

CaoUPnP
ユーザーズ ガイド
Version 1.0
February 24, 2006

【備考】

【改版履歴】

日付	版数	内容
2006-02-24	1.0	初版作成.

目次

1. はじめに	5
2. UPnP セットアップ	6
2.1. UPnP コンポーネントのインストール	6
3. CaoUPnP の概要	8
3.1. 概要	8
3.2. UPnP APIs	9
3.2.1. QueryStateVariable API	9
3.2.2. InvokeAction API	10
3.3. サービス XML	10
3.3.1. serviceStateTable	10
3.3.2. actionList	11
4. CaoUPnPConfig	12
4.1. 概要	12
4.2. 操作方法	12
4.2.1. タブ入力	12
4.2.1.1. エンジンタブ	13
4.2.1.2. コントローラタブ	15
4.2.1.3. アイテムタブ	16
4.2.2. メニュー	19
4.2.2.1. ファイルメニュー	19
4.2.2.2. 編集メニュー	20
4.2.2.3. アクションメニュー	21
4.2.2.4. ヘルプメニュー	23
4.3. レジストリ構成情報	23
5. CaoUPnP チュートリアル	24
5.1. クライアントアプリケーションの作成	24
5.2. CaoUPnPConfig の設定	27
5.3. サンプルの実行	29
6. 注意事項	34

7. 参考情報	35
7.1. クライアントサンプル	35
7.2. UPnP 関連リンク	35

1. はじめに

本書は CAO(Controller Access Object)のクライアントアプリケーションとして位置付けられる CaoUPnP のユーザーズガイドです。

CaoUPnP は CAO モジュールを 1 つの UPnP(Universal Plug and Play)機器であるとみなし、UPnP デバイスの機能を提供します。(図 1-1) CAO は様々な FA 機器との通信を可能にするモジュールなので、その上のアプリケーションである CaoUPnP も当然様々な FA 機器に接続して使うことができます。

UPnP とは、Microsoft によって提唱されている、複雑な設定などを行わなくてもデバイスをネットワーク上に繋げるだけで、シームレスに接続することのできる技術です。

CaoUPnP は UPnP 機器として動作するモジュールなので、ネットワークに接続するだけで、クライアントアプリケーションから利用することが可能となります。これにより、CAO プロバイダへのシームレスな接続が可能になります。

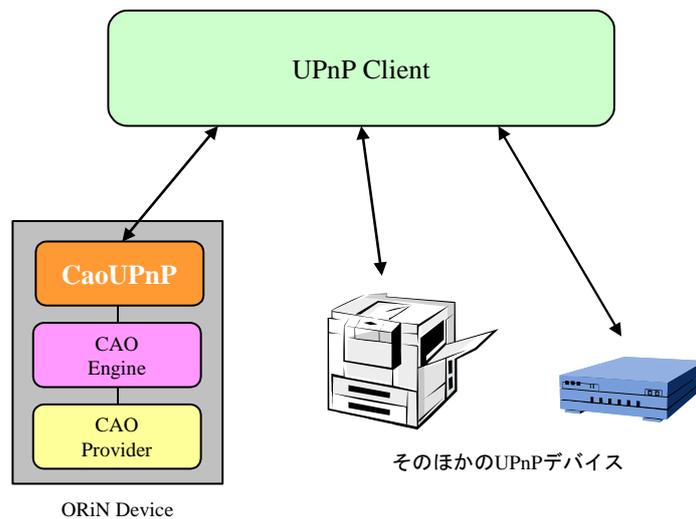


図 1-1 CaoUPnP のイメージ

2. UPnP セットアップ

CaoUPnP を UPnP デバイスとして動作させるためには、UPnP コンポーネントをインストールする必要があります。CaoUPnP は、Windows XP に対応していますので、Windows XP における UPnP コンポーネントのセットアップ方法を解説します。

2.1. UPnP コンポーネントのインストール

WindowsXP では、デフォルトで UPnP がインストールされていないので、UPnP のコンポーネントをインストールしてください。

- (1) 「コントロールパネル」から「プログラムの追加と削除」を開き、「Windows コンポーネントの追加と削除」を選びます。

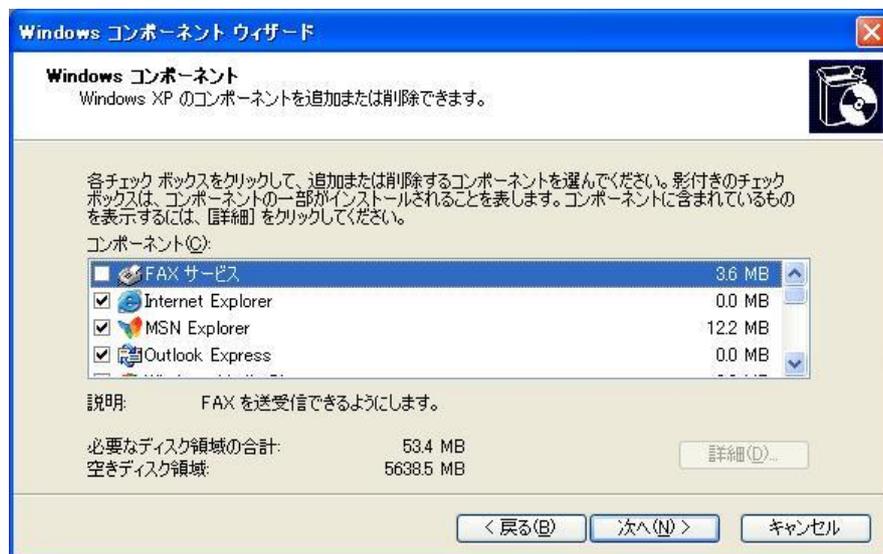


図 2-1 Windows コンポーネントウィザード

- (2) 「ネットワークサービス」を選び、「詳細」ボタンを押します。



図 2-2 ネットワークサービスの追加

- (3) 「ユニバーサル プラグ アンド プレイ」にチェックをつけ、「OK」ボタンを押下します。



図 2-3 ユニバーサルプラグアンドプレイコンポーネントの追加

- (4) 「次へ」ボタンを押下すると、インストールが開始されます。
(5) 「完了」ボタンを押下して完了です。インストーラの指示に従って再起動をおこなってください。

※サービスパック未適用の WindowsXP には、UPnP コンポーネントに[セキュリティホール](#)がありますので、Windows Update を行い修正しておく事を強くお勧めいたします。

3. CaoUPnP の概要

3.1. 概要

CaoUPnP の構成を下図に示します。CaoUPnP は UPnP クライアントと CAO を接続するミドルウェアであり、UPnP クライアントからの要求を CAO の機能を使って処理する UPnP デバイスです。CaoUPnP は1つの独立したプロセス(アウトプロセスサーバ)で、複数の UPnP クライアントに対してサービスを提供します。

CaoSQLConfig with UPnP Option(以下、CaoUPnPConfig)は、CaoSQLConfig を利用し CAO の CaoController オブジェクト、CaoVariable オブジェクトを作成するのに必要なパラメータを設定したり、CaoUPnP の動作オプションを設定するためのツールです。このツールで設定された内容は、レジストリと csq ファイルに保存され、CaoUPnP の起動時に参照します。

