



プロダクション内からの外部 メッセージング・プラットフォーム の使用方法

Version 2023.1

2024-01-02

プロダクション内からの外部メッセージング・プラットフォームの使用法

InterSystems IRIS Data Platform Version 2023.1 2024-01-02

Copyright © 2024 InterSystems Corporation

All rights reserved.

InterSystems®, HealthShare Care Community®, HealthShare Unified Care Record®, IntegratedML®, InterSystems Caché®, InterSystems Ensemble®, InterSystems HealthShare®, InterSystems IRIS®, および TrakCare は、InterSystems Corporation の登録商標です。HealthShare® CMS Solution Pack™ HealthShare® Health Connect Cloud™, InterSystems IRIS for Health™, InterSystems Supply Chain Orchestrator™, および InterSystems TotalView™ For Asset Management は、InterSystems Corporation の商標です。TrakCare は、オーストラリアおよび EU における登録商標です。

ここで使われている他の全てのブランドまたは製品名は、各社および各組織の商標または登録商標です。

このドキュメントは、インターシステムズ社(住所: One Memorial Drive, Cambridge, MA 02142)あるいはその子会社が所有する企業秘密および秘密情報を含んでおり、インターシステムズ社の製品を稼働および維持するためにのみ提供される。この発行物のいかなる部分も他の目的のために使用してはならない。また、インターシステムズ社の書面による事前の同意がない限り、本発行物を、いかなる形式、いかなる手段で、その全てまたは一部を、再発行、複製、開示、送付、検索可能なシステムへの保存、あるいは人またはコンピュータ言語への翻訳はしてはならない。

かかるプログラムと関連ドキュメントについて書かれているインターシステムズ社の標準ライセンス契約に記載されている範囲を除き、ここに記載された本ドキュメントとソフトウェアプログラムの複製、使用、廃棄は禁じられている。インターシステムズ社は、ソフトウェアライセンス契約に記載されている事項以外にかかるソフトウェアプログラムに関する説明と保証をするものではない。さらに、かかるソフトウェアに関する、あるいはかかるソフトウェアの使用から起こるいかなる損失、損害に対するインターシステムズ社の責任は、ソフトウェアライセンス契約にある事項に制限される。

前述は、そのコンピュータソフトウェアの使用およびそれによって起こるインターシステムズ社の責任の範囲、制限に関する一般的な概略である。完全な参照情報は、インターシステムズ社の標準ライセンス契約に記載され、そのコピーは要望によって入手することができる。

インターシステムズ社は、本ドキュメントにある誤りに対する責任を放棄する。また、インターシステムズ社は、独自の裁量にて事前通知なしに、本ドキュメントに記載された製品および実行に対する代替と変更を行う権利を有する。

インターシステムズ社の製品に関するサポートやご質問は、以下にお問い合わせください:

InterSystems Worldwide Response Center (WRC)

Tel: +1-617-621-0700

Tel: +44 (0) 844 854 2917

Email: support@InterSystems.com

目次

1	プロダクション内からの JMS メッセージの取得	1
1.1	組み込みのビジネス・サービス	1
1.2	アダプタの詳細	1
1.3	関連項目	2
2	プロダクションから JMS へのメッセージの送信	3
2.1	アダプタの詳細	3
2.2	JMS メッセージの送信	3
2.3	組み込みのビジネス・オペレーション	4
2.4	関連項目	4
3	プロダクション内からの Kafka メッセージの取得	5
3.1	組み込みのビジネス・サービス	5
3.2	アダプタの詳細	5
3.3	関連項目	6
4	プロダクションから Kafka へのメッセージの送信	7
4.1	アダプタの詳細	7
4.2	Kafka メッセージの送信	7
4.3	組み込みのビジネス・オペレーション	8
4.4	関連項目	8
5	プロダクション内からの RabbitMQ メッセージの取得	9
5.1	組み込みのビジネス・サービス	9
5.2	アダプタの詳細	9
5.2.1	SSL/TLS 構成	10
5.3	関連項目	10
6	プロダクションから RabbitMQ へのメッセージの送信	11
6.1	アダプタの詳細	11
6.1.1	SSL/TLS 構成	11
6.2	RabbitMQ メッセージの送信	12
6.3	組み込みのビジネス・オペレーション	12
6.4	関連項目	12
7	プロダクションから Amazon SNS へのメッセージの送信	13
7.1	アダプタの詳細	13
7.2	組み込みのビジネス・オペレーション	14
7.3	一般的な AWS 設定	14
7.4	関連項目	14
8	プロダクション内からの Amazon SQS メッセージの取得	15
8.1	組み込みのビジネス・サービス	15
8.2	アダプタの詳細	15
8.3	一般的な AWS 設定	16
8.4	関連項目	16
9	プロダクションから Amazon SQS へのメッセージの送信	17
9.1	アダプタの詳細	17
9.2	Amazon SQS メッセージの送信	17
9.3	組み込みのビジネス・オペレーション	18

