

Microsoft
Windows Workflow Foundationについて
～3.0から3.5に進化～

わんくま同盟 東京勉強会#21

2008/06/21

mxb



わんくま同盟 東京勉強会 #21

Microsoft Windows Workflow Foundation(WF)とは

- .NET Framework 3.0から搭載されたWindows プラットフォームでワークフローソリューションを開発するための拡張機能
- WFは、ワークフローベースのアプリケーションの開発および実行のためのAPIおよびツールを提供
- WFはMicrosoft社や他社のワークフロー製品と違い、.NET Frameworkの共通言語ランタイム(以降CLRと略します)上で稼働
- このため、.NET言語との親和性が高い
- .NET言語で作成したカスタムコンポーネントを組み込み、標準提供のコンポーネントと同様に使用することが出来る

WFで使用される用語は主に2つ

• ワークフロー

- ワークフローは、アクティビティのマップとして定義されたヒューマンプロセスまたはシステムプロセスのモデルです。
- ワークフローベースのプログラムは、XAML (eXtensible Application Markup Language) 文書内に指定されます。
- これは、ドメイン固有アクティビティの見地からプログラムの構造を指定する文書です。

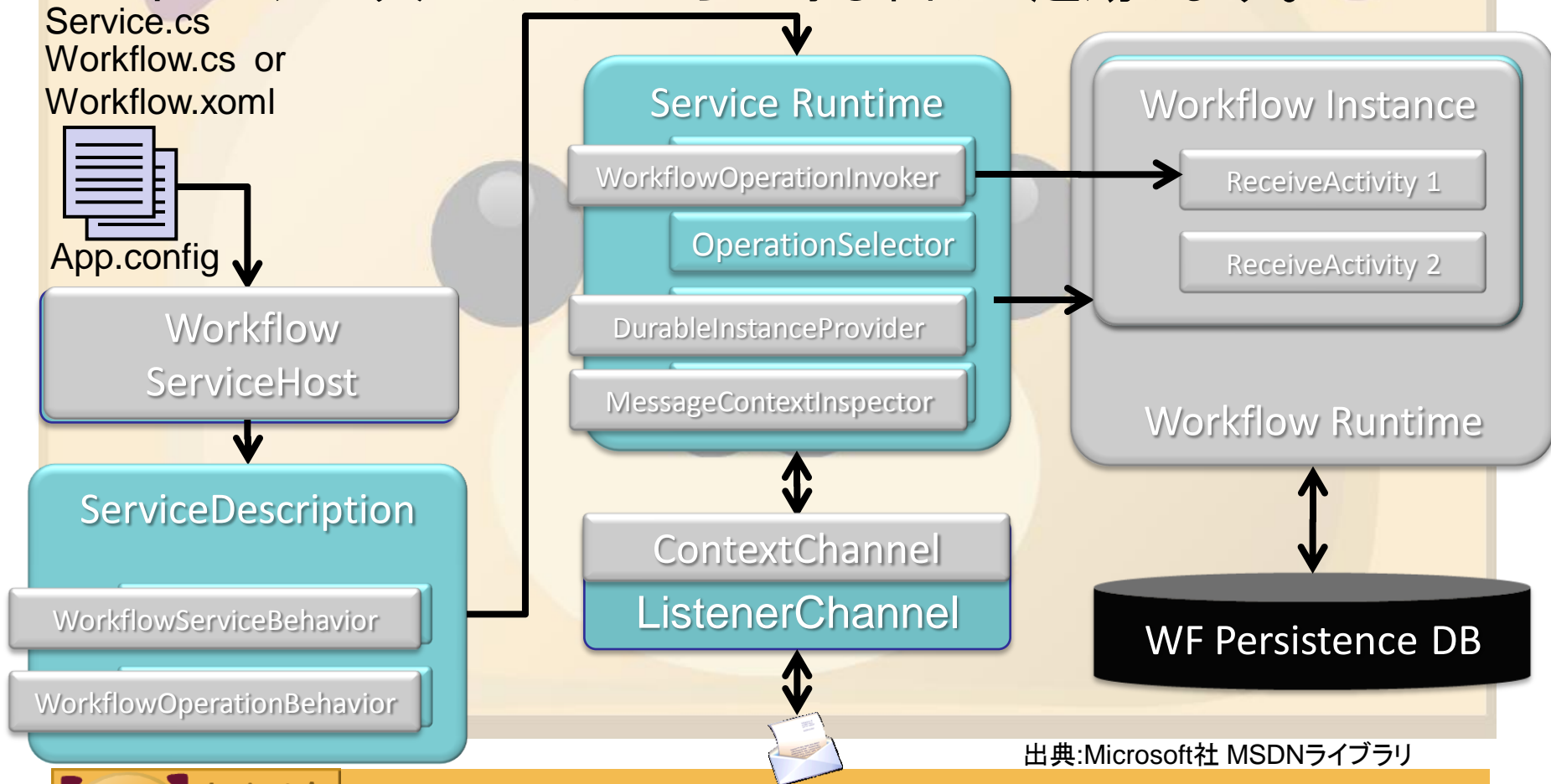
WFで使用される用語は主に2つ

• アクティビティ

- アクティビティは、ワークフローのステップであり、ワークフローの実行、再利用、および構成の単位です。
- アクティビティのマップは、ルール、アクション、状態、およびそれらの関係を表します。アクティビティを配置してデザインされたWFのワークフローは、.NETアセンブリにコンパイルされ、ワークフローランタイムおよびCLRで実行されます。
- アクティビティは、C#あるいはVisual Basicなどの従来のCLRベースのプログラミング言語で実装されます。

WFのアーキテクチャ

- WFはユーザインターフェイスを持っていません。そのため、ホストアプリケーションからの呼び出しで起動します。



出典:Microsoft社 MSDNライブラリ

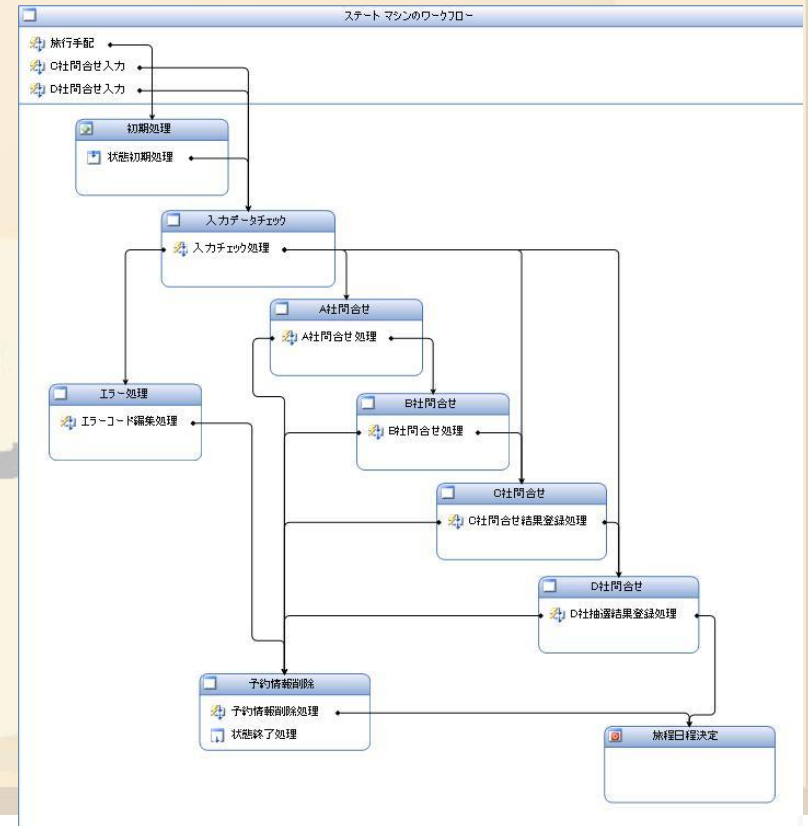
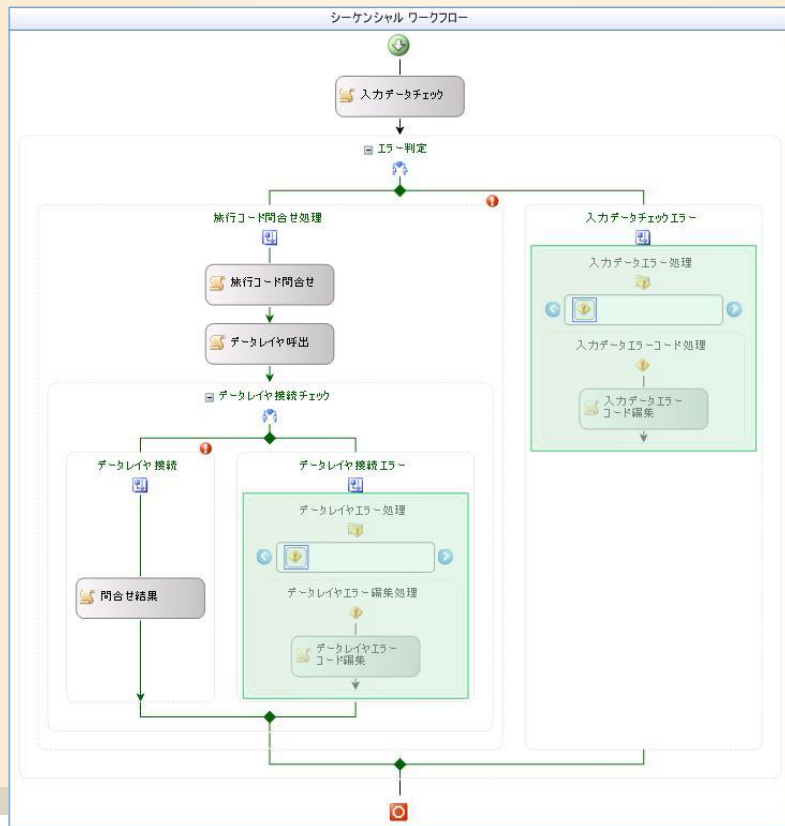


