

CoAP プロバイダ

CoAP(Constrained Application Protocol)

Version 1.0.0

ユーザーズ ガイド

May 15, 2015

【備考】

【改版履歴】

バージョン	日付	内容
1.0.0	2015-05-15	初版.

【対応機器】

機種	バージョン	注意事項

目次

1. はじめに	4
2. プロバイダの概要	5
2.1. 概要	5
2.2. モード	5
2.2.1. サーバモード	5
2.2.2. クライアントモード	5
2.3. メソッド・プロパティ	6
2.3.1. CaoWorkspace::AddController メソッド	6
2.3.2. CaoController::AddCommand メソッド	8
2.3.3. CaoController::AddVariable メソッド	8
2.3.4. CaoController::Execute メソッド	8
2.3.5. CaoController::OnMessage イベント	9
2.3.6. CaoCommand::get_Parameters プロパティ	9
2.3.7. CaoCommand::put_Parameters プロパティ	9
2.3.8. CaoCommand::get_Result プロパティ	9
2.3.9. CaoCommand::Execute メソッド	9
2.3.10. CaoVariable::get_Value プロパティ	9
2.3.11. CaoVariable::put_Value プロパティ	9
2.3.12. CaoMessage::get_Number プロパティ	9
2.3.13. CaoMessage::get_Value プロパティ	9
2.3.14. CaoMessage::get_Destination プロパティ	10
2.3.15. CaoMessage::Reply メソッド	10
3. コマンドリファレンス	11
3.1. コマンド一覧	11
3.2. コマンド詳細	11
付録 A. Option と Payload の指定	13
付録 B. CoAP パラメーター一覧	13

1. はじめに

本書は, CoAP(Constrained Application Protocol)で通信を行う CAO プロバイダのユーザーズガイドです.
本書で扱う CAO プロバイダ(CaoProvCoAP.dll)を CoAP プロバイダと呼びます.

次章に CoAP プロバイダの概要, 3 章にコマンドリファレンスを記載しています.

2. プロバイダの概要

2.1. 概要

CoAP プロバイダは, Ethernet の UDP 通信により CoAP で通信を行う CAO プロバイダです. そのファイル形式は DLL(Dynamic Link Library)であり, CAO エンジンから使用時に動的にロードされます. CoAP プロバイダを使用するにあたっては ORiN2SDK をインストールするか, 下表を参照して手作業でレジストリ登録を行う必要があります.

表 2-1 CoAP プロバイダ

ファイル名	CaoProvCoAP.dll
ProgID	CaoProv.IETF.CoAP
レジストリ登録	regsvr32 CaoProvCoAP.dll
レジストリ登録の抹消	regsvr32 /u CaoProvCoAP.dll

2.2. モード

CoAP プロバイダでは動作モードとしてサーバモードとクライアントモード, 通信モードとして同期モードと非同期モードがあり, 以下に示す 3 つのモードが選択できます.

2.2.1. サーバモード

サーバモードは, 通信相手からのデータ受信を契機に動作するモードです. 通信相手に主体的にデータを送信することはできません. サーバモードでは, 通信モードは非同期モードに固定されます.

サーバモードでは, 通信相手からの受信があったときに `OnMessage` イベントを発生させます.

データ受信時の `OnMessage` イベントで取得した `CaoMessage` オブジェクトの `CaoMessage::Reply()` で応答を送信します.

2.2.2. クライアントモード

クライアントモードは, 通信相手に主体的にデータを送信するモードです. 通信相手からの応答を期待するかどうかで同期モードと非同期モードを選択できます.

•同期モード

データ送信後, 通信相手からの応答を待つモードです. データの送信は `CaoCommand::Execute()`, `CaoVariable::get_Value()`または `CaoVariable::put_Value()`コマンドで行います.

•非同期モード

データ送信後, 通信相手からの応答を待たないモードです. データの送信は `CaoController::Execute()`コマンドで行います. 通信相手から応答があったとしても無視されます.

