

INDUSTRIAL  
EDGE  
SOLUTION  
WITH  
HARD REALTIME  
CAPABILITIES

**RT-edge**

Micronet.Co,

マイクロネット

**Micronet**

INDUSTRIAL EDGE SOLUTION WITH HARD REALTIME CAPABILITIES RT-EDGE

# WEB サーバーコンテナ

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)299-90-8557

## 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。

## 製品ご利用上の注意

(製品ご利用前に必ずお読みください)

本ソフトウェアのご使用の際には本マニュアルをお読みいただくとともにご使用に関しては、本マニュアルをお読みいただき、安全に対し十分に注意を払い、正しく取り扱うようお願いいたします。

## 機能拡張における注意事項



本製品ではご利用になる機能をパラメータ設定によるカスタマイズや、開発キットを利用し新たに機能を開発することによる拡張が可能です。パラメータ設定によりカスタマイズした場合、または開発機能を組み込み、搭載したシステムを再配布する前には必ず十分デバッグを行っていただくようお願いいたします。



本製品にて提供される機能利用により、産業通信により接続される運転中の PLC、ロボット、サーボ制御モーター等装置データにアクセスすることが可能です。制御用メッセージや、レジスタの取り扱いは接続装置に依存するため、装置のマニュアルを十分お読みいただき、ご理解いただいた上で行っていただくようお願いいたします。

ネットワーク経由により遠隔地からの機器・装置に対し制御を行う場合、ノイズその他を原因としたデータ通信異常により機器・装置側のトラブルに対応できない場合もあります。

## ネットワークセキュリティ上の注意事項



本製品を組み込んだシステム運用の際、外部機器からの不正アクセス、DoS 攻撃、コンピュータウィルス等、サイバー攻撃に対し、製品に接続するアクセス先機器やシステムセキュリティを保つため、コンピュータへのアンチウィルスソフトウェアの導入、ファイヤーウォール、VPN ルーター設置等の対策をご検討ください。

## システム設計上の注意事項



本製品を搭載した産業用 PC において強制電源 OFF する操作を行わないでください。強制電源 OFF 操作によりシステムデータ構造の破損や不備が生じ、不安定な挙動や誤動作が発生する可能性があります。

## 本ソフトウェアの適用について

本ソフトウェアをご使用いただく際、ソフトウェアにおける不具合・不備が生じたが発生した場合でも重大な事故に至らない用途であること、および故障、不具合発生時には製品外部にバックアップ、もしくはフェールセーフ機能が設けられていることを条件とさせていただきます。

マイクロネット社(以下 当社)は、当社製品の品質、性能、安全に係る一切の責任(債務不履行責任、瑕疵担保責任、品質保証責任、不法行為責任、製造物責任を含むがそれらに限定されません)を負わないものとします。

## 目次

<b>1. はじめに</b>	<b>5</b>
1.1. 全体構成	5
1.2. 概要	6
1.3. 構成要素	7
1.4. 用語解説	7
1.5. 関連資料	10
1.6. 実行環境	10
<b>2. コンテナ導入のフロー</b>	<b>11</b>
<b>3. インストール</b>	<b>12</b>
3.1. ファイル	12
3.2. ファイル配置	12
3.3. 起動設定	13
3.4. 動作確認	13
<b>4. 設定概要</b>	<b>15</b>
4.1. ECI 設定	15
4.1.1. RTCD 設定	16
4.1.2. コンテナプロパティ設定	17
<b>5. Web サービスコンテナ設定</b>	<b>18</b>
5.1. RTCD 設定概要	18
5.2. ECI(コンテナ設定情報ファイル)の編集	18
5.2.1. 1) タグ生成	18
5.2.2. 2) 入力タグ参照	20
5.2.3. 3) 出力タグ参照	20
5.3. コンテナプロパティ設定	21
<b>6. ASP.net 実行環境構築手順</b>	<b>22</b>
6.1. コンテナアプリケーションの作成	22
6.2. 環境変数の登録	28
6.3. IIS の設定	29
<b>7. 付属サンプルについて</b>	<b>33</b>
7.1. サンプルの概要	33
7.2. IoSample の起動	35
7.3. WebSample の起動	35
7.4. 画面説明	36
7.5. 定周期処理のテクニック	37
<b>8. デバック実行の為の設定</b>	<b>40</b>
8.1. デバック時 IIS Express の 64 ビット設定	40
8.2. スタートページの設定	40
<b>9. サービスインジケータタグ</b>	<b>41</b>
9.1. 一覧	41
9.2. サービス起動状態ステータス (.Status)	41
9.3. サービス異常状態ステータス (.Error)	41

9.4. サービス実行状態ステータス (.Run) .....	42
9.5. サービス実行カウンタ (.Live) .....	42
<b>10. WEB サーバーコンテナ API</b> .....	<b>43</b>
10.1. 関数リスト .....	43
10.1.1. API 機能 .....	43
■ EgWebInit .....	43
<b>11. トラブルシューティング</b> .....	<b>45</b>
11.1. EgStart 起動できません。 .....	45
11.2. スタートページの設定漏れ。 .....	47

## 1. はじめに

### 1.1. 全体構成

RT-edge は「標準コンテナ」「カスタムコンテナ」「RT-edge インターフェース」から構成されています。この構成の内、WEB サーバー コンテナは、「標準コンテナ」に該当し、下図 RT-edge 構成図内の「WEB サーバー」の部分になります。

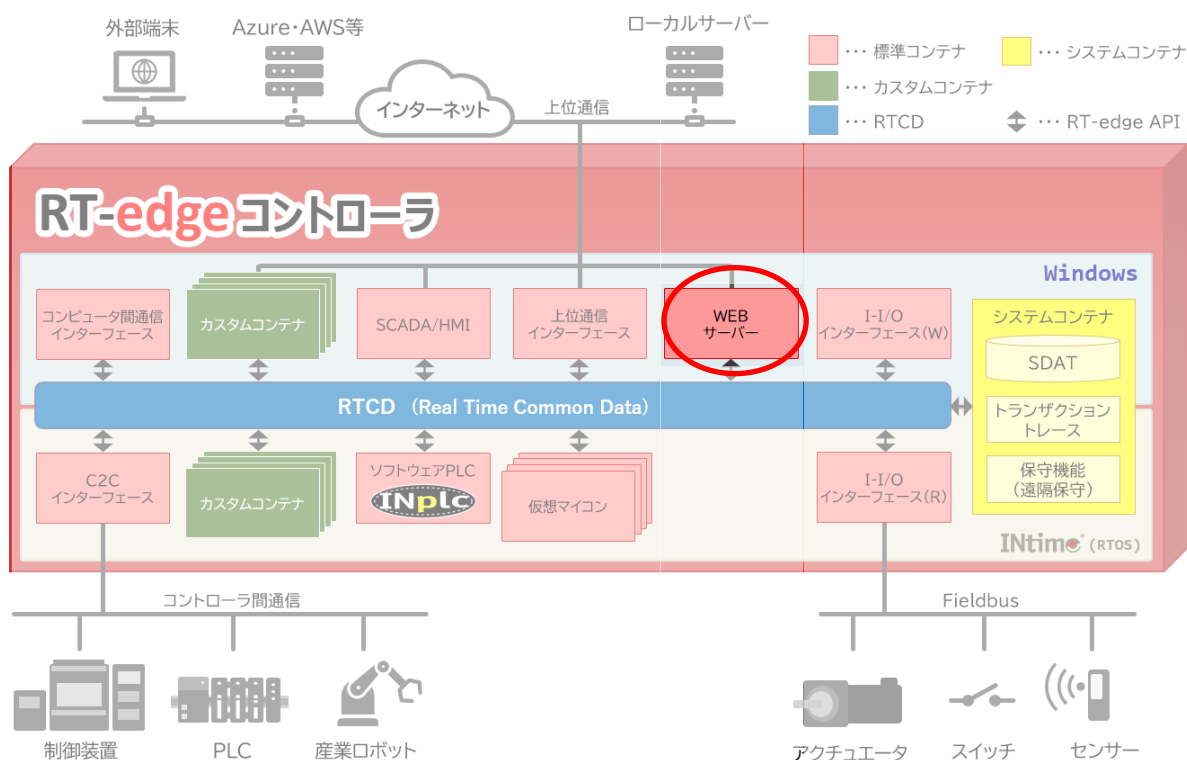


図 1. RT-edge コントローラ構成図

## 1.2. 概要

WEB サーバーコンテナでは、プログラミングで利用可能なキットを提供しています。

このキットを用いて、インターネットブラウザ経由でデータの閲覧、操作が可能な Web フォーム (コンテナサービスアプリケーション)を作成する事ができます。

Web フォームは、Microsoft ASP.NET(以下、ASP.NET)をベースに、IIS(Internet Information Services)経由にて、インターネットブラウザからのデータの閲覧、操作が可能になります。

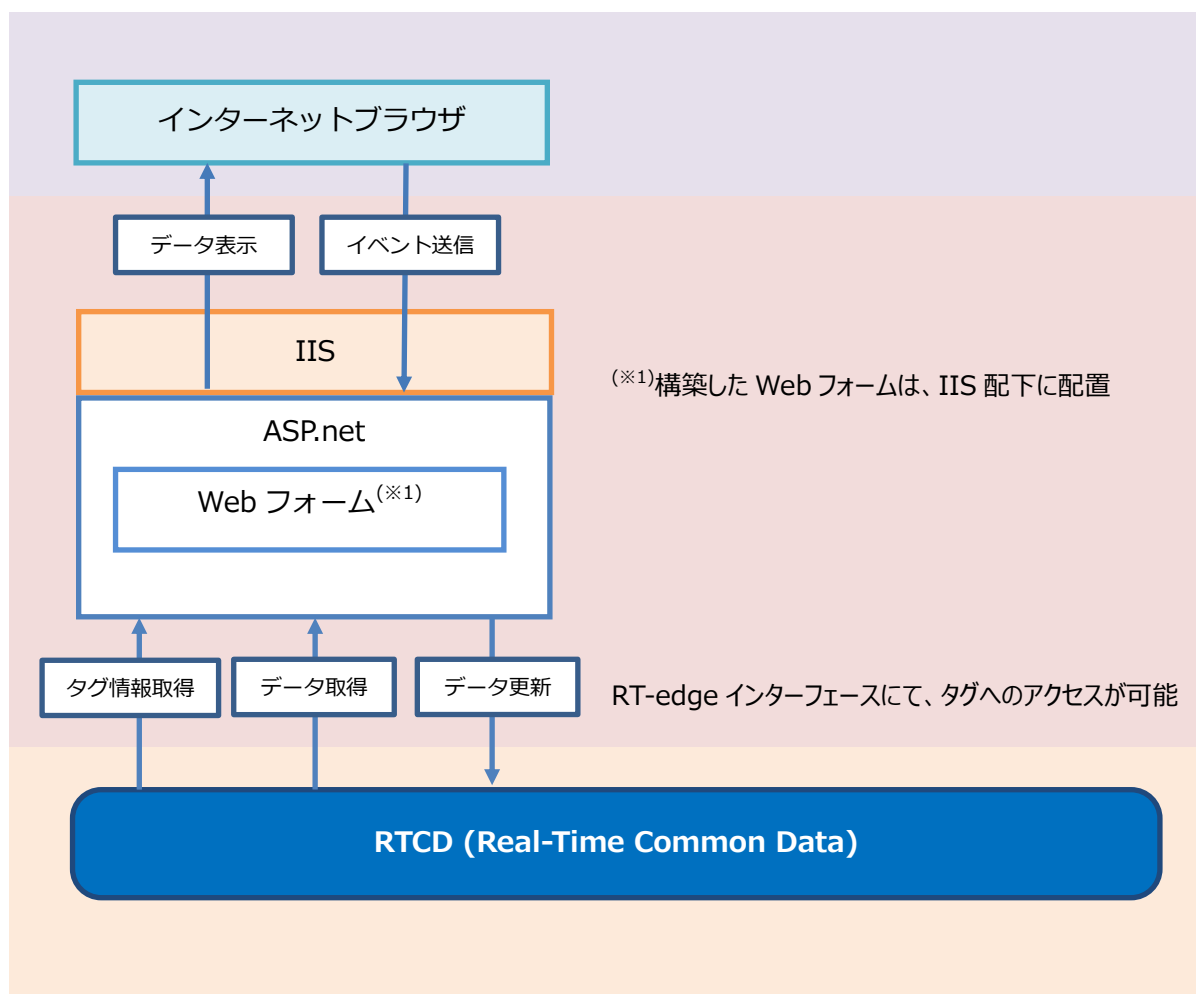


図 2. WEB サーバーコンテナ構成図

- 背景: RT-edge コントローラと同一ネットワーク内の機器のブラウザ
- 背景: WEB サーバーコンテナとして、お客様に作成、設定をしていただく部分
- 背景: RT-edge インターフェース