WFの機能

作れるワークフローは基本的に2種類

シーケンシャル・ワークフロー

ステートマシン・ワークフロー

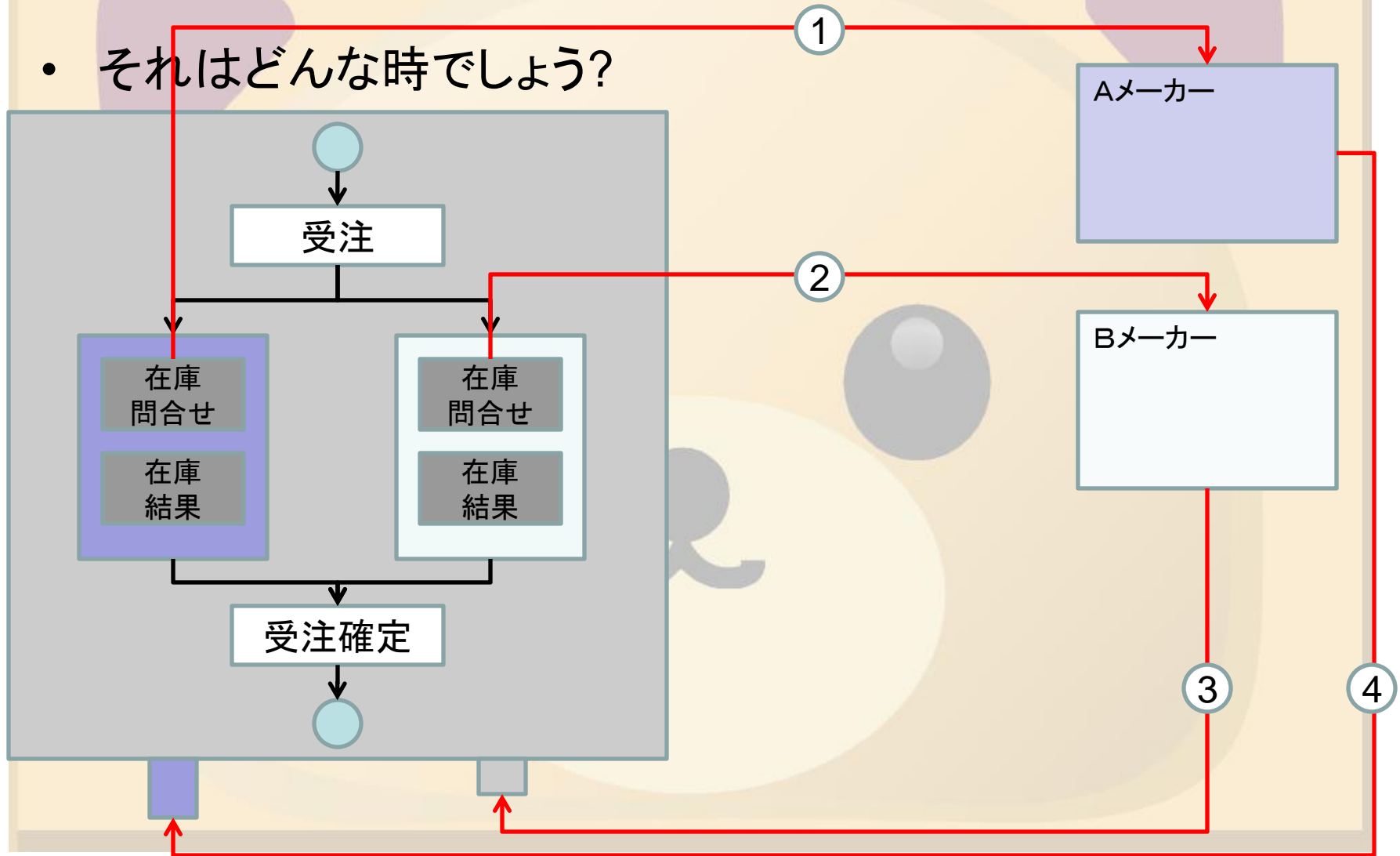


シーケンシャル・ワークフロー

- シーケンシャル・ワークフローは、ワークフローが開始されてから様々なステップ、条件式などが次々に実行され、最後のアクティビティが完了するまで途切れることなく続行される処理に適しています。
- しかし、シーケンシャル・ワークフローはその中で定義された外部接続、外部イベントからの接続、条件により複数の処理の同時実行などにより、内部に定義された実行順序が異なる場合があります。

内部に定義された実行順序が異なる？

- それはどんな時でしょう？



シーケンシャル・ワークフローで 使えるアクティビティ

クラス	説明
CallExternalMethodActivity	ローカル サービスでのメソッドの呼び出しに使用されるワークフロー通信アクティビティを定義します。このアクティビティは、ローカル サービスでワークフローからホストにデータを送信するために使用されます。
CodeActivity	ここにワークフロー内で実行するロジックを記述します。
CompensatableSequenceActivity	SequenceActivity アクティビティの補正可能バージョンを定義します。このクラスは継承できません。
CompensatableTransactionScopeActivity	TransactionScopeActivity アクティビティの補正可能バージョンを定義します。このクラスは継承できません。
ConditionedActivityGroup	制約に基づく実行コンテキストの定義を子アクティビティのセットに提供します。
DelayActivity	タイマを設置し、非同期的にタイマの期限切れを待機するロジックを提供します。このクラスは継承できません。
EventDrivenActivity	実行がイベントによって初期化される Activity をラップします。このクラスは継承できません。これは、イベントの処理に使用される CompositeActivity です。通常、イベントは、遅延タイマの期限切れに応答してホストまたはランタイムによって発生させることができます。EventDrivenActivity は、SequenceActivity から継承されるので、これは、最初のアクティビティを IEventActivity にする必要があるという追加の制約が加えられたシーケンスです。