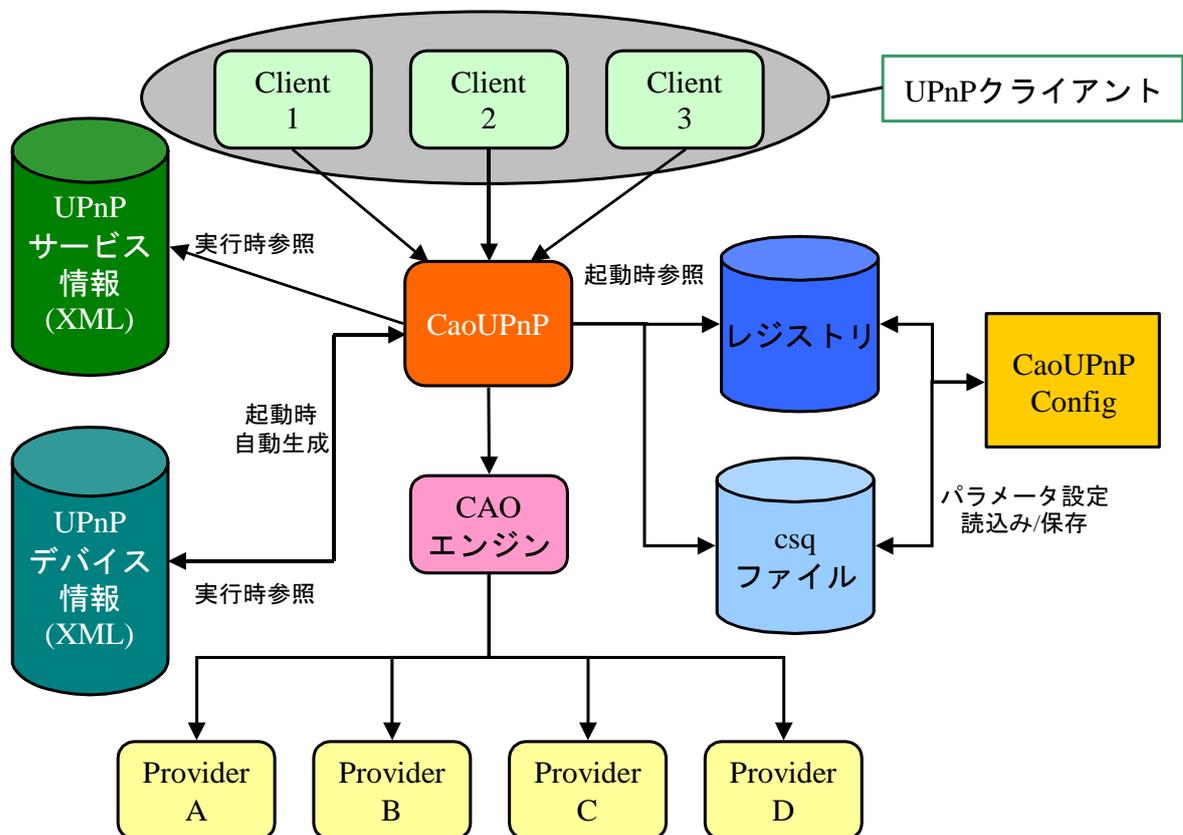


図 3-1 CaoUPnP の動作概要

CaoUPnP を利用したシステムの動作を以下に説明します。

- (1) まず、UPnP クライアントアプリケーションは、SSDP(Simple Service Discovery Protocol)というプロトコルを用いて UPnP デバイスの検索を行います。SSDP では、クライアントアプリケーションからマルチキャストで検索条件を送信し、検索条件にマッチした UPnP 機器(CaoUPnP を含む)が、自身の IP アドレスやポート番号をユニキャストで返信します。
- (2) 次にクライアントアプリケーションは、UPnP 機器のデバイス情報や、サービス情報を XML 形式で取得します。CaoUPnP の場合、これらの情報は CaoUPnPDevice.xml と CaoUPnPService.xml に記述されています。(CaoUPnPService.xml は静的に存在しますが、CaoUPnPDevice.xml は CaoUPnP 起動時に自動的に作成されます)
- (3) クライアントアプリケーションは、取得した情報をもとに、UPnP 機器の提供するサービスを利用します。サービスの実行には、UPnP APIs の Control Point API が利用されます。この API では、SOAP(Simple Object Access Protocol)を利用して、UPnP デバイスのサービスを実行します。
- (4) CaoUPnP では、実行されたサービスに応じて、CAO プロバイダから変数を取得や設定を行います。

3.2. UPnP APIs

クライアントアプリケーションから、UPnP のサービスを実行するには、UPnP APIs を利用します。

クライアントアプリケーションから、IUPnPService::QueryStateVariable, IUPnPService::InvokeAction の 2 つの API を利用することで、CaoUPnP の提供しているサービスを利用することができます。

3.2.1. QueryStateVariable API

QueryStateVariable は UPnP デバイスの指定したステータス変数を取得するための API です。

API の定義は以下のようになります。

[Visual Basic]

```
VARIANT QueryStateVariable(  
    ByVal bstrVariableName as String  
)
```

[Visual C++]

```
HRESULT QueryStateVariable(  
    BSTR bstrVariableName,  
    VARIANT* pvarValue  
);
```

この関数では、引数の“bstrVariableName”に取得したいステータス変数の名前を指定することで、そのステータス変数の値を取得することができます。取得した値は、Visual Basic の場合、関数の戻り値として、Visual C++の場合、引数“pvarValue”に格納されます。

3.2.2. InvokeAction API

InvokeAction は、UPnP デバイスの指定した動作を行うための API です。

API の定義は以下のようになります。

[Visual Basic]

```
VARIANT InvokeAction(
    ByVal bstrActionName as String,
    ByVal vInActionArgs as VARIANT,
    ByRef pvOutActionArgs as VARIANT
)
```

[Visual C++]

```
HRESULT InvokeAction(
    BSTR bstrActionName,
    VARIANT varInActionArgs,
    VARIANT* pvarOutActionArgs,
    VARIANT* pvarRetVal
);
```

この関数では、“bstrActionName”に実行したいメソッド名、“varInActionArgs”にメソッドの引数を与えて実行すると、指定したメソッドが実行されます。

3.3. サービス XML

UPnP クライアントアプリケーションは、サービス XML ファイルから UPnP デバイスが提供しているサービスを知ることができます。

CaoUPnP では、CaoUPnPService.xml ファイルにサービスが記述されています。

サービス XML ファイルには、<serviceStateTable>タグと<actionList>タグが記述されています。

3.3.1. serviceStateTable

<serviceStateTable>タグには、QueryStateVariable API で取得することのできる変数を定義します。

表 3-1 serviceStateTable 一覧

変数名	型	説明
Name	String	コントローラ名
Description	String	コントローラの説明
ItemCount	i4(long)	アイテムの数
ItemNames	String	アイテムの名前一覧
Command	String	コマンド文字列
Result	String	Command に設定されたコマンド文

		字列をコントローラの Execute で実行した実行結果
ItemName	String	現在のアイテム名
ItemValue	String	現在のアイテムの値
ItemAttribute	i4(long)	現在のアイテムの属性
ItemDateTime	Date	現在のアイテムの取得時間
ItemDescription	String	現在のアイテムの説明
ItemAllowedValueMin	r4(float)	現在のアイテムの最小値
ItemAllowedValueMax	r4(float)	現在のアイテムの最大値
ItemAllowedValueStep	r4(float)	現在のアイテムのステップサイズ
ItemAllowedValueList	String	現在のアイテムのリスト
ItemDefaultValue	String	現在のアイテムのデフォルト値

3.3.2. actionList

<actionList>タグには、invokeAction API を用いて実行することのできるコマンドの定義を記述します。

表 3-2 actionList 一覧

コマンド名	引数	関連する変数名	説明
put_Command	[in] strCommand	Command	実行したいコマンド文字列の設定を行います。
put_ItemName	[in] strItemName	ItemName	値を取得するアイテムを選択します。 引数には、CaoUPnPConfig で設定したアイテム名を指定してください。
put_ItemValue	[in] strItemValue	ItemValue	現在のアイテムに値を設定します。

※put_ItemValue で設定する引数は以下の書式で記述します。

<データ型>,<データ列>

この書式の詳細については、『[ORiN2 プログラミングガイド](#)』の「2.2.5.データの記述方式」を参照してください。

4. CaoUPnPConfig

4.1. 概要

CaoSQLConfig の引数に“UPnP”を付けて実行(“CaoSQLComfig.exe UPnP”)することで CaoUPnP 用 CaoSQLConfig(以下 CaoUPnPConfig)を起動します。CaoUPnPConfig は、の CaoUPnP で利用する CAO のコントローラオブジェクト、変数オブジェクトを作成する際に必要なパラメータを設定するツールです。ここで設定した情報はレジストリと csq ファイルに記録され、CaoUPnP の起動時に読み込まれます。