### 1.3. 構成要素

WEB サーバーコンテナは以下の構成になっています。

表 1. 構成一覧

ファイル	説明
DOCRTEGEDGESRV_Web.pdf	本マニュアル
WebSample	WEB サーバーコンテナのサンプル一式を収めたフォルダ

### 1.4. 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 2. 用語集

用語	説明
<b>RT-edge</b>	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。
<b>RT-edge ソフトウェア</b>	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。
<b>IoT ゲートウェイ</b>	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。
<b>IT システム</b>	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。
<b>RTOS プラットフォーム</b>	RTOS(RealTime Operating System)によりプライオリティベースのタスクスケジューリングを行う実行環境。ハードウェアアクセスや粒度が細かく精度の高い要求処理が実行プラットフォームとして適しています。 RT-edge は RTOS プラットフォームとして TenAsys 社製 INtime ソフトウェアを使用します。
<b>GPOS プラットフォーム</b>	GPOS(General Purpose Operating System: 汎用 OS)上で動作する環境。IT 処理全般の実行処理プラットフォームです。RT-edge、GPOS プラットフォーム(Windows OS)として Windows ソフトウェアを使用します。
<b>INtime</b>	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。 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。
<b>RTA</b>	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



用語	説明
	す。
<b>RSL</b>	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。
<b>API</b>	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。
<b>NTX</b>	INtime's Windows NT extension API: INtime 用 Windows NT 拡張 API の略称。NTX 関数は Windows プログラムが INtime リアルタイム環境上で実行するリアルタイムプログラムと通信を可能とする関数セットです。
<b>OPC</b>	主に産業オートメーション分野においてデータ交換を目的とした相互運用標準規格。
<b>OPC UA</b>	OPC UA(OPC Unified Architecture の略) 異なるプラットフォーム間のデータ交換を可能とした信頼性のある産業用通信データ交換標準。インダストリー4.0 の RAMI モデルに採用された規格。
<b>エッジアプリケーション</b>	RT-edge 内コンテナにより集積されたデータ(RTCD)を活用、処理実行するソフトウェアです。
<b>エッジコンピューティング</b>	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。
<b>オンプレミス</b>	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。
<b>クラウド</b>	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。
<b>産業用 PC</b>	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。
<b>データ収集</b>	診断、分析を行う対象となるデータを集積する処理。
<b>データ加工</b>	集積されたデータを利用しやすい形に変更する処理。
<b>産業機器通信インターフェース</b>	各種フィールドバス経由で機器、装置との通信、もしくは直接入出力デバイスの制御を行うインターフェースです。本インターフェースを介し、センサー値の参照やアクチュエータ制御が可能です。
<b>コンテナサービス/EgService</b>	RT-edge システムを構成するコンテナ機能内のインターフェースや、プロセス(rta/exe)を指します。
<b>タグ/EgTag</b>	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。
<b>リンクタグ</b>	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。
<b>データセット/EgDataset</b>	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

用語	説明
	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。
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。
サービスコンフィグファイル	RT-edge フレームワークが、タグやメールボックスなどの RT-edge オブジェクトの生成やサービスの起動を自動処理するための定義を記述した XML 形式のファイルです。
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	<p>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。</p> <p>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、または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機能です。</p> <p>RTCD 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。</p>
標準コンテナ	RT-edge 製品に含まれるコンテナです。
カスタムコンテナ	ユーザーが独自に作成したコンテナです。
IIS	Internet Information Services の略称、Web サーバー用の Windows の機能です。
ASP.net	.NET 環境の Web サーバー上で、Web ページの送信時に動的にプログラムを実行する仕組みです。ASP (Active Server Pages) の後継で、Web アプリケーションの開発・運用に用いられます。

## 1.5. 関連資料

本マニュアル以外にも、RT-edge 製品本体のインストール CD 内に下記のマニュアルがあります。  
必要に応じて合わせてご参照ください。

表 3. 関連資料

名称	ファイル名	内容
ユーザーズマニュアル	DOCRTEGEUSR.pdf	RT-edge システム全般的な内容の説明が記載されています。
API リファレンス マニュアル	DOCRTEGEAPI.pdf	各 API の使用方法が記載されています。
コンテナ作成マニュアル	DOCRTEGESRV.pdf	ローカル環境で動作する製品付属サンプルの構成及び操作方法の記載が含まれています。

## 1.6. 実行環境

WEB サーバー コンテナの動作要件として、以下製品が必要です。

- 1) RT-edge 3.3.0 以降
- 2) .NET Framework 4.6 以上

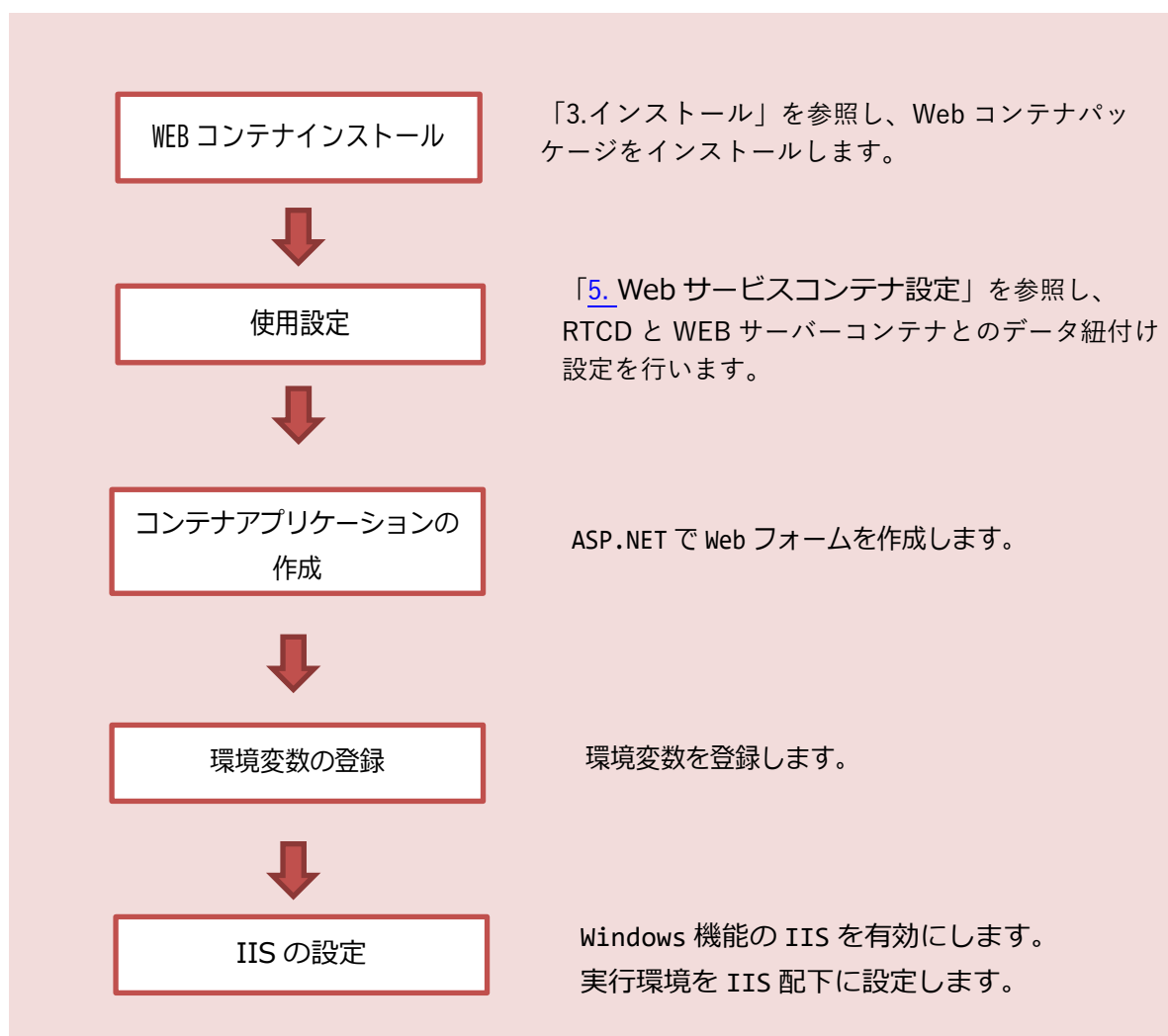


その他実行環境についての制限は RT-edge に準拠します。ユーザーズマニュアルをご参照ください。また、動作可能な CPU は INtime のバージョンにより異なります。詳細は INtime 製品ドキュメントをご参照ください。

## 2. コンテナ導入のフロー

WEB サーバーコンテナを用いて RT-edge システムにブラウザ経由でのデータ閲覧、変更機能を導入する作業は次の通りです。

WEB サーバーコンテナは RT-edge の API を取り込んだプログラムをお客様に作成して頂く必要がある、半成品の製品です。



## 3. インストール

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

### 3.1. ファイル

Web 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：

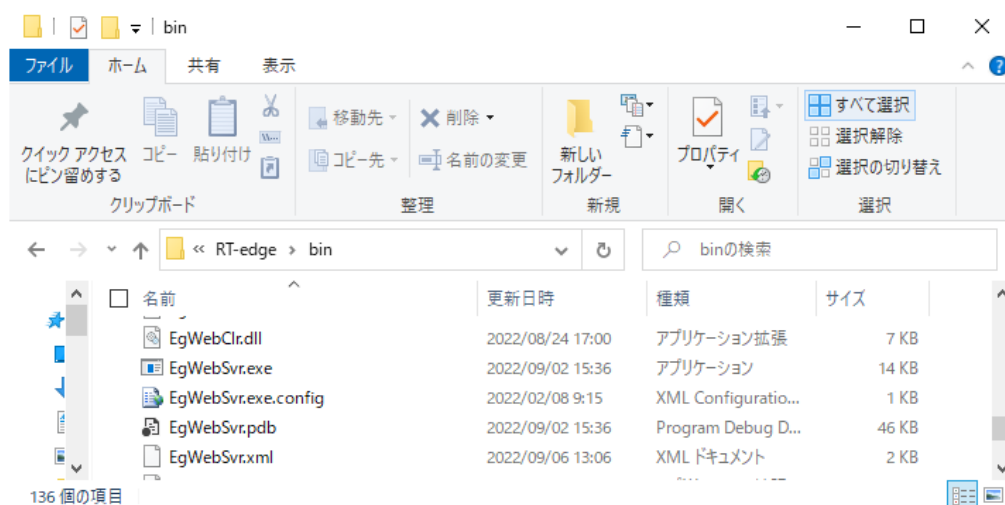
表 4.Web コンテナパッケージコンポーネント

配置先 フォルダ階層	ファイル名	説明
RT-edge¥bin¥	EgWebSvr.exe	Web サービスコンテナにおいて、タグ情報を RTCD に展開、そのタグ情報を Web プログラムに送信するサービスプロセスです。
RT-edge¥bin¥	EgWebSvr.exe.config	EgWebSvr.exe のアプリケーション構成ファイルです。
RT-edge¥bin¥	EgWebSvr.xml	Web サービスコンテナ用 RT-edge 設定情報です。本設定ファイルの構成により Web サービスコンテナが RTCD に生成するタグの設定を行います。
RT-edge¥bin¥	EgWebClr.dll	Web プログラムで使用するファイル。EgWebSvr.exe からのタグ情報取得、タグの読み書きを行う API です。

