



プロダクション内での MQTT アダプタの使用方法

Version 2024.1
2024-06-06

プロダクション内での MQTT アダプタの使用法

InterSystems IRIS Data Platform Version 2024.1 2024-06-06

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 MQTT (Message Queuing Telemetry Transport) の概要	1
2 MQTT アダプタの使用法	3
2.1 MQTT 受信アダプタ	3
2.2 MQTT 受信アダプタを使用するビジネス・サービスの作成	3
2.3 OnProcessInput() メソッドの実装	4
2.4 MQTT 送信アダプタ	4
3 MQTT パススルー・ビジネス・サービスおよびオペレーションの構成と使用	5
3.1 パススルー・ビジネス・サービスの追加と構成	5
3.2 パススルー・ビジネス・オペレーションの追加と構成	7
3.3 MQTT メッセージの形式	8
4 受信 MQTT アダプタと送信 MQTT アダプタの設定	9
4.1 まとめ	9
4.2 クリーン・セッション	9
4.3 クライアント ID	9
4.4 接続タイムアウト	10
4.5 認証情報名	10
4.6 キープ・アライブ	10
4.7 LWT トピック	10
4.8 LWT メッセージ	10
4.9 QOS	10
4.10 保持	10
4.11 SSL 構成名	11
4.12 タイムアウト	11
4.13 トピック	11
4.14 トレース	11
4.15 トレース・レベル	11
4.16 URL	12

1

MQTT (Message Queuing Telemetry Transport) の概要

MQTT (Message Queuing Telemetry Transport) は、多くのデバイスがデータをネットワーク上に発行できるように設計された軽量プロトコルです。MQTT は、帯域幅が制限されているネットワークにおいても高いメッセージ・スループットを実現するよう設計されている軽量メッセージ・プロトコルです。発行および購読のメカニズム、およびブローカ・サーバの使用により、メッセージのパブリッシャおよびサブスクライバのネットワークへの接続が信頼性の低いものであっても、帯域幅の狭いネットワークで信頼性の高いメッセージの通信を実現できます。MQTT は、多くの小さなデバイスがネットワークに情報を発行する IoT に非常に適しています。クライアントはブローカへの単一の接続を持つ任意のデバイスからのメッセージを購読できます。MQTT の詳細は、[MQTT.org](http://mqtt.org) を参照してください。

InterSystems IRIS® データ・プラットフォームでは MQTT 3.1 をサポートしています。MQTT 仕様は、OASIS 標準 <http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqtt-v3.1.1.html> として定義されています。

MQTT アダプタ `EnsLib.MQTT.Adapter.Inbound` は、MQTT connect、subscribe、および receive メッセージを送信します。`EnsLib.MQTT.Adapter.Outbound` は、MQTT publish メッセージを送信します。

これらのアダプタを使用して、カスタムのビジネス・サービスおよびオペレーションを開発したり、組み込みのビジネス・サービス `EnsLib.MQTT.Service.Passthrough` およびビジネス・オペレーション `EnsLib.MQTT.Operation.Passthrough` を使用することができます。相互運用プロダクション外で MQTT プロトコルを使用する場合は、下位レベルの `%Net.MQTT` クラスを使用できます。

MQTT プロトコルでは、MQTT ブローカ・サーバと MQTT クライアントが使用されます。クライアントはメッセージを発行および購読します。クライアントはブローカにメッセージを発行します。また、クライアントはブローカに購読を登録してメッセージを受信します。このアーキテクチャにより、クライアントは、わずか 1 つのブローカへの接続で、さまざまなパブリッシャによって発行されたメッセージを購読できます。

多くの場合は、組み込みのビジネス・サービスおよびオペレーションを使用して MQTT ブローカにアクセスできますが、より複雑な動作が必要な場合、アダプタを使用してカスタム・コードを作成する必要があることがあります。

MQTT クラスは Eclipse の [Paho](#) の [C 言語用 MQTT クライアント・ライブラリ](#)を使用します。

2

MQTT アダプタの使用法

ここでは、MQTT の受信アダプタおよび送信アダプタ `EnsLib.MQTT.Adapter.Inbound` および `EnsLib.MQTT.Adapter.Outbound` の動作と、カスタムのビジネス・サービスおよびオペレーションでこれらを使用する方法について説明します。