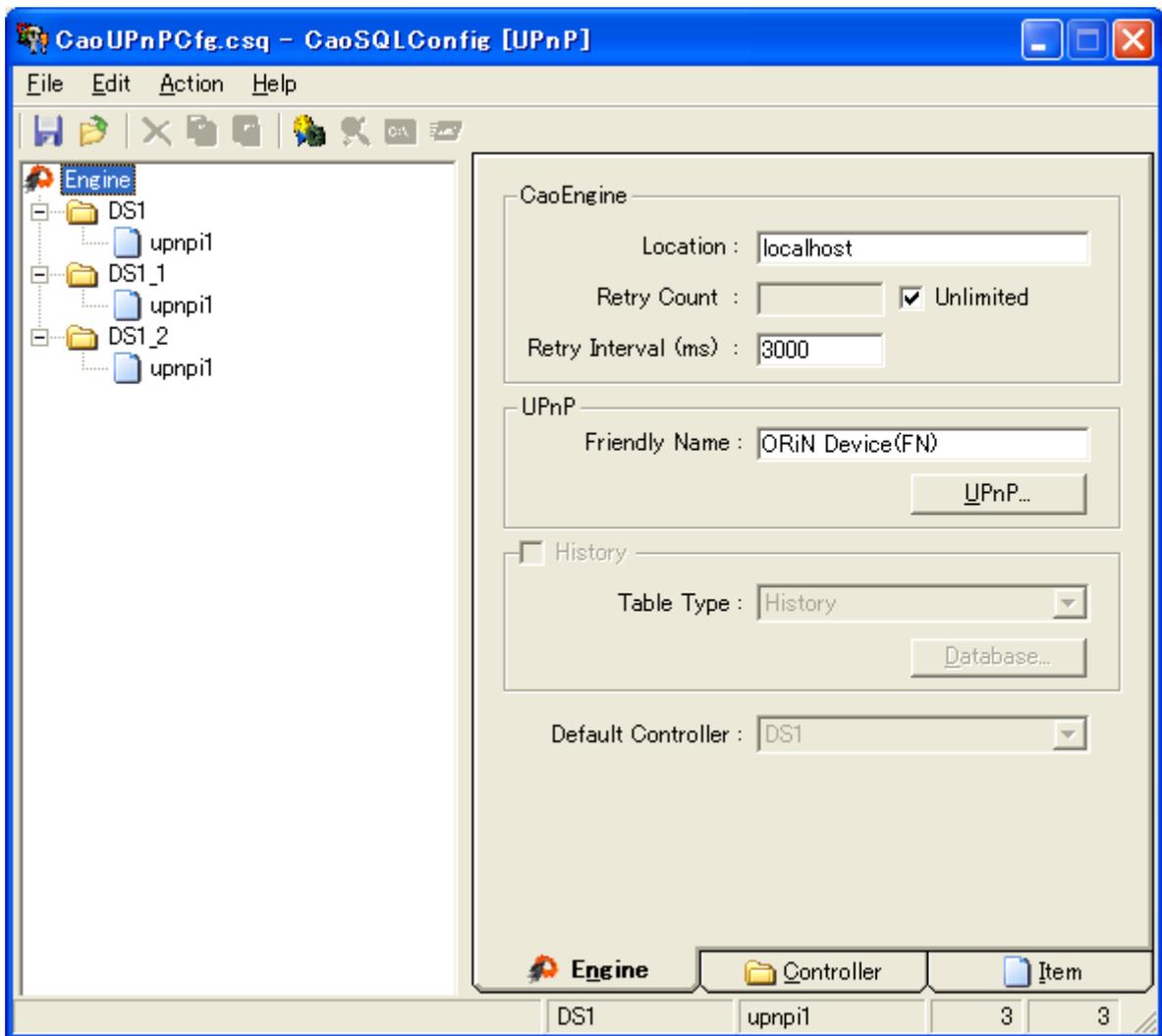


図 4-1 CaoUPnPConfig 画面

4.2. 操作方法

4.2.1. タブ入力

画面右のタブはツリービューのノードを選択状態にすることによってその情報を表示します。エンジンのノードを選択するとタブはCAOエンジンの情報を表示します。コントローラのノードを選択するとタブはそのコン

トローラの情報を表示します。アイテムのノードを選択するとタブはそのアイテムの情報を表示します。

4.2.1.1. エンジンタブ

エンジンタブでは図 4-2 に示した形式で情報を設定・表示します。CAO エンジンの情報は自由に設定できます。

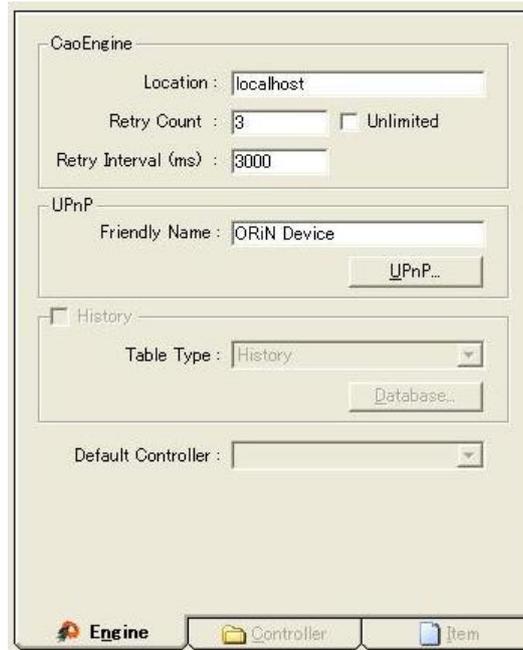


図 4-2 CaoUPnPConfig の起動

ここで設定可能な項目は以下の通りです。

[Cao エンジン起動マシン名] – Location

CaoOPC.exe から起動される Cao.exe の起動コンピュータ名を入力します。何も入力されていない場合はローカルで起動します。

[リトライ回数, 無限リトライ] – Retry Count / Unlimited

CaoController のオープンで失敗した場合リトライすることができます。リトライ回数は無制限をチェックする, 又は回数を入力します。

[リトライ間隔] – Retry Interval(ms)

リトライ間隔は一度失敗してから, リトライをするまでの間隔をミリ秒単位で入力します。

[UPnP デバイス名] – Friendly Name

CaoUPnP の UPnP デバイス名を設定します。UPnP クライアントアプリケーションからは, ここで設定した

名前で, CaoUPnP を判断します. **【入力必須項目】**

[UPnP デバイス情報] – UPnP

CaoUPnP の UPnP デバイスとしてのデバイス情報を確認することができます. この情報は, “CaoUPnPDevice.xml” に記述されます.

Tag	Value
DeviceType	urn:schemas-upnp-org:device:Basic:1
Manufacturer	ORiN Consortium
ManufacturerURL	http://www.orin.jp/
ModelDescription	CAO-UPnP Device
ModelName	CaoUPnP
ModelNumber	CUPP-1
ModelURL	http://www.orin.jp/
SerialNumber	1
UDN	uuid:12333
UPC	1
PresentationURL	sample.html

図 4-3 デバイス情報の表示

以下に各項目の説明を行います.

表 4-1 UPnP のデバイス情報

設定項目	説明
Device Type	UPnP のデバイスタイプを指定します. この項目は UPnP フォーラムかメーカーによって決定され, デバイスごとに固有の値が与えられます. 【変更不可項目】
Manufacturer	デバイスを製作したメーカー名を指定します. 【入力必須項目】
Manufacturer URL	メーカーのウェブサイトへの URL を指定します.
Model Description	デバイスの説明について記述します.
Model Name	デバイスのモデル名を指定します. 【入力必須項目】
Model Number	デバイスのモデル番号を指定します.
Model URL	デバイスを解説するためのウェブサイトへの URL を指定します.
Serial Number	デバイスのシリアルナンバーを指定します.