9.4 一般的な AWS 設定	18
9.5 関連項目	18

1

プロダクション内からの JMS メッセージの取得

InterSystems IRIS は、相互運用プロダクションに、[PEX](#) フレームワーク ([EnsLib.JMSPEX](#)) を使用して実装された JMS 受信アダプタを使用するビジネス・サービスが含まれる場合に JMS コンシューマになります。このアダプタにより、ビジネス・サービスは JMS キューまたはトピックからメッセージを取得できます。プロダクションを JMS コンシューマとして使用する場合には、受信アダプタを利用する[組み込みのビジネス・サービス](#)を使用する、またはアダプタを使用する独自のビジネス・サービスを構築するという、2 つのオプションがあります。

重要 このページでは、インターシステムズが [PEX](#) フレームワークを使用して実装する受信アダプタとビジネス・サービスを JMS で使用する方法を説明します。関連するクラスは [EnsLib.JMSPEX](#) パッケージの一部です。今後のリリースで削除される可能性がある、従来の [EnsLib.JMS](#) パッケージのクラスは使用しないでください。

1.1 組み込みのビジネス・サービス

受信アダプタを使用するカスタムのビジネス・サービスを構築するのではなく、管理ポータル設定を使用して、プロダクションに単に [EnsLib.JMSPEX.Service](#) を追加し、アダプタのプロパティを定義するだけです。ビジネス・サービスが、[\[ビジネスサービス\]→\[呼び出し間隔\]](#) 設定で指定した間隔で JMS サーバからメッセージを取得すると、これらのメッセージは非同期要求を使用してプロダクション内の別のビジネス・ホストに転送されます。これらの要求が転送されるビジネス・ホストは、[\[基本の設定\]→\[ターゲット構成名\]](#) 設定で指定されます。

ビジネス・サービスのプロダクションへの追加に関する基本情報は、“[ビジネス・ホストの追加](#)” を参照してください。

1.2 アダプタの詳細

JMS 受信アダプタは、クラス [EnsLib.JMSPEX.InboundAdapter](#) です。これには以下の設定が含まれており、これらはすべて、管理ポータルにアダプタを使用するビジネス・サービスの設定として表示されます。

- **QueueOrTopicName** は、コンシューマがメッセージを受信する JMS キューまたはトピックを定義します。
- **ReceiveSettings** (オプション) は、メッセージ取得の設定を定義する JSON 文字列です。使用できる設定のリストは、JMS API [JMSReceiveSettings](#) クラスによって使用可能になるプロパティのリストと同じで、各プロパティ名がキーとして機能します。
- **URL** は JMS サーバの URL を定義します。
- **InitialContextFactoryName** は初期 JMS コンテキスト・ファクトリの名前を定義します。

- ・ **ConnectionFactoryName** は JMS 接続ファクトリの名前を定義します。
- ・ **ClientID** は、プロダクションを JMS クライアントとして識別するための文字列を定義します。
- ・ **Credentials** は、JMS クライアントのユーザ名とパスワードに対応するインターシステムズ認証情報を定義します。認証情報の作成の詳細は、["設定で使用する再利用可能項目の定義"](#) を参照してください。

受信アダプタを使用するカスタムのビジネス・サービスの構築に関する一般情報は、["ビジネス・サービスの定義"](#) を参照してください。

1.3 関連項目

- ・ [プロダクションから JMS へのメッセージの送信](#)
- ・ JMS メッセージング API の使用法 (プロダクションなしで使用する場合)

2

プロダクションから JMS へのメッセージの送信

InterSystems IRIS は、相互運用プロダクションに、[PEX](#) フレームワーク (**EnsLib.JMSPEX**) を使用して実装された JMS 送信アダプタを使用するビジネス・オペレーションが含まれる場合に JMS プロデューサになります。このアダプタにより、ビジネス・オペレーションは JMS キューまたはトピックにメッセージを送信できます。プロダクションを JMS プロデューサとして使用する場合には、送信アダプタを利用する組み込みのビジネス・オペレーションを使用する、またはアダプタを使用する独自のビジネス・オペレーションを構築するという、2 つのオプションがあります。

重要 このページでは、インターシステムズが [PEX](#) フレームワークを使用して実装する送信アダプタとビジネス・オペレーションを JMS で使用する方法を説明します。関連するクラスは **EnsLib.JMSPEX** パッケージの一部です。今後のリリースで削除される可能性がある、従来の **EnsLib.JMS** パッケージのクラスは使用しないでください。

2.1 アダプタの詳細

JMS 送信アダプタは、クラス **EnsLib.JMSPEX.OutboundAdapte** です。これには以下の設定が含まれており、これらはすべて、管理ポータルにアダプタを使用するビジネス・オペレーションの設定として表示されます。

