our Community page](#) for



コラム

コンテキストルート名 [imart] で war をデプロイした場合、以下の URL からログインできます。

- <http://localhost:28080/imart/login> (一般ユーザのログイン画面)
- <http://localhost:28080/imart/system/login> (システム管理者のログイン画面)



コラム

localhost の部分は Payara を実行しているサーバの IP アドレスに置き換えてください。

インスタンスのログ

以下のファイルに出力されます。

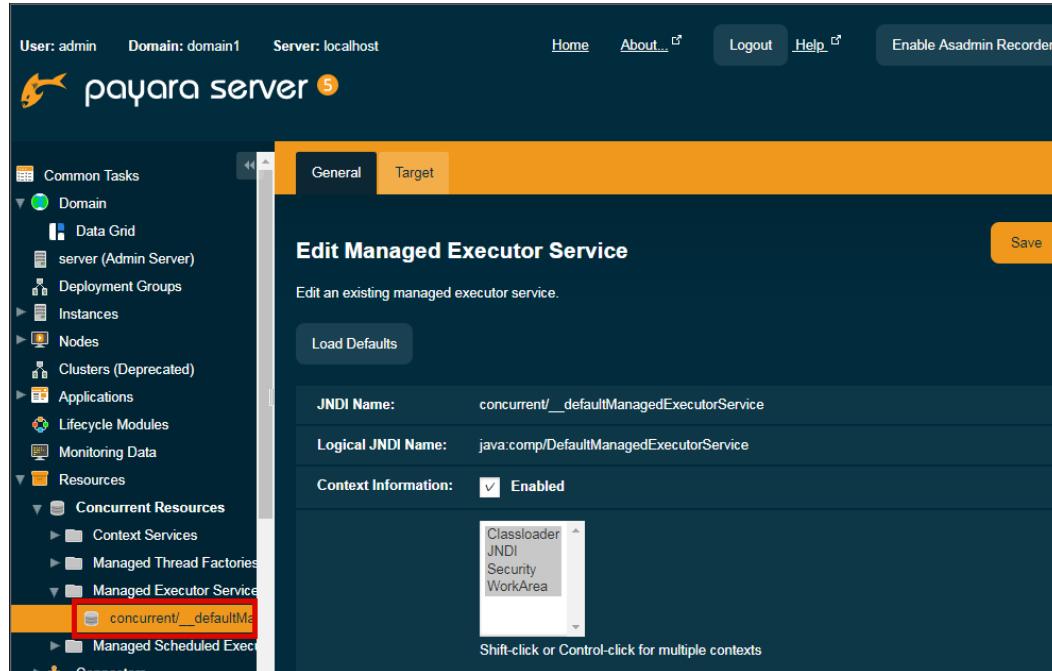
%PAYARA_HOME%/glassfish/nodes/localhost-domain1/imart-instance/logs/server.log

Payara Server では非同期処理のタスク実行エンジンとして Concurrency Utilities for Java EE を利用しています。 非同期処理機能では JNDI を通じて Concurrency Utilities の ExecutorService を取得しています。 本章では ExecutorService の設定を行います。

- 項目
- Executor Service の設定

Executor Service の設定

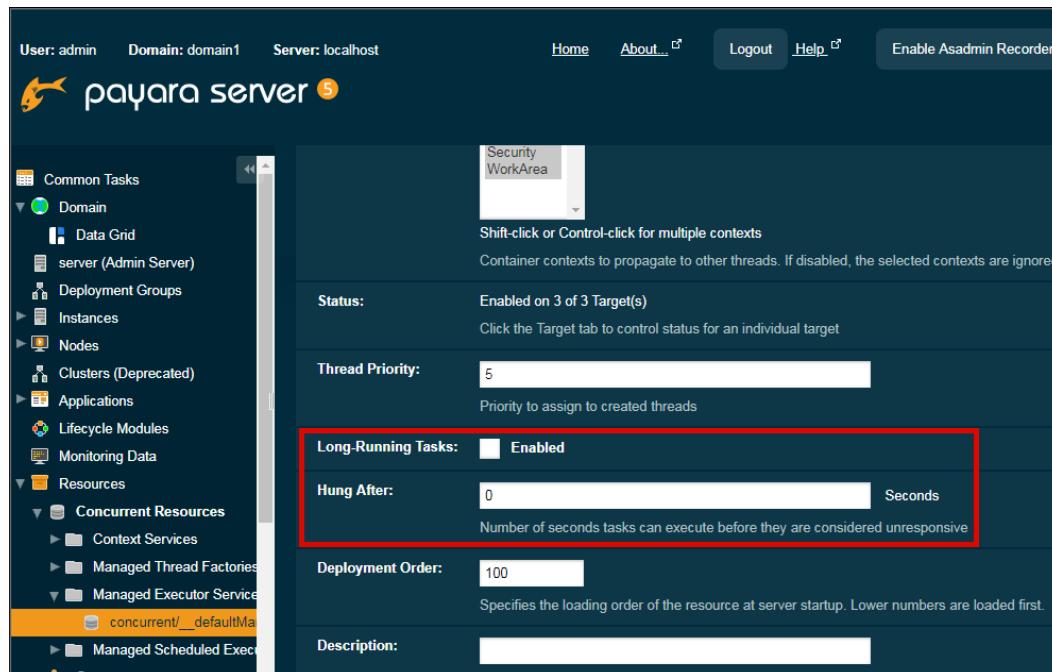
- 左ペインのメニューから [Resources] - [Concurrent Resources] - [Managed Executor Services] - [concurrent/_defaultManagedExecutorService] を選択します。



- 以下の設定を確認します。

Long-Running Tasks Hung After に 1 以上の値を設定する場合、選択して有効化してください。
Hung After に 0 を設定する場合、どちらでも構いません。

Hung After 0 を指定してください。





注意

Hung After に 1 以上の値を設定し、かつ Long-Running Tasks を無効化している場合、非同期スレッドが Hung After 秒以上実行後に以下のような警告が表示されます。

```
[2018-07-04T17:51:24.433+0900] [Payara 5.182] [警告] [AS-CONCURRENT-00001] [javax.enterprise.concurrent]
[tid: _ThreadID=327 _ThreadName=glassfish-internal-managedThreadFactory-Thread-1] [timeMillis:
1530694284433] [levelValue: 900] [[
Task [MyTask@61a84554] has been running on thread [concurrent/_defaultManagedExecutorService-
managedThreadFactory-Thread-1] for 52 seconds, which is more than the configured hung task threshold of 5
seconds in [concurrent/_defaultManagedExecutorService].]]
```

intra-mart Accel Platform のテナント環境を構築します。

具体的な手順は「[テナント環境セットアップ](#)」を参照してください。

付録

Payara のサービス化

Payara のサービス化手順について説明します。

項目

- サービスの登録
- サービスの削除
- サービスの起動
- サービスの終了

サービスの登録

1. create-service domain1 コマンドを実行します。

- Linux

```
$PAYARA_HOME/bin/asadmin create-service domain1
```

- Windows

```
%PAYARA_HOME%\bin\asadmin.bat create-service domain1
```



コラム

管理者権限で実行してください。



コラム

java へのパスが通っていない場合、以下の手順を参考にして、<%PAYARA_HOME%/glassfish/config/asenv.conf> ファイルに AS_JAVA の設定を追加します。