UDN	デバイスのメーカーが決定したユニークな ID を指定します。【入力必須項目】
UPC	統一商品コード(Universal Product Code)を指定します。
Presentation URL	デバイスのプレゼンテーション用の URL を指定します。プレゼンテーションページは、コントロールポイントによって、ブラウザにロードされ、デバイスの情報を見たり、制御したりすることができます。

4.2.1.2. コントローラタブ

コントローラタブでは図 4-4 に示した形式で情報を設定・表示します。コントローラの情報は自由に設定できますが、「CaoUPnP コントローラ名」だけは登録された情報の中でユニークな情報を持つことが要求されるため、メニューの名前変更でしか変更できません。またコントローラ名に関しては大文字と小文字の区別はされず、数値のみの名前は許可しません。また、記号の内、¥\$#:は使用できません。

図 4-4 コントローラタブの設定

ここで設定可能な項目は以下の通りです。

[コントローラ有効] – Enable

このコントローラの有効/無効設定です。無効に設定された場合は CAO-OPC 実行時に起動されません。

[コントローラ名] – Name

CaoUPnP でのコントローラ名です。UPnP クライアントからは、QueryStateVariable の引数に“Name”を設定し実行することで値が取得できます。

[コントローラ名, プロバイダ名, コンピュータ名, オプション]**– Controller Name, Provider Name, Machine, Option**

CaoController を Add する際に渡すパラメータです。入力した値がそのまま使用されます。

[詳細情報] – Description

ここは任意の文字列を格納できるので、そのコントローラの説明などを記述しておくとうりやすいでしょう。UPnP クライアントからは、QueryStateVariable の引数に“Description”を設定し実行することで値が取得できます。

4.2.1.3. アイテムタブ

アイテムタブでは図 4-5 に示すような情報を設定・表示する。コントローラタブと同様にアイテム名のみはメニューの名前変更でないと変更は行えません。またアイテム名に関しては大文字と小文字の区別はされず、数値のみの名前は許可しません。また、記号の内、¥\$#:は使用できません。

図 4-5 アイテムタブの設定

ここで設定できる項目は以下の通りです。

[アイテム有効] - Enabled

このアイテムを有効にするか、しないかの設定です。無効に設定された場合はこのアイテムの情報を使用しません。

[アイテム名] - Name

アイテム名を設定します。UPnP クライアントからは、QueryStateVariable の引数に“ItemName”を設定することで値が取得できます。

[変数名] - Variable Name

CaoVariable オブジェクトの取得(ICaoController::AddVariable)時に渡される引数です。入力された値はそのまま使われます。大文字・小文字は区別されません。

[オプション] - Option

変数に付加したいオプションを指定します。CaoVariable オブジェクトの取得(ICaoController::AddVariable)時の引数に使用されます。

[クラス] - Class

変数クラスの種類を選びます。

[システムオブジェクト名] - Object Name

変数の親オブジェクト名を入力します。但しコントローラの変数オブジェクトは、親コントローラのみ可能なので入力できません。

[詳細情報] - Description

アイテムに関する詳細な情報を設定できます。XML ファイルから登録されるので、不揮発な情報を設定するのに向いています。UPnP クライアントからは、QueryStateVariable の引数に“ItemDescription”を設定し実行することで値が取得できます。

[属性情報] - Attribute

アイテムの属性を設定できます。UPnP クライアントからは、QueryStateVariable の引数に“ItemAttribute”を設定し実行することで値が取得できます。

[Details の設定] - Details

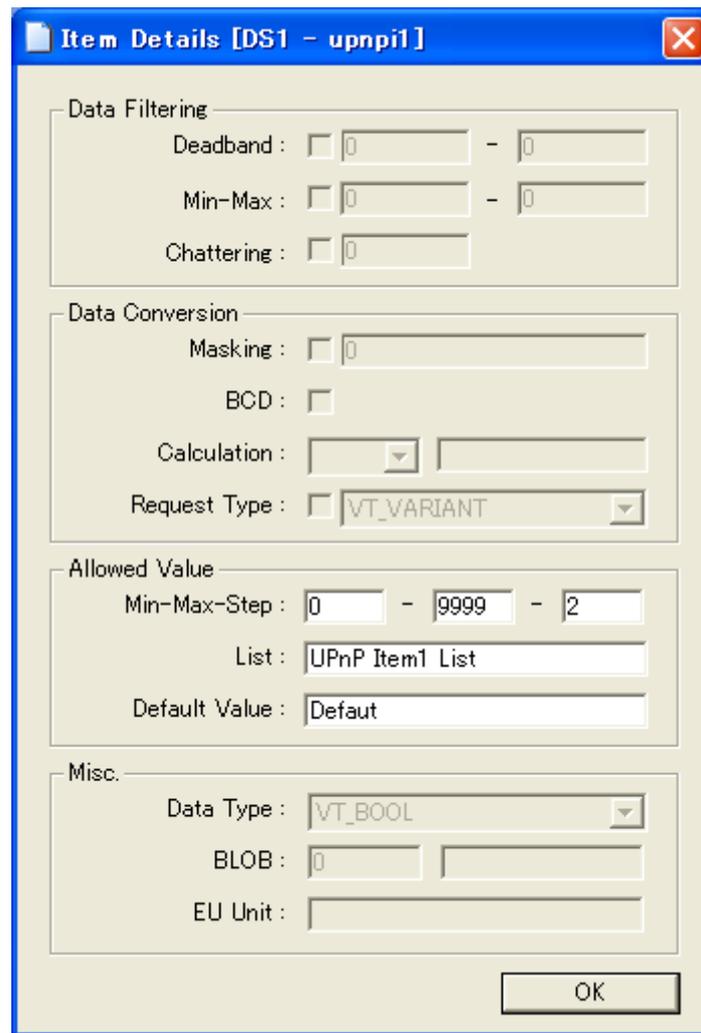


図 4-6 アイテムの詳細設定

[許容値] – Allowed Value**[最小・最大・刻み幅] – Min-Max-Step**

許容値の最小値, 最大値, 刻み幅を設定します。

UPnP クライアントからは, `QueryStateVariable` の引数にそれぞれ “`ItemAllowedValueMin`”, “`ItemAllowedValueMax`”, “`ItemAllowedValueStep`” を設定し実行することで値が取得できます。

[許容値リスト] – List

許容値リストを設定します。UPnP クライアントからは, `QueryStateVariable` の引数に “`ItemAllowedValueList`” を設定し実行することで値が取得できます。

[デフォルト値] – Default Value

許容値の初期値を設定します。UPnP クライアントからは, `QueryStateVariable` の引数に “`ItemDefaultValue`” を設定し実行することで値が取得できます。

4.2.2. メニュー

CaoUPnPConfig における基本的な操作はメニューバーから選択します。一部ツリービューを右クリックして選択できるメニューもありますが、メニューバーから選択できる機能と全く同等です。

4.2.2.1. ファイルメニュー

作成したデータのセーブやロード等の処理を行います。

Open, Save, Save As.. で使用されるファイルの拡張子は“**csq**”です。

Import, Export で使用されるファイルの拡張子は“**csx**”です。

[新規作成] – New

新規に csq ファイルを作成します。

[ファイルから読み込み] – Open...

指定した csq ファイルから設定情報を読み込みます¹。

読み込みをキャンセルした場合には読み込み結果は反映されません。

[ファイル保存] – Save

現在の設定内容を csq ファイルに上書き保存します。

[名前を付けて保存] – Save As...

現在の設定内容を csq ファイルに保存します。

[インポート] – Import...

csx ファイルからノードを選択しているノードに追加します。

[エクスポート] – Export...

選択しているノードを csx ファイルに出力します。

[CSV ファイルからインポート] – Import from CSV file...

csv ファイルからノードを選択しているノードに追加します。

[CSV ファイルにエクスポート] – Export to CSV file...

