

PSLXプラットフォーム計画

C#版

共通モジュール実装マニュアル

第4部

共通コンポーネント API 仕様

バージョン 1.0

2009年6月

NPO 法人ものづくり APS 推進機構

改訂履歴

日付	内容	備考
2009/03/12	バージョン 1.0 ベータ版	
2009/06/07	バージョン 1.0	

もくじ

1. はじめに.....	6
◆ 目的.....	6
◆ 対象とする読者	6
◆ 稼動環境	7
◆ 電子ファイル	7
2. ドキュメントサービスモジュール	8
◆ DocumentManager クラス	8
コンストラクタ	8
メソッド	8
プロパティ	8
イベント	9
◆ TransactionProcess クラス	9
プロパティ	9
メソッド	10
◆ TransactionMessage クラス	10
メソッド	10
プロパティ	10
◆ Document クラス	11
メソッド	11
プロパティ	12
◆ Error クラス	13
プロパティ	13
◆ Header クラス	13
メソッド	13
プロパティ	14
◆ Condition クラス	14
メソッド	15
プロパティ	15
◆ Selection クラス	15
メソッド	16
プロパティ	16
◆ DomainObject クラス	17
プロパティ	17

メソッド	18
◆ Constraint クラス	18
プロパティ	19
メソッド	19
◆ Property クラス	19
プロパティ	19
◆ ProfileDefinition クラス	20
プロパティ	20
メソッド	20
◆ DocumentProfile クラス	21
メソッド	21
プロパティ	21
◆ DomainObjectProfile クラス	21
メソッド	21
プロパティ	21
◆ PropertyProfile クラス	22
メソッド	22
プロパティ	22
◆ ImplementProfile クラス	22
メソッド	22
プロパティ	23
◆ ImplementDocument クラス	23
メソッド	23
プロパティ	24
◆ ImplementProperty クラス	24
プロパティ	24
◆ ImplementAction クラス	24
プロパティ	25
◆ ImplementEvent クラス	25
メソッド	25
プロパティ	25
◆ PpsDocumentsException クラス	26
プロパティ	26
3. メッセージングサービスモジュール	27
◆ TransportMessage クラス	27
コンストラクタ	27

メソッド	27
プロパティ	27
◆ MessageManager クラス.....	27
コンストラクタ	28
メソッド	28
プロパティ	28
◆ MessageConnector クラス	29
メソッド	29
プロパティ	29
イベント	30
◆ PpsMessagingException クラス	30
プロパティ	30
◆ IConnectorEngine インタフェース	30
メソッド	30
プロパティ	30
イベント	31
◆ QueueReceiveArgs クラス (イベント引数)	31
コンストラクタ	31
プロパティ	31

1. はじめに

◆ 目的

PPS 共通コンポーネントは、通信の細かなしくみ、および XML に関する専門知識の浅いアプリケーション・プログラマであっても、容易に PSLX プラットフォーム対応のシステム構築が可能となるように設計されています。また、OASIS PPS 技術委員会が定めた国際標準を準拠したうえで、今後、個々に開発されるさまざまなアプリケーション間の相互接続性を保証するための、共通の実装環境を提供します。

この仕様書は、実装マニュアル第 4 部「共通コンポーネント API 仕様」です。ここでは、第 1 部から第 3 部までで解説した PPS ドキュメントサービス、PPS メッセージングサービスの 2 つの PPS 共通コンポーネントのアプリケーション・インタフェース (API) を体系的に示します。この仕様書は、プログラミングのためのリファレンスマニュアルとして利用することができます。なお、本仕様書とは別に、電子的な API のヘルプファイルが提供されていますので、そちらも合わせて利用してください。

◆ 対象とする読者

(1) 資格

この仕様書は、PSLX プラットフォーム計画プロジェクトに参加している企業の従業員に対して、PSLX プラットフォーム対応ソフトウェアを開発するために公開している文章です。PSLX プラットフォーム計画プロジェクトのメンバー以外であっても、この仕様書を閲覧することは可能ですが、NPO 法人ものづくり APS 推進機構の許可なく複製や再配布を行うことは禁止されています。

(2) 必要とする知識・技術

ソフトウェア開発の一般知識を有する人を対象にした文章です。特に、下記の項目についての知識が必要です。

- C#の言語仕様
- Visual Studio による開発方法
- オブジェクト指向モデリングの概要

◆ 稼働環境

本仕様書に含まれる内容を実行するためには、次のソフトウェア環境が必要となります。

区分	内容
オペレーティングシステム	Windows XP Service Pack 3 以降 Windows 2003 Server 以降
コンポーネント	.NET Framework 2.0 以降
Web ブラウザ	Microsoft Internet Explorer 6,7 および 8 Mozilla Firefox 3.0
開発ツール	Visual Studio 2005 ServicePack1、または Visual Studio 2008

◆ 電子ファイル

本仕様書の内容に相当する電子データとして、共通コンポーネントから自動生成されたヘルプファイルがあります。引数やより詳細な情報は、次のヘルプファイルを参照してください。

コンポーネント	ヘルプファイル名
PPS ドキュメントサービス	Pps.Documents.chm
PPS メッセージングサービス	Pps.Messaging.chm

2. ドキュメントサービスモジュール

◆ DocumentManager クラス

DocumentManager クラスは、PSLX 対応の XML を処理するための管理クラスです。送信や受信のためのメッセージの単位で、XML を生成または解釈します。また、メッセージの処理の単位である TransactionProcess クラスのインスタンスを生成します。

コンストラクタ

コンストラクタ名	引数	使い方
DocumentManager	string id	アプリケーション ID を指定して、インスタンスを生成します。
DocumentManager	なし	インスタンスを生成します

メソッド

メソッド名	引数	戻り値	使い方
Initialize	string profileName	void	プロファイルのファイル名(パス付)を指定して初期化します。
Initialize	Stream stream Stream schema	void	指定したストリームからプロファイルおよびスキーマを読み込み、初期化します。
CreateProfileMessage	Document.ActionTypes type	TransactionMessage	自身のアプリケーションの実装プロファイルを生成します。
ResumeProcess	string id	TransactionProcess	トランザクションIDをもちいて、実行中のトランザクションプロセスを生成し再開します。
CreateProcess	なし	TransactionProcess	トランザクションプロセスを新規に生成し開始します。
	TransactionMessage message	TransactionProcess	受け取った業務メッセージからトランザクションプロセスを生成し開始します。
	string xml	TransactionProcess	受け取った XML テキストからトランザクションプロセスを生成し開始します。
Close	なし	void	ドキュメントマネージャの処理を終了する。カウンタ等を保存します。
Read	string filename	TransactionMessage	ファイル名を指定して業務メッセージを読み込みます。
	Stream stream	TransactionMessage	ストリームを指定して業務メッセージを読み込みます。
	TextReader textReader	TransactionMessage	テキストリーダーを指定して業務メッセージを読み込みます。
ReadImplementProfile	Stream stream	ImplementProfile	ファイルから実装プロファイル(XML データ)を読み込みます
ReadImplementProfile	string filename	ImplementProfile	ファイルから実装プロファイル(XML データ)を読み込みます
ResumeProcess	string id	TransactionProcess	トランザクション処理を再度生成します。
Parse	string xml	TransactionMessage	XML テキストを指定して、対応する業務メッセージを生成します。
CreateProfileMessage	Document.ActionTypes type	TransactionMessage	定義された情報に従って実装プロファイルを生成します。