### 3.2. ファイル配置

Web コンテナパッケージは、ファイルコンポーネントを ZIP 圧縮した形式で配布されます。

.zip ファイル名称	RTedgeWeb.zip
.zip ファイル解凍先	C:¥RT-edge¥bin¥



### 3.3. 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。Web 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、Web サービスコンテナ用のElement (Service Element)を追加します。

```
<?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=" EgWebSvr" Path="EgWebSvr.exe" >
    </Service>
  </Services>
</RTedge>
```

図 1.EgWeb 起動登録

### 3.4. 動作確認

RT-edge ソフトウェアを起動し、Web サービスコンテナが正常に起動することを確認します:

- 3) RT-Edge ソフトウェア(C:¥RT-edge¥bin¥EgBoot.exe)を開始します。
- 4) RT-Edge オブジェクトブラウザ(C:¥RT-edge¥bin¥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) 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. サービスインジケータタグ」エラー! 参照元が見つかりません。を参照ください)。

---

**正常状態**

---

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

Name	Current Value	Type	Source	Comment
SERVICE.EgWebSvr	00000000 (0)	Int32		
SERVICE.EgWebSvr.Status	01 (1)	byte		
SERVICE.EgWebSvr.Error	False (0)	bool		
SERVICE.EgWebSvr.Run	True (1)	bool		
SERVICE.EgWebSvr.Live	0000260f (9743)	UInt32		

56 items available

図 2. RT-edge オブジェクトブラウザ起動時の様子

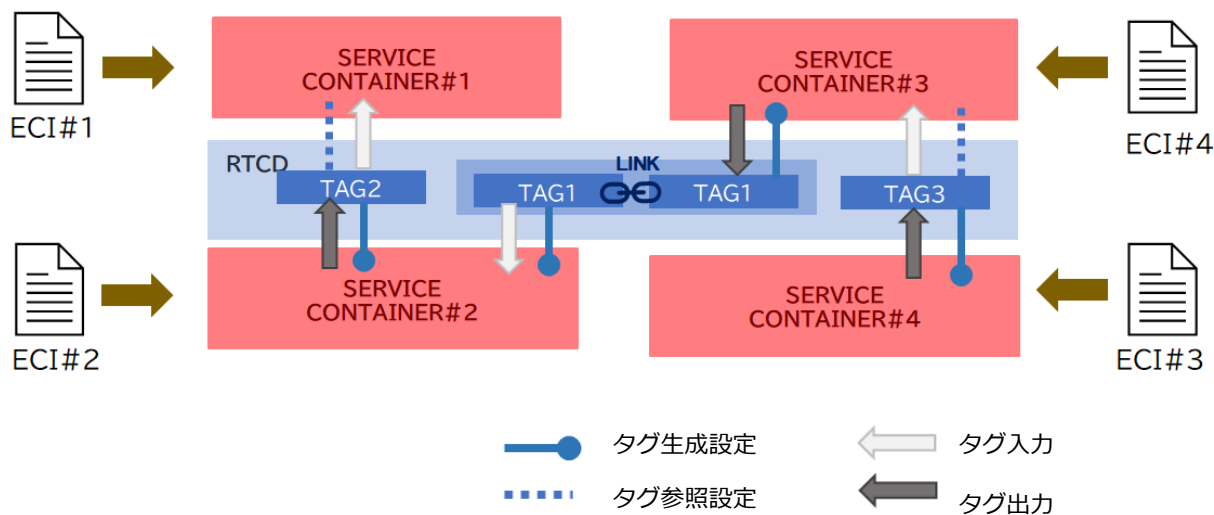
## 4. 設定概要

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 RTCD 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 RTCD 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
RTCD 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>

### 4.1. ECI 設定

RTCD 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。



### 4.1.1. RTCD 設定

RTCD 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><b>生成</b> &lt;Tags&gt;エレメント内に、&lt;Tag&gt;を生成します。 サービスコンテナ独自の名称(一意名)で&lt;Tag Name=&gt;の設定を行います。</p> <p><u>リンクタグ</u></p> <p>&lt;Tages&gt;エレメント内に、&lt;Tag&gt;を生成します。 他サービスコンテナの提供するタグと同名で&lt;Tag Name=&gt;の設定を行います。</p> <p><b>Tag</b></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>: &lt;TagRefs_IN&gt; エレメント内</p> <p><u>出力参照</u>: &lt;TagRefs_OUT&gt; エレメント内</p> <p>&lt;TagRef Name=&gt;に参照するタグを指定します。</p> <p><b>TagRef</b></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>&lt;Datasets&gt;エレメント内に、&lt;Dataset&gt;</p> <p><b>タグ・コレクション定義</b> &lt;Dataset Name=&gt;にデータセット名を指定します。</p> <p>Dataset エレメント内に、&lt;TagRefs&gt;エレメントを作成します。</p> <p>&lt;TagRef Name=&gt;に参照するタグを指定します。</p> <p><b>TagRef</b></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>&lt;Collectors&gt;エレメント内に、&lt;Collector&gt;を作成し</p> <p>&lt;Collector Name=&gt;に名称を設定します (Dataset を収集する機能名)</p> <p><b>Collector</b></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。										

#### 4.1.2. 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>&lt;Tags&gt;エレメント内に、&lt;Tag&gt;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><b>Tag</b></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。

## 5. Web サービスコンテナ設定

### 5.1. RTCD 設定概要

Web サービスコンテナではタグ定義設定を行います:

メモリ領域 を RTCD に公開	ローカルタグ生成 + タグ参照	メモリ領域を RTCD に定義するタグのデータ入出力先と して割り当てる。
---------------------	-----------------------	--

#### ■ 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:EgWebSvr.xml)の編集

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。

#### ■ タグ参照設定

Web サービスコンテナでは、生成したローカルタグに対する内部参照設定が必要です。本設定により生成したタグに対する入出力方向を定義し、タグデータの更新が行われます。

### 5.2. ECI(コンテナ設定情報ファイル)の編集

#### 5.2.1. 1) タグ生成

- 1) ECI コンテナ設定情報(EgWebSvr.xml)ファイルを開きます。
- 2) RTedge > Tags 内に、Tag エlementを追加します。

```
<Tag Name="Web001" Type="6" Address="%D0" Comment="Web サーバー 任意タグ1"/>  
<Tag Name="Web002" Type="11" Address="%L4" Comment="Web サーバー 任意タグ2"/>  
<Tag Name="Web003" Type="10" Address="%D12" Comment="Web サーバー 任意タグ3"/>
```

(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。

表 5. 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

PLC データ型	Type	Size	用途	RT-edge データ型(egTag)
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

#### (ウ) Address 設定

Web サービスコンテナにおける Address 設定は任意項目です。メモとして使用できます。

### 5.2.2. 2) 入力タグ参照

Web ページから入力を行うタグを設定します。

入力が不要な場合、本設定は行いません。

- 1) RTEdge > Services > Service(Name 属性=EgWebSvr) > TagRefs > TagRefs\_IN エlementを追加します。
- 2) TagRefs\_IN Element内に、TagRef Elementを追加します。
- 3) TagRef の Name 属性には、Web サービスコンテナで定義されるローカルタグを設定します。

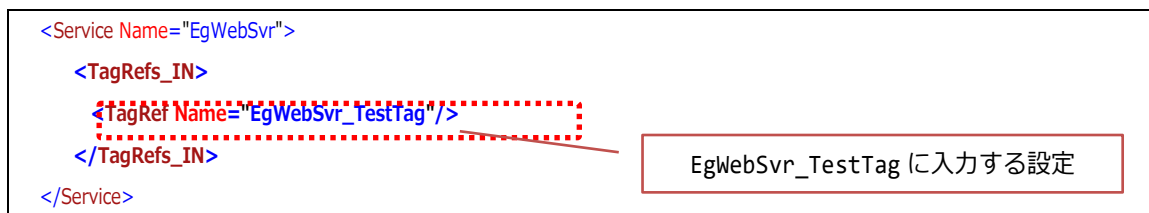


図 4. TagRefs\_IN の設定



存在しないタグを指定した場合は無視されます。

### 5.2.3. 3) 出力タグ参照

Web ページに対して出力を行うタグを設定します。

出力が不要な場合、本設定は行いません。

- 1) RTEdge > Services > Service(Name 属性=EgWebSvr) > TagRefs > TagRefs\_OUT Elementを追加します。
- 2) TagRefs\_OUT Element内に、TagRef Elementを追加します。
- 3) TagRef の Name 属性には、Web サービスコンテナで定義されるローカルタグを設定します。

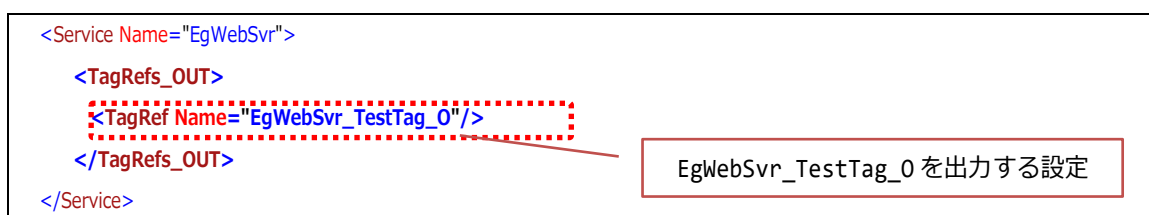


図 5. TagRefs\_OUT の設定



存在しないタグを指定した場合は無視されます。

### 5.3. コンテナプロパティ設定

Web サービスコンテナには、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。

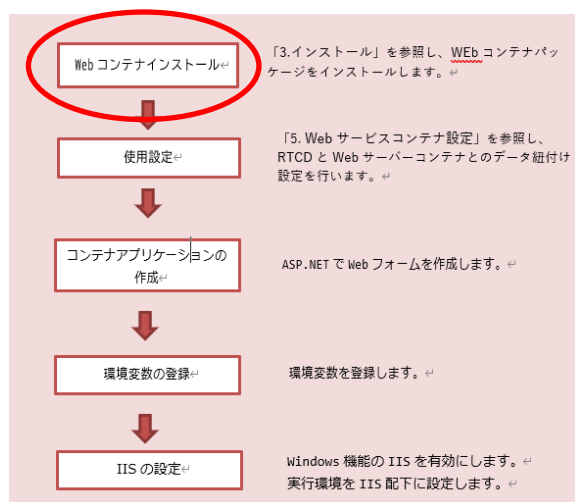
設定	Tag 名	デフォルト設定	内容
サービス 起動設定	SERVICE.EgWebSvr.AutoRun	1 (タグデータ更新開始)	サービス起動時のタグデータ更新自動開始を指定します。
ページオープン カウント	SERVICE.EgWebSvr.PageOpenCnt	0	Web ページのオープン回数のカウント用に準備したタグです。