- ・ **URL** は JMS サーバの URL を定義します。
- ・ **InitialContextFactoryName** は初期 JMS コンテキスト・ファクトリの名前を定義します。
- ・ **ConnectionFactoryName** は JMS 接続ファクトリの名前を定義します。
- ・ **ClientID** は、プロダクションを JMS クライアントとして識別するための文字列を定義します。
- ・ **Credentials** は、JMS クライアントのユーザ名とパスワードに対応するインターシステムズ認証情報を定義します。認証情報の作成の詳細は、[“設定で使用する再利用可能項目の定義”](#) を参照してください。

送信アダプタを使用するカスタムのビジネス・オペレーションの構築に関する一般情報は、[“ビジネス・オペレーションの定義”](#) を参照してください。

2.2 JMS メッセージの送信

送信アダプタにより JMS に送信されるメッセージのクラスは **EnsLib.JMSPEX.Message** です。このクラスには、メッセージを定義するための以下のプロパティが含まれます。

- ・ **destination** は、プロデューサがメッセージを送信する JMS キューまたはトピックを定義します。

- ・ **type** は、メッセージ・タイプを定義します ("Text" または "Bytes")。
- ・ **textBody** または **bytesBody** は、メッセージ本文のコンテンツを定義します。2 つのうち、メッセージの **type** に対応するプロパティを設定します。

properties プロパティを使用して、メッセージにメタデータを添付することもできます。**properties** プロパティは、`%External.Messaging.JMSMessageProperty` オブジェクトの `%ListOfObjects` コレクションを受け入れます。JMS メッセージ・プロパティ・オブジェクトの作成に関する詳細なガイダンスは、JMS API に関するドキュメントを参照してください。JMS メッセージ・プロパティの一般的な情報は、[JMS のドキュメント](#)を参照してください。

送信アダプタの `SendMessage()` メソッドは、唯一の引数として **EnsLib.JMSPEX.Message** オブジェクトを取り、そのメッセージを JMS サーバに送信します。例えば、カスタムのビジネス・オペレーションは、以下を含めることにより、アダプタのメソッドを呼び出すことができます。

```
Do ..Adapter.SendMessage(pRequest)
```

2.3 組み込みのビジネス・オペレーション

送信アダプタを使用するカスタムのビジネス・オペレーションを構築するのではなく、管理ポータルを使用して、プロダクションに単に **EnsLib.JMSPEX.Operation** を追加し、必要な設定を定義するだけです。このビジネス・オペレーションは、プロダクション内の別のビジネス・ホストから要求を受信すると、アダプタの `SendMessage()` メソッドを呼び出します。この要求には、**EnsLib.JMSPEX.Message** のタイプの JMS メッセージを含める必要があります。

ビジネス・オペレーションのプロダクションへの追加に関する基本情報は、["ビジネス・ホストの追加"](#)を参照してください。

2.4 関連項目

- ・ [プロダクション内からの JMS メッセージの取得](#)
- ・ JMS メッセージング API の使用法 (プロダクションなしで使用する場合)

3

プロダクション内からの Kafka メッセージの取得

InterSystems IRIS は、相互運用プロダクションに Kafka 受信アダプタを使用するビジネス・サービスが含まれる場合に Kafka コンシューマになります。このアダプタにより、ビジネス・サービスは Kafka トピックからメッセージを取得できます (Kafka メッセージ・オブジェクトとして)。プロダクションを Kafka コンシューマとして使用する場合には、受信アダプタを利用する [組み込みのビジネス・サービス](#) を使用する、またはアダプタを使用する独自のビジネス・サービスを構築するという、2 つのオプションがあります。

3.1 組み込みのビジネス・サービス

受信アダプタを使用するカスタムのビジネス・サービスを構築するのではなく、管理ポータル設定を使用して、プロダクションに単に `EnsLib.Kafka.Service` を追加し、アダプタのプロパティを定義するだけです。ビジネス・サービスが、[\[ビジネスサービス\]→\[呼び出し間隔\]](#) 設定で指定した間隔で Kafka からメッセージを取得すると、これらのメッセージは非同期要求を使用してプロダクション内の別のビジネス・ホストに転送されます。これらの要求が転送されるビジネス・ホストは、[\[基本の設定\]→\[ターゲット構成名\]](#) 設定で指定されます。

ビジネス・サービスのプロダクションへの追加に関する基本情報は、["ビジネス・ホストの追加"](#) を参照してください。

3.2 アダプタの詳細

Kafka 受信アダプタは、クラス `EnsLib.Kafka.InboundAdapter` です。これには以下の設定が含まれており、これらはすべて、管理ポータルにアダプタを使用するビジネス・サービスの設定として表示されます。

- **Servers** は、Kafka クラスタ内のサーバを指定する、コンマ区切りの IP address:port エントリのリストを定義します。
- **Topic** は、コンシューマがメッセージを取得する Kafka トピックを定義します。
- **GroupID** は、コンシューマのコンシューマ・グループ ID を定義します。
- **ReceiveSettings** (オプション) は、メッセージ取得の設定を定義する JSON 文字列です。使用できる設定のリストは、Kafka API `KafkaReceiveSettings` クラスによって使用可能になるプロパティのリストと同じで、各プロパティ名がキーとして機能します。
- **Credentials** は、Kafka クライアントのユーザ名とパスワードに対応するインターシステムズ認証情報を定義します。認証情報の作成の詳細は、["設定で使用する再利用可能項目の定義"](#) を参照してください。