2.1 MQTT 受信アダプタ

MQTT 受信アダプタは、ブローカ上のメッセージを購読し、そのサブスクリプションからメッセージを受信する役割を果たします。この受信アダプタは以下を実行します。

- ・ 初期化の際、`%Net.MQTT.Client` クライアントを作成します。
- ・ タスクでは、以下を実行します。
 1. ブローカに接続されているかどうかを確認します。初めて呼び出された場合、または接続が失われている場合、アダプタは `%Net.MQTT.Client` クライアントを使用してブローカに接続し、指定されたトピックを購読します。これはオプションでワイルドカードを含めることができます。
 2. `%Net.MQTT.Client` クライアントを使用して、トピックとメッセージを受信します。
 3. 受信呼び出しがタイムアウトしていない場合、ビジネス・サービスの `ProcessInput()` メソッドを呼び出して、メッセージに渡します。

2.2 MQTT 受信アダプタを使用するビジネス・サービスの作成

このアダプタをプロダクションで使用するには、ここに記載されているように新しいビジネス・サービス・クラスを作成します。その後、“[MQTT パススルー・ビジネス・サービスおよびオペレーションの構成と使用](#)” で説明しているパススルー・サービスおよびオペレーションの場合と同様に、これをコンパイルし、プロダクションに追加して、構成します。新しいカスタムのメッセージ・クラスが必要な場合は、“プロダクションの開発”の“[メッセージの定義](#)”の説明のとおりこれを作成します。

ビジネス・サービス・クラスの基本要件を以下に列挙します。

- ・ ビジネス・サービス・クラスは `Ens.BusinessService` を拡張するものでなければなりません。
- ・ クラスの ADAPTER パラメータは `EnsLib.MQTT.Adapter.Inbound` である必要があります。

- ・ クラスには `OnProcessInput()` メソッドを実装する必要があります。これについては、“[OnProcessInput\(\) メソッドの実装](#)” で説明します。
- ・ その他のオプションと一般情報は、“[プロダクションの開発](#)” の “[ビジネス・サービス・クラスの定義](#)” を参照してください。

受信アダプタに関連する設定については、“[MQTT パススルー・ビジネス・サービスおよびオペレーションの構成と使用](#)” および “[MQTT アダプタの設定](#)” を参照してください。

以下の例は、ビジネス・サービス・クラスの全体的な構造を示しています。

Class Definition

```
Class EMQTT.NewService1 Extends Ens.BusinessService
{
Parameter ADAPTER = "EnsLib.MQTT.Adapter.Inbound";

Method OnProcessInput(pInput As EnsLib.MQTT.Message, pOutput As %RegisteredObject) As %Status
{
    set tsc=$$$OK
    //your code here
    Quit tsc
}
```

2.3 OnProcessInput() メソッドの実装

カスタムのビジネス・サービス・クラスでは、`OnProcessInput()` メソッドのシグニチャは以下のようになります。

```
Method OnProcessInput(pInput As EnsLib.MQTT.Message, pOutput As %RegisteredObject) As %Status
```

2.4 MQTT 送信アダプタ

送信アダプタは、ブローカにメッセージを発行する役割を果たします。送信アダプタがメッセージで呼び出されると、以下を実行します。

- ・ ブローカへの接続がまだない場合、これを作成します。
- ・ 入力パラメータにトピックの値が含まれる場合は、これを使用します。トピックが提供されていない場合は、設定で指定されたトピックを使用します。
- ・ 入力パラメータまたは設定からのトピックを使用して、ブローカにメッセージを発行します。