選択しているノードを csv ファイルに出力します。

¹ Win9x/Me では OS の制限から 64K 以上のサイズのファイルの読み込みの場合はデータに欠損が生じる場合があるので注意して下さい。

[終了] – Exit

CaoUPnPConfig のプログラムを終了します。

4.2.2.2. 編集メニュー

アイテムの追加削除などの編集処理を行います。

[コントローラ追加] – Add Controller

CaoUPnP で読み込むコントローラの新規追加を行います。既存のコントローラと同じ名前のコントローラは追加できません(大文字・小文字の区別をしません)。有効な名前が入力されるとツリーに追加され、その詳細情報は画面右のタブから入力します。

[アイテム追加] – Add Item

ツリービューで選択されたノードにアイテムを追加します。選択されたコントローラに既に登録されているアイテムと同じ名前のアイテムは追加できません(大文字・小文字の区別をしません)。有効な名前が入力されるとツリーに追加され、詳細情報は画面右のタブから入力します。

[名前変更] – Rename

ツリービューで選択されたコントローラ/アイテムの名前を変更します。すでに登録されているコントローラ/アイテムの名前に変更はできません(大文字・小文字の区別をしません)。

[削除] – Delete

ツリービューで選択されたノードを削除します。コントローラの場合はコントローラに登録されたアイテムを再帰的に全て削除します。ツリーでアイテムが選択された状態でキーボードから Del キーを押下することによっても同等の処理が行えます。

[ノードのコピー] – Copy

ツリーで選択されたノードをコピーします。コントローラの場合は登録されたアイテムもコピーされます。但し、トリガ情報はコピーされません。下記の“貼り付け”操作でコピーした情報のクローンをノードに追加できます。

[貼り付け] – Paste

コピーされた情報を新しい名前を入力してノード貼り付けます。

[ソート] – Sort

選択されているレベルのノードを昇順ソートします。

4.2.2.3. アクションメニュー

CaoUPnP.exe の実行や実行時の設定に関する設定を行います。設定はレジストリに保存されます。

[環境設定] – Settings

CaoUPnP の環境設定ダイアログが表示されます。

[全般タブ] – General

プロセスの優先度や使用言語など、CaoUPnP 全般に関する設定を行うことができます。

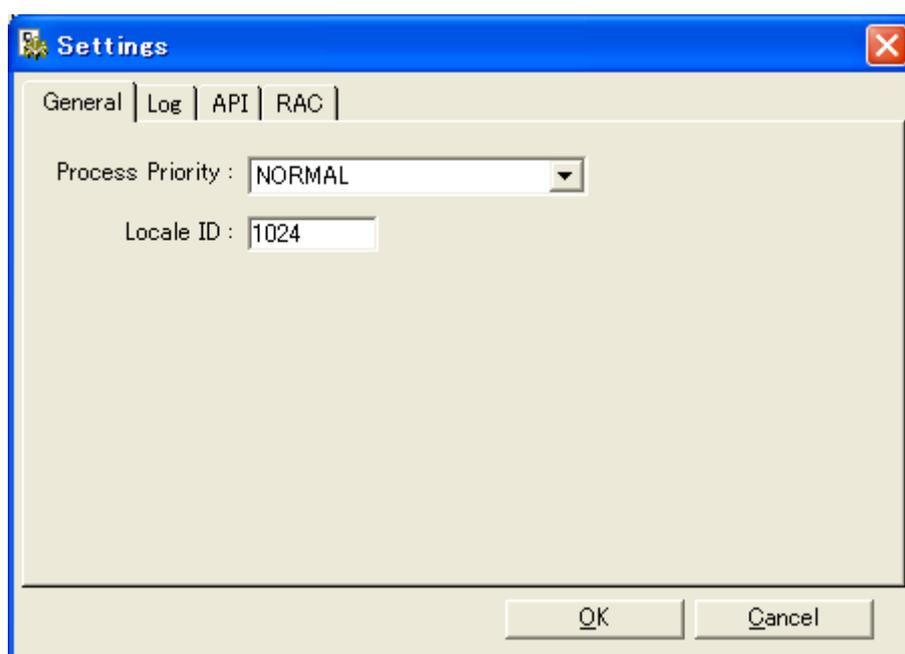


図 4-7 CaoUPnP 環境設定 (General)

[プロセス優先度] – Process Priority

CaoUPnP のプロセス優先度を設定します。優先度に対する調整は以下の通りです。

REAL TIME > HIGH > **NORMAL** > IDEL

[ロケール ID] – LocaleID

使用する言語 ID を設定します。

[ログ設定タブ] – Log

CaoUPnP のログ出力に関する設定を行うことができます。

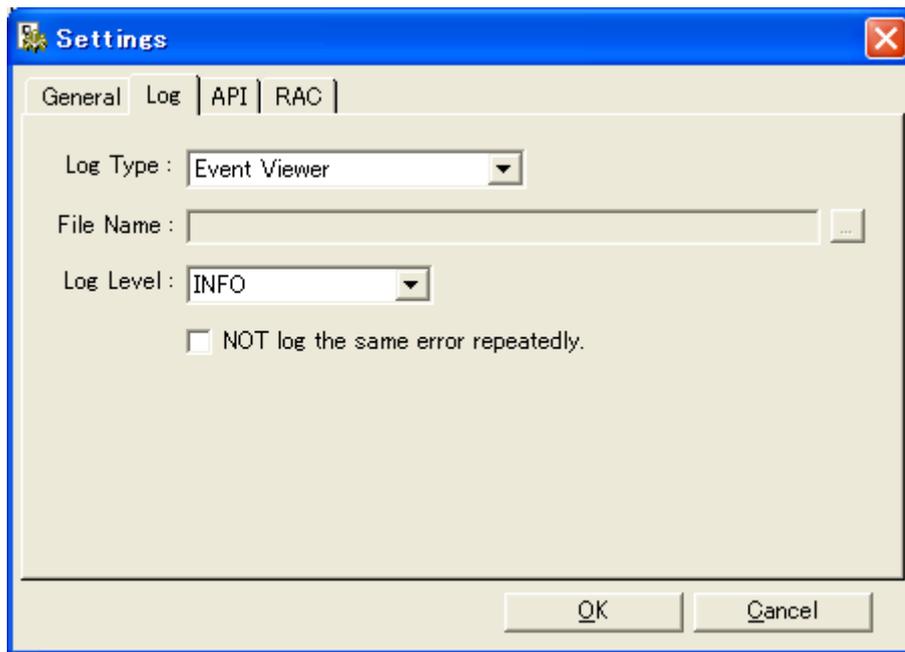


図 4-8 CaoUPnP 環境設定(Log)

[ログタイプ] – Log Type

CaoUPnP.exe のログの出力タイプを選択します。

ログの出力タイプは以下のものを選択することができます。

表 4-2 ログタイプ

出力先	備考
Console	コンソールに出力します
Message Box	メッセージボックスに出力します(サービス起動時)
Event Viewer	イベントビューワに出力します(サービス起動時)
Debug Viewer	デバッグ出力します。
Text File	指定したテキストファイルに出力します。

[ファイル名] – File Name

Log Type を Text File にした場合にログを出力するファイルパスを設定します。

[ログ出力レベル] – Log Level

ログの出力レベルを設定します。ログレベルの設定は、以下の 5 つのレベルから選択することができます。“FATAL”がもっとも深刻度が高く、“DEBUG”に近づくほど深刻度は低くなります。標準では

“INFO”に設定されています。

FATAL > ERROR > WARN > **INFO** > DEBUG

[同エラー出力抑制] – NOT log the same error repeatedly.