受信アダプタを使用するカスタムのビジネス・サービスの構築に関する一般情報は、["ビジネス・サービスの定義"](#) を参照してください。

3.3 関連項目

- ・ [プロダクションから Kafka へのメッセージの送信](#)
- ・ [Kafka メッセージング API の使用法 \(プロダクションなしで使用する場合\)](#)

4

プロダクションから Kafka へのメッセージの送信

InterSystems IRIS は、相互運用プロダクションに Kafka 送信アダプタを使用するビジネス・オペレーションが含まれる場合に Kafka プロデューサになります。このアダプタにより、ビジネス・オペレーションは Kafka トピックにメッセージを送信できます。プロダクションを Kafka プロデューサとして使用する場合には、送信アダプタを利用する組み込みのビジネス・オペレーションを使用する、またはアダプタを使用する独自のビジネス・オペレーションを構築するという、2 つのオプションがあります。

4.1 アダプタの詳細

Kafka 送信アダプタは、クラス `EnsLib.Kafka.OutboundAdapter` です。これには以下の設定が含まれており、これらはすべて、管理ポータルにアダプタを使用するビジネス・オペレーションの設定として表示されます。

- **Servers** は、Kafka クラスタ内のサーバを指定する、コンマ区切りの IP address:port エントリのリストを定義します。
- **ClientID** は、プロデューサの Kafka クライアント ID を定義します。
- **Credentials** は、Kafka クライアントのユーザ名とパスワードに対応するインターシステムズ認証情報を定義します。認証情報の作成の詳細は、[“設定で使用する再利用可能項目の定義”](#) を参照してください。

送信アダプタを使用するカスタムのビジネス・オペレーションの構築に関する一般情報は、[“ビジネス・オペレーションの定義”](#) を参照してください。

4.2 Kafka メッセージの送信

送信アダプタにより Kafka に送信されるメッセージのクラスは `EnsLib.Kafka.Message` です。これには、以下のプロパティが含まれます。

- **topic** は、プロデューサがメッセージを送信する Kafka トピックを定義します。
- **value** (%String) は、Kafka メッセージのコンテンツを定義します。**value** が設定されている場合、**binaryValue** は設定しないでください。
- **binaryValue** (任意の長さのバイナリ・ストリーム) は、メッセージの長さが %String の最大長を超える場合の、Kafka メッセージのコンテンツを定義します。**binaryValue** が設定されている場合、**value** は設定しないでください。
- **key** は、Kafka メッセージのオプションのタグを定義します。

送信アダプタの `SendMessage()` メソッドは、唯一の引数として `EnsLib.Kafka.Message` オブジェクトを取り、そのメッセージを Kafka サーバに送信します。例えば、カスタムのビジネス・オペレーションは、以下を含めることにより、アダプタのメソッドを呼び出すことができます。

```
Do ...Adapter.SendMessage(pRequest)
```

4.3 組み込みのビジネス・オペレーション

送信アダプタを使用するカスタムのビジネス・オペレーションを構築するのではなく、管理ポータル設定を使用して、プロダクションに単に `EnsLib.Kafka.Operation` を追加し、アダプタのプロパティを定義するだけです。このビジネス・オペレーションは、プロダクション内の別のビジネス・ホストから要求を受信すると、アダプタの `SendMessage()` メソッドを呼び出します。この要求には、`EnsLib.Kafka.Message` のタイプの Kafka メッセージを含める必要があります。

ビジネス・オペレーションのプロダクションへの追加に関する基本情報は、“[ビジネス・ホストの追加](#)”を参照してください。

4.4 関連項目

- ・ [プロダクション内からの Kafka メッセージの取得](#)
- ・ [Kafka メッセージング API の使用法 \(プロダクションなしで使用する場合\)](#)

5

プロダクション内からの RabbitMQ メッセージの取得

InterSystems IRIS は、相互運用プロダクションに RabbitMQ 受信アダプタを使用するビジネス・サービスが含まれる場合に RabbitMQ コンシューマになります。このアダプタにより、ビジネス・サービスは RabbitMQ キューからメッセージを取得できます。プロダクションを RabbitMQ コンシューマとして使用する場合には、受信アダプタを利用する[組み込みのビジネス・サービス](#)を使用する、またはアダプタを使用する独自のビジネス・サービスを構築するという、2つのオプションがあります。