プロパティ

プロパティ名	型	意味	使い方
--------	---	----	-----

DocumentIdCounter	int	生成メッセージのカウンタ	生成された業務ドキュメントに一意の ID を与えるためのカウンタ
DocumentIdFormat	string	業務ドキュメントの識別記号のフォーマット文字列	カウンタ(数値)をもちいた文字列を生成するためのフォーマット文字列
FnConfig	string	システム設定用ファイル名	システム設定データ(カウンタなど)を保存するファイル名を指定します
FnPSLXSchema	string	PSLX 用の XML スキーマ	Initialize メソッドを呼ぶ前に、ファイルの格納位置を絶対パスで指定します
Id	string	アプリケーション ID	自身のアプリケーションを送信相手に認識させるために、ユニークな文字列を指定します。
Implement	ImplementProfile	実装プロファイル(定義用)	業務アプリケーションが設定した実装プロファイルを取得します。
MessageIdCounter	int	送信メッセージのカウンタ	送信したメッセージに一意の ID を与えるためのカウンタ
MessageIdFormat	string	メッセージ識別記号のフォーマット文字列	カウンタ(数値)をもちいた文字列を生成するためのフォーマット文字列
Name	string	アプリケーション名	送信者を識別するためのアプリケーション名称
Profile	ProfileDefinition	プロファイルを取得する	業務ドキュメント、業務オブジェクト等に関する定義(プロファイル)を取得します。
TransactionIdCounter	int	生成トランザクションのカウンタ	生成されたトランザクションに一意の ID を与えるためのカウンタ
TransactionIdFormat	string	トランザクション識別記号のフォーマット文字列	カウンタ(数値)をもちいた文字列を生成するためのフォーマット文字列

イベント

メソッド名	引数	戻り値	使い方
TransactionStartHandler	TransactionProcess transaction	void	トランザクション処理を開始します。
TransactionCommitHandler	TransactionProcess transaction	void	トランザクション処理を完了します。
TransactionCancelHandler	TransactionProcess transaction	void	トランザクション処理を中断します。

◆ TransactionProcess クラス

TransactionProcess クラスは、アプリケーションが業務ドキュメントを処理する単位です。サーバの場合には、受信したメッセージとそれに対応して返信するメッセージは同一の TransactionProcess 内で管理されます。また、必要に応じて、アプリケーション・プログラムは、コミットやロールバック等を行うことができます。

プロパティ

プロパティ名	型	意味	使い方
Confirm	ConfirmTypes	確認要求	トランザクションの処理結果に対する確認要求を設定または取得します。 Never: 確認は不要 OnError: エラーが発生した場合のみ通知する Always: 成否にかかわらず必ず確認を返信する
ConfirmReceived	ConfirmTypes	受け取った確認要求	受取った業務メッセージにある確認要求を取得または設定します。
Documents	List<Document>	トランザクション処理における生成した業務ドキュメントのリスト	すでに生成された業務ドキュメントのリストを取得します。ただし、取得した後で、そのリストに対して、削除などの処理を行うことができます。
DocumentsReceived	List<Document>	トランザクション処理にお	受信メッセージの業務ドキュメントのリストを取得し

		ける受信した業務ドキュメントのリスト	ます。
Id	string	トランザクション識別用文字列	特定のアプリケーション ID 内でユニークである識別文字列を返します。ID は、ドキュメントサービスによって設定されます。
InitiatorName	string	送信者名	受信メッセージの場合に、送信元を識別するための ID が設定されています。トランザクションはこのアプリケーション ID とトランザクション ID との組でユニークとなります。
Result	TentativeResults	トランザクション処理結果	トランザクションの処理結果または状態を取得します。

メソッド

メソッド名	引数	戻り値	使い方
CreateDocument	string docName	Document	指定した名前の業務ドキュメントを生成します。
CreateDocument	string docName string optionName	Document	指定した名前の業務ドキュメントを生成します。オプション文字列を指定することができます。
CreateMessage	なし	TransactionMessage	TransactionMessage を生成します。
IsResponseRequired	なし	bool	返信用トランザクションが返信を要求しているかの区分を取得します。受信したメッセージの Action 属性が Get の場合や返信要求属性が Always の場合は、true が自動設定されます。
TransactionStart	なし	void	トランザクション処理を開始します。
TransactionCommit	なし	void	トランザクション処理を完了します。
TransactionCancel	なし	void	トランザクション処理を中断します。
TransactionResume	なし	void	クライアントがトランザクションを再開します。
CheckTransaction	Document document	bool	受け取ったトランザクション内のドキュメントの処理結果を確認します。
ExitTransaction	なし	void	受け取ったトランザクションの処理を終了します。

◆ TransactionMessage クラス

TransactionMessage クラスは、一回に送信または受信する情報を表現します。TransactionMessage のインスタンスには、ユニークな ID が設定されます。トランザクションメッセージには、複数の業務ドキュメントを持ちます。

メソッド

メソッド名	引数	戻り値	使い方
Write	string name	void	XML を指定したファイル名に出力します。
	Stream name	void	XML を指定した Stream に出力します。
	TextWriter name	void	XML を指定した TextWriter に出力します。
XmlString	string	void	XML を文字列として取得します。

プロパティ

プロパティ名	型	意味	使い方
Confirm	TransactionProcess.ConfirmTypes	確認要求	往信メッセージに設定された確認要求を設定または取得します。返信メッセージの場合は、設定した値は無効です。 Never: 確認は不要 OnError: エラーが発生した場合のみ、通知する

			Always: 成否にかかわらず、必ず確認を返信する
ConnectionString	string	内部 DB 接続用の文字列	内部の DB に接続するための文字列。ユーザ認証のための情報などを設定または取得します。
Documents	List<Document>	業務ドキュメントのリスト	トランザクションに含まれる業務ドキュメントの一覧を取得します
Id	string	送信または受信メッセージの ID	トランザクション ID を取得します
Implement	ImplementProfile	実装プロファイル	実装プロファイルを取得します。(送受信用)
ImplementObtained	ImplementProfile	取得した実装プロファイル	取得した実装プロファイルを取得します。
ImplementObtainedAction	Document.ActionTypes	取得した実装プロファイルの要求アクション	取得した実装プロファイルの要求アクションを取得します。
IsCanceled	bool	取り消し状態	取消が指定されているかどうかを取得または設定します。
IsCommitted	bool	確認(完了)状態	コミットメントが指定されているかどうかを取得または設定します。
isStarted	bool	開始状態	開始が指定されているかどうかを取得または設定します。
MessageType	MessageTypes	メッセージ種別	メッセージの種別を取得します。 Profile: 実装プロファイル Transactions: 業務ドキュメントを含むトランザクション
SenderName	string	送信者名	メッセージを生成したアプリケーション ID を取得します。
TransactionId	string	トランザクション ID	トランザクション ID を取得または設定します。

◆ Document クラス

Document クラスは、業務ドキュメントを表します。業務ドキュメントは、サーバあるいはクライアントにおけるアクションの単位であり、Add (追加)、Change (編集)、Remove (削除)、Get (照会)、Show (Get に対する回答)、Sync (イベントの要求)、Notify (通知)、Confirm のいずれかを定義します。業務ドキュメントはその内部に、業務オブジェクトをもつことができます。