同じエラーの出力抑制設定を行います。チェックを入れると、同じエラーが何度発生しても1度しかエラーメッセージを出力しません。

[API 設定タブ] – API

設定できる項目はありません。

[RAC 機能設定タブ] – RAC

設定できる項目はありません。

[サービスの起動] – Service Start

CaoUPnP サービスを起動します。csq ファイル情報は CaoUPnP の起動時に読み込まれる為、csq ファイル情報を更新したい場合は CaoUPnP サービスを再起動する必要があります。

CaoUPnP.exe がサービスとして登録されていない場合や、既にサービスが実行されている場合は使用できません。

[サービスの停止] – Service Stop

CaoUPnP サービスを停止することができます。CaoUPnP.exe がサービスとして登録されていない場合や、既にサービスが停止されている場合は使用できません。

[サービスの再起動] – Service Restart

CaoUPnP サービスを再起動することができます。CaoUPnP.exe がサービスとして登録されていない場合は使用できません。

4.2.2.4. ヘルプメニュー

ヘルプやライセンス登録のメニューです。

[バージョン情報]

バージョン情報を表示します。

4.3. レジストリ構成情報

使用するレジストリは CaoSQLConfig と同じですので、詳しくは『[CaoSQL ユーザーズガイド](#)』の「4.3 レジストリ構成情報」を参照して下さい。

5. CaoUPnP チュートリアル

本章では、CaoUPnP を用いたシステムの構築方法について解説を行います。

ここでは、CaoUPnP にアクセスするための簡単な UPnP クライアントアプリケーションを作成し、CaoUPnPConfig で設定した DataStore プロバイダへのアクセスを行います。

5.1. クライアントアプリケーションの作成

ここでは Visual Basic 6.0 を用いてクライアントアプリケーションの作成を行います。

Visual C++6.0 などの他の環境を利用したい場合は、7 章を参照してください。

(1) プロジェクトの作成

まずは、Visual Basic6.0 を起動し、“標準 EXE”のプロジェクトを新規作成してください。

(2) 参照設定

クライアントアプリケーションから UPnP APIs を利用するためには、UPnP Type Library の参照を追加する必要があります。「プロジェクト」の「参照設定」から“UPnP 1.0 Type Library”を追加してください。



図 5-1 参照設定

(3) フォームの作成

次にクライアントアプリケーションのフォームを作成します。以下に示すような、テキストボックス、ボタン、コンボボックスを持つフォームを作成してください。

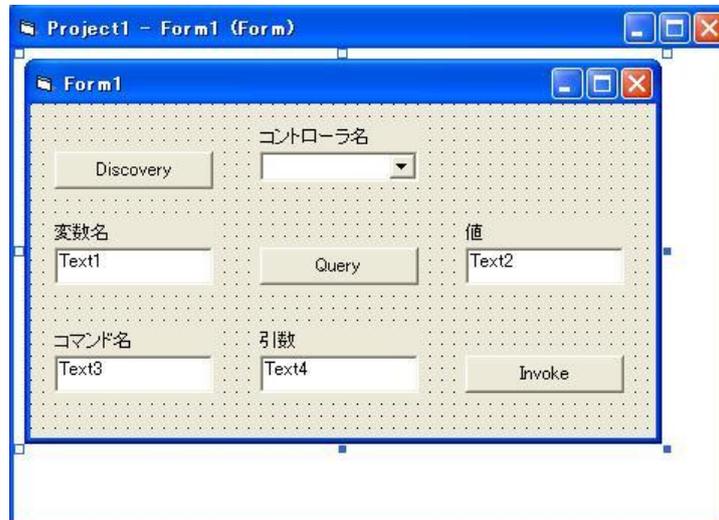


図 5-2 UPnP クライアントのフォーム

(4) コードの作成

以下のようなソースコードを記述してください。

List 5-1 **Form1.frm**

```
Private devices As UPnPDevices
Private services As UPnPServices

Private device As UPnPDevice
Private service As UPnPService

Private serviceTable() As UPnPService

Private devicefinder

Private Sub Form_Load()
    ' Device Finder を生成する ①
    Set devicefinder = CreateObject("UPnP.UPnPDeviceFinder.1")
End Sub

' ネットワーク上の UPnP デバイスを検索する
Private Sub cmdDiscovery_Click()
    cmdDiscovery.Enabled = False
    Combo1.Clear

    ' すべての UPnP デバイスを取得する ②
    Set devices = devicefinder.FindByType("upnp:rootdevice", FD_ANY)

    If devices.count = 0 Then
```

```

        MsgBox ("UPnP デバイスが見つかりませんでした。")
        GoTo Finish
    End If

```

```

' ORiN Device のすべてのサービスを取得する ③
For Each device In devices
    If StrComp(device.FriendlyName, "ORiN Device") = 0 Then
        MsgBox ("ORiN Device を発見しました。")
        Set services = device.services
        Exit For
    End If
Next

```

```

If services Is Nothing Then
    MsgBox "ORiN Deviceが見つかりませんでした。"
    GoTo Finish
End If

```

```

' ORiN デバイスのサービスを取得する ④
ReDim serviceTable(services.count)
Dim cnt As Integer
cnt = 0
For Each service In services
    Combo1.AddItem (service.QueryStateVariable("Name"))
    Set serviceTable(cnt) = service
    cnt = cnt + 1
Next

```

```

Finish:
    cmdDiscovery.Enabled = True
End Sub

```

```

' コンボボックスからサービスを選択する
Private Sub Combo1_Click()
    Set service = serviceTable(Combo1.ListIndex)
    cmdQuery.Enabled = True
    cmdInvoke.Enabled = True
End Sub

```

```

' 変数値を取得する ⑤
Private Sub cmdQuery_Click()
    Text2.Text = service.QueryStateVariable(Text1.Text)
End Sub

```

```

' コマンドを実行する ⑥
Private Sub cmdInvoke_Click()
    Dim actionArgs(1)
    Dim actionOutArgs As Variant
    Dim retVal
    ' コマンドの引数を設定する
    actionArgs(0) = Text4.Text
    ' コマンドを実行する
    retVal = service.InvokeAction(Text3.Text, actionArgs, actionOutArgs)
End Sub

```

以下にソースコードの説明を示します。

1. UPnPDeviceFinder をレイトバインディングで生成します。
2. UPnPDeviceFinder を用いて、UPnP デバイスの検索を行います。
ここでは、FindByType メソッドを用いて、タイプを指定して検索をおこなっています。今回指定したタイプは、“upnp:rootdevice”なので、ネットワーク上に存在するすべての UPnP デバイスを見つけます。
3. 発見した UPnPDevice の中から、Friendly Name が“ORiN Device”のものを探します。
4. “ORiN Device”が持っているサービスを取得し、テーブルに登録します。
5. QueryStateVariable API を用いて、変数の取得を行います。
6. InvokeAction API を用いて、コマンドの実行を行います。

5.2. CaoUPnPConfig の設定

次に CaoUPnPConfig を利用して、CaoUPnP からアクセスすることのできる CAO プロバイダの設定を行います。

(1) CaoUPnPConfig の起動

タスクマネージャで、CaoUPnP.exe が起動していないことを確認してから、CaoUPnPConfig を起動してください。

(2) Controller の追加

「Edit」の「Add Controller」から、コントローラの追加を行います。ここでは、コントローラ名を“DS1”とし、プロバイダには“CaoProv.DataStore”を選択しました。

