

INDUSTRIAL
EDGE
SOLUTION
WITH
HARD REALTIME
CAPABILITIES

RT-edge

Micronet.Co,

マイクロネット
Micronet

INDUSTRIAL REALTIME EDGE COMPUTERS

外部通信 Container




RT-edge Software External Communication I/F Service Container
ユーザーズマニュアル

 **株式会社マイクロネット**
<http://www.mnc.co.jp>

TEL: +81(0)299-90-1733

FAX: +81(0)299-92-8557

本書で使用するマークについて

	ノート: 操作方法や手順等の補足情報や注釈を説明しています。
	情報: 製品を利用する上で有効な豆知識となる説明をしています。
	警告: 製品仕様上注意が必要な事象について説明しています。

Windows、Visual Studio は、米国 Microsoft Corporation の米国およびその他の国における商標または登録商標です。
INtime は、米国 TenAsys Corporation の登録商標です。

TenAsys®, INtime®, eVM® and iRMX® are registered trademarks in USA of the TenAsys Corporation.

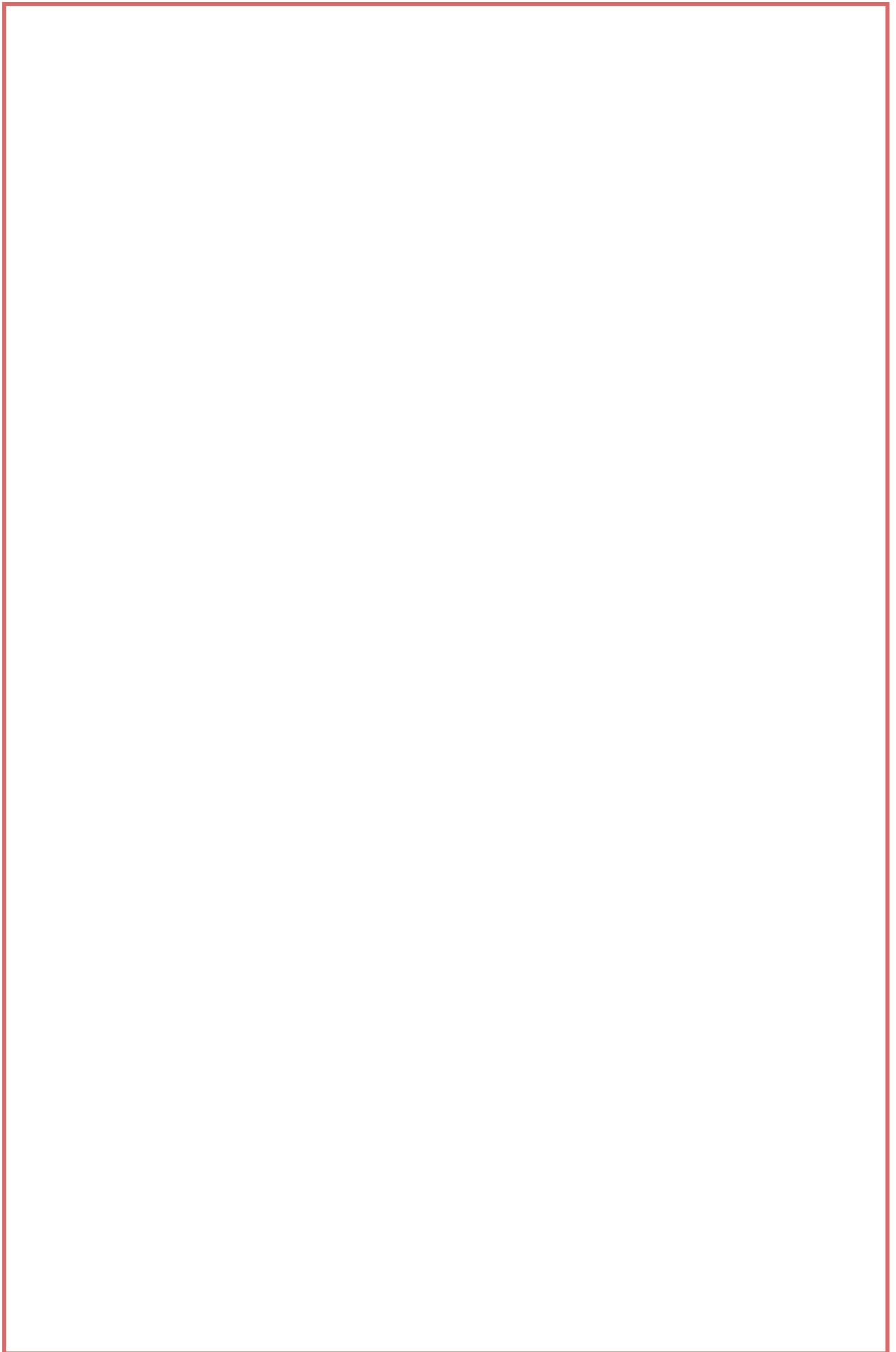
その他、本書に記載されている会社名、商品名は、各社の商標または登録商標です。

本書の内容を無断で転載することは禁止されています。

本書の内容に関しては、予告なしに変更することがあります。あらかじめご了承ください。

目次

用語解説	4
関連資料	6
1. 概要	7
1.1. RT-edge とサービスコンテナ	7
1.2. 外部通信コンテナ	9
2. 仕様	11
2.1. 動作環境	11
2.2. 使用可能タグ数	11
2.3. 対応プロトコルと使用ドライバ	11
3. コンテナ導入フロー	12
4. インストール	13
4.1. ファイル	13
4.2. ファイルのインストール	14
4.3. 起動設定	15
4.4. 動作確認	15
4.5. 終了設定	17
5. 環境設定	18
5.1. Edge コントローラ設定	18
6. 設定概要	19
6.1. ECI 設定	19
7. 外部通信サービスコンテナ設定	22
7.1. RT-edge Object 設定概要	22
7.2. ECI(コンテナ設定情報ファイル)の編集	23
7.3. コンテナプロパティ設定	26
8. 設定サンプル	27
9. 動作確認	28
9.1. 動作確認手順	28
9.2. サービスインジケータタグ	29
9.3. RT-edge タグデータの妥当性について	29
10. 付録	30
10.1. サービスプロパティタグ	30
10.2. サービスメッセージ	32
10.3. ECI Address 定義	33
10.4. 設定サンプル	35
10.5. 環境設定 その他仕様制限事項	37
10.6. トラブルシューティング	38



用語解説

本ドキュメントにおいて使用される用語・略称について説明します:

表 1. 用語集

用語	説明
RT-edge	エッジコンピューティングを軸とする IT の情報処理と、FA における装置・機器の制御を融合し、密度の高い高頻度データ利用を可能とするソフトウェアプラットフォームです。 FA で要求されるハードリアルタイム制御を組み込むことで、情報処理と機器・装置制御を可能とするエッジコントローラを構成することができます。
RT-edge 基本ソフトウェア	RT-edge 機能の核となる機能・ライブラリを実装するパッケージソフトウェア製品です。
IoT ゲートウェイ	IoT において、端末とインターネットを介した遠隔サーバー(クラウド)がデータのやりとりをする際、中継する役割を担う機能。サーバーや送信経路であるインターネット負荷の軽減をします。
IT システム	オンプレミスもしくはクラウドを活用した業務システムやアプリケーション。
INtime	INtime for Windows: Windows と協調動作可能なリアルタイムカーネル拡張ソフトウェアです(RTOS ソフトウェア)。 INtime Distributed RTOS(dRTOS): Windows OS を必要とせず、スタンドアロンで動作するリアルタイム OS です。
RTA	RealTime Application: リアルタイムアプリケーションの略称。INtime 上で動作するローダブルプロセスの拡張子です。INtime 上で動作するローダブルアプリケーションは、RTA という拡張子を持ちます。
RSL	Realtime Shared Library: リアルタイム共有ライブラリの略称。INtime 上でアプリケーションがロード可能なライブラリです。Windows 上で使用される DLL(Dynamic Link Library)のようなものです。RTA から使用されるライブラリインタフェース等は、こちらを使用して作成することができます。
API	Application Programming Interface: アプリケーションプログラミングインタフェースの略称。RT-edge ではデバイスへのアクセスインタフェースとして API ライブラリを提供しています。
エッジアプリケーション	RT-edge 内コンテナにより集積されたデータ(RT-edge Object)を活用、処理実行するソフトウェアです。
エッジコンピューティング	RT-edge 内で稼働する制御コンテナソフトウェアにより装置・機器から収集した高密度なデータをリアルタイムに収集、分析、フィードバックします。IT システムとの情報連携。
オンプレミス	サーバーやソフトウェア等の情報システム、アプリケーション等のソフトウェアを管理する施設内に設置して運用すること。
クラウド	サーバーやストレージ等のインフラやソフトウェアを必要とせず、必要な IT リソースが、インターネットを通じてオンデマンドで得られる形態、サービス。
産業用 PC	高信頼性、耐環境性、長期供給等の特徴をもつ産業用途の PC。
データ収集	診断、分析を行う対象となるデータを集積する処理。
データ加工	集積されたデータを利用しやすい形に変更する処理。
サービス/EgService	RT-edge システムを構成する機能プロセス(rta/exe)です。
タグ/EgTag	瞬時値データ値 1 つを示すオブジェクトです。ユニーク名とグローバルなスコープを持ち、全ての EgService から読み書きが許されたオブジェクトです。タグは生成時にデータ型が確定され変更はできません。
リンクタグ	同一名称のタグを重複生成した場合に自動的に別名称で生成されるタグを指します。 通常のタグと同様、グローバルなスコープを持ち、全ての EgService から読み書きが許されたオブジェクトです。一つのタグに対し、異なるプロパティ情報を定義したい場合に使用します。
データセット/EgDataset	タグ 1 つ以上の組み合わせでデータ並び順(データ構造)を定義する名前付きオブジェクトです。
コレクタ/EgCollector	データセットに定義されたデータ構造に従って、同時刻のバイナリデータ列で生成し、データレコードとしてメールボックスに送信するオブジェクト (スレッド) です。