2.3. メソッド・プロパティ

2.3.1. CaoWorkspace::AddController メソッド

CoAP プロバイダでは AddController 時に、通信用の接続パラメータを参照し、通信の接続を行います。このときオプションで通信形態、接続パラメータ、タイムアウトの設定を指定します。

書式 AddController(<bstrCtrlName:BSTR>,<bstrProvName:BSTR>,
<bstrPcName:BSTR > [,<bstrOption:BSTR>])

bstrCtrlName : [in] コントローラ名
 bstrProvName : [in] プロバイダ名. 固定値 =” CaoProv.IETF.CoAP”.
 bstrPcName : [in] プロバイダの実行マシン名
 bstrOption : [in] オプション文字列

以下にオプション文字列に指定するリストを示します。

表 2-2 CaoWorkspace::AddController のオプション文字列

オプション	説明
Conn=<接続パラメータ>	必須. 接続パラメータを設定します. 詳細は 2.3.1.1 に示します.
Timeout=<タイムアウト時間>	送受信時のタイムアウト時間. (デフォルト:5000 msec)
CoAPClient [=TRUE / FALSE]	動作モードを設定します. TRUE の場合はクライアントモード, FALSE の場合はサーバモードで動作します. (デフォルト:FALSE)
Sync [=TRUE / FALSE]	通信モードを設定します. TRUE の場合は同期モード, FALSE の場合は非同期モードで動作します. 動作モードがサーバモードの場合は, 非同期モードに固定されます. (デフォルト:FALSE)
PacketOpt [=<パケットパラメータ>]	通信パケットの設定をします. (デフォルト:“0:0:0”) (参照 2.3.1.2)
SendRetry=<リトライ回数>	送信時のリトライ回数. 0~7(デフォルト:5) 0 以下の場合, 0 として扱われます. 7 以上の場合, 7 として扱われます.
RecvRetry=<リトライ回数>	受信時のリトライ回数. 0~7(デフォルト:5) 0 以下の場合, 0 として扱われます. 7 以上の場合, 7 として扱われます.

2.3.1.1. Conn オプション

以下に Conn オプションの接続パラメータ文字列を示します。ここで角括弧("[]")内は省略可能を示します。また、各パラメータの解説中の下線部はオプションを指定しなかったときのデフォルト値になります。

•サーバモード

“udp[:<Source Address>[:<Source Port>]]”

- <Source Address> : 複数の NIC を使う場合にこのオプションで IP アドレスを指定して NIC を選択することができます。省略した場合は、自動的に選択されます。ローカルマシンに割り当てられていない IP アドレスを指定したときはエラーを返します。
(デフォルト:255.255.255.255)
- <Source Port> : サーバポート番号。
(デフォルト:5683)

•クライアントモード

“udp[:<Destination Address>[:<Destination Port>[:<Source Address>[:<Source Port>]]]]”

- <Destination Address> : サーバ IP アドレス。
(デフォルト:127.0.0.1)
- <Destination Port> : サーバポート番号。
(デフォルト:5683)
- <Source Address> : 複数の NIC を使う場合にこのオプションで IP アドレスを指定して NIC を選択することができます。省略した場合は、自動的に選択されます。ローカルマシンに割り当てられていない IP アドレスを指定したときはエラーを返します。
(デフォルト:255.255.255.255)
- <Source Port> : クライアントポート番号。
(デフォルト:0)

2.3.1.2. PacketOpt オプション

以下に PacketOpt オプションのパラメータ文字列を示します。ここで角括弧("[]")内は省略可能を示します。また、各パラメータの解説中の下線部はオプションを指定しなかったときのデフォルト値を示します。

“PacketOpt =[<Convert>[:<URIEncode>[:<Space>]]]”

- <Convert> : Payload データ変換。
0:b-CAP モード
- <URIEncode> : URI に含まれる空白文字を<Space>で指定した文字に変換。
1 ビット目:URI パスの空白文字を変換。
2 ビット目:URI クエリの空白文字を変換。