3

MQTT パススルー・ビジネス・サービスおよびオペレーションの構成と使用

ここでは、組み込みの MQTT パススルー・ビジネス・サービスおよび MQTT パススルー・ビジネス・オペレーションの動作について説明します。

3.1 パススルー・ビジネス・サービスの追加と構成

ビジネス・サービスをプロダクションに追加するには、管理ポータルを使用して相互運用対応ネームスペースで以下の操作を行います。

1. MQTT ブローカでユーザ名/パスワード認証を使用している場合、まず、その情報で認証情報を定義する必要があります (管理ポータルの [相互運用性]→[構成]→[認証情報])。TLS 認証を使用している場合は、インターシステムズの “TLS ガイド” を参照してください。
2. `EnsLib.MQTT.Service.Passthrough` ビジネス・サービス・クラスのインスタンスをプロダクションに追加します。
3. 入力を受信できるようにビジネス・サービスを構成します。
 - ・ [基本の設定] 内の [ターゲット構成名] プロパティを設定します。このプロパティは多くのビジネス・サービスに共通で、ビジネス・サービスがそのメッセージをどこに送信するのかを指定します。ターゲットには、1 つ以上のビジネス・プロセスおよびビジネス・オペレーションを指定できます。
 - ・ オプションで [CleanSession] プロパティを設定します。[CleanSession] にチェックが付けられている場合、ブローカは接続から次の接続までの間、サブスクリプションについての情報を保持しません。接続が一時的に切断されている間に発行されたメッセージを受信する場合は、[CleanSession] プロパティにはチェックを付けないでください。
 - ・ [ClientID] プロパティを設定します。これはブローカに対するクライアントを指定します。ASCII でエンコードされ、長さが 1 ～ 23 文字である必要があります。ブローカへの接続を切断してから再接続する場合、ClientID が同じであればその ID 情報を保持し、切断中に送信されたメッセージを返します ([CleanSession] にチェックが付いていない場合)。[ClientID] フィールドを空のままにすると、クライアントによって一意の ClientID が割り当てられます。複数のプロダクションで同じビジネス・サービスを実行している場合、それぞれが異なる ClientID を持つようにしてください。そうでない場合、ブローカはそれらを 1 つのクライアントとして扱い、それらのうちの 1 つにしかメッセージを返しません。
 - ・ [ConnectTimeout] プロパティを設定します。これは、ビジネス・サービスがブローカからの応答確認を待機する秒数を設定します。

- ・ ログイン認証情報を使用している場合、[CredentialsName] プロパティを手順 1 で定義した認証情報名に設定します。
- ・ [KeepAlive] プロパティを、クライアントからブローカへのメッセージの最大間隔 (秒数) に設定します。
- ・ オプションで、[LWTTopic] および [LWTMessage] プロパティを設定します。これらは、接続の Last Will and Testament (遺言) メッセージを設定します。接続が予期せずになされたことをブローカが検出すると、これは指定したトピックとメッセージを使用してメッセージを発行します。
- ・ [QOS] プロパティを設定します。これによって、ブローカとの接続におけるサービス品質が設定されます。値が 0 の場合、ビジネス・サービスはブローカからの確認応答を待機しないことを意味し、値が 1 の場合、ビジネス・サービスはブローカがメッセージを受信したという確認応答を待機することを意味します。
- ・ [Retained] プロパティにはチェックを付けしないでください。これはメッセージを発行するためにのみ使用され、メッセージの購読には使用されません。メッセージの発行時にこのプロパティを設定すると、ブローカにこのメッセージを保持し、これを新しいサブスクライバに返すよう指示します。
- ・ TLS セキュリティを使用している場合は、[SSLConfigName] を TLS 構成の名前に設定します。
- ・ [QOS] を 1 に設定した場合、[タイムアウト] プロパティを、ビジネス・サービスが確認応答を待機する秒数に設定します。
- ・ [トピック] プロパティを、購読するトピックに設定します。トピックは通常、サブトピックのレベルが / (スラッシュ) で区切られた階層文字列です。サブスクリプションでは、トピックにワイルドカードを含めることができます。
 - + (プラス記号) – そのレベルのすべての文字列に一致します。例えば、"/temperatures/+/highest" は、トピック "temperatures/Paris/highest" や "temperatures/Bangkok/highest" に一致しますが、"temperatures/Paris/highest/today" には一致しません。プラス記号は、同じレベル内の他の文字と組み合わせることはできません。
 - # (シャープ記号) – 最後のレベルのみで、任意の文字列および任意のサブトピックに一致します。例えば、"temperatures/Paris/#" は、"temperatures/Paris/highest"、"temperatures/Paris/highest/today"、"temperatures/Paris/lowest/record/thiscentury" のトピックすべてに一致します。
- ・ ビジネス・サービスで行われる MQTT アクションをトレースする場合は、[トレース] プロパティにチェックを付け、[TraceLevel] プロパティを設定します。MQTT トレース・メッセージは、ビジネス・サービスのプロダクション・ログに書き込まれます。[TraceLevel] では以下の値を扱うことができます。
 - 0 – MQTTCLIENTTRACEMAXIMUM、最大レベルのトレース。
 - 1 – MQTTCLIENTTRACEMEDIUM、中レベルのトレース。
 - 2 – MQTTCLIENTTRACEMINIMUM、最小レベルのトレース。
 - 3 – MQTTCLIENTTRACEPROTOCOL、プロトコル・レベルのトレース。
 - 4 – MQTTCLIENTTRACEERROR、エラーのみのトレース。
 - 5 – MQTTCLIENTTRACESEVERE、サーバ・エラーのみのトレース。
 - 6 – MQTTCLIENTTRACEFATAL、致命的なエラーのみのトレース。
- ・ [URL] プロパティを、ブローカの URL とポート番号で設定します。“tcp” または “ssl” の後にドメイン名とポートを “:” で区切って続けます (例: “tcp://MQTTBroker.example.com:1883”)。通常、TLS 対応のエンド・ポイントは 8883 のポートを使用して構成されますが、これは必須ではありません。