用語	説明
メールボックス/EgMailBox	時系列なデータセット、または時系列メッセージを FIFO で蓄えることができ、また受信イベントとして処理できるオブジェクトです。
タグ参照/TagRef	タグの参照として使用するオブジェクトです。タグの名前を保持し値は保持しません。サービスコンフィグファイルでデータセットの収集用タグとして定義することや、サービス内のオブジェクトとして定義することでサービスのメンバ変数として使用することができます。
コレクタ参照/CollectorRef	コレクタの参照として使用するオブジェクトです。コレクタの名前を保持しそれ以外のオブジェクトは保持しません。サービスコンフィグファイルでサービス内のオブジェクトとして定義することでサービス内のメンバ変数として使用することができます。
メッセージ	メールボックスで扱われる 1 レコード分のデータ、またはサービス間のコマンド、応答の電文です。
フレームワーク	フレームワークは、アプリケーションが API を組み合わせて実装するよくある処理についてマクロ化、自動化したものでサービスコンフィグファイルの記述により自動処理させることができます。
RT-edge コンテナ設定情報 (ECI)	RT-edge コンテナが RT-edge Object として展開する入出力データ定義の他、RT-edge コンテナフレームワークが、オブジェクト生成やコンテナサービス等自動処理するための定義設定情報(XML 型式)。
入力	RT-edge システムを中心に見た場合、外部の情報を RT-edge システムへ取り込む方向性のデータの流れを意味します。
出力	RT-edge システムを中心に見た場合、RT-edge システムが持つデータを外部に書き出す方向性のデータの流れを意味します。
RTCD	Realtime Common Data の略称。RT-edge システム上で最もベースとなる共有データ構造機能です。
RT-edge Object	RT-edge システム上で使用可能なオブジェクト群（機能群）の総称です。 例えば、センサーや装置から収集したデータをアプリケーション間で受け渡しを行う場合に使用するタグ、アプリケーション間でメッセージのやり取りを行う場合のメールボックス等、アプリケーション間でデータの受け渡しを行うケースにおいて利用されるオブジェクトです。 RT-edge Object は Windows アプリケーション間、INtime®アプリケーション間、Windows-INtime®アプリケーション間いずれの場合も利用可能です。
通信プロトコル	異なるデバイスやコンピュータシステム、ソフトウェアなどが互いに通信するために制定された規格です。
Edge I/O ポート	通信対象の入出力データエリアを、連続した一つのアドレス空間として定義した RT-edge 独自のアドレス表現です。
SSL	SSL は Secure Socket Layer の略で、インターネット上でデータを暗号化して送受信できるプロトコルです。公開鍵暗号を応用したデジタル証明書による通信相手の認証、共通鍵暗号による通信の暗号化、ハッシュ関数による改ざん検知などの機能を提供しています。
TLS	TLS は Transport Layer Security の略で、SSL の後継規格です。SSL 3.0 が改訂される際に同時に名称が改められて TLS1.0 が生まれました。
FTPS	FTPS は、File Transfer Protocol over SSL/TLS の略で、FTP で送受信するデータを SSL(Secure Socket Layer)/TLS(Transport Layer Security)で暗号化する通信プロトコルです。

関連資料

RT-edge 製品に含まれる資料

表 2 .RT-edge 関連資料

名称	ファイル名	内容
RT-edge ユーザーズマニュアル	DOCRTEGEUSER.pdf	RT-edge システムの全般的な説明が記載されています。
RT-edge API リファレンス	DOCRTEGEAPI.pdf	RT-edge API の使用方法が記載されています。
RT-edge コンテナ作成マニュアル	DOCRTEGGSRV.pdf	RT-edge コンテナの構造、サンプルプロジェクトを利用した作成方法等について記載されています。

1. 概要

1.1. RT-edge とサービスコンテナ

RT-edge とは、エッジコンピューティングを軸とする IT の情報処理と、FA における装置・機器の制御を融合し、密度の高い高頻度データ利用を可能とするソフトウェアプラットフォームです。

RT-edge の利用により、装置やセンサーからの高密度なデータ収集、分析だけでなく、提供される開発ライブラリキットを使用し、タグデータをレジスタとした機器制御を行うハードリアルタイムエッジアプリケーションの開発が可能です。

サービスコンテナ

RT-edge の処理ターゲットは、エッジコンピューティングを軸とした IT 情報処理(IT-Process)と、ミリ秒精度のハードリアルタイム性を要求される FA 制御(FA-Control)に分類され、ターゲットの機能に特化した専門処理サービスをコンテナ(サービスコンテナ)と呼びます。

IT 情報処理ターゲットは上位層にあり、主に外部システムからの要求指示の受付や、外部システムへのデータ公開、通信等を担う要素となります。IT 情報処理サービスコンテナは、制御システムのコンソール画面や外部システムから WEB ブラウザ経由でのアクセス機能、制御データ情報を外部クラウドストレージに保存する機能等、上位システムとの接続・インターフェースを提供します。

一方、FA 制御ターゲットは下位層に位置し、主に通信やハードウェアへの直接 I/O 入出力等により装置・機器制御を担う要素です。FA 制御サービスコンテナは、産業用フィールドバスやコントローラ通信プロトコルによるロボット制御、計測機器からのデータロギング、デジタルパルス出力等、装置・機器へのアクセスを提供します。

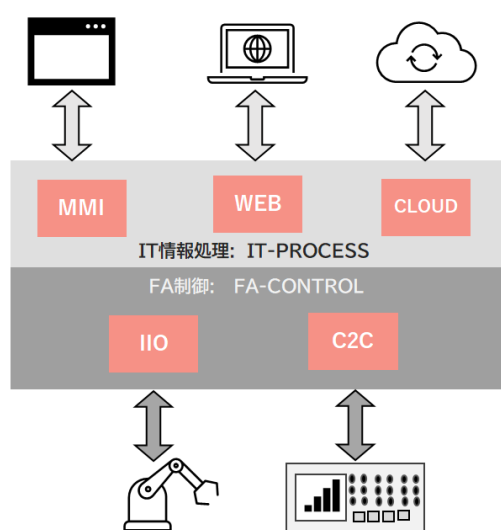


図 1. ターゲットとコンテナ

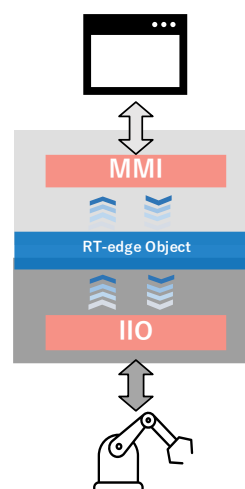


図 2. コンテナの役割



サービスコンテナ例

WEB: IT 情報処理ターゲット内には、IIS(Internet Information Service)を介し、制御情報をインターネット上に公開する WEB サービスコンテナ
 IIO: ロボットアーム制御に特化した産業 I/O サービスコンテナ

サービスコンテナはターゲットに特化した入出力データを RT-edge Object であるシステム内でグローバルにアクセス可能なタグ情報としてリンクし、このタグ情報のコレクションを公開します。

サービスコンテナは、タグ情報コレクションや、動作・挙動を決定するパラメータ設定と、ターゲット処理に特化した一つ以上の実行処理の集合体です:

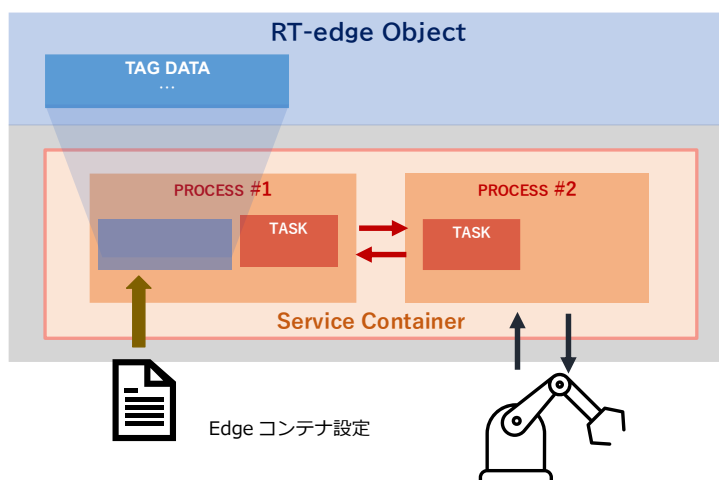


図 3. サービスコンテナの構造

サービスコンテナ例

PROCESS #1

RT-edge Object を使用する処理プロセス (サービスハンドラ)

PROCESS #2

ターゲット特化処理プロセス

Edge コンテナ設定情報

タグデータコレクション等コンテナ設定

1.2. 外部通信コンテナ

外部通信コンテナは、Windows ネットワーク経由で外部サーバーアプリケーションと接続し、RT-edge Object を介したデータ送受信機能を提供します。

MQTT や TCP を用いた通信や、FTPS を用いて外部 FTP サーバーへのファイルのアップロード、またはファイルのダウンロードを行う機能として動作します。

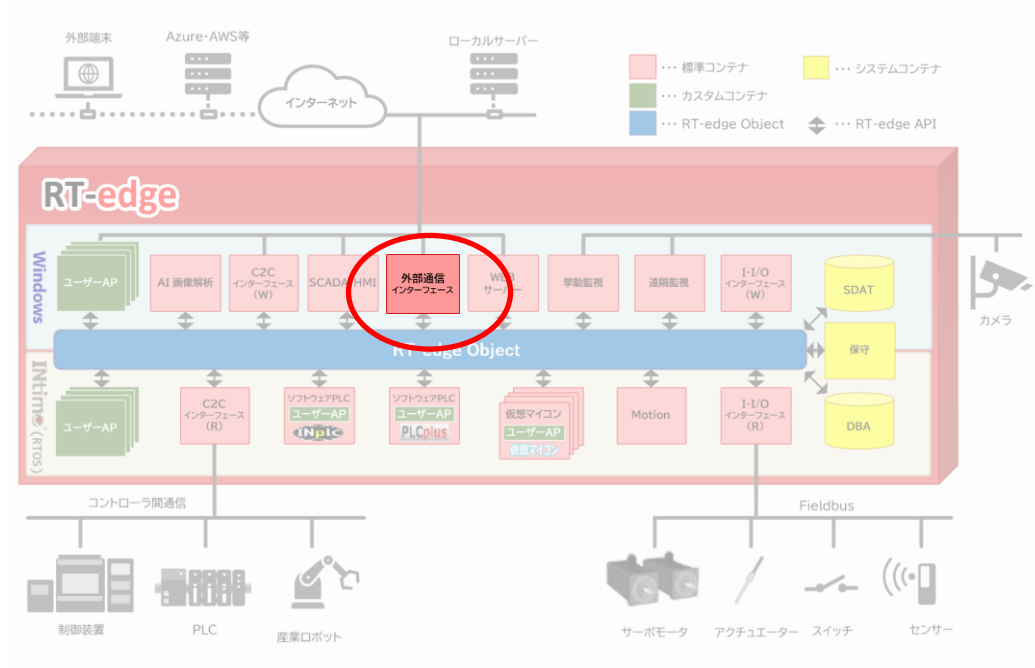


図 4.RT-edge 関連図

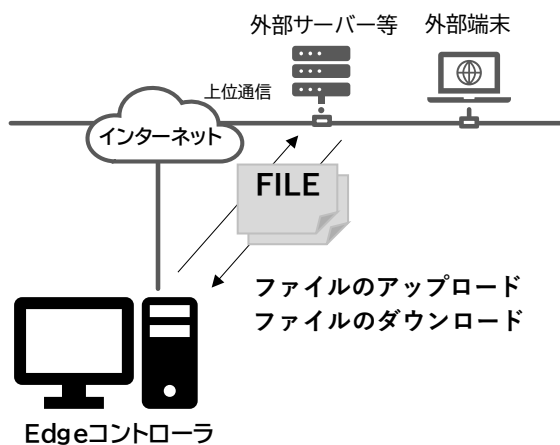


図 5. 外部サーバーを例としたネットワーク接続図

構成要素

外部通信コンテナは以下コンポーネントから構成されます:

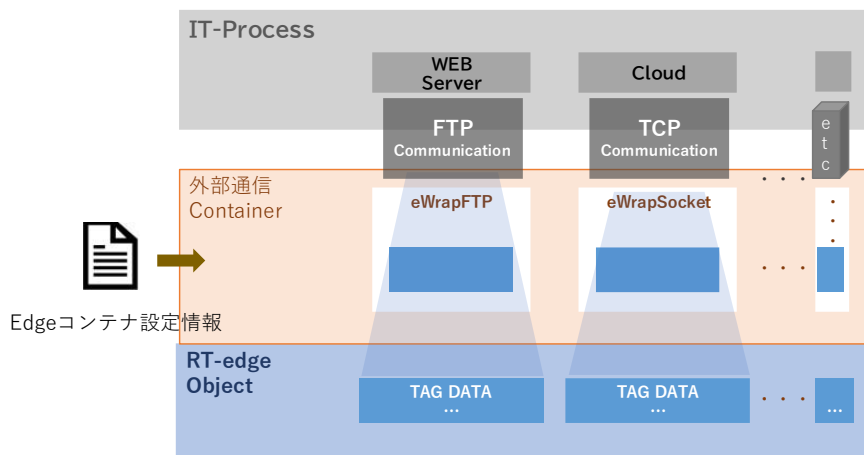


図 6. 外部通信コンテナ構成

表 3. 外部通信コンテナ構成要素

コンポーネント	内容
EgExtIF.exe	RT-edge 外部通信コンテナに含まれるソフトウェアコンポーネントです。 ECI(RT-edge コンテナ設定情報:XML 形式ファイル)を読み込み、各通信プロトコルを利用した外部通信を行い Web サーバーや Cloud などの外部データを RT-edge Object に展開、またサーバーからのファイルダウンロードやサーバーへのファイルのアップロードを行います。
eWrapFTP.dll	RT-edge 外部通信コンテナに含まれるソフトウェアコンポーネントです。 外部サーバーのファイルを RT-edge コントローラへダウンロード、RT-edge 内のファイルを外部サーバーへアップロードすることが可能なライブラリです。
eWrapSocket.dll	RT-edge 外部通信コンテナに含まれるソフトウェアコンポーネントです。 TCP を用いた通信でデータの送受信を行う場合に使用します。 TCP/IP クライアントとして動作します。
eWrapMQTT.dll	RT-edge 外部通信コンテナに含まれるソフトウェアコンポーネントです。 MQTT を用いた通信でデータの送受信を行う場合に使用します。 MQTT Publisher および Subscriber として動作します。
HiFTP.dll	FTP 通信 ドライバ用ライブラリです。
HiTCP.dll	TCP 通信 ドライバ用ライブラリです。
HiMQTT.dll	MQTT 通信 ドライバ用ライブラリです。
RT-edge コンテナ設定情報(ECI)	外部通信コンテナの動作設定情報です。



本ドキュメントでは、主に外部通信コンテナの利用方法について説明します。
RT-edge 基本ソフトウェア、他サービスコンテナについては各々のマニュアルを参照ください。

2. 仕様

2.1. 動作環境

- RT-edge 基本ソフトウェアバージョン 3.4.0 以降
- Windows OS x64
- .Net Framework 4.6

2.2. 使用可能タグ数

使用可能なタグ数は、RT-edge 基本ソフトウェア 仕様に依存します。

2.3. 対応プロトコルと使用ドライバ

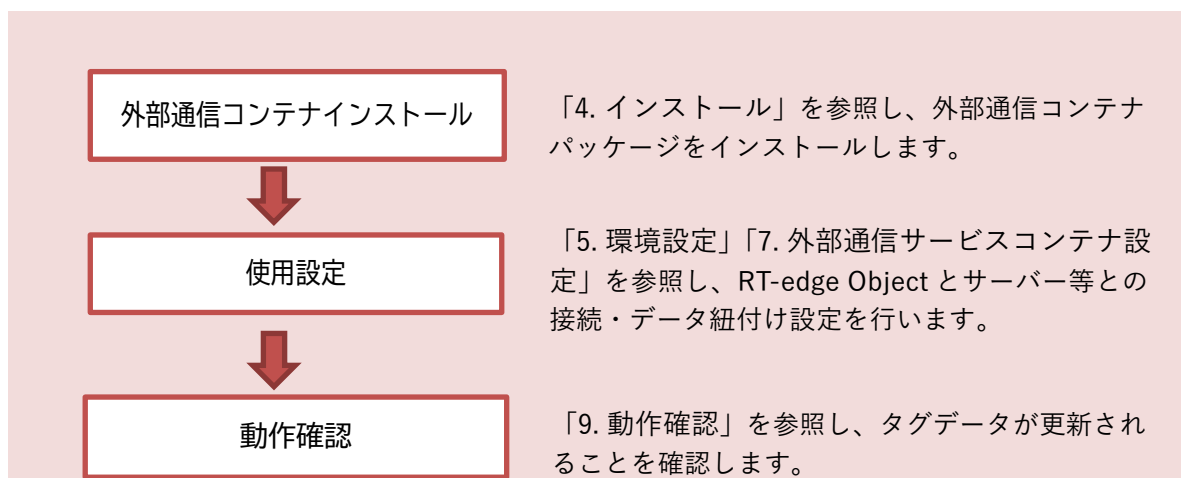
外部通信コンテナでは以下のプロトコルをサポートしています。また使用するドライバ製品とバージョン、対応する通信プロトコル名を記載します。

表 4. 対応プロトコル一覧

No	通信プロトコル名	使用ドライバ	ドライババージョン
1	FTP	micronet 社製 FTP 通信 Driver For Windows 使用	1.0.0 以上
2	TCP	micronet 社製 TCP 通信 Driver For Windows 使用	1.0.0 以上
3	MQTT	micronet 社製 MQTT 通信 Driver For Windows 使用	1.0.0 以上

通信プロトコルごとの仕様・制限事項は 10.5. 環境設定 その他仕様制限事項を参照ください。

3. コンテナ導入フロー



4. インストール

予め RT-edge を「C:\RTedge」にインストールした状態を前提としています。

4.1. ファイル

外部通信コンテナパッケージには以下のファイルコンポーネントが含まれています：

表 5.外部通信コンテナパッケージコンポーネント（全コンポーネント）

配置先 フォルダ階層	ファイル名	説明
RTedge\bin	EgExtIF.exe	各通信プロトコルを利用した外部通信を行い Web サーバーや Cloud などの外部データを RT-edge Object に展開、またサーバーからのファイルダウンロードやサーバーへのファイルのアップロードを行います。
RTedge\bin	eWrapFTP.dll	FTP 通信を利用し、外部サーバーのファイルを RT-edge コントローラへダウンロード、RT-edge 内のファイルを外部サーバーへアップロードすることが可能なライブラリです。
RTedge\bin	eWrapSocket.dll	TCP 通信ドライバと RT-edge Object とを繋ぐライブラリです。TCP 通信でデータの送受信を行う場合に使用します。
RTedge\bin	eWrapMQTT.dll	MQTT 通信ドライバと RT-edge Object とを繋ぐライブラリです。MQTT 通信でデータの送受信を行う場合に使用します。
RTedge\bin	HiFTP.dll	FTP 通信 ドライバ用ライブラリです。
RTedge\bin	HiTCP.dll	TCP 通信 ドライバ用ライブラリです。
RTedge\bin	HiMQTT.dll	MQTT 通信 ドライバ用ライブラリです。
RTedge\bin	EgExtIF.xml	外部通信コンテナ用 RT-edge 設定情報です。本設定ファイルの構成により外部通信コンテナが生成するタグの設定を行います。
RTedge\doc	DOCRTEGESRC_ExtIF.pdf	本ドキュメントです。



各通信プロトコルに対応したライブラリは、必要に応じてインストールを行います。

4.2. ファイルのインストール

外部通信コンテナ用のモジュールインストール方法については、別紙インストール手順書_EXTIFを参照ください。

4.3. 起動設定

RT-edge におけるサービスコンテナ、および関連サービス・アプリケーションの設定は、RT-edge ブートストラッパー設定により行います。外部通信コンテナの起動設定も同様、RT-edge ブートストラッパー設定に準拠します:

- 1) C:\RT-edge\bin\EgBoot.xml をテキストエディタで開きます。
- 2) RTedge エlement内の Services Element内に、外部通信サービスコンテナ用のElement(Service Element)を追加します。Service Name には使用する通信プロトコル名を指定します。また、Argument 属性に通信プロトコルごとに用意されているライブラリの指定を行います。

Argument="-DLL DLL 名"

DLL 名には以下の名前を入力します。

通信プロトコル	RSL 名
FTP	eWrapFTP.dll
TCP	eWrapSocket.dll
MQTT	eWrapMQTT.dll

以下では、FTP 通信プロトコルを用いた外部通信サービスコンテナの起動設定の内容です。

```
<?xml version="1.0" encoding="utf-8"?>
<RTedge xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  :
  <Services>
    <Service Name ="EgBoot" Argument="RTCD=NodeA;TagMaxNum=10000" >
    </Service>
    <Service Name="EgLog" Path="EgLog.exe" Argument="DispNumMax=500" >
    </Service>
    <Service Name="EgTime" Path="EgTime.exe" >
    </Service>
    <Service Name="FTP" Path="EgExtIF.exe" Argument="-DLL eWrapFTP.dll" >
    </Service>
  </Services>
</RTedge>
```

図 7.EgExtIF 起動登録

4.4. 動作確認

RT-edge ソフトウェアを起動し、外部通信サービスコンテナが正常に起動することを確認します。ここでは「図 7.EgExtIF 起動登録」で記載の FTP 通信プロトコルを使用した例を用いて説明します:

- 1) RT-Edge ソフトウェア(C:\RT-edge\EgBoot.exe)を開始します。
- 2) RT-Edge オブジェクトブラウザ(C:\RT-edge\EgBrow.exe)を起動します。