参考 : SSH ノードとなるサーバに Payara をインストールする

サービスの削除

- Linux

```
rm -f /etc/rc.d/init.d/payara_domain1  
rm -f /etc/rc.d/rc0.d/K20payara_domain1  
rm -f /etc/rc.d/rc1.d/K20payara_domain1  
rm -f /etc/rc.d/rc2.d/S20payara_domain1  
rm -f /etc/rc.d/rc3.d/S20payara_domain1  
rm -f /etc/rc.d/rc4.d/S20payara_domain1  
rm -f /etc/rc.d/rc5.d/S20payara_domain1  
rm -f /etc/rc.d/rc6.d/K20payara_domain1
```

- Windows

```
sc delete domain1
```



コラム

管理者権限で実行してください。

サービスの起動

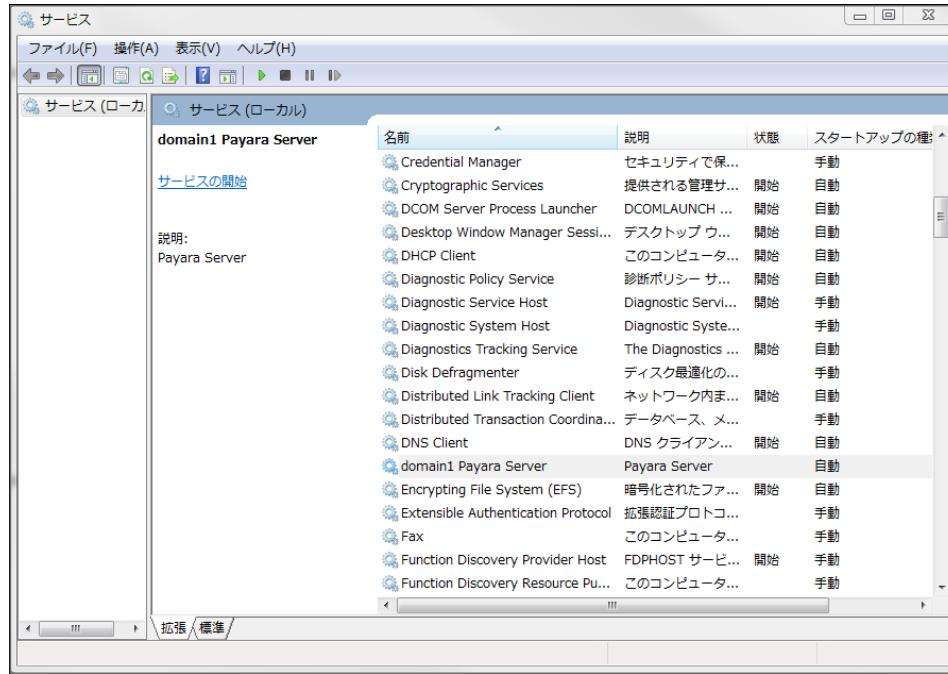
Linux

以下のコマンドを実行してください。

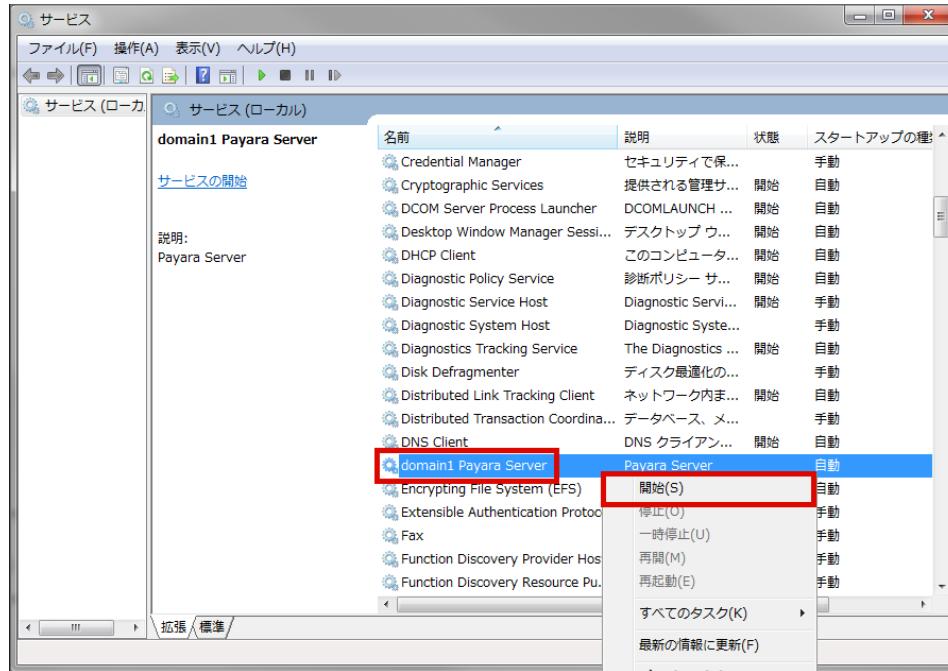
```
service payara_domain1 start
```

Windows

- サービス管理ツール(services.msc)を起動します。



- [domain1 Payara Server] を右クリックし、[開始] をクリックします。



サービスの終了

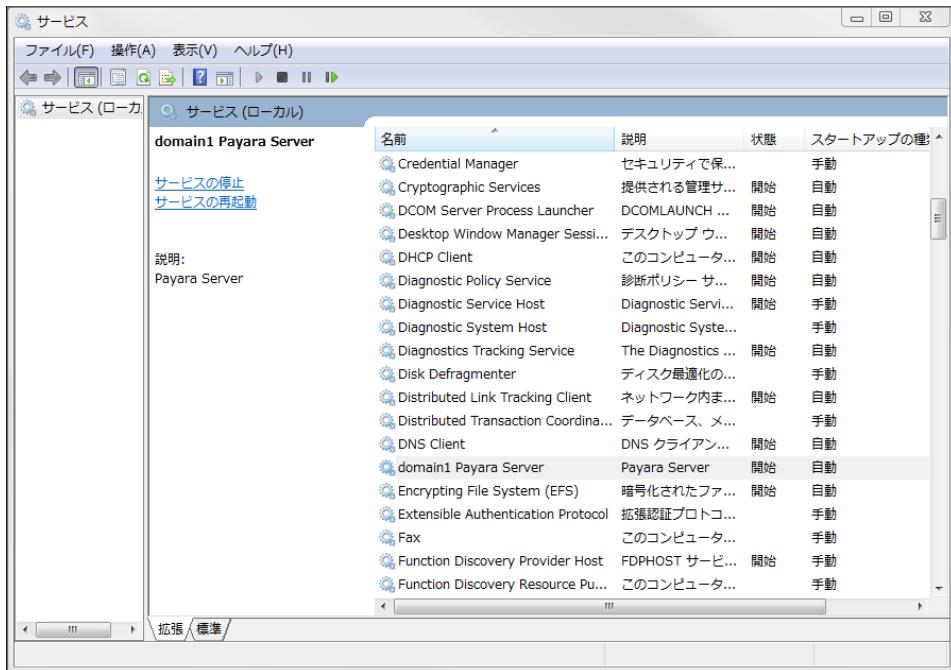
Linux

以下のコマンドを実行してください。

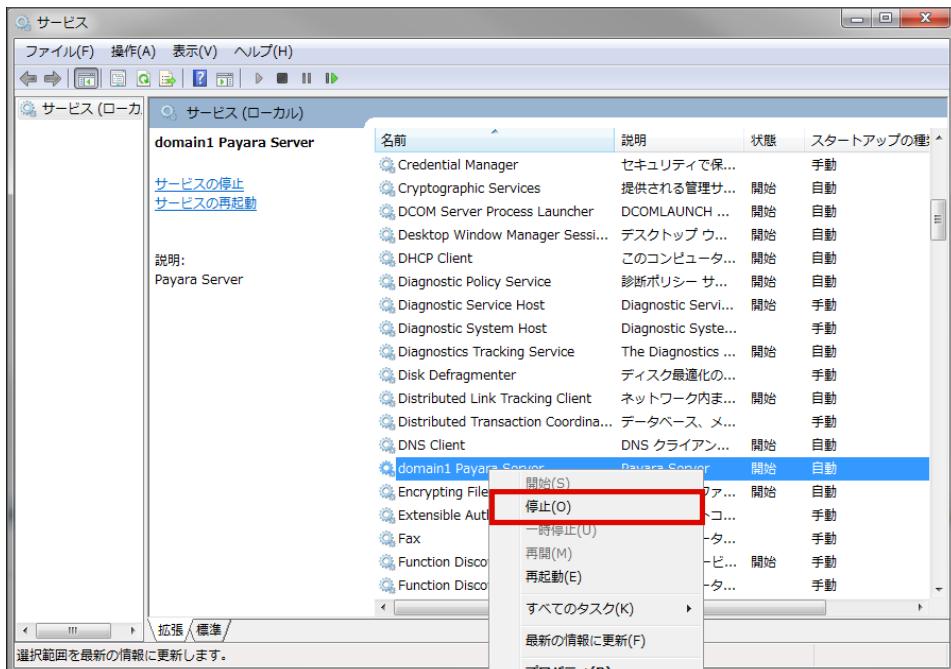
```
service payara_domain1 stop
```

Windows

- サービス管理ツール(services.msc)を起動します。



2. [domain1 Payara Server] を右クリックし、[停止] をクリックします。



NIC が複数ある場合

NIC が複数存在する環境の設定について説明します。

項目

- DAS Bind Address を設定する
- jgroups.bind_addr を設定する

DAS Bind Address を設定する

1. 左ペインのメニューから [Domain] - [Data Grid] を選択します。

User: admin Domain: domain1 Server: localhost

Home About... Logout Help Enable Admin Recorder

payara server 5

Common Tasks Domain Data Grid server (Admin Server) Deployment Groups Instances Nodes Clusters (Deprecated) Applications Lifecycle Modules Monitoring Data Resources Concurrent Resources Connectors JDBC JMS Resources JNDI JavaMail Sessions Resource Adapter Config

Data Grid Instances Configuration

Domain Wide Data Grid Configuration

Enable and configure the settings for the embedded Domain wide Data Grid within Payara.

Data Grid must be enabled to use the Hazelcast Web and EJB session persistence type in the [Availability Service](#).