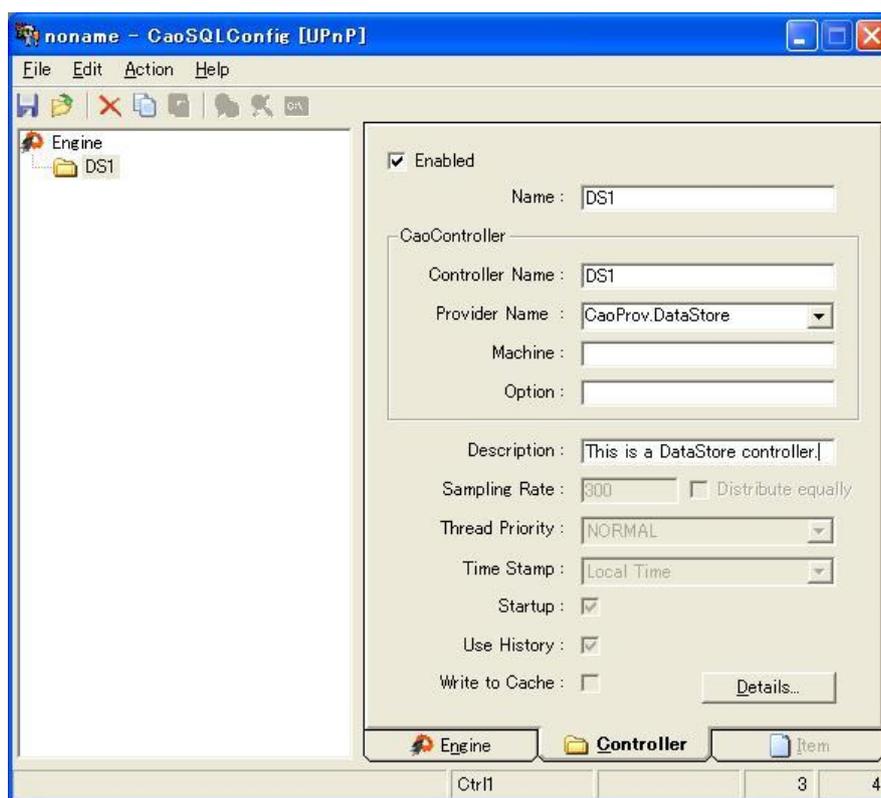


図 5-3 コントローラの追加

(3) Item の追加

メニューの「Edit」から「Add Item」を選択し、アイテムの追加を行います。ここでは、“ItemA”，“ItemB”，“ItemC”を追加しました。

「Description」, 「Details」の項目は自由に設定してください。

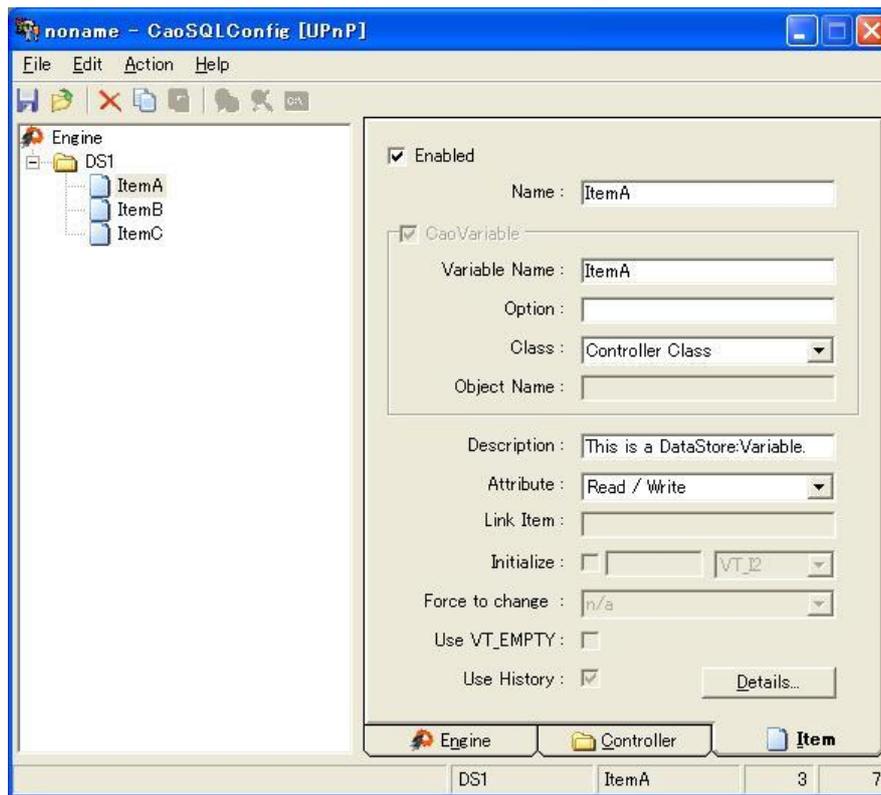


図 5-4 アイテムの追加

(4) 設定の保存

メニューの「File」から「Save as. . .」を選択し、設定の保存を行います。

5.3. サンプルの実行

最後に作成したクライアントアプリケーションの実行を行います。

(1) CaoUPnP.exe の起動

CaoUPnP.exe の起動を行います。CaoUPnP.exe を起動するには、「コントロールパネル」→「管理ツール」→「コンピュータの管理」から“CaoUPnP”を選択してサービスの開始をしてください。もしくはCaoUPnPConfig の「Action」メニューから“Service Start”を選択してサービスを起動して下さい。

(2) クライアントアプリケーションの実行

5.1 節で作成したクライアントアプリケーションを起動してください。起動に成功すると以下のような画面が表示されます。

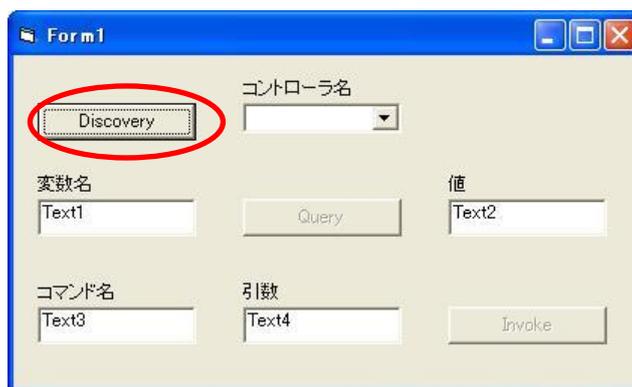


図 5-5 クライアントアプリケーション画面

次に“Discovery”ボタンを押下してください。ネットワーク上に存在する UPnP デバイスの検索を行います。ここで、CaoUPnP が見つかったと、以下のようなダイアログが表示されます。



図 5-6 UPnP デバイスの発見

ここで、コントローラ名コンボボックスをクリックすると、以下のように、CaoUPnPConfig で設定した、コントローラの名前一覧が表示されます。

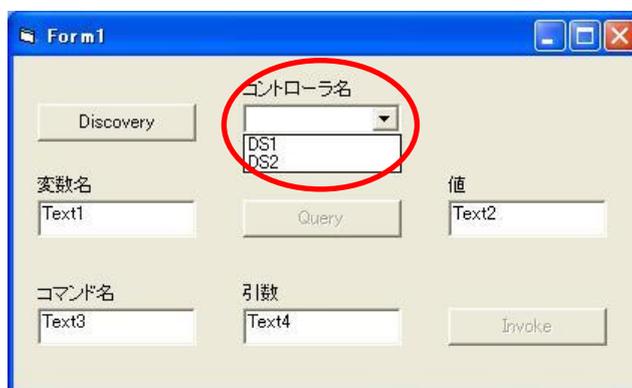


図 5-7 コントローラを選択

コンボボックスからコントローラを選択後、変数名に“ItemCount”を入力し“Query”ボタンを押下すると、値に CaoUPnPConfig で登録したアイテムの個数が表示されます。

今回は、“ItemA”、“ItemB”、“ItemC”の3つを登録したので、“3”が表示されます。

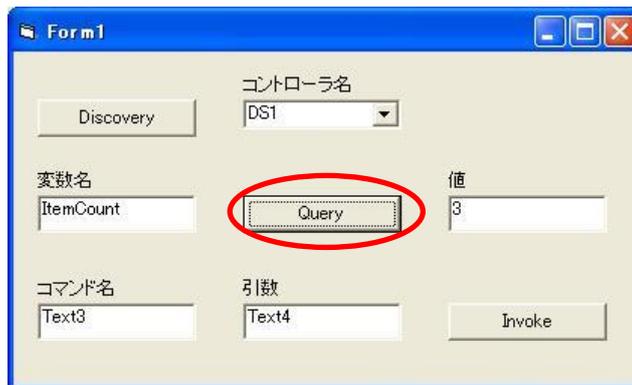


図 5-8 ItemCount の取得

コマンド名に“put_Command”を入力し、引数に適切な文字列を入力して“Invoke”ボタンを押下します。

確認のために変数名に“Command”を入力し、“Query”ボタンを押下すると、先ほど入力した文字列が表示されます。

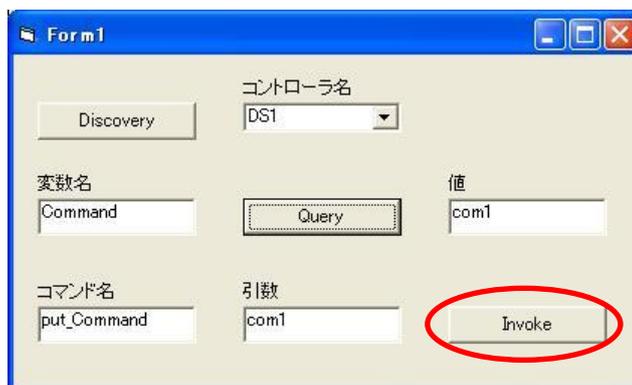


図 5-9 put_Command コマンドの実行

同様に、コマンド名に“put_ItemName”を入力し、引数には、CaoUPnPConfig で登録したアイテム名を入力して“Invoke”ボタンを押下します。