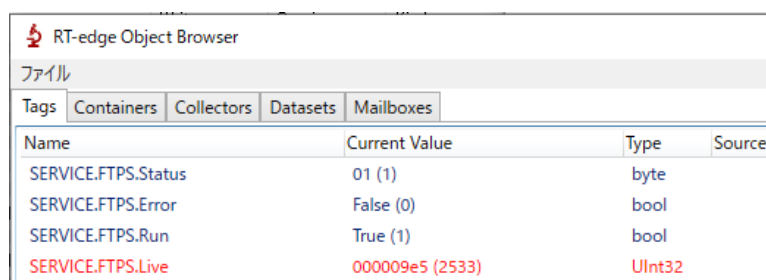


RT-edge 起動直後に RT-edge オブジェクトブラウザを起動すると初期化中の為、想定されるタグが表示されない場合があります。一度 RT-edge オブジェクトブラウザを終了し、再度起動させてください。

- 3) サービスインジケータタグから、正常状態であることを確認します(サービスインジケータタグについては「9.2. サービスインジケータタグ」を参照ください)。

正常状態

- SERVICE.FTP.Status が 01(1) であること
- SERVICE.FTP.Run が true(1) であること
- SERVICE.FTP.Error が False(0) であること
- SERVICE.FTP.Live が 増加していくこと



Name	Current Value	Type	Source
SERVICE.FTPS.Status	01 (1)	byte	
SERVICE.FTPS.Error	False (0)	bool	
SERVICE.FTPS.Run	True (1)	bool	
SERVICE.FTPS.Live	000009e5 (2533)	UInt32	

図 8. RT-edge オブジェクトブラウザ起動時の様子

上記の状態になっていない場合には、以下のトラブルシューティングをご参照ください。

「サービスインジケータの.Error が true になっています。」

「サービスインジケータの.Run が false になっています。」

4.5. 終了設定

RT-edge におけるサービスコンテナ、および関連サービス・アプリケーションの終了は、RT-edge 終了サービス「EgShDown」により行います。外部通信サービスコンテナの終了設定も同様、RT-edge 終了サービス設定に準拠します:

- 1) C:\RTedge\bin\EgShDown.xml をテキストエディタで開きます。
- 2) ArrayOfAnyType エlement 内に、外部通信コンテナの使用プロトコル名を追加します。

```
<anyType xsi:type="xsd:string">FTP</anyType>
<anyType xsi:type="xsd:string">TCP</anyType>
<anyType xsi:type="xsd:string">MQTT</anyType>
```

- 3) 編集を保存し、ファイルを閉じます。
- 4) 追加結果は以下のようになります。

```
...:
<ArrayOfAnyType ...>
  <anyType xsi:type="xsd:string">EgLog</anyType>
  <anyType xsi:type="xsd:string">EgTime</anyType>
  <anyType xsi:type="xsd:string">EgBoot</anyType>
  <anyType xsi:type="xsd:string">FTP</anyType>
  <anyType xsi:type="xsd:string">TCP</anyType>
  <anyType xsi:type="xsd:string">MQTT</anyType>
</ArrayOfAnyType>
```

図 9.EgExtIF コンテナ終了登録

RT-edge ソフトウェアの終了時は、C:\RTedge\bin\EgShDown.exe を実行します。

5. 環境設定

5.1. Edge コントローラ設定

外部通信コンテナでは、各プロトコルに応じて Edge コントローラの IP アドレス等の設定が必要です。通信プロトコルごとに設定する内容、通信プロトコルごとの仕様については、10.5. 環境設定 その他仕様制限事項をご参照ください。

6. 設定概要

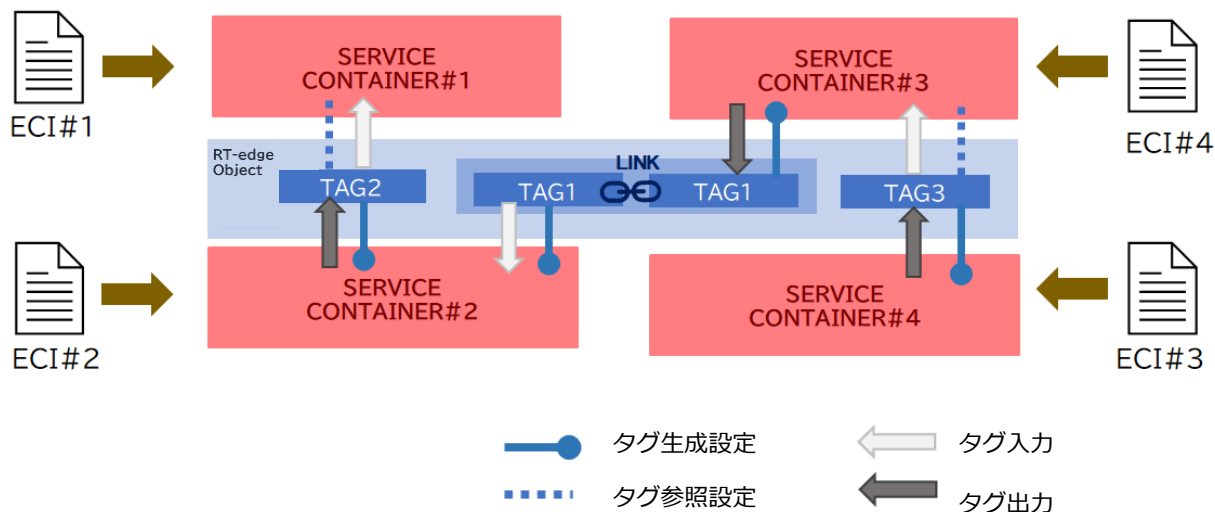
サービスコンテナ設定により担当するターゲットのデータとタグとの接続が可能となります。

サービスコンテナにおける基本設定は、タグ、データセットの定義を主とした RT-edge Object 設定と、サービスコンテナの入出力周期やプライオリティ設定等、コンテナプロパティ設定に分類されます：

設定項目	説明
RT-edge Object 設定	<p><u>タグ設定</u></p> <p>ローカルタグ生成設定・リンクタグ生成設定</p> <p>タグ参照設定</p> <p><u>データセット設定</u></p> <p>タグ・コレクション定義</p> <p>周期・プライオリティ設定</p>
コンテナプロパティ設定	<p>データ更新方式(オンデマンド・サイクリック(周期設定))</p> <p>プライオリティ設定等</p> <p>※コンテナプロパティ値は各サービスコンテナにより実装が異なります。</p>

6.1. ECI 設定

RT-edge Object 設定、プロパティ設定は、コンテナ毎に定義する設定情報(ECI: RT-edge コンテナ設定情報)に基づきます。ECI ファイルは XML 形式のテキストファイルとして生成されています：



各サービスコンテナは、タグ・リンクタグ生成設定を行います。

タグに対する入出力方向設定・参照設定を行います。

RT-edge Object 設定

RT-edge Object 設定では、ECI ファイル内で編集する XML タグの編集要素は以下のように定義されています:

設定項目	設定手順												
タグ設定	<p><u>ローカルタグ</u></p> <p><Tags>エレメント内に、<Tag>を生成します。</p> <p>サービスコンテナ独自の名称(一意名)で<Tag Name=>の設定を行います。</p> <p><u>リンクタグ</u></p> <p><Tages>エレメント内に、<Tag>を生成します。</p> <p>他サービスコンテナの提供するタグと同名で<Tag Name=>の設定を行います。</p> <p>Tag</p> <table> <tr> <th>キーワード</th><th>説明</th></tr> <tr> <td>Name</td><td>公開するタグ名を設定します。</td></tr> <tr> <td>Type</td><td>RT-edge データ型に関連する型定義値を設定します。 ※参照: データ型</td></tr> <tr> <td>Size</td><td>タグデータサイズを指定します。 ※参照: データ型</td></tr> <tr> <td>Address</td><td>サービスコンテナにおけるデータ取得元、宛先となるアドレス情報を指定します。本アドレス書式は、サービスコンテナ毎に異なります。</td></tr> <tr> <td>Comment</td><td>タグに対するコメントを設定します。</td></tr> </table>	キーワード	説明	Name	公開するタグ名を設定します。	Type	RT-edge データ型に関連する型定義値を設定します。 ※参照: データ型	Size	タグデータサイズを指定します。 ※参照: データ型	Address	サービスコンテナにおけるデータ取得元、宛先となるアドレス情報を指定します。本アドレス書式は、サービスコンテナ毎に異なります。	Comment	タグに対するコメントを設定します。
キーワード	説明												
Name	公開するタグ名を設定します。												
Type	RT-edge データ型に関連する型定義値を設定します。 ※参照: データ型												
Size	タグデータサイズを指定します。 ※参照: データ型												
Address	サービスコンテナにおけるデータ取得元、宛先となるアドレス情報を指定します。本アドレス書式は、サービスコンテナ毎に異なります。												
Comment	タグに対するコメントを設定します。												
参照	<p><u>入力参照</u>: <TagRefs_IN> エレメント内</p> <p><u>出力参照</u>: <TagRefs_OUT> エレメント内</p> <p><TagRef Name=>に参照するタグを指定します。</p> <p>TagRef</p> <table> <tr> <th>キーワード</th><th>説明</th></tr> <tr> <td>Name</td><td>参照するタグを指定します。</td></tr> </table>	キーワード	説明	Name	参照するタグを指定します。								
キーワード	説明												
Name	参照するタグを指定します。												
データセット設定	<p><Datasets>エレメント内に、<Dataset></p> <p>タグ・コレクション定義</p> <p><Dataset Name=>にデータセット名を指定します。</p> <p>Dataset エレメント内に、<TagRefs>エレメントを作成します。</p> <p><TagRef Name=>に参照するタグを指定します。</p> <p>TagRef</p> <table> <tr> <th>キーワード</th><th>説明</th></tr> <tr> <td>Name</td><td>参照するタグを指定します。</td></tr> </table> <p>※Dataset 内に TagRef オブジェクトを列挙します。</p>	キーワード	説明	Name	参照するタグを指定します。								
キーワード	説明												
Name	参照するタグを指定します。												

設定項目	設定手順										
周期・プライオリティ 設定	<p><Collectors>エレメント内に、<Collector>を作成し</p> <p><Collector Name=>に名称を設定します (Dataset を収集する機能名)</p> <p>Collector</p> <table> <tr> <th>キーワード</th><th>説明</th></tr> <tr> <td>Name</td><td>Dataset 収集機能名を指定します</td></tr> <tr> <td>Interval</td><td>収集周期を指定します (1ms 単位)</td></tr> <tr> <td>Priority</td><td>プライオリティを設定します</td></tr> <tr> <td>DatasetName</td><td>収集するデータセット名を指定します。</td></tr> </table>	キーワード	説明	Name	Dataset 収集機能名を指定します	Interval	収集周期を指定します (1ms 単位)	Priority	プライオリティを設定します	DatasetName	収集するデータセット名を指定します。
キーワード	説明										
Name	Dataset 収集機能名を指定します										
Interval	収集周期を指定します (1ms 単位)										
Priority	プライオリティを設定します										
DatasetName	収集するデータセット名を指定します。										