Data Grid must be enabled to use the Hazelcast Web and EJB session persistence type in the Availability Service. Data Grid must be enabled to use [JCache \(JSR 107\)](#) capabilities and apis in your applications.

Restart Data Grid Applies the data grid configuration change immediately, without needing a restart of the Server. WARNING this can cause data grid wide data loss if a large number of instances restart the data grid immediately.

DAS Bind Address The network interface address the DAS Data Grid should bind to

DAS Public Address The network address that other instances should use to connect to the DAS. If left blank the value Public Bind Address will be used.

- [DAS Bind Address] に使用する NIC の IP アドレスを設定し、[Save] をクリックします。

User: admin Domain: domain1 Server: localhost

Home About... Logout Help Enable Admin Recorder

payara server 5

Common Tasks Domain Data Grid server (Admin Server) Deployment Groups Instances Nodes Clusters (Deprecated) Applications Lifecycle Modules Monitoring Data Resources Concurrent Resources Connectors JDBC JMS Resources JNDI JavaMail Sessions Resource Adapter Config

Data Grid Instances Configuration

Domain Wide Data Grid Configuration

Save

Enable and configure the settings for the embedded Domain wide Data Grid within Payara.

Data Grid must be enabled to use the Hazelcast Web and EJB session persistence type in the [Availability Service](#).

Data Grid must be enabled to use the Hazelcast Web and EJB session persistence type in the Availability Service. Data Grid must be enabled to use [JCache \(JSR 107\)](#) capabilities and apis in your applications.

Restart Data Grid Applies the data grid configuration change immediately, without needing a restart of the Server. WARNING this can cause data grid wide data loss if a large number of instances restart the data grid immediately.