メソッド

メソッド名	引数	戻り値	使い方
SetAppData	string name, string val	void	アプリケーション固有の情報を設定します。(拡張領域)name は属性名、val は値です。
GetAppData	string name	string	アプリケーション固有の情報を取得します。(拡張領域)なお、属性名は、あらかじめオフライン情報として相手に通知しておく必要があります。
CreateError	なし	Error	空の Error をドキュメントに新規追加します。
CreateDomainObject	なし	DomainObject	空の DomainObject をドキュメントに新規追加します。
CreateHeader	なし	Header	空の Header をドキュメントに新規追加します。(ヘッダは1つしか追加できない)
	string objectName	Header	業務ドキュメントとは異なるオブジェクトクラスに属するヘッダを生成します。
CreateCondition	なし	Condition	空の Condition をドキュメントに新規追加します。
	string id	Condition	ID 制約をもった制約を新規追加します。
CreateSelection	なし	Selection	空の Selection をドキュメントに新規追加します。
	Selection.SelectionType	Selection	セレクションタイプを同時に追加します。

	pes type		
GetDomainObjectProfile	なし	DomainObjectProfile	該当する業務オブジェクトの定義情報を取得します。

プロパティ

プロパティ名	型	意味	使い方
Id	string	ドキュメントの識別 ID	アプリケーション ID 内で一意となるように設定してください。共通モジュールは、TransactionManager クラスの DocumentIdCounter プロパティと DocumentIdFormat プロパティを使って自動的に一意となる ID を割り当てます。
Action	ActionTypes	アクション種類	アクション種別を設定します。 Add: 追加 Change: 編集 Remove: 削除 Get: 情報要求 Show: Get に対する回答 (Get を受け取ったサーバは、Show で返事をしなければならない。) Sync: イベントの要求 Notify: 通知 Confirm: Add, Change, Remove および Sync の Confirm 要求に対する回答
EventName	string	イベントの識別名	イベントの識別名を取得または設定します。
DocumentName	string	業務ドキュメント名	業務ドキュメント名 (Product, OperationSchedule など) を取得します。(業務ドキュメント名は CreateDocument メソッドで指定します)
ObjectName	string	業務オブジェクト名	業務オブジェクト名を取得します。業務オブジェクト名は業務ドキュメント名 (DocumentName) が決まると、PSLX プロファイルにより一意に決まる (profile-pslx.xml に記述されています。プロファイル定義ファイル名は、TransactionManager の Initialize メソッドの引数で与えている)。業務ドキュメント名に従った、業務オブジェクトが設定されています。
PrimitiveName	string	プリミティブ要素名	プリミティブ要素名を取得します。 【PLSX 規約】業務ドキュメント名 (DocumentName) が決まると、プリミティブ要素名は PSLX プロファイルにより一意に決まる (profile-pslx.xml に記述されている。プロファイル定義ファイル名は、TransactionManager の Initialize メソッドの引数で与えている)。業務ドキュメント名に従った、プリミティブ要素名が設定されている。
MainHeader	Header	Header 要素	ヘッダ要素を取得します。
Errors	List<Error>	Error 要素	アクション種類が、Show と、Confirm の場合のみ使用します。 受信したドキュメントの処理中にエラーが発生した場合に、エラー内容を通知します。
TentativeResult	TransactionProcess.TentativeResults	トランザクション処理結果	トランザクション処理を行っている場合、その一時的結果を取得します。 Success: 成功 Failure: 失敗 Skipped: 未処理
DomainObjects	List<DomainObject>		業務ドキュメントの中身を表す業務オブジェクトを取得します。業務オブジェクトは、CreateObject メソッドで生成することができます。
Conditions	List<Condition>	Condition 要素	アクション種類が、Change, Remove, Get の場合に使用します。対象とするドメインオブジェクトの条件指定 (SQL の where 文に相当)。Condition 要素が複数ある場合には、それらを“OR”の関係になっている意味となります。
Selections	List<Selection>	Selection 要素	アクション種類が Get または、Change の時のみに

			使用します。Getまたは、Chnageの対象となるプロパティ名です。
Reference	Document	ドキュメントが参照する相手のドキュメント	返信においては、対応する往信内のドキュメントを設定または取得します。また、その場合に、逆に往送側のドキュメントに、返信の対応するドキュメントを設定または取得します。
ReferenceId	string	ドキュメントの参照先ドキュメント ID	ドキュメントの参照先ドキュメント ID を取得します。
Option	string	オプション名	適用ロジックのためのオプション識別文字を取得します。
Profile	DocumentProfile	業務ドキュメントの定義情報	業務ドキュメントの定義情報を取得します。

◆ Error クラス

Error クラスは、業務ドキュメントに含まれ、相手から受け取った業務ドキュメントの処理においてエラーまたは警告が発生した場合に、相手に返信するためのものです。ひとつの業務ドキュメントに複数設定が可能です。受け取った業務ドキュメントの確認要求が On Error の場合で、エラー情報が 1 つ以上存在する場合には、返信メッセージを返さなければなりません。

プロパティ

プロパティ名	型	意味	使い方
ErrorId	string	エラー識別番号	ドキュメント内でユニークな ID を取得します。
ReferenceInfo	string	エラーとなった業務メッセージの参照情報	エラーの発生場所を特定するために有用な任意の記述を取得または設定します。
ErrorCode	string	エラーコード	PPS 仕様書に準拠した 000 から 012 までのエラーコードを取得または設定します。ただし現時点では規約ではありません。
Location	string	エラーの場所情報	ドキュメント内でのエラーの発生位置(行数など)を示す任意の情報を取得または設定します。
ErrorType	ErrorTypes	エラー種別	エラー種別を取得または設定します。 Error: エラー(省略値) Warning: 警告
Description	string	任意の説明	任意の説明情報を取得または設定します。

◆ Header クラス

Header クラスは、業務ドキュメントに設定するヘッダ情報であり、業務ドキュメントに 1 つだけ設定可能です。ヘッダには、業務ドキュメントの本体として設定された業務オブジェクトと同じ種類の業務プロパティを設定する場合と、ヘッダ独自に異なる種類の業務オブジェクトの内容を設定する場合があります。

メソッド

メソッド名	引数	戻り値	使い方
GetProperty	string name	Property	すでに生成した指定名のプロパティを検索し返し

			ます。
CreateProperty	string name	Property	指定名のプロパティを追加します。指定したプロパティ名がすでに存在する場合は例外が発生します。中間オブジェクトの階層にかかわらず設定可能。指定する名前は、プロファイルに登録済みであるか、あるいは“user:”ではじまる文字列でなければなりません。

プロパティ

プロパティ名	型	意味	使い方
ObjectId	string	対象オブジェクトの ID	ドキュメントのヘッダに表示する業務オブジェクト (ボディ部の業務オブジェクトと異なる種類である場合もあるので注意) の ID を取得または設定します。
ObjectName	string	対象となる業務オブジェクト名称	ドキュメントのヘッダに表示する業務オブジェクト (ボディ部の業務オブジェクトと異なる種類である場合もあるので注意) の名称を取得または設定します。
Title	string	業務ドキュメントのタイトル	ドキュメントに設定する任意のタイトル文字列を取得または設定します。
TotalObjectCount	int	業務オブジェクト数	業務オブジェクトの総数を設定または取得します。親の業務ドキュメントのアクション種別が Get の場合は、要求する最大件数を表します。Show の場合は、照会内容条件に該当する業務オブジェクトの総数を表します (ただし、一回の返信にすべてのオブジェクトを含まない場合があります、その場合にはボディ部の業務オブジェクト数と異なる場合があるので注意のこと。複数回に分けて送信する場合には、offset 属性との組み合わせを行います。)
Offset	int	オフセット (読み飛ばし) 数	オフセット数を設定または取得します。親の業務ドキュメントのアクション種別が Get の場合は、オフセット (読み飛ばし) を依頼する数。Show の場合は、返信する業務オブジェクトが先頭を読み飛ばした数。一般にオフセットを行う場合はソート属性を付加し、そうでない場合は ID で昇順にソートされているものとします。
Properties	List<Property>	ヘッダ用プロパティのリスト	ヘッダ用の業務オブジェクトの内容を表すプロパティのリストを取得します。ヘッダの場合は、同一名称のプロパティが複数存在する場合があります。
this[string name]	object	プロパティ内容	プロパティの値を設定または取得します。