コンテナプロパティ設定

サービスコンテナプロパティ設定値は、Tag として登録されており、サービスコンテナ実装毎に数や種類は異なります。規定値プロパティは、SERVICE.キーワードをプリフィックスとしたタグ名で登録されています:

設定項目	設定手順												
コンテナプロパティ値	<p><Tags>エレメント内に、<Tag>において、SERVICE. キーワードをプリフィックスとしたタグは、コンテナプロパティタグです:</p> <p>Tag</p> <table> <tr> <th>キーワード</th><th>説明</th></tr> <tr> <td>Name</td><td>SERVICE. キーワードをプリフィックスとした名称で設定されています。</td></tr> <tr> <td>Type</td><td>RT-edge データ型に関連する型定義値を設定します。</td></tr> <tr> <td>Size</td><td>タグデータサイズを指定します。</td></tr> <tr> <td>Value</td><td>設定値</td></tr> <tr> <td>Comment</td><td>タグに対するコメントを設定します。</td></tr> </table>	キーワード	説明	Name	SERVICE. キーワードをプリフィックスとした名称で設定されています。	Type	RT-edge データ型に関連する型定義値を設定します。	Size	タグデータサイズを指定します。	Value	設定値	Comment	タグに対するコメントを設定します。
キーワード	説明												
Name	SERVICE. キーワードをプリフィックスとした名称で設定されています。												
Type	RT-edge データ型に関連する型定義値を設定します。												
Size	タグデータサイズを指定します。												
Value	設定値												
Comment	タグに対するコメントを設定します。												



コンテナプロパティ設定については、各サービスコンテナに付与するユーザーズマニュアルを参照してください。

7. 外部通信サービスコンテナ設定

外部通信コンテナにおける RT-edge Object 設定には、アップロード・ダウンロードするファイルパスや外部サーバー等の I/O 情報と RT-edge Object との接続設定が含まれます。プロパティ設定では接続先となる URL や、接続サーバー機器等の IP、ポート番号など、サービスコンテナに特化した設定が含まれます。

7.1. RT-edge Object 設定概要

外部通信サービスコンテナではタグ定義設定を行います：

外部サーバー 等とデータのやり取りを行う I/O メモリ情報をタグに公開	ローカルタグ生成 + タグ参照	外部サーバー等の I/O メモリ情報をタグ(RT-edge Object)のデータ入出力先として割り当てる。
外部サーバー 等とファイルのやり取りするパス情報をタグに公開		外部サーバーへのアップロードファイル、外部サーバーからのダウンロードファイルのパスをタグに定義する

■ ローカルタグ生成

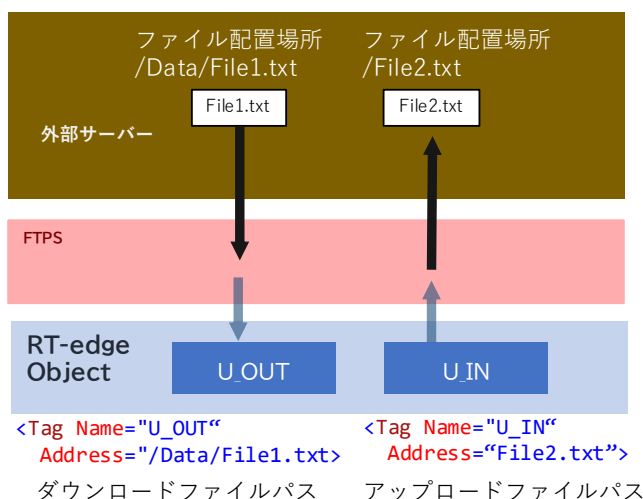
RT-edge システム内で一意となるタグを定義し、データ取得元、データ宛先の設定を行います。

データ取得元とデータ宛先設定

ECI(Edge コンテナ設定情報:EgExtIf.xml)の編集

ここでは FTP を用いたファイルのアップロード・ダウンロードの設定を例に記載します。

ローカルタグとして定義したタグのデータ取得元/宛先を設定します。外部通信サービスコンテナ用 ECI(Edge コンテナ設定情報:EgExtIf.xml)に、タグを定義し、アップロードファイルのパス、ダウンロードファイルのパスを割り当てます。



ローカルタグを生成し、アップロードファイルのパス(ローカルファイルのパス)、もしくはダウンロードファイルのパス(サーバー上のファイルパス)を設定します。

図 11. FTP を介したファイル操作

7.2. ECI(コンテナ設定情報ファイル)の編集

ECI の編集を行います。下記では FTP 通信プロトコルを利用した例を記載します。

1) タグ生成

- 1) ECI コンテナ設定情報(EgExtIF.xml) ファイルを開きます。
- 2) RTedge > Tags 内に、Tag エレメントを追加します。

<Tag Name="TempData" Type="1" Address="/Data/TempData.txt" Comment="温度記録情報ファイル"/>	①
<Tag Name="ErrorLog" Type="1" Address="/Data/ErrorLog.txt" Comment="エラーログファイル"/>	②
<Tag Name="SettingFile" Type="1" Address="/Setting/SettingFile.txt" Comment="設定ファイル"/>	③

- ① コントローラ内に保存されている「温度記録情報ファイル」をサーバーのルートフォルダ以下の指定フォルダ(Address 定義)のパスにアップロードする設定を行います。
- ② コントローラ内に保存されている「エラーログファイル」をサーバーのルートフォルダ以下の指定フォルダ(Address 定義)のパスにアップロードする設定を行います。
- ③ サーバーに保存されている「設定ファイル」をコントローラにダウンロードする設定を行います。

(ア) Name 設定

RT-edge のタグ名を指定します。

(イ) Type 設定

以下の表を参考に、入出力を行うデータ型(Type)を指定します。FTP 通信で用いるタグの Type は"1"を指定してください。

表 6. データ型に対応する Type 一覧

PLC データ型	Type	Size	用途	RT-edge データ型(egTag)
BOOL	Type="1"	Size="1"	BOOL 値	Boolean
INT	Type="4"	Size="2"	符号付き 16bit 整数	Int16
DINT	Type="6"	Size="4"	符号付き 32bit 整数	Int32
LINT	Type="8"	Size="8"	符号付き 64bit 整数	Int64
UINT	Type="5"	Size="2"	符号なし 16bit 整数	UInt16
UDINT	Type="7"	Size="4"	符号なし 32bit 整数	UInt32
ULINT	Type="9"	Size="8"	符号なし 64bit 整数	UInt64
REAL	Type="10"	Size="4"	単精度実数(32bit)	Float
LREAL	Type="11"	Size="8"	倍精度実数(64bit)	Double
WORD	Type="5"	Size="2"	符号なし 16bit 整数	UInt16
DWORD	Type="7"	Size="4"	符号なし 32bit 整数	UInt32
LWORD	Type="9"	Size="8"	符号なし 64bit 整数	UInt64
-	Type="2"	Size="1"	符号付き 8bit 整数	SByte
-	Type="3"	Size="1"	符号なし 8bit 整数	Byte

(ウ) Address 設定

外部通信コンテナにおける Address 設定は以下 2 通りのフォーマットにそって設定します:

① パス指定

Address = "パス"

主に、FTP を用いたファイルのアップロード・ダウンロードに使用します。

② 領域指定

Address = "FTP/16#00ff"

①

②

表 7. Address フォーマット

	名称	設定値	説明
①	通信プロトコル名	※説明欄参照	通信プロトコル名を指定します。 指定可能な通信プロトコル名は付録 10.3. ECI Address 定義を参照ください。
②	Edge I/O ポート オフセット値	※説明欄参照	Edge I/O ポートのオフセット値を指定します。 オフセット値は、10 進数表記、16 進数表記に対応しています。16 進数表記の場合は先頭に"16#"を付けることで、16 進数と解釈します。 10 進数表記：255 16 進数表記：16#00ff Edge I/O ポートのアドレス定義は通信プロトコルごとに異なります。詳細は付録 10.3. ECI Address 定義を参照ください。

2) 入力タグ参照

外部サーバーや、その他外部機器から入力を行うタグを設定します。
外部サーバーや外部機器からの入力が不要な場合、本設定は行いません。

- 1) RTEdge > Services > Service(Name 属性="プロトコル名") > TagRefs > TagRefs_IN エレメントを追加します。
- 2) TagRefs_IN エレメント内に、TagRef エレメントを追加します。
- 3) TagRef の Name 属性には、使用するローカルタグを設定します。

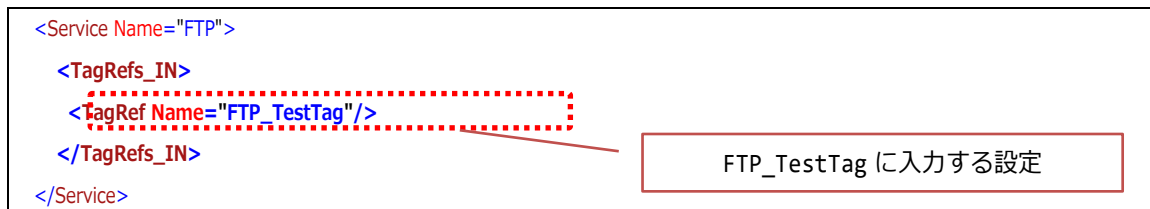


図 12. TagRefs_IN の設定



存在しないタグを指定した場合は無視されます。

3) 出力タグ参照

外部サーバーやその他外部機器等に対して出力を行うタグを設定します。
出力処理が不要な場合、本設定は行いません。

- 1) RTEdge > Services > Service(Name 属性="プロトコル名") > TagRefs > TagRefs_OUT エレメントを追加します。
- 2) TagRefs_OUT エレメント内に、TagRef エレメントを追加します。
- 3) TagRef の Name 属性には、使用するローカルタグを設定します。

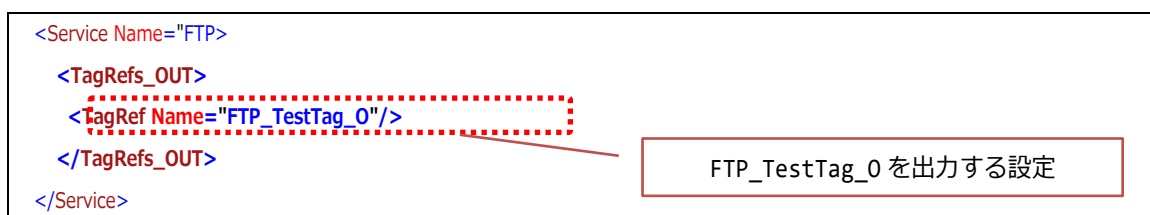


図 13. TagRefs_OUT の設定



存在しないタグを指定した場合は無視されます。

7.3. コンテナプロパティ設定

外部通信コンテナには、以下の主要なプロパティ設定があります。サービスコンテナのプロパティ設定はタグの一部として構成されています。以下にコンテナで共通なコンテナプロパティ定義を記載します。