DAS Bind Address The network interface address the DAS Data Grid should bind to

DAS Public Address The network address that other instances should use to connect to the DAS. If left blank the value for Public Bind Address will be used.

jgroups.bind_addr を設定する

- 左ペインのメニューから [Configurations] - [imart-config] を選択し、[System Properties] をクリックします。

User: admin Domain: domain1 Server: localhost

Home About... Logout Help Enable Admin Recorder

System Properties

Configuration Name: imart-config

Dynamic Reconfiguration:

Additional Properties (11)

Select	Instance Variable Name	Default Value
<input type="checkbox"/>	ASADMIN_LISTENER_PORT	24848
<input type="checkbox"/>	HTTP_LISTENER_PORT	28080
<input type="checkbox"/>	HTTPS_LISTENER_PORT	28481

Save

2. [Add Property] をクリックします。

User: admin Domain: domain1 Server: localhost

Home About... Logout Help Enable Admin Recorder

System Properties

Configuration Name: imart-config

Dynamic Reconfiguration:

Additional Properties (11)

Select	Instance Variable Name	Default Value
<input type="checkbox"/>	ASADMIN_LISTENER_PORT	24848
<input type="checkbox"/>	HTTP_LISTENER_PORT	28080
<input type="checkbox"/>	HTTPS_LISTENER_PORT	28481

Add Property Delete Properties

Save

3. 下記のように入力し、[Save] をクリックします。

Instance Variable Name	jgroups.bind_addr
Default Value	使用する NIC の IP アドレスを設定します。 例として、ここでは 172.16.0.2 と入力します。

The screenshot shows the Payara Admin Console interface. On the left, there's a sidebar with various service configurations like Admin Service, Data Grid, Request Tracing, etc. The main area is titled 'System Properties' for the 'imart-config' configuration. It shows a table of properties with columns for 'Select', 'Instance Variable Name', and 'Default Value'. The 'jgroups.bind_addr' property is highlighted with a red box, showing its value as '172.16.0.2'. A large orange 'Save' button is located at the top right of the form.

Select	Instance Variable Name	Default Value
<input type="checkbox"/>	jgroups.bind_addr	172.16.0.2
<input type="checkbox"/>	ASADMIN_LISTENER_PORT	24848
<input type="checkbox"/>	HTTP_LISTENER_PORT	28080

Payara のクラスタリング

Payara のクラスタリング環境の構築手順について説明します。

項目

- DAS ノードとなるサーバに Payara をインストールする
- SSH ノードとなるサーバに Payara をインストールする
- DAS ノードの Payara を起動する
- DAS ノードの Payara の管理コンソールにアクセスする
- SSH ノードの設定を行う
- SSH ノードインスタンスを作成する
- Deployment Group を作成する
- Deployment Group を終了する
- Deployment Group を起動する
- Domain Data Grid を確認する
- デプロイメントグループを JDBC リソースの対象に設定する
- アプリケーションをデプロイする
- インスタンスの起動確認

本章では以下の 2 台構成でセットアップを行う手順について説明します。

	マシン1	マシン2
プライベートIPアドレス	172.16.0.2	172.16.0.3
構成	DAS + インスタンス1 インスタンス2	
名称	DAS ノード	SSH ノード

本章では、マシン1を「DAS ノード」、マシン2「SSH ノード」を呼びます。

DAS ノードとなるサーバに Payara をインストールする

1. DAS ノードとなるサーバに以下のセットアップを実施します。

- Payara のインストール
- Payara の設定



コラム

DAS ノードとなるサーバにはスタンダードアローンと同様のセットアップを実施します。

SSH ノードとなるサーバに Payara をインストールする

- SSH ノードとなるサーバに以下のセットアップを実施します。

- [Payara をインストールする](#)



コラム

SSH ノードとなるサーバには Payara のインストールと JDBC ドライバの配置を実施します。



コラム

SSH ノード上の Payara の起動や停止は DAS ノードとなるサーバから実施します。そのため、このステップでは Payara と JDBC ドライバの配置のみを行ってください。

- <%PAYARA_HOME%/glassfish/config/asenv.conf> ファイルに AS_JAVA の設定を追加します。

```
...  
AS_IMQ_LIB="../../../mq/lib"  
AS_IMQ_BIN="../../../mq/bin"  
AS_CONFIG="../../../config"  
AS_INSTALL.."."  
AS_DEF_DOMAINS_PATH="../domains"  
AS_DEF_NODES_PATH="../nodes"  
AS_DERBY_INSTALL="../../../javadb"  
AS_H2_INSTALL="../../../h2db"  
AS_JAVA="/usr/local/jdk"
```



コラム

AS_JAVA に指定する Java のパスには、\$AS_JAVA/bin/java ファイルが存在するようなパスを設定してください。

例として、ここでは /usr/local/jdk と設定します。

この設定により、/usr/local/jdk/bin/java が利用されます。

DAS ノードの Payara を起動する

- DAS ノードとなるサーバを起動します。以下の手順を参考にしてください。

- [Payara の起動と停止](#)

DAS ノードの Payara の管理コンソールにアクセスする

- DAS ノードとなるサーバの管理コンソールを開きます。以下の手順を参考にしてください。

- [管理コンソールにアクセス](#)

SSH ノードの設定を行う

- 左ペインのメニューから [Nodes] を選択し、[New] をクリックします。

- 下記のように入力し、[OK] をクリックします。

Name	ノードの名前を入力します。 例として、ここでは remote-node1 と入力します。
Type	SSH
Node Host	172.16.0.3
Node Directory	未入力で構いません。
Installation Directory	SSH ノードとなるサーバ 上の Payara のインストールパスを入力します。 例として、ここでは /var/payara と入力します。
Install Payara Server	選択しないでください。
Force	選択しないでください。
SSH Port	22

SSH User Name	root
SSH User Authentication	Password
SSH User Password	root ユーザのパスワードを入力します。
Configuration	imart-config を選択してください。 Reference the Selected Configuration を選択してください。



コラム

SSH Port, SSH User Name, SSH User Authentication 等の SSH の設定については、DAS ノードとなるサーバから SSH ノードとなるサーバへのアクセスが可能な SSH の情報を設定してください。公開鍵認証の場合 SSH User Authentication は Password ではなく Key File を選択してください。この場合、Key File には鍵ファイルへのパスを指定してください。



コラム

SSH User Name には、SSH ノード上の Payara ディレクトリ、ストレージディレクトリ等へのアクセス権があるユーザを指定してください。

User: admin Domain: domain1 Server: 172.16.0.2

New Node

Create a node on which Payara Server instances can reside.

Name*: remote-node1

Type: SSH

Node Host: 172.16.0.3

Node Directory:

OK

User: admin Domain: domain1 Server: 172.16.0.2

New Node

Create a node on which Payara Server instances can reside.