5.1 組み込みのビジネス・サービス

受信アダプタを使用するカスタムのビジネス・サービスを構築するのではなく、管理ポータル設定を使用して、プロダクションに単に `EnsLib.RabbitMQ.Service` を追加し、アダプタのプロパティを定義するだけです。ビジネス・サービスが、**[ビジネスサービス]→[呼び出し間隔]** 設定で指定した間隔で RabbitMQ からメッセージを取得すると、これらのメッセージは非同期要求を使用してプロダクション内の別のビジネス・ホストに転送されます。これらの要求が転送されるビジネス・ホストは、**[基本の設定]→[ターゲット構成名]** 設定で指定されます。

ビジネス・サービスのプロダクションへの追加に関する基本情報は、“[ビジネス・ホストの追加](#)” を参照してください。

5.2 アダプタの詳細

RabbitMQ 受信アダプタは、クラス `EnsLib.RabbitMQ.InboundAdapter` です。これには以下の設定が含まれており、これらはすべて、管理ポータルにアダプタを使用するビジネス・サービスの設定として表示されます。

- **Queue Name** は、コンシューマがメッセージを受信する RabbitMQ キューを定義します。
- **ExchangeName** (オプション) は、キューにメッセージをルーティングする RabbitMQ エクスチェンジを定義します。
注釈 RabbitMQ がメッセージをルーティングする方法の詳細は、[RabbitMQ のドキュメント](#)を参照してください。
- **BindingKeys** (オプション) は、プロダクションがメッセージを受信するキューを、ユーザが指定したエクスチェンジに結合するキーを定義します。

- ・ **ReceiveSettings** (オプション) は、メッセージ取得の設定を定義する JSON 文字列です。使用できる設定のリストは、RabbitMQ API `RabbitMQReceiveSettings` クラスによって使用可能になるプロパティのリストと同じで、各プロパティ名がキーとして機能します。
- ・ **MQHost** は RabbitMQ サーバのホスト名または IP アドレスを定義します。
- ・ **MQPort** は、RabbitMQ と通信するためのポート番号を定義します。
- ・ **MQVirtualHost** (オプション) は、RabbitMQ の仮想ホスト名を定義します。
- ・ **Credentials** は、RabbitMQ クライアントのユーザ名とパスワードに対応するインターシステムズ認証情報を定義します。認証情報の作成の詳細は、[“設定で使用する再利用可能項目の定義”](#) を参照してください。

受信アダプタを使用するカスタムのビジネス・サービスの構築に関する一般情報は、[“ビジネス・サービスの定義”](#) を参照してください。

5.2.1 SSL/TLS 構成

RabbitMQ アダプタは両方とも (受信と[送信](#))、SSL/TLS を使用して RabbitMQ と接続するようプロダクションを構成するための追加設定を受け入れます。これらの設定は、アダプタを使用するビジネス・サービスの設定として管理ポータルにも表示されます。

これらの設定の詳細リストについては、RabbitMQ API を使用して RabbitMQ クライアントを構成する手順を参照してください。

5.3 関連項目

- ・ [プロダクションから RabbitMQ へのメッセージの送信](#)
- ・ [RabbitMQ メッセージング API の使用法 \(プロダクションなしで使用する場合\)](#)

6

プロダクションから RabbitMQ へのメッセージの送信

InterSystems IRIS は、相互運用プロダクションに RabbitMQ 送信アダプタを使用するビジネス・オペレーションが含まれる場合に RabbitMQ パブリッシャになります。このアダプタにより、ビジネス・オペレーションは RabbitMQ エクスチェンジにメッセージを送信できます。プロダクションを RabbitMQ パブリッシャとして使用する場合には、送信アダプタを利用する組み込みのビジネス・オペレーションを使用する、またはアダプタを使用する独自のビジネス・オペレーションを構築するという、2 つのオプションがあります。

6.1 アダプタの詳細

RabbitMQ 送信アダプタは、クラス `EnsLib.RabbitMQ.OutboundAdapter` です。これには以下の設定が含まれており、これらはすべて、管理ポータルにアダプタを使用するビジネス・オペレーションの設定として表示されます。

- ・ **MQHost** は RabbitMQ サーバのホスト名または IP アドレスを定義します。
- ・ **MQPort** は、RabbitMQ と通信するためのポート番号を定義します。
- ・ **MQVirtualHost** (オプション) は、RabbitMQ の仮想ホスト名を定義します。
- ・ **Credentials** は、Kafka クライアントのユーザ名とパスワードに対応するインターシステムズ認証情報を定義します。認証情報の作成の詳細は、[“設定で使用する再利用可能項目の定義”](#) を参照してください。

送信アダプタを使用するカスタムのビジネス・オペレーションの構築に関する一般情報は、[“ビジネス・オペレーションの定義”](#) を参照してください。

6.1.1 SSL/TLS 構成

RabbitMQ アダプタは両方とも (受信と送信)、SSL/TLS を使用して RabbitMQ と接続するようプロダクションを構成するための追加設定を受け入れます。これらの設定は、アダプタを使用するビジネス・サービスの設定として管理ポータルにも表示されます。