表 8.コンテナプロパティ設定

設定	Tag 名	デフォルト設定	内容
サービス 起動設定	SERVICE.<通信プロトコル名>.AutoRun	1 (タグデータ更新開始)	サービス起動時のタグデータ更新自動開始を指定します。
タグ更新周期	SERVICE.<通信プロトコル名>.Cycle	1000	タグデータの入出力周期を設定します。 (ms)



コンテナプロパティタグはあらかじめ定義されており、名称は、SERVICE. から開始しています。



IP アドレスやポート番号といったプロパティ設定は通信プロトコルごとに異なります。通信プロトコルごとに設定が必要なプロパティタグの一覧は付録 10.1. サービスプロパティタグを参照ください。

8. 設定サンプル

ここでは外部通信コンテナ FTP の設定を例に説明します。その他のサンプルは付録 10.4. 設定サンプルを参照ください。

1) FTP 転送時のモード設定(RT-edge インストール PC)

FTP 転送時のモード設定を Tag に定義します。

表 9.FTP 転送設定情報

Tag 名	タグタイプ	値 デフォルト値	タグ概要	備考
SERVICE.{CN}.ASCIIMode	bool	FALSE	FTP 転送時のアスキーモード(またはバイナリモード)の指定	TRUE : アスキーモード FALSE : バイナリモード
SERVICE.{CN}.PASVMode	bool	FALSE	FTP 転送時のパッシブモード(またはアクティブモード)の指定	TRUE : パッシブモード FALSE : アクティブモード
SERVICE.{CN}.SSLMode	bool	FALSE	SSL 有効(無効)の指定	TRUE : SSL 有効 FALSE : SSL 無効
SERVICE.{CN}.KeepAliveMode	bool	FALSE	要求の完了後に FTP サーバーへの制御接続を閉じるかどうか	TRUE : 接続状態を保つ FALSE : 接続を閉じる
SERVICE.{CN}.LocalFolder	string	""	FTP 通信を行う作業用ローカルフォルダ	
SERVICE.{CN}.Url	string	""	FTP サーバー接続 Url	
SERVICE.{CN}. Username	string	""	ユーザー名	
SERVICE.{CN}. Password	string	""	パスワード	

{CN} : コンテナ名

2) タグ設定

「7.1. RT-edge Object 設定」を参考に、タグの生成・入出力設定を行います。各タグの Address にダウンロードパス、アップロードパス、または Edge I/O ポート(情報領域)のアドレス値を設定します。

```
<Services>
  <Service Name="FTP" >
    <Tags>
      <Tag Name="SERVICE.FTP.LocalFolder" Type="1" Address="C:/FTP_Local/" Comment="FTP 通信を行う作業用ローカルフォルダ"/>
      ...
      <Tag Name="TempData" Type="1" Address="TempData.txt" Comment="温度記録情報ファイル"/>
      <Tag Name="ErrorLog" Type="1" Address="ErrorLog.txt" Comment="エラーログファイル"/>
      <Tag Name="SettingFile" Type="1" Address="SettingFile.txt" Comment="設定ファイル"/>
      <Tag Name="ConnectStatus" Type="7" Address="FTP/16#4000" Comment="情報領域 接続ステータス設定ファイル"/>
    </Tags>
    <TagRefs_IN>
      <TagRef Name="SettingFile"/>
      <TagRef Name="ConnectStatus"/>
    </TagRefs_IN>
    <TagRefs_OUT>
      <TagRef Name="TempData"/>
      <TagRef Name="ErrorLog"/>
    </TagRefs_OUT>
  </Service>
</Services>
```

9. 動作確認

9.1. 動作確認手順

ここでは外部通信コンテナ FTP の設定を例に説明します。

RT-edge ソフトウェアを起動し、外部通信コンテナ FTP が正常に起動することを確認します：

- 1) Edge コントローラと外部端末の接続設定を行います。
- 2) Edge コントローラと外部端末をインターネット経由で接続します。

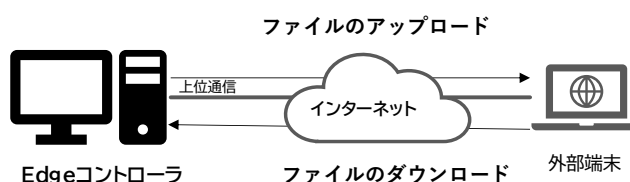


図 14. 接続構成図

- 3) RT-Edge ソフトウェア(C:\RTedge\EgBoot.exe)を開始します。
- 4) RT-edge オブジェクトブラウザ(C:\RTedge\EgBrow.exe)を起動します。



RT-edge 起動直後に RT-edge オブジェクトブラウザを起動すると初期化中の為、想定されるタグが表示されない場合があります。一度 RT-edge オブジェクトブラウザを終了し、再度起動させてください。

- 5) RT-edge コンテナ設定情報(ECI)で定義したタグが表示されていることを確認します。

図 15. コンテナ生成タグイメージは、RT-edge Object Browser のスクリーンショットを示しています。画面には「Tags」タブが選択されており、以下のタグ一覧が表示されています。

Name	Current Value	Type	Source	Comment
SERVICE.FTPS.MboxColSize	00000800 (2048)	UInt32		
SERVICE.FTPS.MboxNameValue	00000029 (41)	UInt32		
SERVICE.FTPS.MboxRowSize	00000400 (1024)	UInt32		
SERVICE.FTPS.Mode	01 (1)	byte		1=入出力Auto
SERVICE.FTPS.OutThread00_Cnt	0000000d (13)	UInt32		
SERVICE.FTPS.OutThread00_LastError	00000000 (0)	UInt32		
SERVICE.FTPS.OutThread00_Time	0.00000000	float		
SERVICE.FTPS.OutThread00_TimeMax	0.00000000	float		
SERVICE.FTPS.OutThread00_TimeMin	0.00000000	float		
SERVICE.FTPS.OutThreadNum	01 (1)	byte		Number of output threads
SERVICE.FTPS.Password	False (0)	bool	mnc	パスワード
SERVICE.FTPS.PASVMode	True (1)	bool		FTP転送時のパッシブモードの指
SERVICE.FTPS.ProcessMode	00 (0)	byte		システム利用 変更不可
SERVICE.FTPS.Run	True (1)	bool		
SERVICE.FTPS.Status	01 (1)	byte		
SERVICE.FTPS.Timeout	00002710 (10000)	UInt32		Driver API Timeout time in
SERVICE.FTPS.Url	False (0)	bool	ftp://192.168	FTPサーバー接続 Url
SERVICE.FTPS.Username	False (0)	bool	FTP	ユーザー名
SettingFile	False (0)	bool	SettingFile.tx	設定ファイル
TempData	False (0)	bool	TempData.txt	温度記録情報ファイル

画面下部には「CSV出力」「一覧再取得」ボタンと「72 items available」と表示されています。

図 15. コンテナ生成タグイメージ

9.2. サービスインジケータタグ

動作確認で必要となるサービスインジケータタグについて説明します:

表 10. サービスインジケータ共通タグ一覧

ステータスインジケータ Tag 名	備考
SERVICE.<通信プロトコル名>.Status	現在のサービス起動状態を示します
SERVICE.<通信プロトコル名>.Error	現在のサービスエラー状態を示します
SERVICE.<通信プロトコル名>.Run	現在のデータタグ参照・更新動作の状態を示します
SERVICE.<通信プロトコル名>.Live	サービスが健全であることを示すカウンタ

上記 Tag の詳細はユーザーズマニュアルを参照ください。

9.3. RT-edge タグデータの妥当性について

タグリンクされた RT-edge タグのデータは、サービスインジケータが以下の状態になっている時、妥当であると判断出来ます。

1. SERVICE.<通信プロトコル名>.Error = FALSE (エラーが発生していない)
2. SERVICE.<通信プロトコル名>.Run = TRUE (タグ参照・更新が実行されている)

上記の状態になっていない場合には、タグデータの入出力に何らかの問題が発生しています。詳細は、以下のトラブルシューティングをご参照ください。

「サービスインジケータの.Error が true になっています。」

「サービスインジケータの.Run が false になっています。」

10. 付録

10.1. サービスプロパティタグ

外部通信コンテナでは以下のサービスプロパティタグを使用します。

共通

表 11. サービスプロパティタグ一覧(共通)

ステータスプロパティ Tag 名	Type	初期値	概要
SERVICE.<通信プロトコル名>.Cycle	UInt32	1000 (ms)	動作サイクルタイムを指定します。 1~65535(ms) (1ms の指定は可能ですが、使用するタグ数が多いほど Read/Write 処理に時間を要します。タグ数に応じて Cycle 時間を増やしてください。)
SERVICE.<通信プロトコル名>.IntervalCycle	UInt32	1000 (ms)	インターバルでの処理をする際の周期を指定します。
SERVICE.<通信プロトコル名>.Timeout	UInt32	1000 (ms)	API 呼び出し時のタイムアウト時間を指定します。
SERVICE.<通信プロトコル名>.ExecuteReConnect	Boolean	0	エラー時、再接続するか否かを指定します。1=再接続処理を実施。0=再接続処理を実施しない(デフォルト)
SERVICE.<通信プロトコル名>.Mode	Byte	1	動作モードを指定します。1=入出力 Auto (システムで使用する為、変更不可)
SERVICE.<通信プロトコル名>.AutoRun	Boolean	1	動作自動スタートの有効無効を指定します。1=起動時に開始 (システムで使用する為、変更不可)
SERVICE.<通信プロトコル名>.InThreadNum	Byte	1	入力動作スレッド生成数を指定します。 (システムで使用する為、変更不可)
SERVICE.<通信プロトコル名>.InThreadNum	Byte	1	入力動作スレッド生成数を指定します。 (システムで使用する為、変更不可)
SERVICE.<通信プロトコル名>.OutThreadNum	Byte	1	出力動作スレッド生成数を指定します。 (システムで使用する為、変更不可)
SERVICE.<通信プロトコル名>.InPriority	Byte	149	入力動作スレッドプライオリティを指定します。
SERVICE.<通信プロトコル名>.OutPriority	Byte	148	出力動作スレッドプライオリティを指定します。