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。

## 6. ASP.net 実行環境構築手順

### 6.1. コンテナアプリケーションの作成



#### 1) ソリューションの作成

Visual Studio 2017 を起動して、新規作成で下記のテンプレートを使用します。  
「Visual C#」「Web」「ASP.NET Web アプリケーション(.NET Framework)」

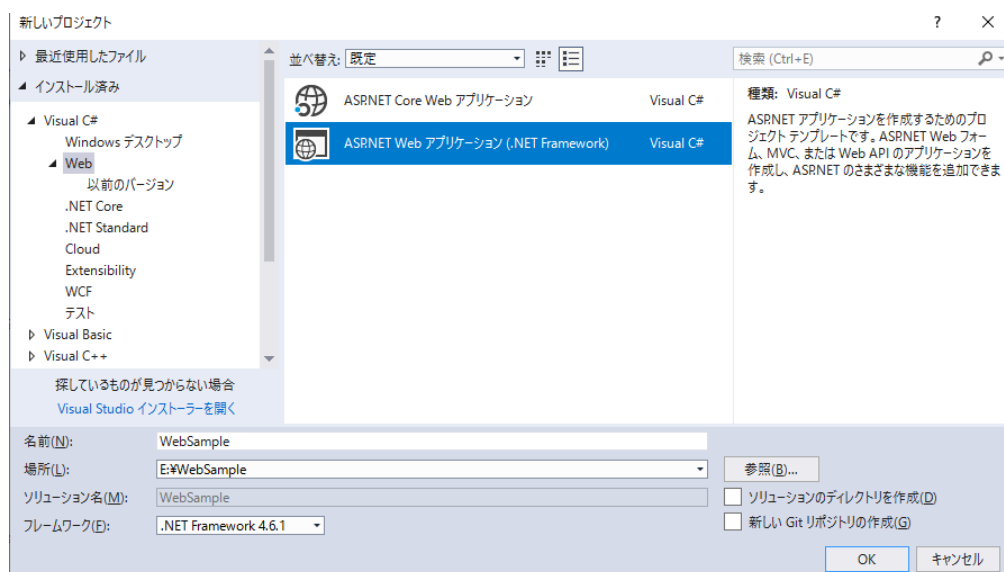


図 3. 新規作成時のテンプレート

## 2) Web フォームの追加

Web フォームを追加します。

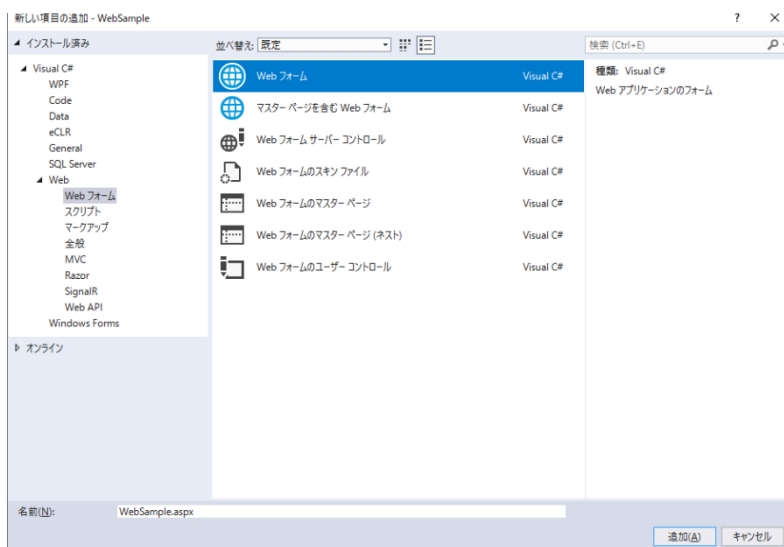


図 4.Web フォームの追加

Web フォームを追加すると「.ASPX」の拡張子で空の画面が追加されますので、これに RT-edge で管理しているタグデータを表示するラベルやボタン等を追加します。

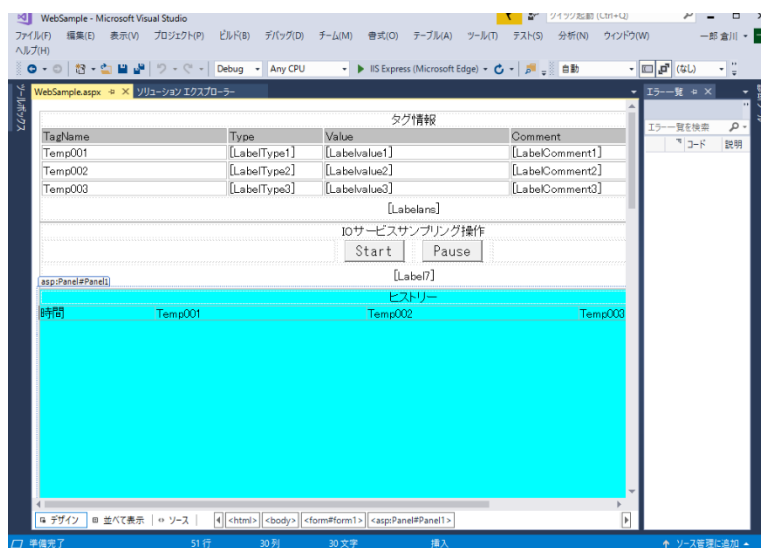


図 5.Web フォームの編集



## 3) 参照設定の追加

RT-edge のインストールフォルダー内の「egapiWrap.dll」「EgWebClr.dll」を参照設定に追加します。

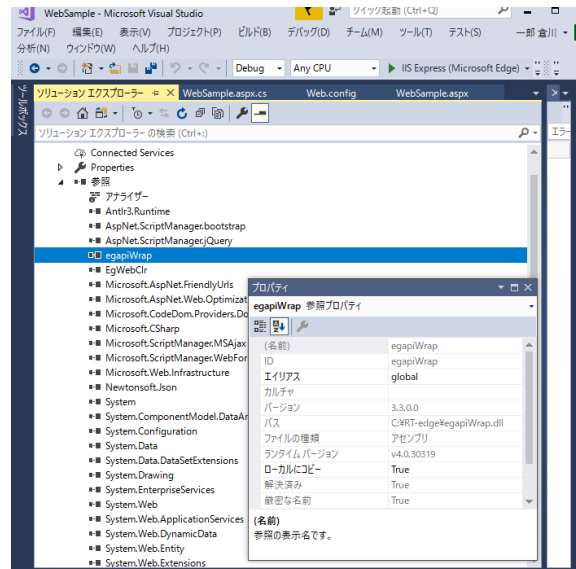


図 6.参照設定

## 4) API 宣言記述の追加

Web フォームを追加すると「.ASPX.CS」拡張子のファイルが追加されます。このファイルにイベントの処理を作成します。

RT-edge で管理しているタグデータにアクセスする為に RT-edge API を用意しています。

WEB アプリケーションで使用頻度の高い API について以下に記載します。

API の詳細については「[API リファレンス マニュアル](#)」を参照してください。

using ステートメントを追加します。

## ステートメント追加

```
...
using RTedge; // RT-edge system API
...
```

API の宣言をします。

## API の宣言

```
...
edge_API EGAPI = new edge_API(); //APIの宣言
...
```

## 5) タグへの読み書き

下記はタグの列挙体取得の API 呼び出しの例です。。

### Value 値の取得

```
//ページロードイベント
protected void Page_Load(object sender, EventArgs e)
{
    //最初の画面表示のみの処理を記述
    if (IsPostBack == false)
    {
        //初期化してタグリストを取得する
        if (EgWebClr.EgWeb_API.EgWebInit() != 0) return;
    }
}
```

## 6) タグへの読み書き

下記はタグ名を指定してタグの値を取得する例です。値を条件判定に使用したり、画面のラベルに表示したりして使用します。

### Value 値の取得

```
//タグの値取得関数
//引数    タグ名
//戻り値  タグの値
protected string proc_ValueGet(string sttagname)
{
    //この関数の戻り値
    string stans = "";
    //APIのステータス
    int ians = 0;
    //APIのタグの値
    object value = new object();
    //APIでタグの値を読み込む
    ians = EGAPI.EgTagRead(sttagname, ref value);
    if (ians == 0){stans = value.ToString();}
    return stans;
}
```

下記はタグ名を指定してタグの値をセットする例です。

#### Value 値のセット

```
//タグのセット関数
//引数    タグ名
//戻り値  ステータス
protected int proc_ValueSet(string sttagname, int idata)
{
    //APIの戻り値の変数
    int ians = 0;
    //APIでタグの値を書き込む
    ians = EGAPI.EgTagWrite(sttagname, idata);
    return ians;
}
```

下記は受信待ちをしているスレッドのメールボックスに対してコマンドを送信する例です。

#### メールボックス送信

```
//スタートボタンのクリックイベント
protected void ButtonStart_Click(object sender, EventArgs e)
{
    //WEBアクセスが許可中か
    if (proc_ValueGet("SERVICE.IoSmp1.WebOpe") == "True")
    {
    }
    else
    {
        //両方のボタン使用不可
        ButtonStart.Enabled = false;
        ButtonPause.Enabled = false;
        return;
    }
    //[IoSmp1]コンテナにSTART CMDを送る
    //C版 引数(メールボックスハンドル, 自分のコンテナ名, 自分のコンテナ名
    //      サイズ, コマンド番号、 データ、 データサイズ)
    //C#版 引数(相手のコンテナ名, 自分のコンテナ名, コマンド番号、 データ
```

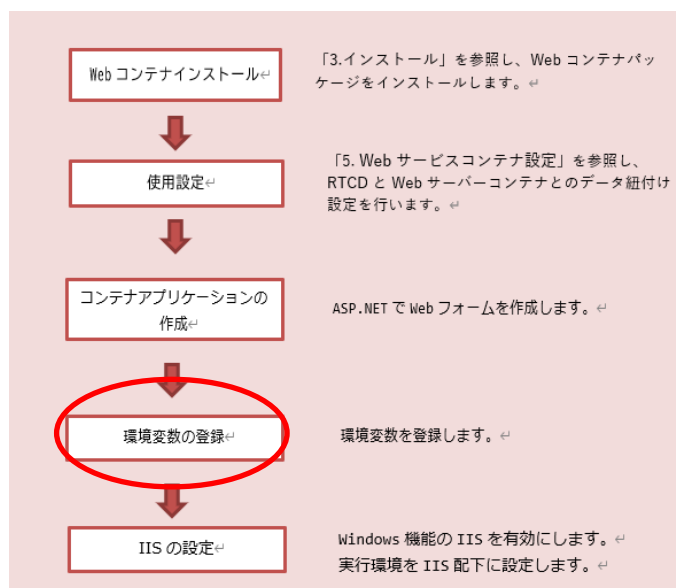
(指定なし可))

```
int ret = EGAPI.EgMailboxSend("IoSmp1", "SERVICE.IoSmp1.Run",  
                               (int)EGDEFINE.EGMSG.EM_SERVICE_RUN);  
  
ButtonStart.Enabled = false;  
ButtonPause.Enabled = true;  
}
```