これらの設定の詳細リストについては、RabbitMQ API を使用して RabbitMQ クライアントを構成する手順を参照してください。

6.2 RabbitMQ メッセージの送信

送信アダプタにより RabbitMQ に送信されるメッセージのクラスは `EnsLib.RabbitMQ.Message` です。このクラスには、以下を含め、メッセージを定義するためのいくつかのプロパティが含まれます。

- ・ **exchange** は、パブリッシャがメッセージを送信する RabbitMQ エクスチェンジを定義します。
- ・ **routingKey** は、エクスチェンジがメッセージをルーティングするために使用するルーティング・キーを定義します。
- ・ **deliveryMode** は、メッセージを永続的なものとして扱うか (値が 2 の場合)、一時的なものとして扱うか (値が 1 の場合) を定義します。
- ・ **contentEncoding** は、メッセージ・コンテンツのエンコードを定義します (UTF-8 など)。
- ・ **encodedContent** は、**contentEncoding** によって指定されたとおりにエンコードされるメッセージのコンテンツを定義します。

このメッセージ・クラスが利用できるメッセージ・プロパティの詳細な説明は、[RabbitMQ のドキュメント](#)を参照してください。

送信アダプタの `SendMessage()` メソッドは、唯一の引数として `EnsLib.RabbitMQ.Message` オブジェクトを取り、そのメッセージを RabbitMQ サーバに送信します。例えば、カスタムのビジネス・オペレーションは、以下を含めることにより、アダプタのメソッドを呼び出すことができます。

```
Do ...Adapter.SendMessage(pRequest)
```

6.3 組み込みのビジネス・オペレーション

送信アダプタを使用するカスタムのビジネス・オペレーションを構築するのではなく、管理ポータル設定を使用して、プロダクションに単に `EnsLib.RabbitMQ.Operation` を追加し、アダプタのプロパティを定義するだけです。このビジネス・オペレーションは、プロダクション内の別のビジネス・ホストから要求を受信すると、アダプタの `SendMessage()` メソッドを呼び出します。この要求には、`EnsLib.RabbitMQ.Message` のタイプの RabbitMQ メッセージを含める必要があります。

ビジネス・オペレーションのプロダクションへの追加に関する基本情報は、“[ビジネス・ホストの追加](#)”を参照してください。

6.4 関連項目

- ・ [プロダクション内からの RabbitMQ メッセージの取得](#)
- ・ RabbitMQ メッセージング API の使用法 (プロダクションなしで使用する場合)

7

プロダクションから Amazon SNS へのメッセージの送信

Amazon SNS は、パブリッシャからのメッセージをサブスクライバに提供するクラウド・サービスです。SNS 送信アダプタを使用するビジネス・オペレーションを含む相互運用プロダクションを作成することにより、SNS パブリッシャとしてインターシステムズ製品を構成できます。ビジネス・オペレーションおよびアダプタに不慣れな場合は、“[相互運用プロダクションの概要](#)”を参照してください。

SNS に送信されるメッセージには、トピック、件名、メッセージのコンテンツの 3 つのコンポーネントがあります。パブリッシャによって SNS に送信される各メッセージは、特定のトピックに関連付けられている必要があります。SNS は、特定のトピックをサブスクライブしているサブスクライバにメッセージをプッシュします。SNS に送信されたメッセージには、件名も含めることができます。メッセージが電子メールとしてサブスクライバに配信されると、SNS はこの件名を件名行として使用します。

7.1 アダプタの詳細

SNS 送信アダプタの ObjectScript クラスは `EnsLib.AmazonSNS.OutboundAdapter` です。[組み込みのビジネス・オペレーション](#)を使用している場合、この送信アダプタの詳細について知る必要はありません。ビジネス・オペレーションがこれを処理します。ただし、カスタムのビジネス・オペレーションを記述している場合は、これがアダプタの `Publish()` メソッドを呼び出して、SNS にメッセージを送信することになります。カスタムのビジネス・オペレーションからこのメソッドを呼び出すには、以下を追加します。

ObjectScript

```
Set tSC = ..Adapter.Publish(..ARNTopic, request.Message, ..Subject)
```

説明：

- ・ **ARNTopic** は、SNS トピックを定義するビジネス・オペレーションのプロパティです。必須。
- ・ **request.Message** は、SNS に送信されるメッセージを含むプロダクション要求オブジェクトのプロパティです。必須。
- ・ **Subject** は、SNS サブジェクトを定義するビジネス・オペレーションのプロパティです。オプション。

これがどのように機能するのかわを示した例は、`EnsLib.AmazonSNS.BusinessOperation` のソース・コードを参照してください。カスタムのビジネス・オペレーションの作成の詳細は、“[ビジネス・オペレーションの定義](#)”を参照してください。

注釈 SNS 送信アダプタは、InterSystems PEX フレームワークを使用して開発されているため、このアダプタの ObjectScript ソース・コードは、他のアダプタとは異なるように見えます。例えば、アダプタのメソッドは、実際には Java PEX コンポーネント内に記述されたメソッドのラップです。