Installation Directory: /var/payara

Install Payara Server: Enabled

SSH

Force: Enabled

SSH Port: 22

SSH User Name: root

OK

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

payara server 5

Common Tasks

- Domain
 - Data Grid
 - server (Admin Server)
- Deployment Groups
- Instances
- Nodes**
 - localhost-domain1
 - Clusters (Deprecated)
 - Applications
 - Lifecycle Modules
 - Monitoring Data
- Resources
- Concurrent Resources
- Connectors
- JDBC
- JMS Resources
- JNDI
- JavaMail Sessions
- Resource Adapter Configs

SSH

Force: Enabled
Specifies whether the node is created even if validation of the node's parameters fails.

SSH Port:
The port to use for SSH connections to this node's host. The default is 22.

SSH User Name:
The user that is to run the process for connecting to this node's host through SSH. The default user that is running the DAS process.

SSH User Authentication:

Select how the SSH user is authenticated when logging in to this node's host.

SSH User Password:
Type the password that the SSH user will use for logging in to the host that this node represents.

OK

3. SSH ノードを登録できました。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

payara server 5

Common Tasks

- Domain
 - Data Grid
 - server (Admin Server)
- Deployment Groups
- Instances
- Nodes**
 - New...
 - Delete
 - Delete and Uninstall
- Clusters (Deprecated)
- Applications
- Lifecycle Modules
- Monitoring Data

Nodes

A node represents a host on which the Payara Server software is installed. A node must exist for every host on which Payara Server resides.

Select	Name	Node Host	Type	Instances
<input type="checkbox"/>	localhost-domain1	localhost	CONFIG	imart-instance Running ✓
<input type="checkbox"/>	remote-node1	172.16.0.3	SSH	

4. [Ping] をクリックすることで、SSH の疎通確認を行えます。

The screenshot shows the Payara Server Admin Console interface. The left sidebar menu is expanded, showing 'Common Tasks', 'Domain' (with 'Data Grid', 'server (Admin Server)', 'Deployment Groups', 'Instances'), 'Nodes' (selected), 'Clusters (Deprecated)', 'Applications', 'Lifecycle Modules', 'Monitoring Data', 'Resources' (with 'Concurrent Resources', 'Connectors', 'JDBC', 'JMS Resources', 'JNDI', 'JavaMail Sessions', 'Resource Adapter Configs'), and 'Instances'. The main content area has a heading 'Payara Server Instances' with the sub-instruction 'Create and manage Payara server instances. Running instances cannot be deleted.' Below this is a table titled 'Server Instances (1)'. The table has columns: Select, Name, LB Weight, Configuration, Node, and Status. It contains one row for 'imart-instance' with LB Weight 100, Configuration 'imart-config', Node 'localhost-domain1', and Status 'Running'. A red box highlights the 'Ping' button in the 'Action' column for the 'remote-node1' row. Above the table, a message box says 'Successfully made SSH connection to node remote-node1 (172.16.0.3)' with a green checkmark icon.

SSH ノードインスタンスを作成する

- 左ペインのメニューから [Instances] を選択し、[New] をクリックします。

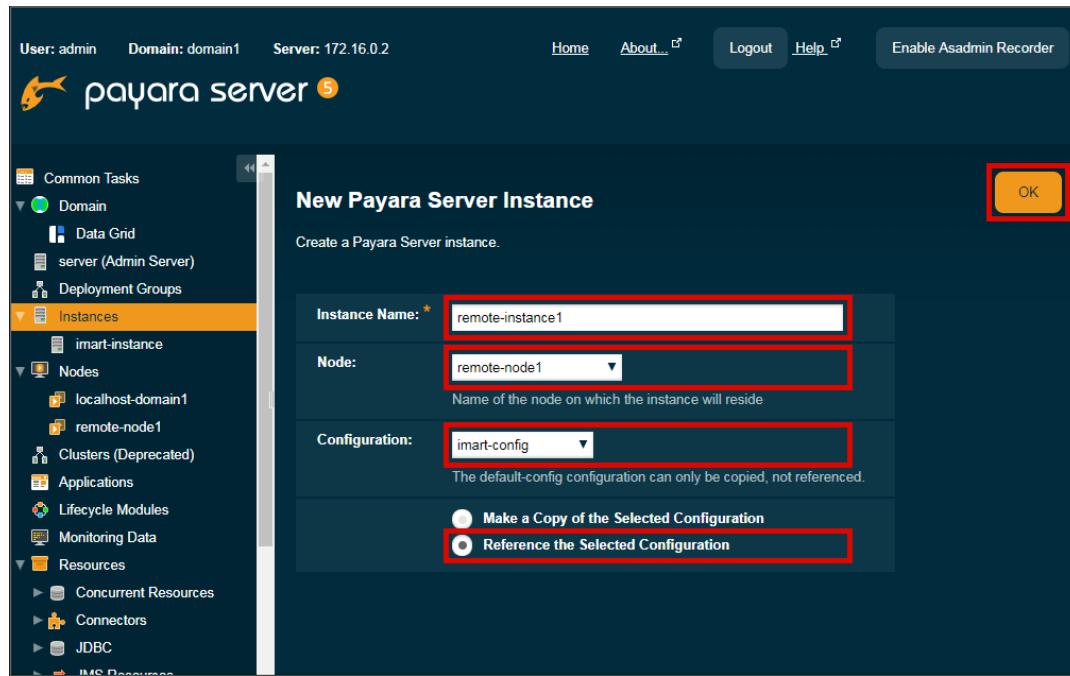
The screenshot shows the Payara Server Admin Console interface. The left sidebar menu is expanded, showing 'Common Tasks', 'Domain' (with 'Data Grid', 'server (Admin Server)', 'Deployment Groups'), 'Instances' (selected), 'Nodes', 'Clusters (Deprecated)', 'Applications', 'Lifecycle Modules', 'Monitoring Data', 'Resources' (with 'Concurrent Resources', 'Connectors', 'JDBC', 'JMS Resources', 'JNDI', 'JavaMail Sessions', 'Resource Adapter Configs'), and 'Instances'. The main content area has a heading 'Payara Server Instances' with the sub-instruction 'Create and manage Payara server instances. Running instances cannot be deleted.' Below this is a table titled 'Server Instances (1)'. The table has columns: Select, Name, LB Weight, Configuration, Node, and Status. It contains one row for 'imart-instance' with LB Weight 100, Configuration 'imart-config', Node 'localhost-domain1', and Status 'Running'. A red box highlights the 'New...' button in the top toolbar of the 'Server Instances' section.