4. [基本設定] でビジネス・サービスを有効にします。

5. プロダクションを実行します。

注釈 TLS の使用時は、証明書ファイルへの相対パスではなく、絶対パスを使用する必要があります。詳細は、“セキュリティ管理ガイド”の“TLS 構成の作成または編集”を参照してください。

3.2 パススルー・ビジネス・オペレーションの追加と構成

ビジネス・オペレーションをプロダクションに追加するには、管理ポータルを使用して相互運用対応ネームスペースで以下の操作を行います。

1. MQTT ブローカでユーザ名/パスワード認証を使用している場合、まず、その情報で認証情報を定義する必要があります（管理ポータルの [Interoperability]→[構成する]→[認証情報]）。TLS 認証を使用している場合は、“セキュリティ管理ガイド”の“TLS ガイド”を参照してください。
2. `EnsLib.MQTT.Operation.Passthrough` ビジネス・オペレーション・クラスのインスタンスをプロダクションに追加します。
3. メッセージを発行できるようにビジネス・オペレーションを構成します。
 - ・ オプションで [CleanSession] プロパティを設定します。[CleanSession] プロパティにチェックが付けられている場合、ブローカは接続から次の接続までの間、セッションについての情報を保持しません。
 - ・ [ClientID] プロパティを設定します。これはブローカに対するクライアントを指定します。ASCII でエンコードされ、長さが 1 ～ 23 文字である必要があります。ブローカへの接続を切断してから再接続する場合、ClientID が同じであれば、ブローカはその ID 情報を保持します。[ClientID] フィールドを空のままにすると、クライアントによって一意のクライアント ID が割り当てられます。
 - ・ [ConnectTimeout] プロパティを設定します。これは、ビジネス・サービスがブローカからの応答確認を待機する秒数を設定します。
 - ・ ログイン認証情報を使用している場合、[CredentialsName] プロパティを手順 1 で定義した認証情報名に設定します。
 - ・ [KeepAlive] プロパティを、クライアントからブローカへのメッセージの最大間隔（秒数）に設定します。
 - ・ オプションで、[LWTTopic] および [LWTMessage] プロパティを設定します。これらは、接続の Last Will and Testament（遺言）メッセージを設定します。接続が予期せずに失われたことをブローカが検出すると、これは指定したトピックとメッセージを使用してメッセージを発行します。
 - ・ [QOS] プロパティを設定します。これによって、ブローカとの接続におけるサービス品質が設定されます。値が 0 の場合、ビジネス・オペレーションはブローカからの確認応答を待機しないことを意味し、値が 1 の場合、ビジネス・オペレーションはブローカがメッセージを受信したという確認応答を待機することを意味します。
 - ・ 保持するメッセージを発行する場合、[Retained] プロパティにチェックを付けます。各サブスクリプションで保持されるメッセージは 1 つのみです。これは、最初のサブスクライバに送信されるメッセージです。その他のメッセージは、サブスクリプションが開始された後で発行される場合にのみ、サブスクライバに送信されます。
 - ・ TLS セキュリティを使用している場合は、[SSLConfigName] を TLS 構成の名前に設定します。
 - ・ [QOS] を 1 に設定した場合、[タイムアウト] プロパティを、ビジネス・サービスが確認応答を待機する秒数に設定します。
 - ・ [トピック] プロパティを設定します。これにより、メッセージの発行に使用するデフォルトのトピックが指定されます。トピックは通常、サブトピックのレベルが /（スラッシュ）で区切られた階層文字列です。オペレーションは、その入力パラメータで指定されたトピックを使用します。入力パラメータにトピックが含まれていない場合、デフォルトのトピックが使用されます。メッセージの発行時、トピックにワイルドカードを含めることはできません。
 - ・ ビジネス・サービスやビジネス・オペレーションで行われる MQTT アクションをトレースする場合は、[トレース] プロパティを選択して、[TraceLevel] プロパティを設定します。MQTT トレース・メッセージは、プロダクション・ログに書き込まれます。[TraceLevel] では以下の値を扱うことができます。