確認のために変数名に“ItemName”を入力し、“Query”ボタンを押下すると、現在のアイテム名が表示されます。

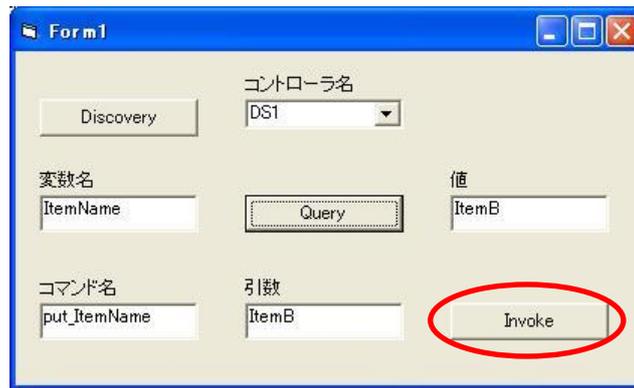


図 5-10 put_ItemName コマンドの実行

さらに、コマンド名に“put_ItemValue”を入力し、引数には、変数値を入力して“Invoke”ボタンを押下します。

確認のために変数名に“ItemValue”を入力し、“Query”ボタンを押下すると、現在のアイテム値が表示されます。

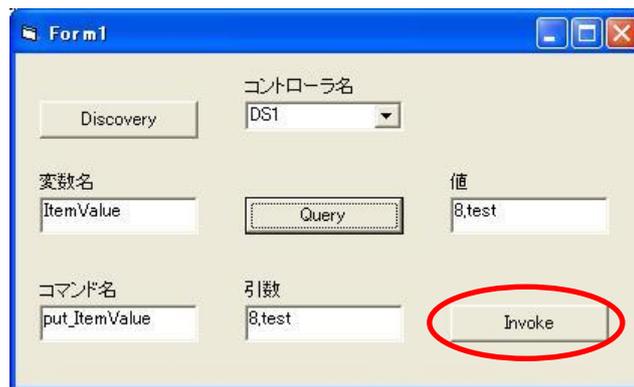


図 5-11 put_ItemValue コマンドの実行

(3) CaoTester による動作確認

CaoTester ツールを利用して、DataStore の変数オブジェクトに値が書き込まれていることを確認します。

まずは、コントローラ名“DS1”、プロバイダ名“CaoProv.DataStore”で、AddContorller を行って下さい。次に、変数名“ItemA”で AddVariable を行い、Get ボタンを押下します。変数に値が書き込まれていることを確認してください。

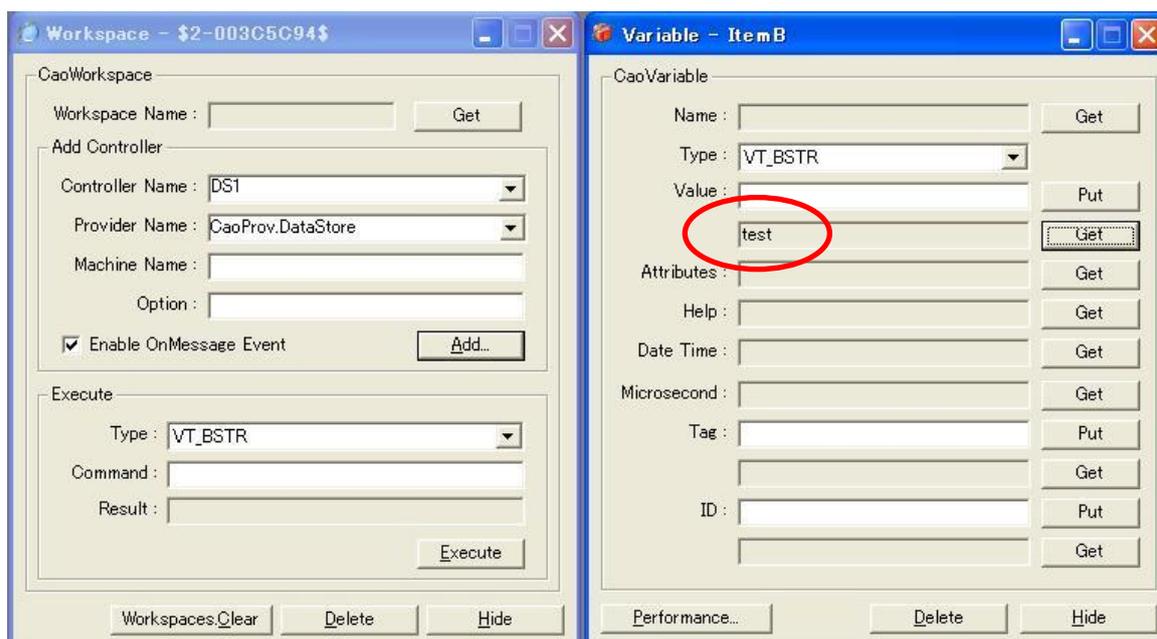


図 5-12 CaoTester による確認

6. 注意事項

- (1) CaoUPnPConfig で csq ファイルを生成/保存してから, CaoUPnP を起動して下さい.
- (2) CaoUPnP.exe, CaoUPnPDevice.dll, CaoUPnPService.xml は 同一フォルダに置いて下さい.
presentationURL のファイルはこれらのファイルと 同一フォルダ以下に置いて下さい.
- (3) presentationURL は 相対パスで記述して下さい.
- (4) ファイアウォールソフトがインストールされている場合, UPnP で使用するポート(**2869**)を空けてください.
- (5) 不正終了等でネットワーク上に CaoUPnP が残っている場合, [コンピュータの管理]→[サービスとアプリケーション]→[サービス]から[**Universal Plug and Play Device Host**]サービスを再起動して下さい.

7. 参考情報

7.1. クライアントサンプル

- (1) 「Platform SDK for Windows XP SP2」に入っているサンプル(VB, CPP)

「Platform SDK for Windows XP SP2」は以下の URL から入手できます。

<http://www.microsoft.com/msdownload/platformsdk/sdkupdate/XPSP2FULLInstall.htm>

UPnP クライアントのサンプルは以下のフォルダに存在します。

%Platform SDK for Windows XP SP2 インストールフォルダ%\Samples\NetDS\upnp\GenericUCP

※VB の場合, FunctionDelegator フォルダ内の TLBTypes.olb, VBoostTypes6.olb を参照設定し, exe ファイルを生成して下さい。

※生成されたサンプル exe ファイルと同フォルダに devtype.txt, udn.txt ファイルを置いてください。

- (2) MSDN のサンプル(VBScript)

以下の URL を参照して下さい。

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/upnp/upnp/creating_a_presentation_web_page.asp

- (3) Intel が配布している「Intel Tools for UPnP Technologies」の中にある DeviceSpy.exe を利用すれば, 容易にテストを行うことができます。

以下の URL から入手できます。

http://www.intel.com/cd/ids/developer/asmo-na/eng/downloads/upnp/tools/index.htm#anchor_3

こちらを利用するには Microsoft .NET Framework version 1.1 以上が必要です。

7.2. UPnP 関連リンク

UPnP Forum :

<http://www.upnp.org/>

MSDN(UPnP):

<http://www.microsoft.com/japan/whdc/device/netAttach/UPnP/default.mspix>

MSDN(UPnP APIs):

http://msdn.microsoft.com/library/en-us/upnp/upnp/universal_plug_and_play_start_page.asp

MSDN(UPnP クライアントサポート):

<http://www.microsoft.com/japan/technet/itsolutions/network/plan/insidenet/sohonet/upnpsup.mspix>