- 以下のように入力し、[OK] をクリックします。

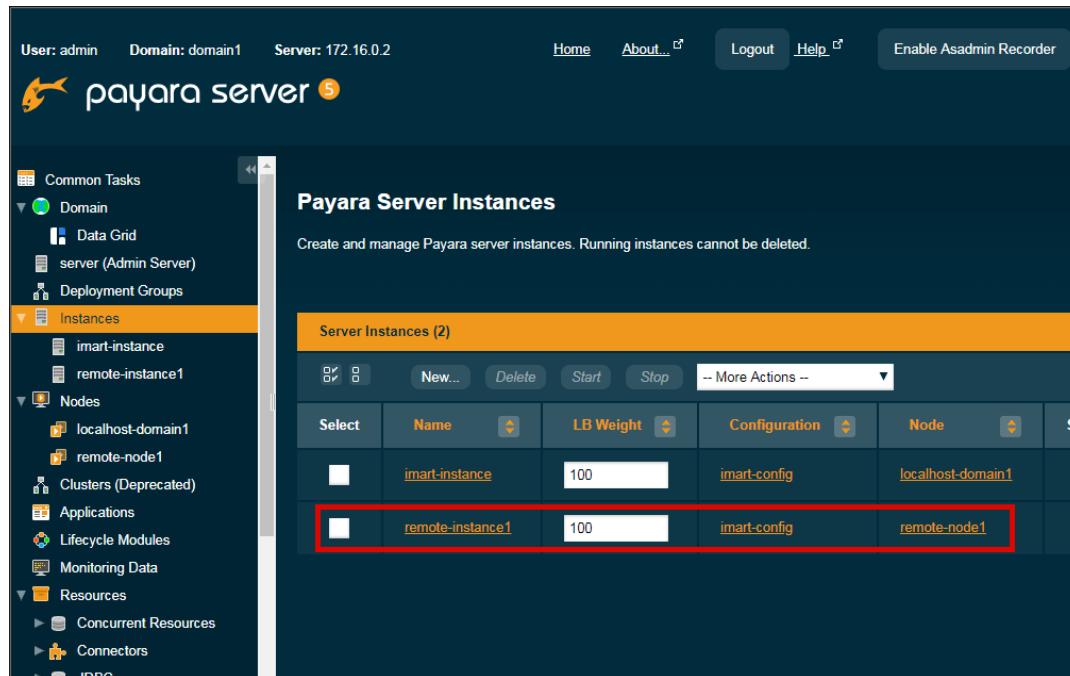
Instance Name インスタンスの名前を入力します。
例として、ここでは `remote-instance1` と入力します。

Node `remote-node1`

Configuration `imart-config` を選択してください。
Reference the Selected Configuration を選択してください。



3. インスタンスを登録できました。



Deployment Group を作成する

- 左ペインのメニューから [Deployment Groups] を選択し、[New] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Asadmin Recorder

Deployment Groups

Create and manage Payara Server deployment groups. A deployment group is a loose collection of Instances used for deployment and lifecycle control.

Deployment Groups (0)		
	New...	Delete Start Deployment Group Stop Deployment Group -- More Actions --
Select	Name	Instances
No items found.		

2. 以下のように入力し、[OK] をクリックします。

Deployment Group Name	デプロイメントグループの名前を入力します。 例として、ここでは imart-deployment-group と入力します。
Selected Instances	[imart-instance] と [remote-instance1] を追加します。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Asadmin Recorder

New Deployment Group

Creates a new Deployment Group

Deployment Group Name *	
<input type="text" value="imart-deployment-group"/>	

Available Instances	
<input type="button" value="Add >"/> <input type="button" value="Add All >>"/> <input type="button" value="< Remove"/> <input type="button" value="<< Remove All"/>	

Selected Instances	
<input type="text" value="imart-instance"/> <input type="text" value="remote-instance1"/>	

Server Instances to Be Created (0)			
New...	Delete		
Select	Instance Name	Weight	Node
No items found.			

3. デプロイメントグループを登録できました。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Deployment Groups

Create and manage Payara Server deployment groups. A deployment group is a loose collection of Instances used for deployment and lifecycle control.

Deployment Groups (1)			
	New...	Delete	Start Deployment Group
Select	Name	Instances	
<input type="checkbox"/>	imart-deployment-group	imart-instance	Running
		remote-instance1	Stopped

Deployment Group を終了する

- 左ペインのメニューから [Deployment Groups] を選択し、[imart-deployment-group] を選択し、[Stop Deployment Group] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Deployment Groups

Create and manage Payara Server deployment groups. A deployment group is a loose collection of Instances used for deployment and lifecycle control.

Deployment Groups (1)			
	New...	Delete	Start Deployment Group
<input checked="" type="checkbox"/>	imart-deployment-group	imart-instance	Running
		remote-instance1	Stopped

- デプロイメントグループを終了できました。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Deployment Groups

Create and manage Payara Server deployment groups. A deployment group is a loose collection of Instances used for deployment and lifecycle control.

Deployment Groups (1)						
	New...	Delete Start Deployment Group Stop Deployment Group -- More Actions --				
Select	Name	Instances				
<input type="checkbox"/>	imart-deployment-group	<table border="1"> <tr> <td>imart-instance</td> <td>● Stopped</td> </tr> <tr> <td>remote-instance1</td> <td>● Stopped</td> </tr> </table>	imart-instance	● Stopped	remote-instance1	● Stopped
imart-instance	● Stopped					
remote-instance1	● Stopped					

Deployment Group を起動する

- 左ペインのメニューから [Deployment Groups] を選択し、[imart-deployment-group] を選択し、[Start Deployment Group] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Deployment Groups

Create and manage Payara Server deployment groups. A deployment group is a loose collection of Instances used for deployment and lifecycle control.

Deployment Groups (1)						
	New...	Delete Start Deployment Group Stop Deployment Group -- More Actions --				
Select	Name	Instances				
<input checked="" type="checkbox"/>	imart-deployment-group	<table border="1"> <tr> <td>imart-instance</td> <td>● Stopped</td> </tr> <tr> <td>remote-instance1</td> <td>● Stopped</td> </tr> </table>	imart-instance	● Stopped	remote-instance1	● Stopped
imart-instance	● Stopped					
remote-instance1	● Stopped					

- デプロイメントグループを開始できました。

User: admin Domain: domain1 Server: 172.16.0.2 Home About... Logout Help Enable Admin Recorder

Deployment Groups

Create and manage Payara Server deployment groups. A deployment group is a loose collection of Instances used for deployment and lifecycle control.