上記例は停止しているサービスに対して、再開するメールを送信しています。。

## 6.2. 環境変数の登録



RT-edge の Web サーバーコンテナでは、API ファイルの配置場所を特定するのに環境変数を使用します。未登録の場合は登録してください。

- 1) システム環境変数へ下記情報を登録します。  
変数名：RTEGDE  
値：RT-edge 製品をインストールしたフォルダーのパス

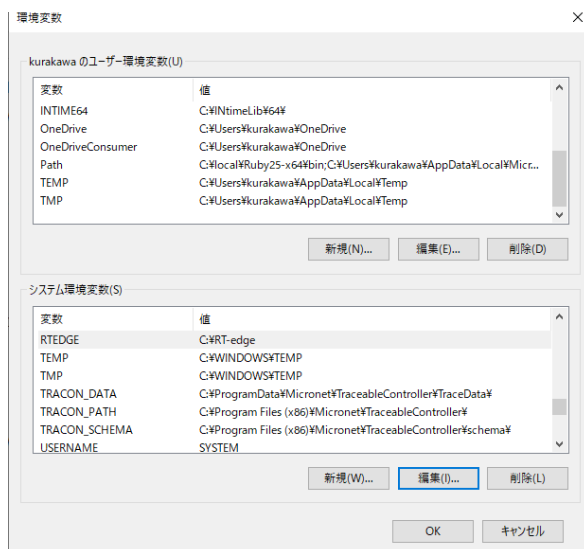
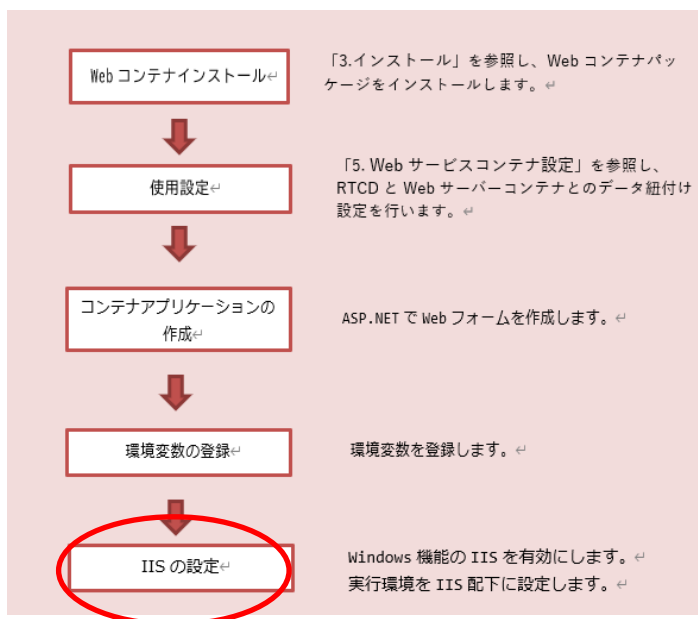


図 7. 環境変数の登録画面

- 2) 登録後は設定を反映する為に PC の再起動をお願いします。

### 6.3. IIS の設定



- 1) [Windows の機能]の画面で[インターネットインフォメーションサービス]の下記図の項目にチェックをいれて機能を有効にします。

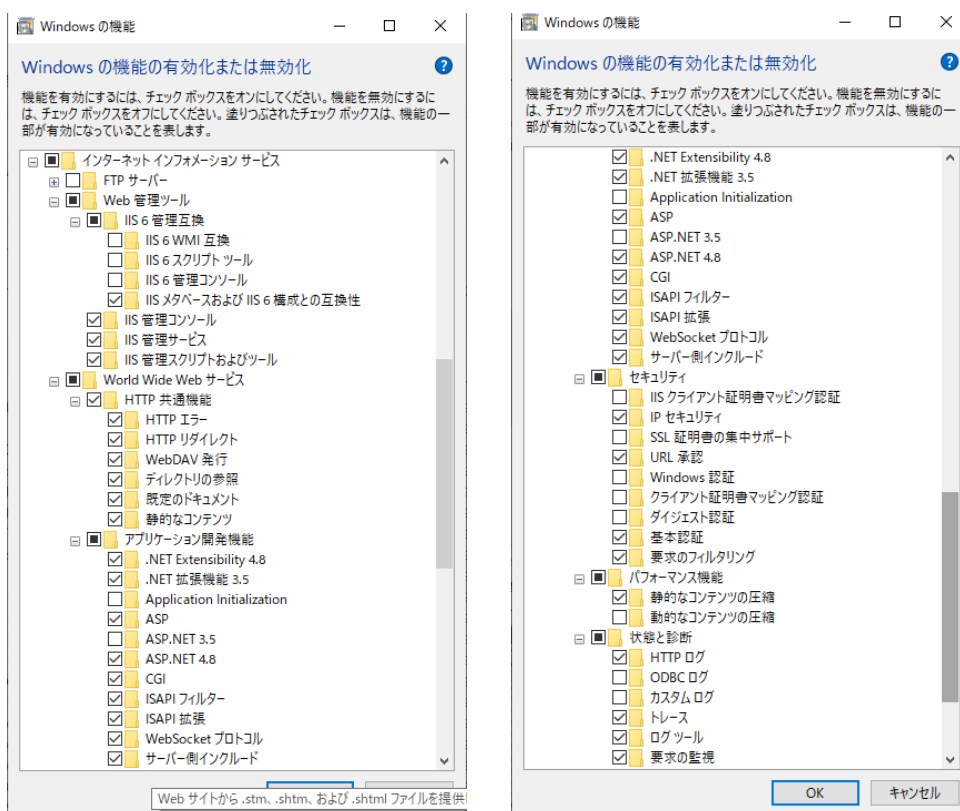


図 8. Windows の機能

2) 検索に[iis]と入力して[IIS マネージャー]を選択します。



図 9. IIS の起動

- 3) [IIS マネージャー] 画面で[Default Web Site]を選択して右クリックして、[アプリケーションの追加]を選択します。

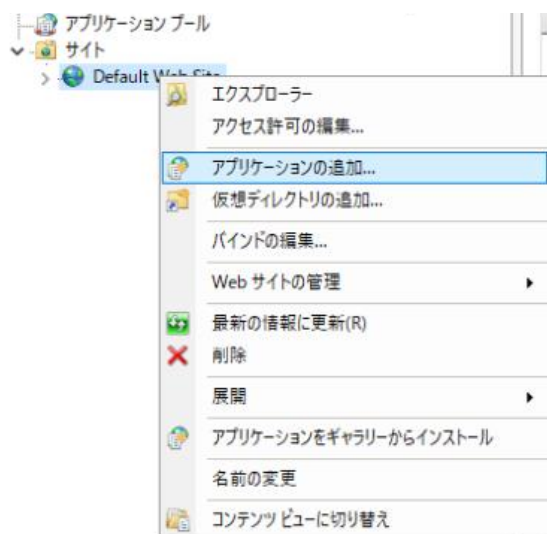


図 10. アプリケーションの追加の選択

- 4) [アプリケーションの追加]画面で下記情報を入力して登録します。  
 エイリアス： 任意の登録名（ブラウザから呼び出すときの URL となります）  
 物理パス： WebSample の物理パス

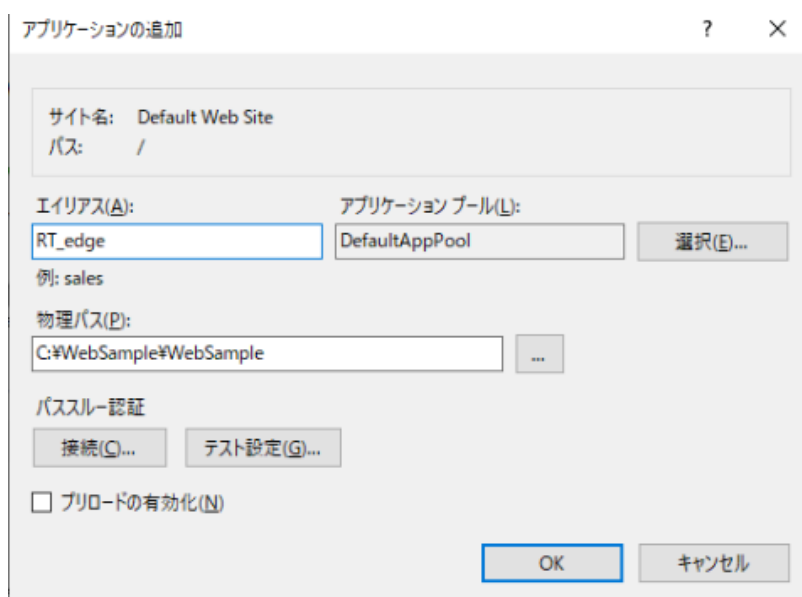


図 11. アプリケーションの追加



- 5) [IIS マネージャー] 画面で[アプリケーションプール]を選択、該当のアプリケーションプールを右クリックして[詳細設定]選択します。

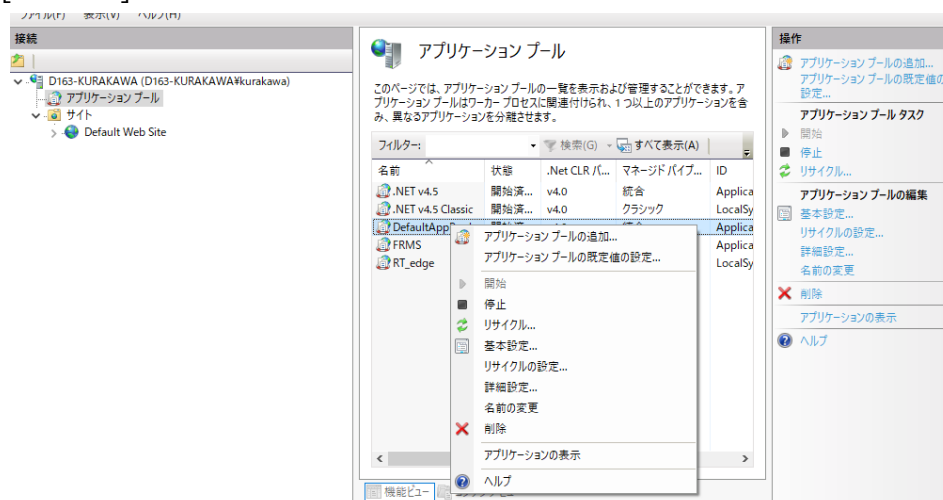


図 12. アプリケーションプールの選択

- 6) [プロセスモデル]の[ID]の設定を[LocalService]に設定します。

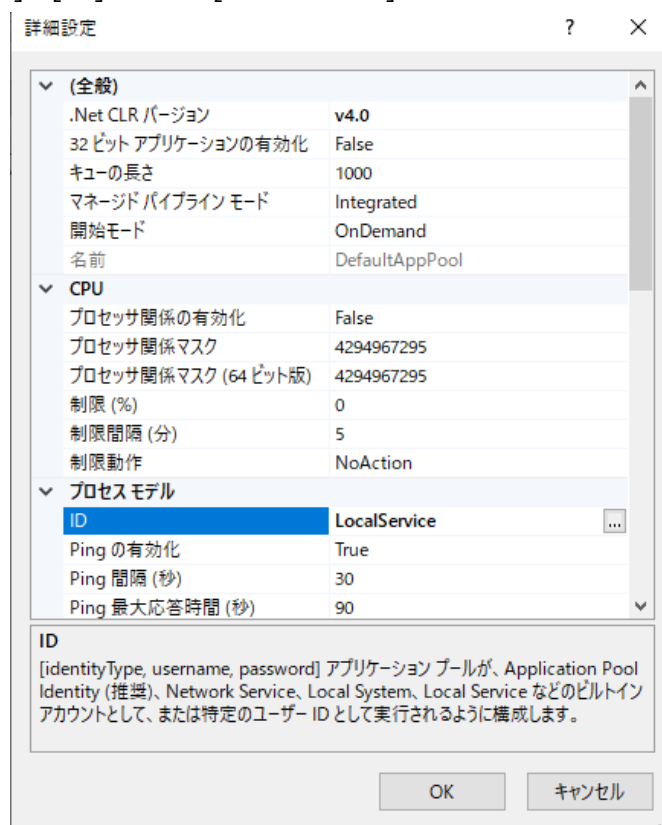


図 13. 詳細設定

## 7. 付属サンプルについて

### 7.1. サンプルの概要

サンプルは温度計の変化をリアルタイムに表示する処理をイメージしています。

疑似的に温度変化を発生させて表示しています。

サンプルは疑似的に温度変化を発生させるサンプル (IoSample) と IIS 配下に設定してブラウザに表示する (WebSample) の二つの構成になっています。