◆ Condition クラス

Condition クラスは、条件を設定するためのクラスであり、サーバ側が保有する複数の業務オブジェクトの中から、特定の業務オブジェクトを選択する必要がある場合や、サーバ側の業務オブジェクトにおいて、複数型の業務プロパティが存在する場合に、それらの中から必要とするものを選択する場合に利用します。前者は、業務オブジェクトにおいて CreateCondition メソッドで生成され、後者は Selection オブジェクトにおいて、CreateCondition メソッドで生成されます。

メソッド

メソッド名	引数	戻り値	使い方
SetConstraint	string name, object value, Constraint.ConstraintT ypes type	Constraint	Condition クラスの Constraint リストに新規 Constraint(絞りこみ条件)を追加します。プロパティ名は PSLX 業務オブジェクトのプロパティ名指定してください。値は、プロパティのデータ型が char 型の場合 string を、qty 型の場合は decimal を、time 型の場合には DateTime 型の値をセットしてください。 制約条件(type)は、次の値を指定します。 EQ:等しい GT:より大きい LT:より小さい GE:以上 LE:以下 NE:等しくない なおプロパティのデータ型が char の場合には EQ または NE のみが指定できます。

プロパティ

プロパティ名	型	意味	使い方
Id	string	ID 条件	検索対象オブジェクトの ID を取得または設定します。SQL 文における Where ID= ○○ に相当します。
Wildcard	string	ワイルドカード指定プロパティ名	Wildcard を適用するプロパティ名を取得または設定します。プロパティ名は PSLX 業務オブジェクトのプロパティ名であり、業務プロパティのデータ型が char 型のもののみ、記述可能です。
Value	string	条件の値	Wildcard で指定したプロパティの Wildcard(*)を含む検索文字列を取得または設定します。
Version	string	バージョン指定	データのバージョンを取得または設定します。 "latest"=最新バージョン "earliest"=最古バージョン "user:"+任意文字列=アプリケーションごとに定義した表現
Constraints	List<Constraint>	条件のリスト	対象オブジェクトの絞り込み条件のリストを取得します。Condition 内の Constraints は AND 条件で対象 Object を絞り込みます。
this[string name]	object	プロパティ内容	SetConstraint において条件が EQ である場合に、業務プロパティ名と値によって制約内容を取得または設定します。また、指定した業務プロパティ名に対応して、制約として設定された値を取得します。

◆ Selection クラス

Selection クラスは、対象とする業務プロパティの種類を指定する場合に利用します。このクラスのオブジェクトは、業務ドキュメントの Action 種別が照会 (Get) または修正 (Change) の場合のみ定義してください。複数型のプロパティのオブジェクトを選択するために、このオブジェクトの下位に Condition を設定する場合には、この Selection オブジェクトは中間オブジェクトごとに分けて生成してください。このクラスのオブジェクトは業務ドキュメント (Document) オブジェクトにおいて、CreateSelection メソッドによって定

義されます。

メソッド

メソッド名	引数	戻り値	使い方
CreateCondition	なし	Condition	セレクション内のコンディションデータを生成します。ここで生成したコンディションオブジェクトに対して、制約を設定していく。この制約の内容によって、サーバ側で選択される業務プロパティが異なります。
CreateProperty	string name	Property	Action 種類が照会 (Change) の場合に照会するプロパティの名称を設定します。業務オブジェクトがもつすべての業務プロパティが指定可能(階層上の位置を考慮する必要はありません。)
GetDomainObjectProfile	なし	DomainObjectProfile	複数型の業務プロパティを扱う場合に、それらが共通して所属する中間オブジェクトの定義情報を取得する。これは照会ドキュメント内で利用されます。

プロパティ

プロパティ名	型	意味	使い方
SelectionType	SelectionTypes	セレクションの種類	<p>アクション種別が Change の場合</p> <p>(1) Undefined: 対象プロパティが単数型の場合には、常に上書き (修正) となるので定義する必要がなく、Undefined となる。定義しても無視される。</p> <p>(2) Insert: 複数型業務プロパティを挿入する。</p> <p>(3) Update: 複数型業務プロパティを修正する。</p> <p>(4) Delete: 複数型業務プロパティを削除する。</p> <p>なお上記以外は、無視されます。既定値は Undefined です。</p> <p>アクション種別が Get の場合</p> <p>(1) None: 照会する業務プロパティが Properties リストによって記述されていることを示す (これは、デフォルト値)。</p> <p>(2) Typical: 典型的な業務プロパティ項目の照会を要求する。どのプロパティを返すかは、アプリケーションが任意に決定できる。(実装プロファイルで宣言する?)</p> <p>(3) All: すべての業務プロパティ項目の照会を要求する。</p>
DataCount	int	選択するデータの数 (依頼時)	アクション種別が Get (照会) の場合に、返信メッセージに含める業務オブジェクト数の最大値を指定します。(0だと無限大を表すものとします。)
DataOffset	int	選択するデータのオフセット (依頼時)	アクション種別 Get (照会) の場合に、返信メッセージに含める業務オブジェクトを先頭からオフセットする数を表します。(読み飛ばす数に相当します)。
Properties	List<Property>	選択内容を示すプロパティリスト	アクション種別 Get (照会) の場合でかつ SelectionType が None である場合に、照会する業務プロパティを個別に指定します。ここでは項目名のみ指定し、値は設定する必要はありません。アクション種別が Change (修正) の場合には、修正する業務プロパティの種類と修正後の値を指定します。Condition 要素による指定によって選択されたすべての業務オブジェクトに対して、同様に同じ修正依頼が発行されます。
Conditions	List<Condition>	選択要素の条件リスト	アクション種別が Change (修正) の場合は、該当する業務オブジェクトが確定した後に、対象とする業務プロパティが複数型である場合、その中のどの

			業務プロパティが修正の対象となるかを指定します。ここで指定した業務プロパティ名と値によって、その業務プロパティをもつ中間オブジェクトが選択され、その下位にある値が Property 要素の指定にしたがって変更されます。
<code>this[string index]</code>	<code>object</code>	プロパティ内容	アクション種別が修正 (Change) の場合は、修正対象となる業務プロパティ名の修正後の値を取得または設定します。

◆ DomainObject クラス

DomainObject クラスは、業務ドキュメント内に定義される業務オブジェクトを表現します。これは、階層構造となっています。トップレベルの DomainObject は、業務オブジェクトに対応し、これらのオブジェクトは、さらに下位に DomainObject のリストを持ち、それによって階層化が可能となります。上位の業務オブジェクトから一つ下位の業務オブジェクト（中間オブジェクト）を識別する方法としては、要素名と修飾名を用います。