3 ビット目: URI ホストの空白文字を変換
(デフォルト: 0)

<Space> : URI に含まれる空白文字を変換する文字を指定.
0: 空白文字のまま
1: ”+”に変換

2.3.2. CaoController::AddCommand メソッド

AddCommand メソッドを呼び出すと CaoCommand オブジェクトが取得できます。CaoCommand オブジェクトは同期モードのみで取得可能です。CaoController クラスの AddCommand メソッドの引数は、コマンド名(BSTR 型)を指定します。ここで指定する"コマンド名"は任意の文字列で特に何の制限もありません。例えば、AddCommand ("Cmd1")を指定することができます。

書式 AddCommand(<bstrName:BSTR> [,<bstrOption:BSTR>])

bstrName : [in] コマンド名
bstrOption : [in] オプション文字列

2.3.3. CaoController::AddVariable メソッド

AddVariable メソッドを呼び出すと CaoVariable オブジェクトが取得できます。CaoVariable オブジェクトは同期モードのみで取得可能です。CaoController クラスの AddVariable メソッドの引数は、変数名(BSTR 型)を指定します。ここで指定する"変数名"は CoAP の URI パスとして使用されます。例えば、AddVariable ("Var1")を指定すると、通信相手に URI パスとして”/Var1”が通知されます。

書式 AddVariable(<bstrName:BSTR> [,<bstrOption:BSTR>])

bstrName : [in] 変数名
bstrOption : [in] オプション文字列

2.3.4. CaoController::Execute メソッド

コマンドを実行します。CaoController::Execute メソッドは非同期モードでのみ実行可能です。

Execute メソッドの引数は、コマンドを BSTR、パラメータを VARIANT 配列で指定します。

書式 [<vntRet:VARIANT> =] Execute(<bstrCmd:BSTR> [,<vntParam:VARIANT>])

bstrCmd : [in] コマンド
vntParam : [in] パラメータ
vntRet [out] 戻り値

コマンドの実行に必要なパラメータおよび取得する結果は 3 章を参照してください。

2.3.5. CaoController::OnMessage イベント

同期モードのときはこのイベントは発生しません。

サーバモードの CoAP プロバイダが通信相手からデータを受信すると、CaoController クラスの OnMessage イベントが発生します。

2.3.6. CaoCommand::get_Parameters プロパティ

CaoCommand オブジェクトに現在設定されているパラメータを取得します。

2.3.7. CaoCommand::put_Parameters プロパティ

CaoCommand オブジェクトにパラメータを設定します。

パラメータは CoAP の Option と Payload に対応します。

パラメータの指定方法は「付録 A Option と Payload の指定」を参照してください。

2.3.8. CaoCommand::get_Result プロパティ

直前の CaoCommand::Execute メソッドの実行結果を取得します。

2.3.9. CaoCommand::Execute メソッド

コマンドを実行します。

Execute メソッドの引数は、コマンドを LONG で指定します。コマンドは CoAP の Code に対応します。送信可能な CoAP の Code 一覧は表 3-2 を参照してください。

書式 Execute(<IMode:LONG>)

IMode : [in] コマンド

2.3.10. CaoVariable::get_Value プロパティ

通信相手に CoAP の Code("Get")要求を発行します。

2.3.11. CaoVariable::put_Value プロパティ

通信相手に CoAP の Code("Put")要求を発行します。

2.3.12. CaoMessage::get_Number プロパティ

通信相手から受信した CoAP の Type と Code に対応します。上位 4 バイトが Type, 下位 4 バイトが Code を表します。

2.3.13. CaoMessage::get_Value プロパティ

通信相手から受信した CoAP の Payload に対応します。

2.3.14. GaoMessage::get_Destination プロパティ

通信相手から受信した CoAP の Option に対応します。

2.3.15. GaoMessage::Reply メソッド

通信相手に OnMessage イベントの処理結果を通知します。

書式 Reply(< vntData:VARIANT>)

vntData : [in] 応答データ

応答データは以下の形式で指定します。

vntData : 応答データ(VT_VARIANT | VT_ARRAY:1 または 2 要素)

[0]実行結果(VT_I4)

イベント処理が正常終了した場合は 0 を, 異常終了した場合は対応する HRESULT を指定.