WebSample は製品付属サンプルが作成したタグにアクセスする処理を、開発言語 ASP.NET プログラムで作成しています。

#### 1) IoSample の実行環境

Web サーバーコンテナインストール CD 内の IoSample.zip も中の

¥EgWeb¥IoSample

Rt-edge のインストールフォルダにコピーして使用します。

#### 2) WebSample の実行環境

Web サーバーコンテナインストール CD 内の EgWeb.zip も中の

¥EgWeb¥WebSample

任意の位置にコピーして IIS 配下に設定して使用します。(6.3 IIS の設定を参照)

- 3) 下記の構成図のピンク色の部分が Web サーバー コンテナの部分です。  
 ピンク色以外の部分がローカルサンプルの部分です。

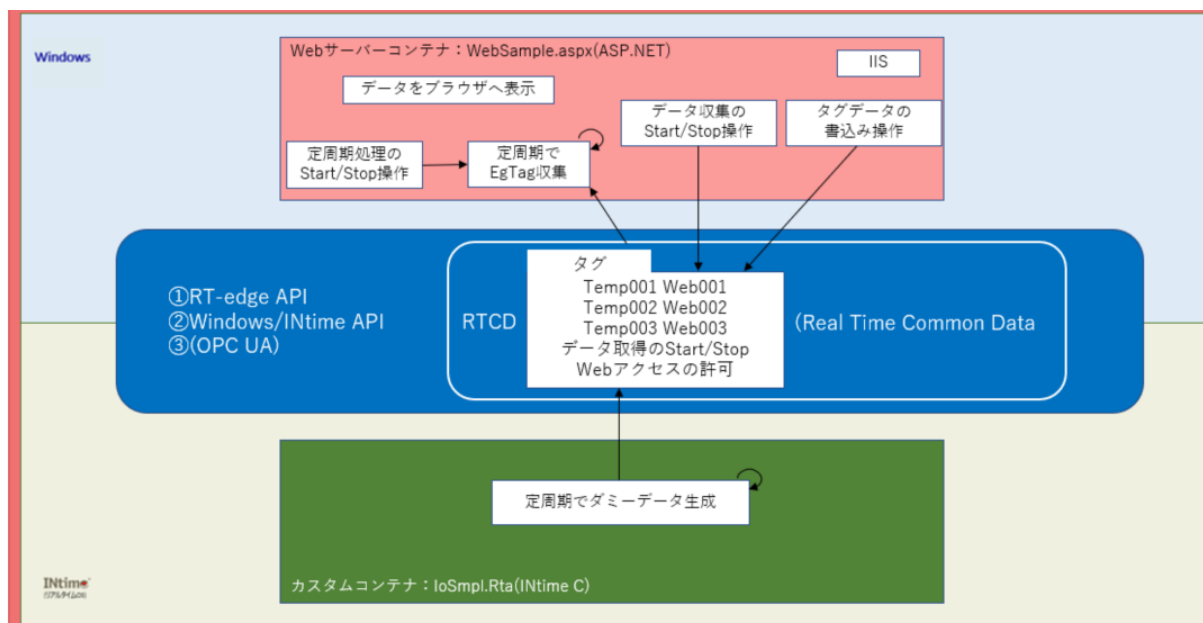


図 14. サンプルの構成図

#### 色分けの説明

- Web サーバーコンテナ 部分
- カスタムコンテナ 部分
- RT-edge インターフェース 部分

## 7.2. IoSample の起動

RT-edge インストールフォルダの EgBoot.exe を実行して起動します。

## 7.3. WebSample の起動

ブラウザの URL 欄に[XXX.XXX.X.XXX/RT\_edge/WebSample.aspx]と入力すると WebSample の画面を表示します。

XXX.XXX.X.XXX : IIS を設定した PC の IP アドレス

RT\_edge : エイリアス名

WebSample.aspx : Web フォーム名

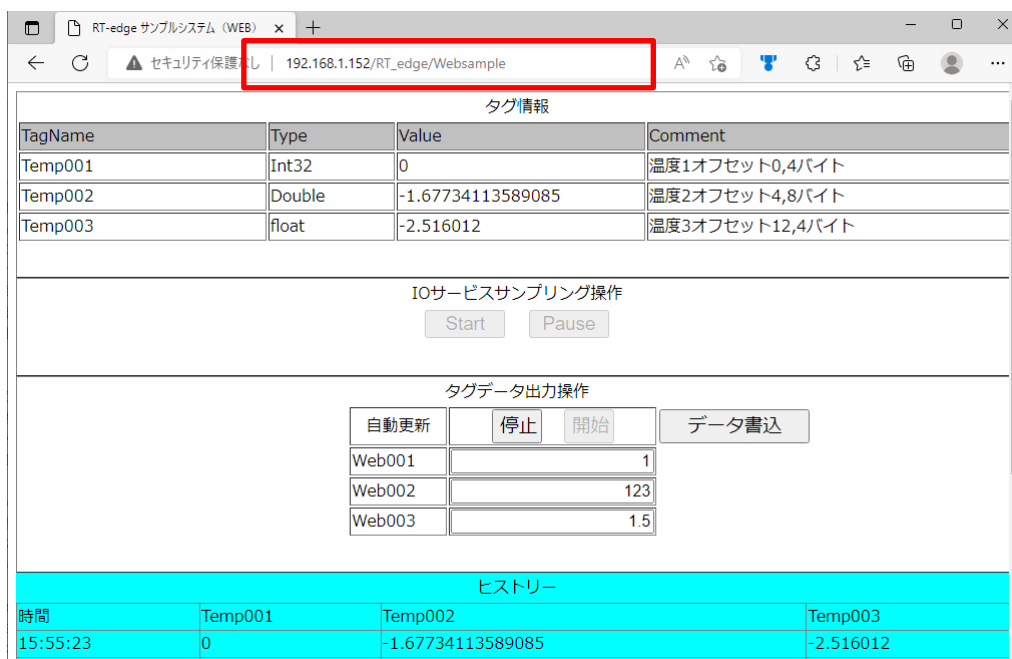


図 15. 起動の例

7.4. 画面説明

1 秒周期でタグデータを取得して画面に表示します。

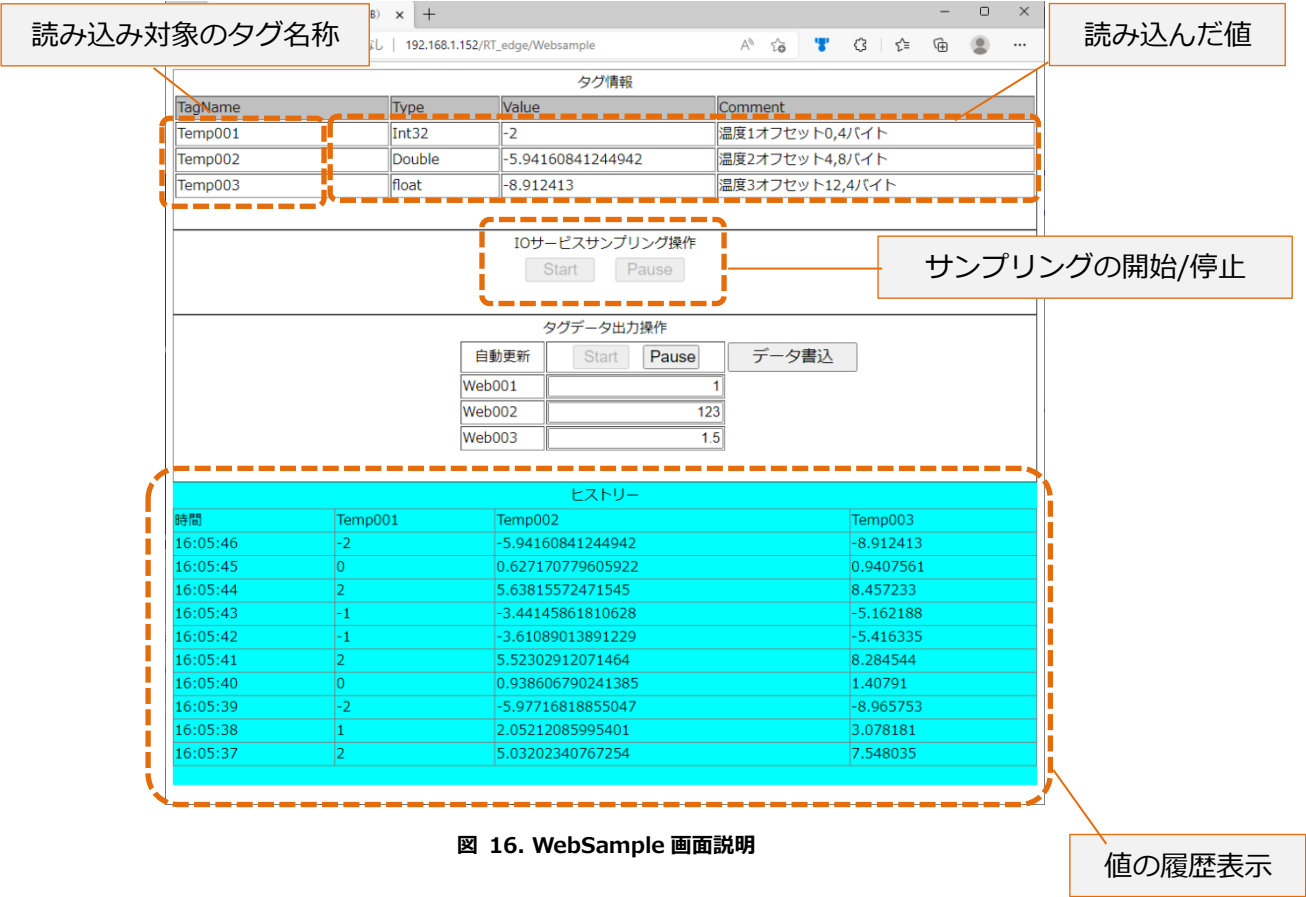


図 16. WebSample 画面説明

IO サンプリング操作について  
初期状態は操作不可となっています。  
[ローカルサンプル]の画面が[WEB アクセス禁止中]となっていますので  
[許可へ変更]ボタンをクリックすると操作可能となります。

Pause : 一時停止  
Start : 再開

## 7.5. 定周期処理のテクニック

ASP.NET の Web フォームでは、クライアント上で発生したイベントは、いったんサーバー側にポストバックされたうえで、サーバー側で処理される。そして、サーバー側で処理した結果を基に、ページ全体をもう一度描画し直すと動作をします。このようなポストバック処理は負荷が高く、イベントが発生するたびにページ全体をリフレッシュすることから画面のちらつきが発生します。