7.2 組み込みのビジネス・オペレーション

インターシステムズは、カスタム・コードを記述することなく、SNS にメッセージを公開するために使用できる、組み込みのビジネス・オペレーションを提供しています。SNS 送信アダプタを使用しているため、プロダクションを SNS パブリッシャにするために必要なのは、この組み込みのビジネス・オペレーションのみです。ただ、`EnsLib.AmazonSNS.BusinessOperation` をプロダクションに追加し、このビジネス・オペレーションに `EnsLib.AmazonSNS.PublishRequest` 要求を送信するよう、その他のビジネス・ホストを構成するだけです。ビジネス・オペレーションのプロダクションへの追加に関する情報は、“[ビジネス・ホストの追加](#)”を参照してください。

`EnsLib.AmazonSNS.BusinessOperation` ビジネス・オペレーションには、メッセージを SNS トピックに関連付けるために使用されるプロパティ `ARNTopic` が含まれます。管理ポータルの `[SNS]→[ARNTopic]` 設定を使用してトピックを指定できます。

`[SNS]→[件名]` 設定を使用して、ビジネス・オペレーションによって送信されるメッセージに件名を付与できます。件名の定義はオプションです。

メッセージのトピックと件名はビジネス・オペレーションで定義される一方、メッセージの実際のコンテンツは、`EnsLib.AmazonSNS.PublishRequest` 要求でビジネス・オペレーションに送信されて、この要求オブジェクトを別のビジネス・ホストから送信するようにプロダクションが構成されます。

7.3 一般的な AWS 設定

SNS 送信アダプタは、一般的な AWS プロパティを含む一般的なアダプタ・クラスを拡張します。これらの AWS プロパティは、送信アダプタを使用するビジネス・オペレーションをプロダクションに追加する際、管理ポータルでビジネス・オペレーションの `[AWS]` 設定を使用して設定できます。

- ・ **CredentialsFile** – 空白の場合、Amazon は既定の認証情報プロバイダ・チェーンを使用して、SNS へのアクセスに必要な認証情報を取得します。AWS 認証情報ファイルを使用する場合は、そのパス名を入力します。
- ・ **リージョン** – アクセスする AWS リージョンを指定します。リージョンのリストは、“[Amazon Regions, Availability Zones, and Local Zones](#)”を参照してください。

7.4 関連項目

- ・ [Amazon SNS メッセージング API の使用法 \(プロダクションなしで使用する場合\)](#)

8

プロダクション内からの Amazon SQS メッセージの取得

InterSystems IRIS は、相互運用プロダクションに RabbitMQ 受信アダプタを使用するビジネス・サービスが含まれる場合に RabbitMQ コンシューマになります。このアダプタにより、ビジネス・サービスは RabbitMQ キューからメッセージを取得できます。プロダクションを RabbitMQ コンシューマとして使用する場合には、受信アダプタを利用する[組み込みのビジネス・サービス](#)を使用する、またはアダプタを使用する独自のビジネス・サービスを構築するという、2つのオプションがあります。

8.1 組み込みのビジネス・サービス

受信アダプタを使用するカスタムのビジネス・サービスを構築するのではなく、管理ポータル設定を使用して、プロダクションに単に `EnsLib.AmazonSQS.Service` を追加し、アダプタのプロパティを定義するだけです。ビジネス・サービスが、[\[ビジネスサービス\]→\[呼び出し間隔\]](#) 設定で指定した間隔で Amazon SQS からメッセージを取得すると、これらのメッセージは非同期要求を使用してプロダクション内の別のビジネス・ホストに転送されます。これらの要求が転送されるビジネス・ホストは、[\[基本の設定\]→\[ターゲット構成名\]](#) 設定で指定されます。

ビジネス・サービスのプロダクションへの追加に関する基本情報は、“[ビジネス・ホストの追加](#)”を参照してください。

8.2 アダプタの詳細

Amazon SQS 受信アダプタは、クラス `EnsLib.AmazonSQS.InboundAdapter` です。これには以下の設定が含まれており、これらはすべて、管理ポータルにアダプタを使用するビジネス・サービスの設定として表示されます。

- **Queue** は、コンシューマがメッセージを受信する Amazon SQS キューを定義します。
- **DeleteAfterReceive** は、プロダクションがメッセージを受信した後、そのメッセージをキューから削除するかどうかを決定します。
- **ReceiveSettings** (オプション) は、メッセージ取得の設定を定義する JSON 文字列です。使用できる設定のリストは、Amazon SQS API `SQSReceiveSettings` クラスによって使用可能になるプロパティのリストと同じで、各プロパティ名がキーとして機能します。

受信アダプタを使用するカスタムのビジネス・サービスの構築に関する一般情報は、“[ビジネス・サービスの定義](#)”を参照してください。

8.3 一般的な AWS 設定

Amazon SQS 受信アダプタは、一般的な AWS プロパティを含む一般的なアダプタ・クラスを拡張します。これらの AWS プロパティは、受信アダプタを使用するビジネス・サービスをプロダクションに追加する際、管理ポータルでビジネス・サービスの [AWS] 設定を使用して設定できます。