背景がグレーの Tag はシステムで利用しているため、値の変更はできません。

FTP

FTP を利用時のサービスプロパティタグは以下の通りです。

表 12. FTP 利用時のサービスプロパティタグ一覧

ステータスプロパティ Tag 名	Type	初期値	概要
SERVICE.FTP.ASCIIMode	Bool	FALSE	FTP 転送時のアスキーモード(またはバイナリモード)を指定します。 TRUE : アスキーモード FALSE : バイナリモード
SERVICE.FTP.PASVMode	Bool	FALSE	FTP 転送時のパッシブモード(またはアクティブモード)を指定します。 TRUE : パッシブモード FALSE : アクティブモード
SERVICE.FTP.SSLMode	Bool	FALSE	SSL 有効(無効)を指定します。
SERVICE.FTP.KeepAliveMode	Bool	FALSE	要求後に FTP サーバーへの制御接続を閉じるかどうかを指定します。 TRUE : 接続状態を保つ FALSE : 接続を閉じる

ステータスプロパティ Tag 名	Type	初期値	概要
<code>SERVICE.FTP.LocalFolder</code>	Bool	""	FTP 通信を行う作業用ローカルフォルダを Address 属性に指定します。
<code>SERVICE.FTP.Url</code>	Bool	""	FTP サーバー接続 Url を Address 属性に指定します。
<code>SERVICE.FTP.Username</code>	Bool	""	接続に必要なユーザー名を Address 属性に指定します。
<code>SERVICE.FTP.Password</code>	Bool	""	接続に必要なパスワードを Address 属性に指定します。

TCP

表 13.TCP 利用時のサービスプロパティタグ一覧

ステータスプロパティ Tag 名	Type	初期値	概要
<code>SERVICE.TCP.KeepAlive</code>	Bool	"0"	KeepAlive 設定を有効にするか否かを指定します。 KeepAlive は、通信開始からしばらくして相手からの通信が途絶えた際に、相手が活着ているかを確認する仕組みです。有効の場合は、相手と通信できない場合、コネクションを切断します。
<code>SERVICE.TCP.KeepAliveTime</code>	Uint16	"10000"	KeepAliveTime の指定をします。 KeepAliveTime は、無通信状態になってから最初の Keep Alive パケットを送信するまでの時間です。
<code>SERVICE.TCP.KeepAliveInterval</code>	Uint16	"10000"	KeepAliveInterval を指定します。 KeepAliveTime が期限切れになった後にキープアライブが再送信される時間間隔です。
<code>SERVICE.TCP.LocalIP</code>	Bool	""	接続元 IP アドレスを指定します。
<code>SERVICE.TCP.ReUseAddr</code>	Bool	"0"	ポート再利用有効
<code>SERVICE.TCP.TargetIP</code>	Bool	""	接続先 IP アドレスを指定します。

MQTT

表 14.MQTT 利用時のサービスプロパティタグ一覧

ステータスプロパティ Tag 名	Type	初期値	概要
<code>SERVICE.MQTT.UseWebSocket</code>	Bool	"0"	通信方式で WebSocket を使用するか否かを指定します。
<code>SERVICE.MQTT.User</code>	Bool	"0"	接続時のユーザー名を Address 属性に指定します。
<code>SERVICE.MQTT.Pass</code>	Bool	"0"	接続時のパスワードを Address 属性に指定します。
<code>SERVICE.MQTT.ServerSetting</code>	Bool	"0"	IP:ポート番号を Address 属性に指定します。 IP とポート番号の区切り文字は":"です。 例) Address="127.0.0.1:1883"

10.2. サービスメッセージ

外部通信コンテナでは以下のメッセージに対する処理が実装されています:

共通メッセージ

表 15.共通メッセージ一覧

メッセージ名	番号	説明
EM_SERVICE_STOP	101	サービスを終了させます。
EM_SERVICE_RUN	102	データ更新処理を開始します。
EM_SERVICE_PAUSE	103	データ更新処理を一時停止します。再開するためには、EM_SERVICE_RUN を送信してください。

各メッセージ送信後のサービスインジケータタグは以下の状態に遷移します:

表 16.メッセージ送信後のサービスインジケータタグの状態一覧

サービスインジケータタグ	初期化完了 (.AutoRun=True)	初期化完了 (.AutoRun=False)	EM_SERVICE _STOP	EM_SERVICE _RUN	EM_SERVICE _PAUSE
SERVICE.<通信プロトコル名>.Status	1	1	0	1	1
SERVICE.<通信プロトコル名>.Run	TRUE	FALSE	FALSE	TRUE	TRUE
SERVICE.<通信プロトコル名>.Live	増加	停止	停止	増加	増加
SERVICE.<通信プロトコル名>.Error	FALSE	FALSE	FALSE	FALSE	FALSE

10.3. ECI Address 定義

FTP

通信プロトコル名 : FTP

Edge FTP I/O ポート アドレス構造 :

FTP の Edge I/O ポートは、情報領域のみ使用します。情報領域は、以下の表のとおり 0x4000～0x4354 の範囲の値を指定します。

Offset	区分	型	説明	サイズ(byte)
0x4000	接続情報	UInt32	接続ステータス(0:オープン済み, 1:未初期化, 2:未オープン)	4
0x4004		string	接続先Uri	256
0x4104		string	ユーザー名	256
0x4204		string	パスワード	256
0x4304	ダウンロード情報	UInt32	累計ダウンロードファイル数	4
0x4308		UInt32	エラー番号(最後に発生したエラー)	4
0x430C		long	エラー回数	8
0x4314		long	エラー時刻(1970-01-01T00:00:00Zからの経過秒)	8
0x431C	アップロード情報	UInt32	累計アップロードファイル数	4
0x4320		UInt32	エラー番号(最後に発生したエラー)	4
0x4324		long	エラー回数	8
0x432C		long	エラー時刻(1970-01-01T00:00:00Zからの経過秒)	8
0x4334	統計情報	long	初期化回数	8
0x433C		long	接続回数	8
0x4344		long	切断回数	8
0x434C		long	読み回数	8
0x4354		long	書き回数	8

図 16.Edge FTP I/O ポート アドレス構造

TCP

通信プロトコル名 : TCP

Edge TCP I/O ポート アドレス構造 :

TCP の Edge I/O ポートは、情報領域のみ使用します。情報領域は、以下の表のとおり 0x4000～0x4030 の範囲の値を指定します。

Offset	説明	サイズ(byte)
0x4000	接続ステータス(0⇒切断, 1⇒接続試行中, 2⇒通信中)	4
0x4004	現在受信可能なバイト数	4
0x4008	トータル送信バイト数	8
0x4010	トータル受信バイト数	8
0x4018	最終の送信バイト数	4
0x401C	最終の受信バイト数	4
0x4020	統計Init関数コール回数	4
0x4024	統計Open関数コール回数	4
0x4028	統計Close関数コール回数	4
0x402C	統計Read関数コール回数	4
0x4030	統計Write関数コール回数	4

図 17.Edge TCP I/O ポート アドレス構造

MQTT

通信プロトコル名 : MQTT

Edge MQTT I/O ポート アドレス構造 :

MQTT の Edge I/O ポートは、情報領域のみ使用します。情報領域は、以下の表のとおり 0x4000~0x5014 の範囲の値を指定します。

情報領域	Offset	区分	型	説明	サイズ(byte)
	0x4000	接続情報	UInt32	接続ステータス0:オープン済み 1:未初期化 2:未オープン	4
	0x4004		string	接続先Uri (SJis)	256
	0x4104		string	ユーザー名(SJis)	256
	0x4204		string	パスワード(SJis)	256
	0x5000	統計情報	UInt32	init_cnt API呼び出し回数	4
	0x5004		UInt32	open_cnt API呼び出し回数	4
	0x5008		UInt32	iclosecnt API呼び出し回数	4
	0x500C		UInt32	read_cnt API呼び出し回数	4
	0x5010		UInt32	write_cnt API呼び出し回数	4

図 18.Edge MQTT I/O ポート アドレス構造

10.4. 設定サンプル

FTP

8. 設定サンプルで記載の通りです。

TCP

ここでは外部通信コンテナ TCP のタグ設定を ECI(xml)の記載を元に説明します。

TCP での送受信の単位はデータセット(Dataset)です。内訳はデータセット(Dataset)内のタグ参照(TagRef)を用いて定義します。

1) タグ設定

「7.1. RT-edge Object 設定」を参考に、タグの生成・入出力設定を行います。

ここでは入出力対象のタグを作成し、Address に入出力対象のデータセット名を設定、または Edge I/O ポート(情報領域)のアドレス値を設定します。

データセットとタグの設定例を以下に記載します。

```
<Services>
  <Service Name="TCP" >
    <Tags>
      ...
      <Tag Name="TCP_OutputTag01" Type="5" Value="0" Comment="送信用データ1"/>
      <Tag Name="TCP_OutputTag02" Type="5" Value="0" Comment="送信用データ2"/>
      <Tag Name="TCP_OutputTag03" Type="5" Value="0" Comment="送信用データ3"/>
      <Tag Name="TCP_OutputTag04" Type="5" Value="0" Comment="送信用データ4"/>
      <Tag Name="TCP_OutputTag05" Type="5" Value="0" Comment="送信用データ5"/>
      <Tag Name="TCP_InputTag01" Type="5" Value="0" Comment="受信用データ1"/>
      <Tag Name="TCP_InputTag02" Type="5" Value="0" Comment="受信用データ2"/>
      <Tag Name="TCP_InputTag03" Type="5" Value="0" Comment="受信用データ3"/>
      <Tag Name="TCP_InputTag04" Type="5" Value="0" Comment="受信用データ4"/>
      <Tag Name="TCP_InputTag05" Type="5" Value="0" Comment="受信用データ5"/>
      <Tag Name="TCP_InputDataSet" Type="1" Address="TCP_INPUT" Comment="受信用データセット"/>
      <Tag Name="TCP_OutputDataSet" Type="1" Address="TCP_OUTPUT" Comment="送信用データセット"/>
    </Tags>
    <Datasets>
      <Dataset Name="TCP_INPUT">
        <TagRefs>
          <TagRef Name="TCP_InputTag01"/>
          <TagRef Name="TCP_InputTag02"/>
          <TagRef Name="TCP_InputTag03"/>
          <TagRef Name="TCP_InputTag04"/>
          <TagRef Name="TCP_InputTag05"/>
        </TagRefs>
      </Dataset>
      <Dataset Name="TCP_OUTPUT">
        <TagRefs>
          <TagRef Name="TCP_OutputTag01"/>
          <TagRef Name="TCP_OutputTag02"/>
          <TagRef Name="TCP_OutputTag03"/>
          <TagRef Name="TCP_OutputTag04"/>
          <TagRef Name="TCP_OutputTag05"/>
        </TagRefs>
      </Dataset>
    </Datasets>
    <TagRefs_IN>
      <TagRef Name="TCP_InputDataSet"/>
    </TagRefs_IN>
    <TagRefs_OUT>
      <TagRef Name="TCP_OutputDataSet"/>
    </TagRefs_OUT>
  </Service>
</Services>
```