[1]戻り値(VT_VARIANT)

イベント処理の結果, 通信相手へデータを送信する必要がある場合指定. 必要ない場合は省略可能.

3. コマンドリファレンス

3.1. コマンド一覧

表 3-1 CaoController::Execute メソッドのコマンド一覧

カテゴリ	コマンド	機能	
	Get	指定したリソースを取得要求する	P. 11
	Post	指定したリソースを処理要求する	P. 11
	Put	指定したリソースを設定要求する	P. 12
	Delete	指定したリソースを削除要求する	P. 12

3.2. コマンド詳細

Get

構文

object.Get <Data>

引数

<Data> = VT_VARIANT

CoAP の Option と Payload に対応します。パラメータの指定方法は「付録 A Option と Payload の指定」を参照してください。

戻り値

なし

説明

指定したリソースの取得要求を発行します。

Post

構文

object.Post <Data>

引数

<Data> = VT_VARIANT

CoAP の Option と Payload に対応します。パラメータの指定方法は「付録 A Option と Payload の指定」を参照してください。

戻り値

なし

説明

指定したリソースの処理要求を発行します。

Put

構文 `object.Put <Data>`

引数 <Data> = VT_VARIANT

CoAP の Option と Payload に対応します。パラメータの指定方法は「付録 A Option と Payload の指定」を参照してください。

戻り値 なし

説明 指定したリソースの設定要求を発行します。

Delete

構文 `object.Delete <Data>`

引数 <Data> = VT_VARIANT

CoAP の Option と Payload に対応します。パラメータの指定方法は「付録 A Option と Payload の指定」を参照してください。

戻り値 なし

説明 指定したリソースの削除要求を発行します。

付録A. Option と Payload の指定

CoAP プロバイダは、CoAP の Option と Payload を 1 つの VARIANT 型で指定する方法を以下のフォーマットで規定しています。

Data(VARIANT)

- | | |
|----------------------------------|--|
| 1. Option を 1 つだけ指定 | (VT_VARIANT VT_ARRAY:2 要素) |
| | [0] Option の値 |
| | [1] Option のデータ
(表 3-2 参照) |
| 2. Option を 1 つだけ指定
+ Payload | (VT_VARIANT VT_ARRAY:3 要素) |
| | [0] Option の値 |
| | [1] Option のデータ |
| | [2] Payload (VT_VARIANT) |
| 3. Option を複数指定 | (VT_VARIANT VT_ARRAY:n 要素) |
| | [0..n] フォーマット 1. (VT_VARIANT VT_ARRAY) |
| 4. Option を複数指定
+ Payload | (VT_VARIANT VT_ARRAY:2 要素) |
| | [0] フォーマット 3. (VT_VARIANT VT_ARRAY) |
| | [1] Payload (VT_VARIANT) |

付録B. CoAP パラメーター一覧

ここでは CoAP でよく使用するパラメータを示します。詳細は <http://coap.technology/> を参照してください。

表 3-2 CoAP パラメーター一覧

パラメータ	名称	値	説明
Code	GET	1	指定したリソースを取得要求する。
	POST	2	指定したリソースを処理要求する。
	PUT	3	指定したリソースを設定要求する。
	DELETE	4	指定したリソースを削除要求する。
Option	Uri-Host	3	URI のホストを指定する。(VT_BSTR)
	Uri-Port	7	URI のポート番号を指定する。(VT_BSTR)
	Uri-Path	11	URI のパスを指定する。(VT_BSTR)
	Content-Format	12	コンテンツの形式を指定する。(VT_UI4)
	Uri-Query	15	URI のクエリを指定する。(VT_BSTR)