プロパティ

プロパティ名	型	意味	使い方
Id	<code>object</code>	業務オブジェクトの ID	これは業務プロパティの中で ID としてプロフィールに定義されているものの値。サーバ上で業務オブジェクトを検索する際の主キーに相当します。もし、プロフィールに ID が定義されていない場合 null を返します。
Parent	<code>DomainObject</code>	親オブジェクト	所属する上位の業務オブジェクトまたは中間オブジェクトを取得します。最上位のオブジェクト(業務オブジェクト)の場合は null を返します。
LocalProperties	<code>List<Property></code>	所有するプロパティのリストを取得する	直接所有する業務プロパティのリストを取得します。これは、XML 構造において属性として値が設定される場合に相当します。すべての業務プロパティは最終的に、この localProperties に行き着きます。
PropertyObjects	<code>List<DomainObject></code>	所有する中間オブジェクトのリストを取得する	直接所有する中間オブジェクトのリストを取得します。XML 構造上、業務プロパティが要素の下位に設定される場合には、この中間オブジェクトが挿入される。
ElementName	<code>string</code>	該当する XML 要素名を取得する	該当する XML 要素名を取得します。たとえば、業務オブジェクトが Material の場合は、Compose, Relation, Set などが相当します。これは、Modifier とペアで利用される。(ElementName と Modifier が等しい場合は、同一の中間オブジェクトとして認識されます)
Modifier	<code>string</code>	該当する XML 要素の type 属性値を取得する	該当する XML 要素の type 属性値を取得します。たとえば、業務オブジェクトが Material の場合には、"pps:child" などが相当します。これは、ElementName とペアで利用されます。(ElementName と Modifier が等しい場合は、同一の中間オブジェクトとして認識されます。)
Profile	<code>DomainObjectProfile</code>		この業務オブジェクトの定義情報であるプロフィールを取得します。
<code>this[string index]</code>	<code>object</code>	プロパティ内容	プロパティ名を指定して業務プロパティの内容を取得または設定します。主に単数形の業務プロパティ

			イ用に用いるが、複数形の場合は先頭のデータもみを対象とする場合には利用可能です。
--	--	--	--

メソッド

メソッド名	引数	戻り値	使い方
CreateProperty	string name	Property	指定した名称のプロパティを下位に設定します。あらかじめプロファイルにおいて定義された業務プロパティの名称以外は設定できず例外が発生します。直下にはない業務プロパティも中間オブジェクト内を探して設定します。
GetAllProperties	なし	Property[]	その時点で設定されているすべての業務プロパティのリストを取得します。中間オブジェクトがある場合はその内部を再帰的に探します。
GetProperties	なし	Property[]	その時点で設定されている業務プロパティのうち、複数型についてはその先頭のものを含むリストを返します。
GetGetPropertyObjectProfiles	なし	DomainObjectProfile[]	現在値が設定されている中間オブジェクトをすべて列挙します。中間オブジェクトは、重複が取り除かれています。複数型の業務プロパティの値が複数定義されている場合でも、1と数えます。
GetPropertyObjects	DomainObjectProfile prof	DomainObject[]	指定した中間オブジェクトの定義情報を持つ中間オブジェクトのリストを取得します。すべての複数型の業務プロパティを対象とする処理を行う場合に使用します。取得した個々の中間オブジェクトにそれぞれ個別に値が設定されています。
GetPropertyObjects	string elementName string modifier	DomainObject[]	指定した要素名と修飾名を持つ中間オブジェクトのリストを取得します。すべての複数型の業務プロパティを対象とする処理を行う場合に使用します。取得した個々の中間オブジェクトにそれぞれ個別に値が設定されています。
GetPropertyObjects	string name	DomainObject[]	指定した業務プロパティ名を持つ第一レベルのすべての中間オブジェクトを取得します。すべての複数型の業務プロパティを対象とする処理を行う場合に使用します。取得した個々の中間オブジェクトにそれぞれ個別に値が設定されています。
GetPrimaryKey	なし	Property	主キーとなるプロパティを取得します。主キーが存在しない場合、あるいは設定されていない場合は null を返します。
CreatePropertyObjects	string elementName string modifier int size	DomainObject[]	複数型の業務プロパティの場合に、要素名と修飾名によって指定された指定数の中間オブジェクトを生成します。修飾名 (Modifier) を指定しない場合は null としてください。
CreatePropertyObjects	string name int size	DomainObject[]	複数型の業務プロパティの場合に、プロパティ名によって確定する第一レベルの中間オブジェクト (Spec, Compose など) を、指定された指定数だけ生成します。
RemovePropertyObjects	string elementName string modifier	void	業務プロパティが複数型の場合の関数。要素名と修飾名によって指定された中間オブジェクトをすべて削除します。
RemovePropertyObjects	string name	void	業務プロパティが複数型の場合の関数。プロパティ名で指定した情報を削除する。削除にともなって、他の枝がない中間オブジェクトは削除されず。

◆ Constraint クラス

Constraint クラスは、Condition クラスに設定される個々の条件を表現するための内部

クラスです。このクラスのオブジェクトの生成は、Condition クラスがもつ SetConstraint メソッドで行います。

プロパティ

プロパティ名	型	意味	使い方
PropertyName	string	絞り込み対象のプロパティ	PSLX 業務オブジェクトで使用されているプロパティ名を使用してください。
DataType	Property.DataTypes	制約値のデータ属性を取得する	Char, Qty, Time のいずれか。これは業務プロパティの定義の時点であらかじめ決定されています。
Value	object	絞り込む値	プロパティのデータ型により、次のデータを使ってください。char 型の場合は string、qty 型の場合は decimal、time 型の場合は DateTime 型となります。
ConstraintType	ConstraintTypes	絞り込み条件	EQ: 等しい GT: より大きい LT: より小さい GE: 以上 LE: 以下 NE: 等しくない

メソッド

メソッド名	引数	戻り値	使い方
GetProperty	なし	Property	制約を内部で表現しているプロパティオブジェクトを取得します。これは、コンディション情報をヘッダ情報にコピーする際に利用します。

◆ Property クラス

Property クラスは、業務オブジェクトである DomainObject クラスが持つ業務プロパティを表します。また、Header クラスの下位にも設定することが可能です。このクラスのインスタンスは、データの型や具体的な値を持ちます。

プロパティ

プロパティ名	型	意味	使い方
PropertyUseType	PropertyTypes	プロパティの種別	プロパティの種別を取得または設定します。このプロパティが Header 要素の子として定義されている場合に設定します。 Target: 業務プロパティがヘッダ用業務オブジェクトのプロパティ Condition: 業務プロパティが Get で照会したときの条件で指定したプロパティ Selection: 業務プロパティが Get で照会したときの Selection 項目であるプロパティ
Name	string	プロパティ名を取得する	業務プロパティ名 (PSLX プロファイルにあらかじめ指定されている名称) です。
Path	string	プロパティの XPath 式	業務プロパティの XML 上での定義方法を表す XPath 式を取得します
Value	object	プロパティの値	業務プロファイルの値を取得または設定します。
SortType	SortTypes	ソートの方向	ソートの方向を取得または設定します。Action が Get (照会) の場合に、Selection 要素の下位に設定した業務プロパティには次の値を設定することができます。これにしたがって、照会されたアプリ

			ケーションは回答メッセージ内の業務オブジェクトをソートしなければなりません。 Asc:昇順 Desc:降順
CalculationType	CalculationTypes	列集計の方法	列集計の方法を取得または設定します。Action 種別が Get(照会)の場合に、Selection 要素の中で次の値を設定します。回答メッセージ(Show)の中では、Header 要素の中で設定します。この属性は SortType 属性と併用できません。Sum: 合計を計算する Ave: 平均を計算する Max: 最大値(最遅値)を計算する Min: 最小値(最早値)を計算する Count: データの数量を計算する
Display	string	表示用の文字列	業務プロパティの表示名を設定または取得します。
DataType	DataTypes	データ型を取得する	データ型を取得します。 char: 文字列 qty: 数量 time: 日時時刻
Multiple	bool	複数型かどうかの区分を取得する	業務プロパティが複数型であるかどうかを取得または設定します。
Profile	PropertyProfile	業務プロパティの定義情報を得る	業務プロパティの定義情報を取得します。