これを改善する為のサンプルではクライアント・コールバックを利用しています。

クライアント・コールバックでは、イベント発生時に必要なデータだけをサーバーに送信し、必要なデータだけその応答として受け取ることができます。

クライアント・コールバックを利用することで最小限のデータをサーバーから取得してページに表示しています。

定周期処理のながれ

- 1) ページ上に表示されない隠しボタンを配置する（例では Button\_ibent というボタン）
- 2) Page\_Load イベントにクライアント・コールバックイベントを作成して、隠しボタンのクリックイベントに設定する。
- 3) サーバー側にイベント発生時にページに表示する為に必要なデータを取得する記述をする。
- 4) ページに定周期のイベント記述する（setInterval 命令を使用）
- 5) 定周期イベントの中で隠しボタンのクリックイベントを呼ぶように記述する。
- 6) クリックイベントによって取り込んだデータをページの表示項目にセットする。

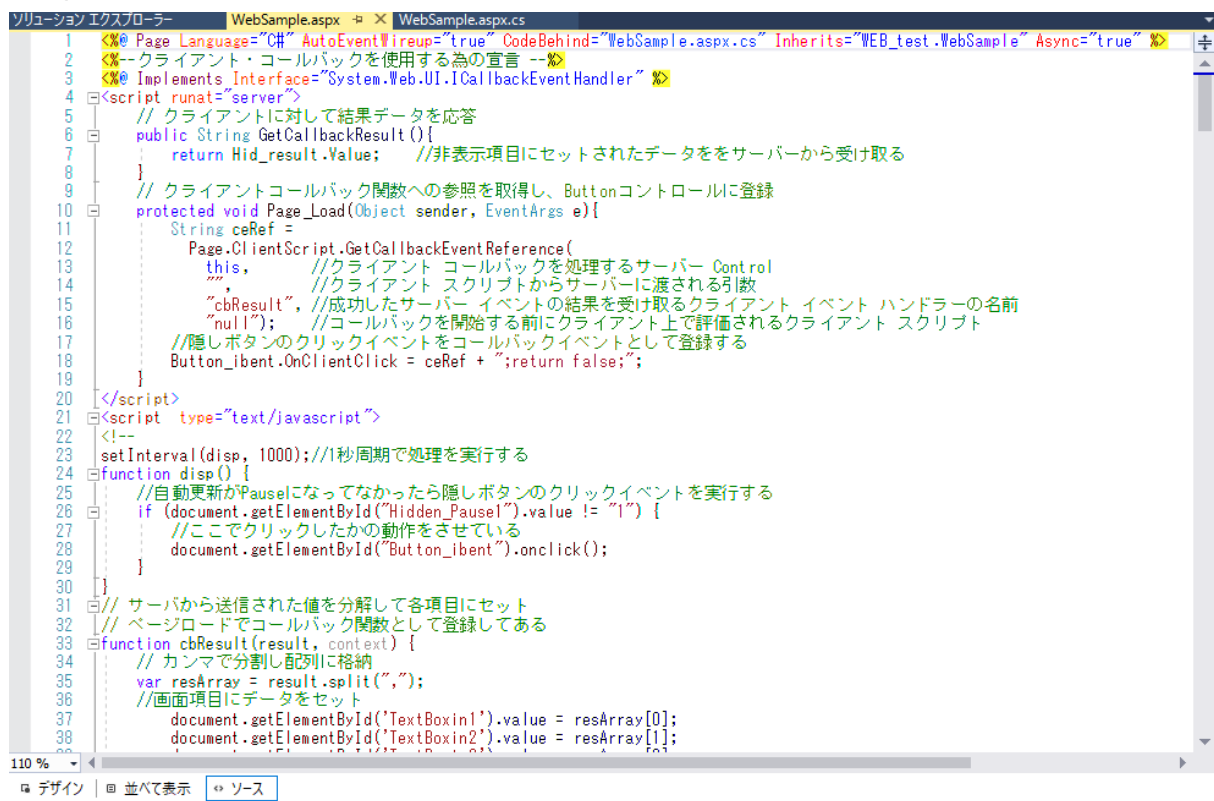


図 17. ページ側のクライアント・コールバックの記述例

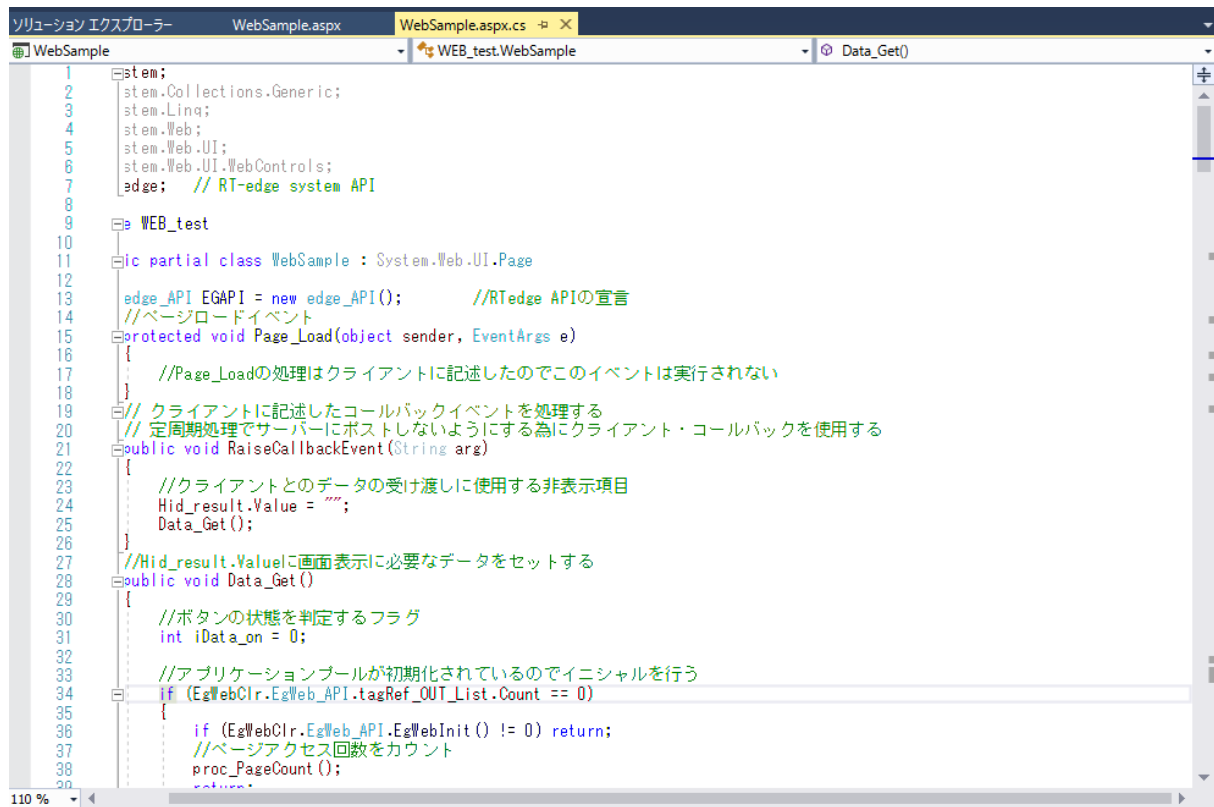


図 18. サーバー側のクライアント・コールバックの記述例

## 7.6. サンプル動作中の IIS の設定

IIS のアプリケーションプールの設定にリサイクル処理の設定があります。

既定値の設定だと 1740 分（29 時間）毎にリサイクルが実行されます。

リサイクルが実行されるとブラウザとサーバー間の接続が切断されます。

長時間サンプルを実行する場合は 0 分に設定を変更してリサイクルを停止してください。

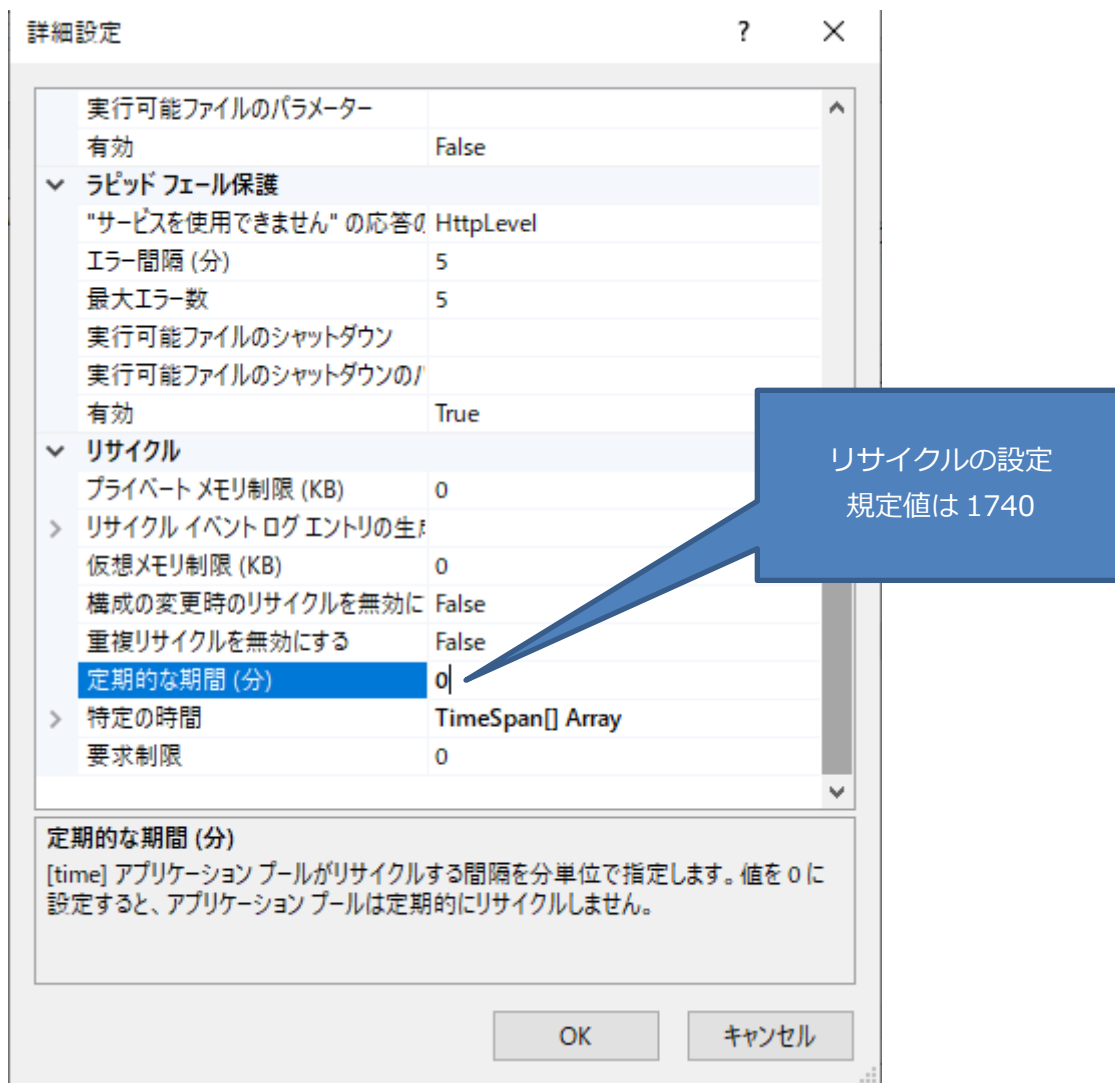


図 19. サンプル実行時のリサイクル設定例



## 8. デバック実行の為の設定

### 8.1. デバック時 IIS Express の 64 ビット設定

[ツール]タブ[オプション]を選択して[オプション]メニューの[プロジェクトおよびソリューション] [Web プロジェクト] の[Web サイトおよびプロジェクト用 IIS Express の 64 ビットバージョンを使用する]の項目にチェックを入れます。