- 0 - MQTTCLIENTTRACEMAXIMUM、最大レベルのトレース。
 - 1 - MQTTCLIENTTRACEMEDIUM、中レベルのトレース。
 - 2 - MQTTCLIENTTRACEMINIMUM、最小レベルのトレース。
 - 3 - MQTTCLIENTTRACEPROTOCOL、プロトコル・レベルのトレース。
 - 4 - MQTTCLIENTTRACEERROR、エラーのみのトレース。
 - 5 - MQTTCLIENTTRACESEVERE、サーバ・エラーのみのトレース。
 - 6 - MQTTCLIENTTRACEFATAL、致命的なエラーのみのトレース。
- ・ [URL] プロパティを、ブローカの URL とポート番号で設定します。“tcp” または “ssl” の後にドメイン名とポートを “:” で区切って続けます (例: “tcp://MQTTBroker.example.com:1883”)。通常、TLS 対応のエンド・ポイントは 8883 のポートを使用して構成されますが、これは必須ではありません。

4. [基本の設定] でビジネス・オペレーションを有効にします。

5. プロダクションを実行します。

注釈 TLS の使用時は、証明書ファイルへの相対パスではなく、絶対パスを使用する必要があります。詳細は、“セキュリティ管理ガイド” の “TLS 構成の作成または編集” を参照してください。

3.3 MQTT メッセージの形式

パススルー・ビジネス・サービスが `EnsLib.MQTT.Message` メッセージを送信し、パススルー・ビジネス・オペレーションが `EnsLib.MQTT.Message` メッセージを受信します。`EnsLib.MQTT.Message` クラスは、`Ens.StringContainer` クラスに [トピック] プロパティを追加して拡張されたものです。MQTT トピックは、[トピック] プロパティに格納され、MQTT メッセージは、文字列コンテナ値に格納されます。

4

受信 MQTT アダプタと送信 MQTT アダプタの設定

ここでは、MQTT の受信および送信アダプタの設定に関するリファレンス情報を提供します。
“[プロダクションの管理](#)” の “[すべてのプロダクションに含まれる設定](#)” も参照してください。

4.1 まとめ

MQTT アダプタには以下の設定があります。

グループ	設定
MQTT	[クリーン・セッション] 、 [認証情報名] 、 [クライアント ID] 、 [接続タイムアウト] 、 [キープ・アライブ] 、 [LWT トピック] 、 [LWT メッセージ] 、 [QOS] 、 [保持] 、 [SSL 構成名] 、 [タイムアウト] 、 [トピック] 、 [トレース] 、 [トレース・レベル] 、 [URL]

残りの設定はすべてのビジネス・サービスに共通のものです。詳細は、“[すべてのビジネス・サービスに含まれる設定](#)” を参照してください。

4.2 クリーン・セッション

ブローカに対するセッションでクリーン・セッションを使用するかどうかを決定します。

4.3 クライアント ID

ブローカに対してこのクライアントを特定する文字列です。ASCII でエンコードされ、長さが 1 ～ 23 文字である必要があります。

これを指定しないと、クライアントによって一意の ID が作成されます。

4.4 接続タイムアウト

接続タイムアウトです。ビジー・サーバに接続すると時間がかかり、このタイムアウトを使用して、早期接続障害を回避できます。接続の試行が失敗となるまでの待機秒数を指定します。