◆ ProfileDefinition クラス

ProfileDefinition クラスは、アプリケーション・プロファイルを定義するためのクラスです。各業務アプリケーションは必ず、最低1つの特定の業務プロファイルを指定しなければなりません。業務ドキュメント、業務オブジェクトなどは、このプロファイルの中で定義されます。

プロパティ

プロパティ名	型	意味	使い方
Name	string	プロファイル名	プロファイルの名称を取得します。
Prefix	string	プロファイルのプリフィックス文字列	プロファイルのプリフィックス文字列を取得します。
Objects	List<DomainObjectProfile>	プロファイルオブジェクトのリスト	プロファイルに含まれる業務オブジェクトのリストを取得します。
Documents	List<DocumentProfile>	プロファイルドキュメントのリスト	プロファイルに含まれる業務ドキュメントのリストを取得します。

メソッド

メソッド名	引数	戻り値	使い方
Read	string fileName	void	指定したファイルからプロファイル情報(ドキュメント情報とオブジェクト情報)を読み込みます。
	Stream stream	void	指定した Stream からプロファイル情報を読み込みます。
GetDocument	string name	DocumentProfile	プロファイルの中から該当ドキュメントを取得します。
GetObject	string name	DomainObjectProfile	プロファイル情報から該当オブジェクトのデータを取得します。

◆ DocumentProfile クラス

DocumentProfile クラスは、業務ドキュメントの定義を記述するためのクラスです。プロファイルの定義ファイルに記述された業務ドキュメントの定義がこのクラスによって表現されます。1つの DocumentProfile は、1つの DomainObjectProfile を指定し、これによって業務ドキュメントと業務オブジェクトの対応関係が規定されます。

メソッド

メソッド名	引数	戻り値	使い方
SetUserProperty	string name, Property.DataTypes type	void	ユーザ独自のプロパティを定義します。

プロパティ

プロパティ名	型	意味	使い方
Name	string	業務ドキュメント名	業務ドキュメント名を取得します。
ObjectName	string	業務オブジェクト名	業務オブジェクト名を取得します。
Category	string	カテゴリ名	カテゴリ名を取得または設定します。
Description	string	任意の説明	任意の説明を取得または設定します。

◆ DomainObjectProfile クラス

DomainObjectProfile クラスは、業務オブジェクトの定義を表現するためのクラスです。アプリケーション・プロファイルは、あらかじめ選択可能な業務ドキュメントの候補をこのクラスによって定義します。

メソッド

メソッド名	引数	戻り値	使い方
GetProperty	string name	PropertyProfile	指定したプロパティ名のプロパティ定義情報を得る。下位を再帰的に検索します。
GetAllProperties	なし	PropertyProfile []	その時点で設定されているすべてのプロパティのリストを取得します。中間オブジェクトがある場合はその内部を再帰的に探します。

プロパティ

プロパティ名	型	意味	使い方
Name	string	業務オブジェクト名	業務オブジェクト名を取得します。
PrimaryKey	PropertyProfile	主キーとなるプロパティ定義	主キーとなるプロパティ定義を設定または取得します。
LocalProperties	List<PropertyProfile>	業務プロパティ定義のリスト(属性)	業務プロパティ定義のリスト(属性用)を取得します。
PropertyObjects	List<DomainObjectProfile>	下位の中間オブジェクト定義のリスト	下位の中間オブジェクト定義のリストを取得します。
Primitive	string	プリミティブ要素名	プリミティブ要素名を取得します。
Description	string	任意の説明	任意の説明を取得または設定します。
ElementName	string	業務オブジェクトの要	業務オブジェクトが持つ要素名を取得します。

		素名	
Modifier	string	業務オブジェクトの修飾名	業務オブジェクトが持つ修飾名を取得します。

◆ PropertyProfile クラス

PropertyProfile クラスは、業務オブジェクトの定義として、設定可能な業務プロファイルの定義を指定するためのクラスです。それぞれの業務オブジェクトは、あらかじめ設定可能な業務プロファイルがアプリケーション・プロファイルの中で指定されています。業務プロファイルの定義としては、複数型の区別やデータ型などがあります。

メソッド

メソッド名	引数	戻り値	使い方
GetMultipleDefinition	out string elementName, out string modifier	DomainObjectProfile	プロパティが属する中間オブジェクトの定義情報を取得します。 elementName: 要素名 modifier: 修飾子名
		DomainObjectProfile	プロパティが属する中間オブジェクトの定義情報を取得します。

プロパティ

プロパティ名	型	意味	使い方
Name	string	名称	名称を設定または取得します。
Path	string	パス情報	パス情報を設定または取得します。
AttName	string	XML 属性名	XML 属性名を取得します。
Multiple	bool	複数型かどうか	複数型であるかどうかを設定または取得します。
MultiLmt	int	複数許可数	複数許可数(0 は不許可または無限大)
Data Type	Property.DataTypes	データ型	データ型
Extended	bool	拡張区分かどうか	拡張された区分であるかどうかを取得または設定します。
Description	string	任意の説明	任意の説明を取得します。
ParentObject	DomainObjectProfile	所属する業務オブジェクト定義	所属する業務オブジェクト定義を取得します。

◆ ImplementProfile クラス

ImplementProfile クラスは、アプリケーション・プログラムが、独自にもつ機能を表現するためのクラスです。ここで指定する内容は、アプリケーション・プロファイルの中であらかじめ定義された内容を取捨選択することで表現される内容となります。実際に処理が可能な業務ドキュメントを、実装ドキュメント (ImplementDocument) として複数設定することができます。

メソッド

メソッド名	引数	戻り値	使い方
AddDocument	string name, string option	ImplementDocument	実装ドキュメントを設定します。

GetDocument	string name, string option	ImplementDocument	業務ドキュメントを取得します。
	string name	ImplementDocument	業務ドキュメントを取得します。
ClearDocuments	なし	void	実装ドキュメントの設定をクリアします。

プロパティ

プロパティ名	型	意味	使い方
Id	string	業務アプリケーション ID	業務アプリケーション ID を取得します。
Name	string	業務アプリケーション名	業務アプリケーション名を取得します。
Documents	List<ImplementDocument>	実装ドキュメントのリスト	実装ドキュメントのリストを取得します。

◆ ImplementDocument クラス

ImplementDocument クラスは、業務アプリケーションが実際に実装した業務ドキュメントを指定するためのクラスです。このクラスによって定義されたものを実装ドキュメントと呼びます。実装アクション (ImplementAction)、実装イベント (ImplementEvent)、そして実装プロパティ (ImplementProperty) を複数設定することができます。

メソッド

メソッド名	引数	戻り値	使い方
AddAction	Document.ActionTypes type, ImplementAction.Roles role, int level	ImplementAction	実装アクションを追加します。 type: アクション種別 role: 役割 level: 実装レベル
AddProperty	string name, string title, bool use, bool typical, bool multi, string description	ImplementProperty	実装プロパティを追加します。 name: プロファイル名 title: 表示名 use: 必須区分 typical: 標準区分 multi: 複数区分 description: 説明
AddPropertyExtended	string name, string title, bool use, bool typical, Property.DataTypes dataType, string description	ImplementProperty	独自の実装プロパティを追加します。name: プロファイル名 title: 表示名 use: 必須区分 typical: 標準区分 dataType: データ型 description: 説明
AddPropertyLinked	string name, string document, string key, string title, bool use, bool typical, string description	ImplementProperty	連結した実装プロパティを追加します。 name: プロファイル名 document: 連結ドキュメント名 key: 連結キー title: 表示名 use: 必須区分 typical: 標準区分
GetProperty	string name	ImplementProperty	指定した名称の業務プロパティの実装定義を取得します。ユーザ定義プロパティの場合はプリフィックスをつけてください。
AddEvent	string name, string property,	ImplementEvent	実装イベントを追加します。