入出力対象のデータセット名を指定

入力データ分解後のタグを指定

出力データ結合元のタグを指定

入力対象のデータセット名が書かれた Tag を指定

出力対象のデータセット名が書かれた Tag を指定

MQTT

ここでは外部通信コンテナ MQTT のタグ設定を ECI(xml)の記載を元に説明します。

1) タグ設定

「7.1. RT-edge Object 設定」を参考に、タグの生成・入出力設定を行います。タグ名には MQTT 通信で使用するトピック名を指定します。

```
<Services>
  <Service Name="MQTT" >
    <Tags>
      ...
      <Tag Name="ESP32/01/temp" Type="7" Value="0" Comment=""/>
      <Tag Name="ESP32/02/temp" Type="7" Value="0" Comment=""/>
      <Tag Name="ESP32/03/temp" Type="7" Value="0" Comment=""/>
      <Tag Name="pub/byte" Type="3" Value="30" Comment=""/>
      <Tag Name="pub/int16" Type="4" Value="40" Comment=""/>
      <Tag Name="pub/uint32" Type="7" Value="70" Comment=""/>
    </Tags>
    <TagRefs_IN>
      <TagRef Name="ESP32/01/temp"/>
      <TagRef Name="ESP32/02/temp"/>
      <TagRef Name="ESP32/03/temp"/>
    </TagRefs_IN>
    <TagRefs_OUT>
      <TagRef Name="pub/byte"/>
      <TagRef Name="pub/int16"/>
      <TagRef Name="pub/uint32"/>
    </TagRefs_OUT>
  </Service>
</Services>
```

10.5. 環境設定 その他仕様制限事項

FTP

- 1) 指定したファイルが存在しない場合は、アップロード/ダウンロードを行いません。

TCP

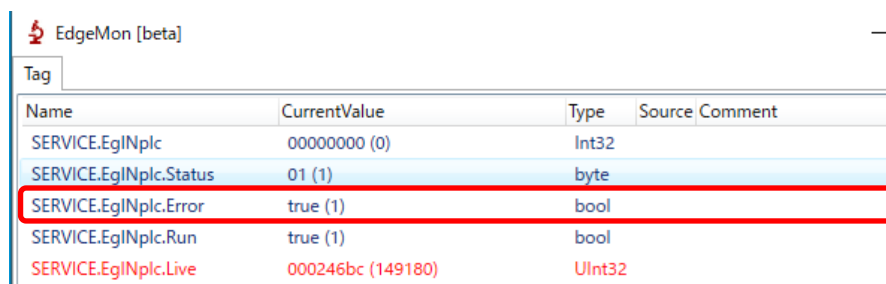
なし

MQTT

- 1) TagRef_IN で指定したタグは MQTT のサブスクライバーとして扱われたため、更新がかかった最新のデータのみを取得します。コンテナ起動時に古い値を取得することはできません。

10.6. トラブルシューティング

サービスインジケータの>Error が true になっています。



Name	CurrentValue	Type	Source	Comment
SERVICE.EglNplc	00000000 (0)	Int32		
SERVICE.EglNplc.Status	01 (1)	byte		
SERVICE.EglNplc.Error	true (1)	bool		
SERVICE.EglNplc.Run	true (1)	bool		
SERVICE.EglNplc.Live	000246bc (149180)	UInt32		

図 19. Error インジケータが true

「.Error」タグが True になる場合、何れかの通信機器との接続に問題がある状態となります。

通信機器と RT-edge コンテナ設定情報(ECI)ファイルについて、以下をご確認ください:

- 1) ネットワーク設定 (IP アドレス・ポート番号) は正しいか?
- 2) 通信設定 (ネットワークアドレス・ノードアドレス・ユニット No) 等は正しいか?
- 3) 物理的に接続されているか (ケーブル断線・通信機器の電源が入っていないなど) ?

サービスインジケータの.Run が false になっています。

- 原因：サービス開始要求を受け付けられていません。

サービス開始要求が無い、初期化処理が終わらずサービス開始要求を受け付けられていません。

- 対応：サービス開始要求を行います。

アプリケーションから「EM_SERVICE_RUN」メッセージを送る、または EgFINS.xml にて「.AutoRun」プロパティの Value を「1」とします。

RT-edge オブジェクトブラウザにサービスインジケータが表示されません。

- 原因：外部通信コンテナ初期化処理中です。

外部通信コンテナ初期化処理中では、タグが生成されていません。

- 対応：終了させ、10 秒程度待ったのち再度 RT-edge オブジェクトブラウザを起動させます。

外部通信コンテナ初期化処理完了後に再度起動させます。

タグデータが更新されません。

- 原因：タグ参照・更新設定がされていません。

サービスインジケータ「.Run」が TRUE にも関わらずタグデータが更新されない場合、タグ参照・更新設定がされていない可能性があります。

- 対応：タグ参照・更新設定を行います。

設定方法については「7.2. ECI(コンテナ設定情報ファイル)の編集」を参照ください。

起動時にコンソールが表示され、「wrong number of arguments (given 0, expected 1+)」が出力されます

- 原因：DLL のロード設定がされていません。

EgExtIf.exe の起動設定において、通信プロトコルごとの DLL のロード設定がされていない可能性があります。

- 対応：タグ参照・更新設定を行います。

Argument 属性を設定する必要があります。設定方法については「4.3. 起動設定」を参照ください。

各エラーコードの原因・対応方法

何らかの問題が発生しサービス異常状態ステータス「.Error」タグが TRUE になっている場合、エラーコードから理由を確認可能です。最後に発生したエラーコードは「.〇〇〇LastErr」タグで確認可能です。各エラーコードの内容・対応方法を以下に記載します。

(〇〇〇にはスレッド名、処理名が入ります)

表 17.エラーコード共通

エラーコード	内容	対策
0x00000000	正常	

表 18.エラーコード FTP

エラーコード	内容	対策
0x00000000	正常	
0xC0000000	初期化が未完了です。	(内部エラー)
0xC0000001	既にオープン済みです。	RT-edge システムの再起動、または PC を再起動してください。
0xC0000002	FTP サーバーとの接続に失敗しました。	FTP サーバーが起動していない可能性があります。確認してください。
0xC0000003	要求がタイムアウトしました。	通信に失敗しています。ネットワーク接続が正しいか確認してください。
0xC0000004	オープン処理が完了していません。	ログを確認し他にエラーが出力されていないか確認してください。
0xC0000005	アドレスに範囲外の値が指定されました。	アドレスの値を確認してください。
0x80000004	指定したローカル FTP フォルダが見つかりません。	フォルダが存在するか、またはアクセス権を確認してください。
0x80000005	ユーザー名が指定されていません。	ユーザー名を指定してください。
0x80000006	パスワードが指定されていません。	パスワードを指定してください。
0x80000008	サーバーアドレスが指定されていません。	サーバーアドレスを指定してください。
0x80000009	FTP サーバーとの接続に失敗しました。	ホスト名が正しくありません。確認してください。
0x8000000A	FTP サーバーへのログインに失敗しました。	ユーザー名またはパスワードに誤りがある可能性があります。確認してください。
0x8000000B	ローカル FTP フォルダへのアクセスに失敗しました。	フォルダが存在するか、またはアクセス権を確認してください。
0x8000000E	(内部エラー)	(内部エラー)
0x80000010	(内部エラー)	(内部エラー)
0x80000011	(内部エラー)	(内部エラー)
0x80000013	SSL 証明書が一致しません。	-
0x80010000～ 0x8001FFFF	FTP 通信エラー	(内部エラー)
0x80020000～ 0x8002FFFF	WEB 通信エラー	(内部エラー)

表 19.エラーコード TCP

エラーコード	内容	対策
0x00000000	正常	
0xC0000000	初期化が未完了です。	(内部エラー)
0xC0000001	接続に失敗しました。	ネットワークの接続を確認後、RT-edge システムの再起動、または PC を再起動してください。
0xC0000002	通信の切断が発生しました。	ネットワークの接続を確認してください。
0xC0000003	要求がタイムアウトしました。	通信に失敗しています。ネットワーク接続が正しいか

エラーコード	内容	対策
		確認してください。
0xC0000004	オープン処理が完了していません。	ログを確認し他にエラーが出力されていないか確認してください。
0xC0000005	アドレスに範囲外の値が指定されました。	アドレスの値を確認してください。

表 20.エラーコード MQTT

エラーコード	内容	対策
0x00000000	正常	
0xC0000000	初期化が未完了です。	(内部エラー)
0xC0000001	既にオープン済みです。	RT-edge システムの再起動、または PC を再起動してください。
0xC0000002	サーバー(ブローカー)との接続に失敗しました。	サーバー(ブローカー)が起動していない可能性があります。確認してください。
0xC0000003	要求がタイムアウトしました。	通信に失敗しています。ネットワーク接続が正しいか確認してください。
0xC0000004	オープン処理が完了していません。	ログを確認し他にエラーが出力されていないか確認してください。
0xC0000005	アドレスに範囲外の値が指定されました。	アドレスの値を確認してください。
0x80000005	ユーザー名が指定されていません。	ユーザー名を指定してください。
0x80000006	パスワードが指定されていません。	パスワードを指定してください。
0x80000007	不明な設定項目が指定されました。	(内部エラー)
0x8000000D	サーバーアドレスが指定されていません。	サーバーアドレスを指定してください。
0x8000000E	(内部エラー)	(内部エラー)
0x80000010	(内部エラー)	(内部エラー)
0x80000011	(内部エラー)	(内部エラー)
0x80000012	(内部エラー)	(内部エラー)

更新履歴

版	日付	更新説明
1	2022.12	初回版
2	2023.04	サービスプロパティタグに以下を追加 SERVICE.<通信プロトコル名>.ExecuteReConnect
3	2023.07	誤字を修正
4	2024.08	誤字を修正
5	2025.8	RTCD の名称を「RT-edge Object」に変更

INDUSTRIAL REALTIME EDGE COMPUTERS

外部通信 Container ユーザーズマニュアル

発行元:株式会社マイクロネット

TEL: +81(0)299-90-1733

FAX: +81(0)299-92-8557

- ・ 本書の内容、及び付属のソフトウェアの全部または一部を無断で転載することは禁止しております。
- ・ 本製品の内容については、将来予告なしに変更することがあります。
- ・ 本製品の内容について万一ご不審な点や記載もれなどお気づきの点がございましたら、お手数ですが、当社までご連絡ください。
- ・ Windows XP、Windows 7、Windows 8、Windows 10 等、Windows は、米国 Microsoft Corporation における登録商標です。
- ・ Visual Studio、Visual C++等は、米国、およびその他の国における Microsoft Corporation の登録商標です。
- ・ INtime は米国 TenAsys における登録商標です。
- ・ その他、記載されている会社名、製品名は、各社の商標又は登録商標です