- ・ **CredentialsFile** – 空白の場合、Amazon は既定の認証情報プロバイダ・チェーンを使用して、SQS へのアクセスに必要な認証情報を取得します。AWS 認証情報ファイルを使用する場合は、そのパス名を入力します。
- ・ **リージョン** – アクセスする AWS リージョンを指定します。リージョンのリストは、“[Amazon Regions, Availability Zones, and Local Zones](#)” を参照してください。

8.4 関連項目

- ・ [プロダクションから Amazon SQS へのメッセージの送信](#)
- ・ Amazon SQS メッセージング API の使用法 (プロダクションなしで使用する場合)

9

プロダクションから Amazon SQS へのメッセージの送信

InterSystems IRIS は、相互運用プロダクションに Amazon SQS 送信アダプタを使用するビジネス・オペレーションが含まれる場合に Amazon SQS プロデューサになります。このアダプタにより、ビジネス・オペレーションは Amazon SQS キューにメッセージを送信できます。プロダクションを Amazon SQS プロデューサとして使用する場合には、送信アダプタを利用する組み込みのビジネス・オペレーションを使用する、またはアダプタを使用する独自のビジネス・オペレーションを構築するという、2 つのオプションがあります。

9.1 アダプタの詳細

Amazon SQS 送信アダプタは、クラス `EnsLib.AmazonSQS.OutboundAdapter` です。

送信アダプタを使用するカスタムのビジネス・オペレーションの構築に関する一般情報は、“[ビジネス・オペレーションの定義](#)” を参照してください。

9.2 Amazon SQS メッセージの送信

送信アダプタにより Amazon SQS に送信されるメッセージのクラスは `EnsLib.AmazonSQS.Message` です。このクラスには、以下を含め、メッセージを定義するためのいくつかのプロパティが含まれます。

- `queue` は、プロデューサがメッセージを送信する Amazon SQS キューを定義します。
- `body` は、メッセージのコンテンツを定義します。

このクラスがメッセージ・オブジェクト・プロパティを介して利用できるメッセージ識別子の詳細な説明は、[Amazon SQS のドキュメント](#)を参照してください。

`messageAttributes` プロパティを使用して、メッセージのカスタム・メタデータを指定することもできます。`messageAttributes` プロパティは、`%External.Messaging.SQSMessageAttribute` オブジェクトの `%ListOfObjects` コレクションを受け入れます。SQS メッセージ属性オブジェクトの作成に関する詳細なガイダンスについては、Amazon SQS API に関するドキュメントを参照してください。メッセージ属性を使用してカスタム・メタデータを添付する場合の一般情報は、[Amazon SQS のドキュメント](#)を参照してください。

送信アダプタの `SendMessage()` メソッドは、唯一の引数として `EnsLib.AmazonSQS.Message` オブジェクトを取り、そのメッセージを Amazon SQS サーバに送信します。例えば、カスタムのビジネス・オペレーションは、以下を含めることにより、アダプタのメソッドを呼び出すことができます。

```
Do ..Adapter.SendMessage(pRequest)
```

9.3 組み込みのビジネス・オペレーション

送信アダプタを使用するカスタムのビジネス・オペレーションを構築するのではなく、管理ポータルを使用して、プロダクションに単に `EnsLib.AmazonSQS.Operation` を追加し、必要な設定を定義するだけです。このビジネス・オペレーションは、プロダクション内の別のビジネス・ホストから要求を受信すると、アダプタの `SendMessage()` メソッドを呼び出します。この要求には、`EnsLib.AmazonSQS.Message` のタイプの Amazon SQS メッセージを含める必要があります。

ビジネス・オペレーションのプロダクションへの追加に関する基本情報は、“[ビジネス・ホストの追加](#)”を参照してください。

9.4 一般的な AWS 設定

SQS 送信アダプタは、一般的な AWS プロパティを含む一般的なアダプタ・クラスを拡張します。これらの AWS プロパティは、送信アダプタを使用するビジネス・オペレーションをプロダクションに追加する際、管理ポータルでビジネス・オペレーションの [AWS] 設定を使用して設定できます。

- ・ **CredentialsFile** – 空白の場合、Amazon は既定の認証情報プロバイダ・チェーンを使用して、SQS へのアクセスに必要な認証情報を取得します。AWS 認証情報ファイルを使用する場合は、そのパス名を入力します。
- ・ **リージョン** – アクセスする AWS リージョンを指定します。リージョンのリストは、“[Amazon Regions, Availability Zones, and Local Zones](#)”を参照してください。

9.5 関連項目

- ・ [プロダクション内からの Amazon SQS メッセージの取得](#)
- ・ [Amazon SQS メッセージング API の使用法 \(プロダクションなしで使用する場合\)](#)