Select	Name	Instances
<input type="checkbox"/>	imart-deployment-group	imart-instance remote-instance1
		✓ Running ✓ Running

Domain Data Grid を確認する

- 左ペインのメニューから [Domain] - [Data Grid] を選択し、[Data Grid Instances] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2 Home About... Logout Help Enable Admin Recorder

Data Grid Instances Configuration

Domain Wide Data Grid Configuration

Enable and configure the settings for the embedded Domain wide Data Grid within Payara. Data Grid must be enabled to use the Hazelcast Web and EJB session persistence type in the Availability Service. Data Grid must be enabled to use the Hazelcast Web and EJB session persistence type in the Availability Service. Data Grid can use JCache (JSR 107) capabilities and apis in your applications.

Restart Data Grid <input type="checkbox"/>	Applies the data grid configuration change immediately, without needing a restart of the Server. WARNING this can cause data grid wide data loss if a large number of instances restart the data grid immediately.
DAS Bind Address <input type="text"/>	The network interface address the DAS Data Grid should bind to
DAS Public Address <input type="text"/>	The network address that other instances should use to connect to the DAS. If left blank the value Public Bind Address will be used.

- [imart-instance] と [remote-instance1] がクラスタリングされていることを確認できます。

User: admin Domain: domain1 Server: 172.16.0.2

Data Grid Instances Configuration

Data Grid Instances

A list of the Data Grid Instances visible to this domain

All Data Grid Instances (3)						
Name	Type	Group	Host	HTTP Ports	HTTPS Ports	Admin Port
remote-instance1	INSTANCE	imart-config	/172.16.0.3	28081	28182	24849
imart-instance	INSTANCE	imart-config	/172.16.0.2	28080	28181	24848
server	DAS	server-config	/172.16.0.2	8080	8181	4848

User: admin Domain: domain1 Server: 172.16.0.2

Logout Help Enable Admin Recorder

payara server 5

Common Tasks Domain Data Grid

- server (Admin Server)
- Deployment Groups
- Instances
- Nodes
 - localhost-domain1
 - remote-node1
- Clusters (Deprecated)
- Applications
- Lifecycle Modules
- Monitoring Data
- Resources
 - Concurrent Resources
 - Connectors
 - JDBC
 - JMS Resources
- INCI

Payara Server Instances (3)

Instance Name	Group	Host Name	HTTP Ports	HTTPS Ports	Admin Port
remote-instance1	imart-config	/172.16.0.3	28081	28182	24849
imart-instance	imart-config	/172.16.0.2	28080	28181	24848
server	server-config	/172.16.0.2	8080	8181	4848

Payara Micro Instances (0)

Command: Send Asadmin Command

Select	Instance Name	Group	Host Name	HTTP Ports	HTTPS Ports	Admin Port	Hazelcast Port	Lite Mode
No Micro instances in the cluster								



コラム

Payara 5 の Domain Data Grid 機能として、DAS ノード(Instance Name=server)もクラスタリングに参加します。

デプロイメントグループを JDBC リソースの対象に設定する

- 左ペインのメニューから [Resources] - [JDBC] - [JDBC Resources], 対象となる JDBC リソースをクリックします。例として、ここでは jdbc/postgresql を選択します。選択後、[Target] - [Manage Targets] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Resource Targets

List resource targets.

JNDI Name: jdbc/postgresql

Targets (2)

Select	Target Name	Enabled
<input type="checkbox"/>	server	✓
<input type="checkbox"/>	imart-instance	✓

Manage Targets... Enable Disable

- [imart-deployment-group] を [Selected Targets] に追加します。その他は必要に応じて追加します。追加後、[Save] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Manage Resource Targets

Manage the targets (clusters and stand-alone server instances) on which the resource is available.

JNDI Name: jdbc/postgresql

Targets

Available Targets:

- jdbc/_TimerPool
- jdbc/_default
- jdbc/_derby
- jdbc/postgresql

Add > Add All >> < Remove << Remove All

Selected Targets:

- server
- imart-instance
- remote-instance1
- imart-deployment-group

- デプロイメントグループを JDBC リソースの対象に追加できました。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Asadmin Recorder

Resource Targets

List resource targets.

JNDI Name: jdbc/postgresql

Targets (2)

Select	Target Name	Enabled
<input type="checkbox"/>	server	✓
<input type="checkbox"/>	imart-deployment-group	✓

アプリケーションをデプロイする

- 左ペインのメニューから [Applications] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Asadmin Recorder

Applications

Applications can be enterprise or web applications, or various kinds of modules. Restart an application or module by clicking on it. This action will apply only to the targets that the application or module is enabled on.

Deployed Applications (0)

Select	Name	Deployment Order	Time Taken To Deploy (Milliseconds)	Date/Time When Application was deployed	Status	Engines
No items found.						

- [Deploy] をクリックします。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Applications

Applications can be enterprise or web applications, or various kinds of modules. Restart an application or module by clicking on its name. This action will apply only to the targets that the application or module is enabled on.

Deployed Applications (0)						
		Deploy...	Undeploy	Filter:		
Select	Name	Deployment Order	Time Taken To Deploy (Milliseconds)	Date/Time When Application was deployed	Status	Engines
No items found.						

3. 以下のように入力します。

Location	[ファイルを選択] より、war ファイルを選択してアップロードします。 または、war ファイルを DAS ノードとなるサーバに別途アップロードし、Browse Files よりアップロードした war ファイルを選択します。
Type	Web Application を選択します。
Context Root	コンテキストルート名を入力します。例として、ここでは imart を設定します。
Application Name	アプリケーション名を入力します。例として、ここでは imart を設定します。
Availability	選択します。Availability を有効化しない場合、セッションレプリケーションを行えません。
Precompile JSPs	選択します。

User: admin Domain: domain1 Server: 172.16.0.2

Home About... Logout Help Enable Admin Recorder

Deploy Applications or Modules

Specify the location of the application or module to deploy. An application can be in a packaged file or specified as a directory.