4.5 認証情報名

MQTT ブローカへのアクセスに使用される認証情報の値セットの ID 名です。認証情報項目で定義されるユーザ名とパスワードは ASCII でエンコードされる必要があります。ブローカでログイン認証情報が不要な場合、これは必要ありません。

4.6 キープ・アライブ

クライアントからブローカへのメッセージの最大間隔 (秒数) を指定します。

4.7 LWT トピック

LWT (Last Will and Testament、遺言) トピックです。指定する場合、ASCII でエンコードする必要があります。ネットワーク・エラーによりクライアントが切断された場合、LWTMessage は LWTTopic のサブスクライバに配信されます。

4.8 LWT メッセージ

LWT (Last Will and Testament、遺言) メッセージです。指定する場合、ASCII でエンコードする必要があります。

4.9 QOS

必要なサービス品質を特定します。以下の 2 つの値のいずれかとなります。

- ・ 0 – QOSFireAndForget、ブローカからの応答を待機しません。
- ・ 1 – QOSWaitForDelivery、ブローカからの応答を待機し、応答がなければエラーを発行します。

4.10 保持

ブローカに、メッセージを保持する必要があるかどうかを示すフラグです。

4.11 SSL 構成名

ブローカとの通信に使用する TLS 構成の ID 名です。TLS 通信が必要な場合にのみ使用されます。

注釈 TLS の使用時は、証明書ファイルへの相対パスではなく、絶対パスを使用する必要があります。詳細は、“セキュリティ管理ガイド”の“TLS 構成の作成または編集”を参照してください。

4.12 タイムアウト

メッセージの送信（確認応答）または受信を待機する秒数で表されるタイムアウト値です。

4.13 トピック

発行または購読するトピックの名前です。トピックは ASCII でエンコードする必要があります。トピックは通常、サブトピックのレベルが /（スラッシュ）で区切られた階層文字列です。サブスクリプションでは、トピックにトピック・レベルとしてワイルドカードを含めることができます。

- ・ +（プラス記号）－そのレベルのすべての文字列に一致します。例えば、“/temperatures/+/highest”は、トピック“temperatures/Paris/highest”や“temperatures/Bangkok/highest”に一致しますが、“temperatures/Paris/highest/today”には一致しません。プラス記号は、同じレベル内の他の文字と組み合わせることはできません。
- ・ #（シャープ記号）－最後のレベルのみで、任意の文字列および任意のサブトピックに一致します。例えば、“temperatures/Paris/#”は、“temperatures/Paris/highest”、“temperatures/Paris/highest/today”、“temperatures/Paris/lowest/record/thiscentury”のトピックすべてに一致します。

4.14 トレース

MQTT トレース機能を有効または無効にします。MQTT トレース・メッセージは、ビジネス・サービスまたはオペレーションのプロダクション・ログに書き込まれます。

4.15 トレース・レベル

MQTT ライブラリのトレース・レベルです。これを設定すると、トレース情報の必要な詳細がログに記録されます。トレース・レベルでは以下の値を扱うことができます。これらは `%Net.MQTT` インクルード・ファイルで定義されます。

- ・ 0 – MQTTCLIENTTRACEMAXIMUM、最大レベルのトレース。
- ・ 1 – MQTTCLIENTTRACEMEDIUM、中レベルのトレース。
- ・ 2 – MQTTCLIENTTRACEMINIMUM、最小レベルのトレース。
- ・ 3 – MQTTCLIENTTRACEPROTOCOL、プロトコル・レベルのトレース。
- ・ 4 – MQTTCLIENTTRACEERROR、エラーのみのトレース。

- ・ 5 – MQTTCLIENTTRACESEVERE、サーバ・エラーのみのトレース。
- ・ 6 – MQTTCLIENTTRACEFATAL、致命的なエラーのみのトレース。

4.16 URL

通信するブローカの URL です。“tcp” または “ssl” の後にドメイン名とポートを “:” で区切って続けます (例 : “tcp://BIGBADAPPLE.local:1883”)。通常、TLS 対応のエンド・ポイントは 8883 のポートを使用して構成されますが、これは必須ではありません。**URL** は ASCII でエンコードする必要があります。