	ImplementEvent.EventTypes type, TimeSpan duration, string description		
GetEvent	string name	ImplementEvent	実装イベントを名前を指定して取得します。

プロパティ

プロパティ名	型	意味	使い方
Name	string	業務ドキュメント名	業務ドキュメント名を取得します。
Option	string	業務オプション名	業務オプション名を取得します。
ProfileStr	string	アプリケーション・プロファイル名	アプリケーションプロファイル名を設定または取得します。
Actions	List<ImplementAction>	実装アクションのリスト	実装アクションのリストを取得します。
Properties	List<ImplementProperty>	実装プロパティのリスト	実装プロパティのリストを取得します。
Events	List<ImplementEvent>	実装イベントのリスト	実装イベントのリストを取得します。
Definition	DocumentProfile	定義データ	属する業務ドキュメント定義を取得します。

◆ ImplementProperty クラス

ImplementProperty クラスは、実装ドキュメント (ImplementDocument) のオブジェクトがもつ業務プロパティを定義するためのクラスです。このクラスによって指定された業務オブジェクトをアプリケーション・プログラムは実装し、処理を行います。

プロパティ

プロパティ名	型	意味	使い方
DataType	Property.DataTypes	データ型	データ型(拡張プロパティの場合のみ有効)を設定または取得します。
Definition	PropertyProfile	定義データ	属する業務プロファイル定義を取得します。
Description	string	説明	説明を設定または取得します。
Extended	bool	拡張区分	拡張区分(拡張であるかどうか)を設定または取得します。
ExtendedName	string	業務プロパティ名	業務プロパティ名(プレフィックスを含む)を取得します。
Link	string	業務ドキュメントの連結情報	業務ドキュメントの連結情報を設定または取得します。
Multi	bool	複数区分	複数区分(複数型であるかどうか)を設定または取得します。
Name	string	業務プロパティ名	業務プロパティ名を設定または取得します。
Title	string	表示タイトル	表示タイトルを設定または取得します。
Typical	bool	標準区分	標準区分(標準的であるかどうか)を設定または取得します。
Use	bool	必須区分	必須区分(必須であるかどうか)を設定または取得します。

◆ ImplementAction クラス

ImplementAction クラスは、実装ドキュメントごとに、業務アプリケーションが対応可

能なアクションを定義するためのクラスです。アクションには、Add（追加）、Change（編集）、Remove（削除）、Get（照会）、Sync（イベントの要求）、そして Notify（通知）を設定可能です。これらは、サーバの場合に受取るメッセージ、クライアントの場合は送信するメッセージの分類となります。

プロパティ

プロパティ名	型	意味	使い方
Action	Document.ActionTypes	アクション種別	アクション種別を取得します。
Level	int	実装レベル	実装レベルを取得します。 0:未実装 1:部分実装 2:フル実装
Role	RoleTypes	役割	役割 (Server または Client) を取得します。

◆ ImplementEvent クラス

ImplementEvent クラスは、業務アプリケーション・プログラムが行うイベント通知サービスについて、個々の実装ドキュメントについて指定するためのクラスです。実装イベントの定義では、イベントの種類や、イベント判定のための条件などを指定します。

メソッド

メソッド名	引数	戻り値	使い方
CreateCondition	なし	Condition	業務オブジェクト選択情報を設定します。
	string id	Condition	業務オブジェクト選択情報を設定します。(ID 指定)
CreateSelection	なし	Selection	業務プロパティ選択情報を設定します。
	string name, object value	Selection	業務プロパティ選択情報を設定します。
SetConstraint	object value Property.DataTypes dataType Constraint.ConstraintTypes constraintType	Constraint	制約を追加します。

プロパティ

プロパティ名	型	意味	使い方
Conditions	List<Condition>	オブジェクトの選択条件	オブジェクトの選択条件を取得します。
Constraints	List<Constraint>	イベント判定式のリスト	イベント判定式のリストを取得します。
Cycle	TimeSpan	イベントサイクル	イベントサイクルを設定または取得します。
CycleSpecified	bool	イベントサイクルが定義されているか	イベントサイクルが定義されているかどうかを設定または取得します。
Description	string	イベントの内容説明	イベントの内容説明を設定または取得します。
Expire	DateTime	イベントの有効期限	イベントの有効期限を設定または取得します。
ExpireSpecified	bool	イベント有効期限が定義されているか	イベント有効期限が定義されているかどうかを設定または取得します。
Name	string	イベント名称	イベント名称を設定または取得します。
Property	string	プロパティ名	プロパティ名を設定または取得します。
Selections	List<Selection>	プロパティの選択条件	プロパティの選択条件を取得します。
Start	DateTime	イベント監視の開始日時	イベント監視の開始日時を設定または取得しま

			す。
StartSpecified	bool	イベント監視の開始日時が定義されているか	イベント監視の開始日時が定義されているかどうかを設定または取得します。
Type	EventTypes	イベントの種別	イベントの種別を設定または取得します。 True: 真の場合 False: 偽の場合 Enter: 偽から真となった場合 Leave: 真から偽となった場合 Change: 値が変更された場合 Add: オブジェクトまたはプロパティが追加された場合 Remove: オブジェクトまたはプロパティが削除された場合 Always: 監視が実行された都度

◆ PpsDocumentsException クラス

PpsDocumentException クラスは、PPS ドキュメントサービスのコンポーネント内で発生した例外を捕捉するために利用するクラスです。

プロパティ

プロパティ名	型	意味	使い方
Message	string	メッセージ	例外の内容を取得します。

3. メッセージングサービスモジュール

◆ TransportMessage クラス

TransportMessage クラスは、メッセージマネージャがメッセージキューシステムとの間で送受信するメッセージの単位となります。メッセージ本体と、トランスポート ID、そして送信者の情報によって構成されます。このクラスは、PPS ドキュメントサービスにおける TransactionMessage クラスに対応するものです。

コンストラクタ

メソッド名	引数	使い方
TransportMessage		コンストラクタ(これは利用しないこと)

メソッド

メソッド名	引数	戻り値	使い方
SaveMessage	string filename	void	メッセージの内容をテキストファイルに保存します
	なし	void	メッセージの内容をファイルに保存します。ファイル名は、ID をもとに自動設定される。

プロパティ

プロパティ名	型	意味	使い方
TransportId	string	トランスポートID	トランスポートID(送信と受信と対応づけるために利用)を設定または取得します。設定しない場合には、マネージャクラスが設定します。送信と受信と対応づけるために利用します。
SenderId	string	送信者ID	送信者の物理ID(往復の場合は送信者の受信キュー名となる)を設定または取得します。往復の場合は送信者の受信キュー名を設定または取得します。設定しない場合には、マネージャクラスが設定します。
ReplyTo	string	返信用キュー名称	このメッセージに対する返信メッセージを送る先のアドレスを設定または取得します。クライアントからサーバへ返信を要求する際に、ここにキュー名を設定します。
Message	string	メッセージ内容	メッセージ内容を設定または取得します。メッセージ本体となる文字列を設定または取得します。

◆ MessageManager クラス

MessageManager クラスは、メッセージの送受信を管理するクラスです。このクラスによって、メッセージの送信や、1つ以上のキューの監視などが実行されます。送信は、一時的にキューを利用するが、受信は、あらかじめキューごとに受信用のキュークラス (Me

ssageConnector) を生成し、それを利用してキューの監視を行います。

コンストラクタ

メソッド名	引数	使い方
MessageManager		メッセージマネージャを生成します。
	string configPath	メッセージマネージャを生成します。指定したファイルから設定ファイルを読み込みます。