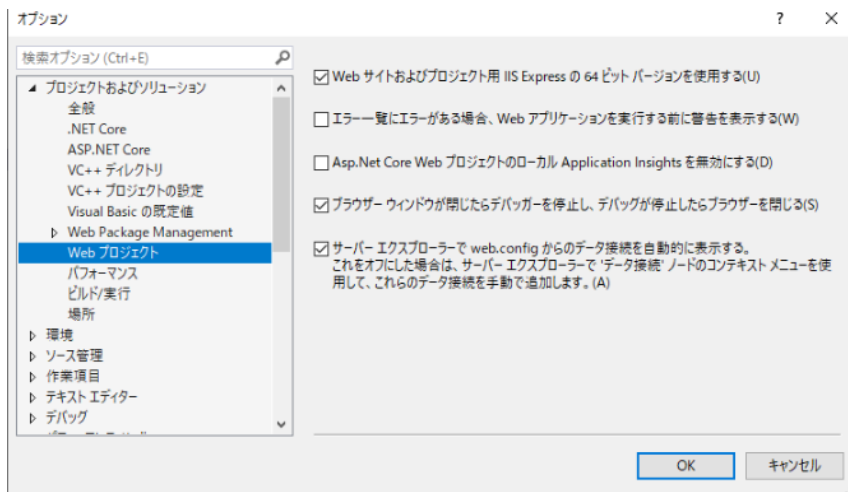


図 20. オプション設定



RT-edge の API は 64 ビット用に作成されていますが、Visual Studio でのデバックで使用している IIS の初期値は 32 ビットのため、デバックが実行できません。この設定をすることによりデバック実行が可能になります。

### 8.2. スタートページの設定

[ソリューションエクスプローラー]を表示して[WebSample.aspx]を選択して右クリック、[スタートページに設定]を選択します。デバック実行をする時にこの設定が必要になります。

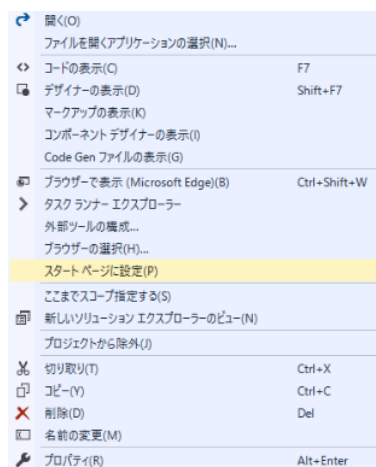


図 21. スタートページの設定

## 9. サービスインジケータタグ

### 9.1. 一覧

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

### 9.2. サービス起動状態ステータス (.Status)

Tag		備考
Name	<code>SERVICE.EgWebSvr.Status</code>	現在のサービス起動状態を示します
Type	<code>uint8</code>	
Size	<code>1</code>	
Value	<code>0</code>	
値	意味	備考
0	サービスは起動していません	Web サービスコンテナ終了処理で設定されます。
1	サービスは起動されています	Web サービスコンテナ起動処理で設定されます。

### 9.3. サービス異常状態ステータス (.Error)

Tag		備考
Name	<code>SERVICE.EgWebSvr.Error</code>	現在のサービスのエラー発生状態を示します
Type	<code>bool</code>	
Size	<code>1</code>	
Value	<code>0 (false)</code>	初期値
値	意味	備考
0 (false)	エラーはありません	
1 (true)	1 つ以上の通信障害が生じています。	「

## 9.4. サービス実行状態ステータス (.Run)

Tag		備考
Name	SERVICE.EgWebSvr.Run	現在のタグ参照・更新動作の状態を示します
Type	bool	
Size	1	
Value	0 (false)	初期値

値	意味	備考
0 (false)	タグ参照・更新は停止しています	タグ参照・更新を行わない状態。サービスメッセージ EM_SERVICE_PAUSE 受信後、タグ参照・更新トリガスレッド動作時に設定されます。
1 (true)	タグ参照・更新は活性化しています。	タグ参照・更新可能な状態。サービスメッセージ EM_SERVICE_RUN 受信後、タグ参照・更新トリガスレッド動作時に設定されます。

## 9.5. サービス実行カウンタ (.Live)

Tag		備考
Name	SERVICE.EgWebSvr.Live	サービスが健全であることを示すカウンタ
Type	uint32	
Size	4	
Value	0	初期値

値	意味	備考
0~0xffffffff	指定周期の満了回数	サービスプロセスの健全性確認手段として用意されています。値が変化していることで健全であることを表します。

## 10. WEB サーバーコンテナ API

### 10.1. 関数リスト

以下リストの API を提供します:

#### 10.1.1. API 機能

API 関数	説明
EgWebInit	タグの列挙体の取得処理

#### ■ EgWebInit

タグの列挙体を取得します:

#### シンタックス

```
int32_t EgWebInit ();
```

#### 戻り値

ステータス値

エラーコード	説明
0x00000001	正常終了時

#### TagRef 構造体

```
public struct TagRef
{
    public string Name;           //名称
    public ushort Size;          //サイズ
    public ushort Type;          //データの型
    public string Address;       //アドレス指定
}
```

#### TagRef 構造体 フィールド

型	シンボル	内容
string	Name	名称
ushort	Size	データのサイズ
ushort	Type	データの型
string	Adress	アドレス指定

**TagRef 構造体リスト**

```
//xml記載済みの参照Tagリスト
```

```
public static List<TagRef> tagRef_IN_List = new List<TagRef>();  
public static List<TagRef> tagRef_OUT_List = new List<TagRef>();  
public static List<TagRef> tagRef_CNTL_List = new List<TagRef>();
```

**TagRef 構造体 フィールド**

シンボル	内容
tagRef_IN_List	入力タグに指定したタグのリスト
tagRef_OUT_List	出力タグに指定したタグのリスト
tagRef_CNTL_List	サービスインジケータータグのリスト

## 11. トラブルシューティング

### 11.1. EgStart 起動できません。

EgStart.exe 起動時に以下のメッセージが表示され、起動に失敗します。

「Failed to initialize. Please check the environment」というメッセージが表示されます。

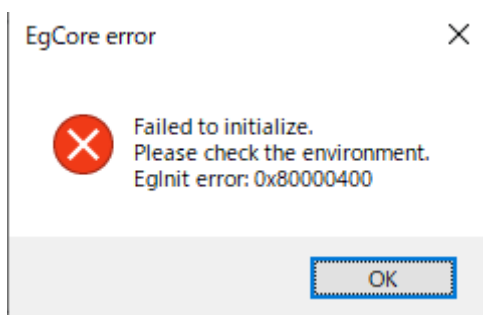


図 22. 起動時のエラー

#### ■ 原因 : INtime カーネルが起動していません。

WEB サーバーコンテナは INtime カーネル上で動作します。そのため事前に INtime カーネルを起動しておく必要があります。Windows タスクトレイ上の INtime Status Monitor が起動状態になっていることを確認します。



図 23. INtime カーネル停止中



図 24. INtime カーネル起動中

#### ■ 対応 : INtime カーネルを起動させます

以下の手順で起動を確認させます。

- 1) Windows タスクトレイ上の、INtime Status Monitor を右クリックします。
- 2) 表示されるメニュー上から NodeA を開始させます。



上記手順にて INtime カーネルが起動しない場合は、ライセンスアクティベート状態や INtime に割り当てるメモリの設定などを確認ください。また INtime カーネル起動時のトラブルについては、マイクロネット HP にも紹介しておりますので参照ください。

デバック時 IIS Express の 64 ビット設定の設定漏れ。

WebSample のデバック実行時に以下のメッセージが表示され、起動に失敗します。

「間違ったフォーマットのプログラムを読み込もうとしました」というメッセージが表示されます。

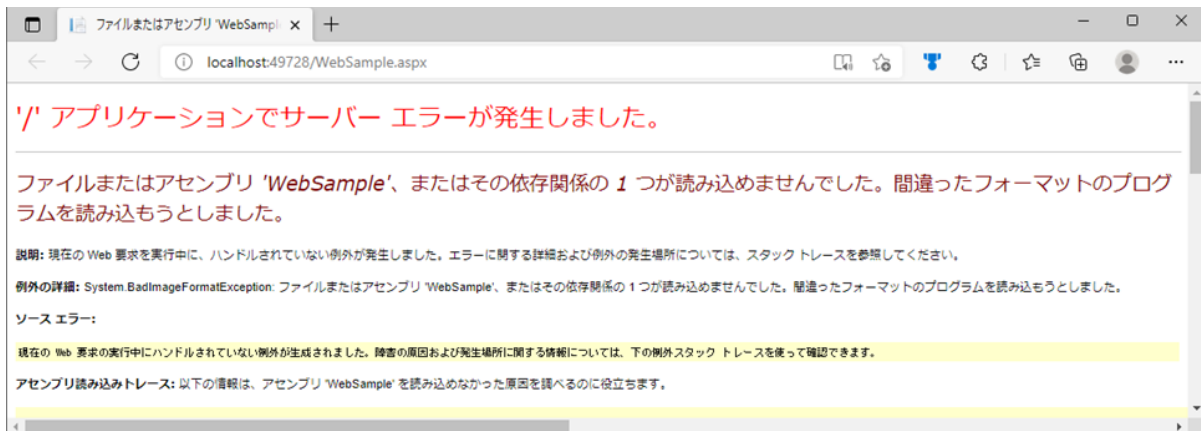


図 25. 起動時のエラー

■ 原因 : デバック時 IIS Express の 64 ビット設定がされていません。

■ 対応 : 設定します。

本マニュアルの「5.1. デバック時 IIS Express の 64 ビット設定」を参照してください。

## 11.2. スタートページの設定漏れ。

WebSample のデバック実行時に以下のメッセージが表示され、起動に失敗します。

「HTTP Error 403.14 - Forbidden」というメッセージが表示されます。

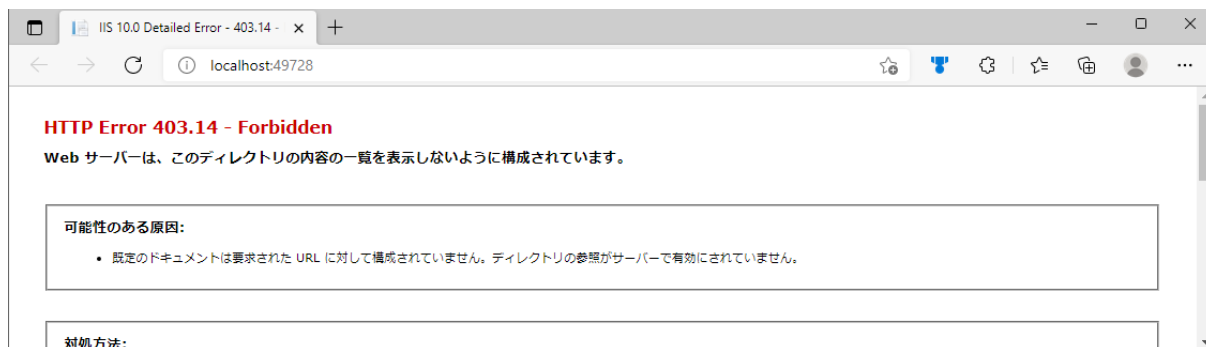


図 26. 起動時のエラー

■ 原因：スタートページの設定がされていません。

■ 対応：設定します。

本マニュアルの「5.2. スタートページの設定」を参照してください。



## 更新履歴

版	日付	更新説明
1	2022.09	初回版

REALTIME SERVICE for Windows

# WEB サーバーコンテナ 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)299-90-8557

- ・ 本書の著作権は、マイクロネットに帰属します。
- ・ 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。