* Indicates required field

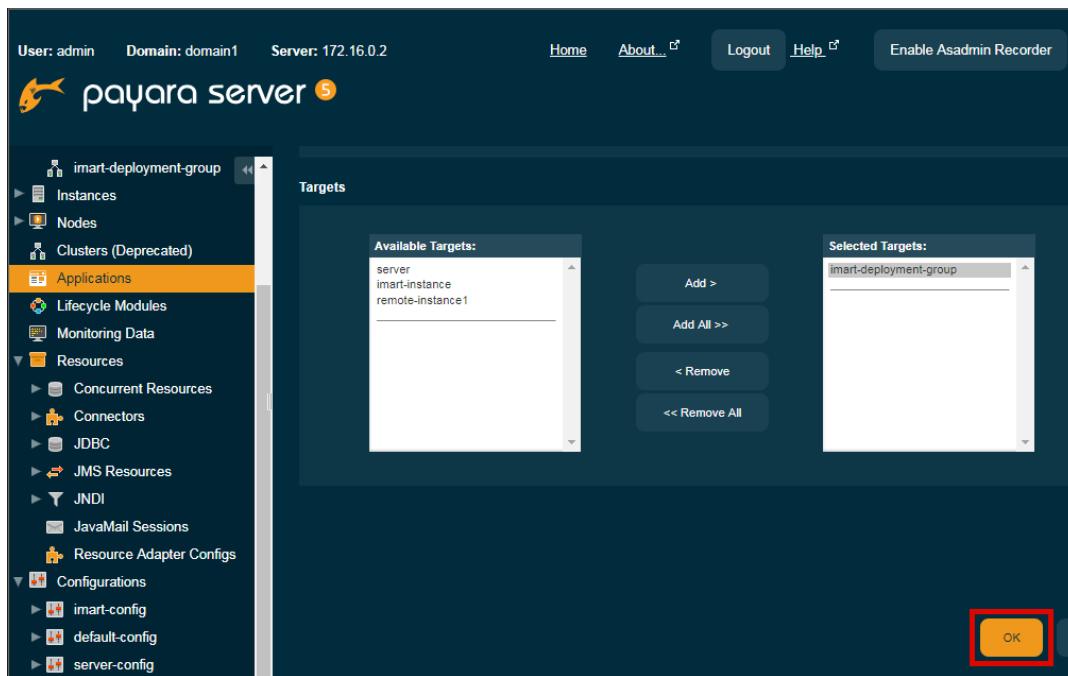
Location:	<input checked="" type="radio"/> Packaged File to Be Uploaded to the Server <input type="radio"/> Local Packaged File or Directory That Is Accessible from Payara Server
Type:	<input checked="" type="radio"/> Web Application
Context Root:	imart
Application Name:	imart
Status:	Enabled

The screenshot shows the Payara Server Admin Console interface. On the left, a sidebar menu lists various server components: Common Tasks, Domain (Data Grid, server (Admin Server)), Deployment Groups, Instances, Nodes, Clusters (Deprecated), Applications (selected), Lifecycle Modules, Monitoring Data, Resources (Concurrent Resources, Connectors, JDBC, JMS Resources, JNDI, JavaMail Sessions, Resource Adapter Configs), and Configurations (imart-config, default-config, server-config). The main panel displays deployment configuration for the selected 'Applications' node. It includes sections for Status (Enabled), Implicit CDI (Enabled), CDI Development Mode (Enabled), Availability (Enabled, highlighted with a red box), Precompile JSPs (Enabled, highlighted with a red box), Run Verifier (unchecked), and Force Redeploy (unchecked). A 'Keep State' button is at the bottom.

4. [Selected Targets] に [imart-deployment-group] を追加します。

The screenshot shows the Payara Server Admin Console interface. The sidebar menu is identical to the previous screenshot. The main panel shows the 'Targets' configuration for the 'Applications' node. It includes fields for Keep State (unchecked), Deployment Order (empty), Libraries (empty), and Description (empty). Below these, the 'Targets' section shows two lists: 'Available Targets' (server, imart-instance, remote-instance1) and 'Selected Targets' (imart-deployment-group, highlighted with a red box). Buttons for 'Add >', 'Add All >>', and '< Remove' are visible between the lists.

5. [OK] をクリックし、デプロイします。



6. デプロイできました。

The screenshot shows the 'Applications' list in the Payara Admin Console. The 'Deployed Applications (1)' table displays the following information for the 'imart' application:

Select	Name	Deployment Order	Time Taken To Deploy (Milliseconds)	Date/Time When Application was deployed	Status	Engines	Logs
<input type="checkbox"/>	imart	100	685	2018/06/29 17:47:50	Enabled on 1 of 1 Target(s)	web	Logs

インスタンスの起動確認

1. 「[インスタンスの起動確認](#)」と同様の手順でインスタンスが起動しているポートを確認します。

User: admin Domain: domain1 Server: 172.16.0.2 Home About... Logout Help Enable Admin Recorder

payara server 5

Common Tasks Domain Deployment Groups Instances **imart-instance** remote-instance1 Nodes Clusters (Deprecated) Applications Lifecycle Modules Monitoring Data Resources Concurrent Resources Connectors JDBC JMS Resources INDI

General Applications Resources Properties Monitoring Batch JMS Physical Destinations

System Properties Instance Properties

System Properties

Instance Name: imart-instance

Additional Properties (11)

Select	Instance Variable Name	Current Value	Override Value
<input type="checkbox"/>	ASADMIN_LISTENER_PORT	24848	
<input type="checkbox"/>	HTTP_LISTENER_PORT	28080	

Save



コラム

DAS ノードとなるサーバ上で 28080 ポートで起動していることが確認できます。

User: admin Domain: domain1 Server: 172.16.0.2 Home About... Logout Help Enable Admin Recorder

payara server 5

Common Tasks Domain Deployment Groups Instances **remote-instance1** imart-instance Nodes Clusters (Deprecated) Applications Lifecycle Modules Monitoring Data Resources Concurrent Resources Connectors JDBC JMS Resources INDI

General Applications Resources Properties Monitoring Batch JMS Physical Destinations

System Properties Instance Properties

System Properties

Instance Name: remote-instance1

Additional Properties (11)

Select	Instance Variable Name	Current Value	Override Value
<input type="checkbox"/>	ASADMIN_LISTENER_PORT	24849	24849
<input type="checkbox"/>	HTTP_LISTENER_PORT	28081	28081



コラム

SSH ノードとなるサーバ上で 28081 ポートで起動していることが確認できます。

2. 上記の手順で確認できた以下の二つの URL に対してロードバランスを行ってください。

- <http://172.16.0.2:28080/imart>
- <http://172.16.0.3:28081/imart>



注意

Payara のクラスタリング環境を運用する場合は、必ず Web Server やロードバランサをステイックセッションとし、同一セッションのリクエストが同じ Payara に送信されるように設定してください。

Apache Solr

Apache Solr は、「IM-ContentsSearch」を利用するために必要です。



コラム

プロジェクトの作成とモジュールの選択で IM-ContentsSearch アプリケーションを選択しない場合、Apache Solr のセットアップは不要です。

- ・ セットアップで困ったら . . .
- ・ アップデート パッチの適用