メソッド

メソッド名	引数	戻り値	使い方
CreateMessage	string msg	TransportMessage	メッセージを生成します。送信用メッセージを生成します。
ResponseMessage	string msg TransportMessage receive	TransportMessage	受信に対する返信メッセージを送信します。
SendMessage	TransportMessage message, string requestQueueName	void	メッセージを送信する。メッセージを送信します。 message: 送信メッセージ requestQueueName:送信用キュー名
SendTextMessage	string messageString, string requestQueueName	void	メッセージを送信する。メッセージを送信します。 messageString: 送信メッセージ requestQueueName:送信用キュー名
	string messageString string requestQueueName string returnQueueName	void	メッセージを送信します。
ReceiveMessage	なし	TransportMessage	メッセージを受信キューから受信する。メッセージを指定キューから受信し、そのメッセージオブジェクトを返します。メッセージがない場合は null が返されます。
	string name	TransportMessage	メッセージを指定キューから受信する。サーバが受信メッセージに対応した返信メッセージを送信します。
ReceiveTextMessage	なし	string	メッセージを受信キューから受信します。メッセージを指定キューから受信しその内容を返す。メッセージがない場合は null が返されます。
	string name	string	メッセージを指定キューから受信します。
GetReceiver	string name		生成済みのメッセージ受信用のキューを得る。ない場合は null を返します。
CreateReceiver	なし	MessageConnector	メッセージ受信用のキューを生成します。キュー名称は ReplyTo になります。
	string name	MessageConnector	メッセージ受信用のキューを生成します。キューの名称を指定して、メッセージ受信用のキューを生成します。
SyncTextMessage	string messageString, string requestQueueName	string	同期形式でテキストメッセージを送信する同期形式でテキストメッセージを送信します。戻り値には、返信メッセージの内容が設定されます。
SyncMessage	TransportMessage sendMessage, string requestQueueName	TransportMessage	同期形式でテキストメッセージを送信する同期形式でメッセージを送信します。戻り値には、返信メッセージの内容が設定されます。
CloseAll	なし	void	受信用キューをすべて閉じます。受信用キューがある場合、それらをすべて閉じます。

プロパティ

プロパティ名	型	意味	使い方
Location	string	サーバのURLまたはキューマネージャ名	サーバのURLまたはキューマネージャ名を設定または取得します。デフォルト値として、"tcp://localhost:61616"が設定されています。MQ

			サーバを特定するための文字列であり各 MQ サーバの種類(メーカ)ごとに設定内容が異なります。この値は設定ファイルによって指定することも可能です。
ConnectorName	string	受信用キュー名称	管理するキューの名称を設定または取得します。デフォルトとして受信をチェックするキュー名称。クライアントの場合は、デフォルトの返信用キューとなります。
ConnectorEngineAssembly	string	コネクタエンジンのアセンブリ名	キューのメーカ提供実行ファイル名を取得します。設定は、定義ファイルにて行います。デフォルト値には、“PPS.MsgQueue.ActiveMQ.dll”が設定されています。キューのメーカ提供実行ファイル名を取得します。この値は、設定ファイルによって指定します。(取得のみ)
ConnectorEngineClassName	string	コネクタエンジンのクラス名	キューのメーカ提供クラス名を取得します。設定は、定義ファイルにて行います。デフォルト値には、“PPS.MsgQueue.ActiveMQ”が設定されています。キューのメーカ提供実行クラス名を取得します。この値は、設定ファイルによって指定します。(取得のみ)
SamplingInterval	int	ポーリングサイクル	ポーリングサイクルを設定または取得します。同期処理を行う場合(SyncMessage)の内部のキューをチェックする間隔
SyncTimeout	int	同期処理用タイムアウト時間	同期処理用タイムアウト時間を設定または取得します。同期処理を行う場合(SyncMessage)の返信の待ち最大待ち時間
SynchronizingObject	Control	同期処理の返信先コントロール	同期処理の返信先コントロールを設定または取得します。返信イベントを Windows フォーム上で処理する場合に、フォームクラスを継承したオブジェクトを設定します。(スレッドの切り替え処理用)
SenderId	string	送信者 ID	送信者 ID を設定または取得します。もし設定されない場合は、マネージャが自動でユニークな値を設定します。メッセージ送信者の ID であり、返信の場合の返信アドレスともなります。
Properties	NameValueCollection	設定値	各エンジン固有の設定値を取得または設定します。

◆ MessageConnector クラス

MessageConnector クラスは、メッセージの受信を行うために利用します。メッセージを監視するキューごとにマネージャオブジェクト (MessageManager) によって生成されます。

メソッド

メソッド名	引数	戻り値	使い方
Open	なし	void	受信キューの監視を開始します。
Close	なし	void	受信キューの監視を終了します。
GetMessage	なし	TransportMessage	受信キューからメッセージをひとつ取り出します。メッセージが存在しない場合は、null を返します。

プロパティ

プロパティ名	型	意味	使い方
--------	---	----	-----

IsConnected	bool	監視状態かどうか	エンジンがメッセージの受信監視を行っているかどうかを取得します。
Name	string	コネクタ名称	コネクタ名称を取得します。

イベント

メソッド名	引数	型	使い方
ReceiveMessage	MessageConnector sender, TransportMessage message	ReceiveMessageHandler	キューからメッセージを受け取ったときに発生します。このイベントに ReceiveMessageHandler をもちいてコールバック関数を設定すると、メッセージの到着時にその関数が呼ばれる。

◆ PpsMessagingException クラス

PpsMessagingException クラスは、PPS メッセージングサービスのコンポーネント内で発生した例外を捕捉するために利用するクラスです。

プロパティ

プロパティ名	型	意味	使い方
Message	string	メッセージ	例外の内容を取得します。

◆ IConnectorEngine インタフェース

IConnectorEngine は、メーカー別エンジンとのインタフェースです。次の仕様は、共通コンポーネントと各メーカー用エンジンとのインタフェースです。各メーカー用エンジンは、このインタフェースに合わせて開発する必要があります。

メソッド

メソッド名	引数	戻り値	使い方
GetMessage	なし	TransportMessage	キューにある先頭のメッセージを取り出します。
SendMessage	TransportMessage message	void	レスポンスキューとメッセージ ID を指定して、メッセージを送信します。
Open	なし	void	受信キューの監視を開始します。
Close	なし	void	受信キューの監視を終了します。

プロパティ

プロパティ名	型	意味	使い方
ServerName	string	サーバ名	サーバの名称を設定または取得します。
ConnectorName	string	キューの名称	キューの名称の設定または取得します。
Properties	NameValueCollection	設定値	エンジン固有の設定値を取得または設定します。
IsListening	bool	キューの状態	キューがリスン(監視)状態にあるかを取得します。

イベント

メソッド名	引数	型	使い方
ReceiveMessage	QueueReceiveArgs e	QueueReceiveHandler	キューからメッセージを受け取ったときに発生します。このイベントに ReceiveMessageHandler をもちいてコールバック関数を設定すると、メッセージの到着時にその関数が呼ばれます。

◆ QueueReceiveArgs クラス (イベント引数)

QueueReceiveArgs クラスは、コネクタエンジンがデータ転送エンジンからメッセージを受取るときのパラメータとなるクラスです。

コンストラクタ

コンストラクタ名	引数	使い方
QueueReceiveArgs	TransportMessage message	QueueReceiveArgs を生成します。

プロパティ

プロパティ名	型	意味	使い方
Message	TransportMessage	受け取ったメッセージ	受け取ったメッセージを表す TransportMessage