EventHandlingScopeActivity	子アクティビティの実行と共にイベント処理を有効にします。このクラスは継承できません。
FaultHandlerActivity	FaultType プロパティで指定された型の Exception を操作するアクティビティ クラスを表します。このクラスは継承できません。



シーケンシャル・ワークフローで 使えるアクティビティ

クラス	説明
HandleExternalEventActivity	ローカル サービスによって発生するイベントの処理に使用されるワークフロー通信アクティビティを定義します。
IfElseActivity	条件に従って、IfElseBranchActivity 型の 2 つ以上のアクティビティの中から 1 つのアクティビティを実行します。このクラスは継承できません。
InvokeWebServiceActivity	プロキシ クラスを介して Web サービスを呼び出し、指定されたとおりにパラメータの引き渡しと受け取りを行います。このクラスは継承できません。
InvokeWorkflowActivity	ワークフローを非同期的に実行します。このクラスは継承できません。
<i>ListenActivity</i>	アクティビティを続行する前に発生する可能性があるイベントのいずれかを待機するようワークフローを設定します。このクラスは継承できません。
ParallelActivity	一連の子アクティビティを同時に実行します。このクラスは継承できません。
PolicyActivity	1 つのステップまたはアクティビティとしてワークフローの実行の一部で実行される Rule クラスのインスタンスのコレクションを表します。
ReplicatorActivity	子アクティビティの複数のインスタンスを実行します。
SequenceActivity	1 つの定義済み順序に従って、子アクティビティを実行します。
SynchronizationScopeActivity	ワークフローのうち、共有変数へのアクセスを制御する必要がある範囲を表します。 SynchronizationScopeActivity の複数のインスタンスが同じ変数にアクセスする場合、それぞれのアクティビティが同時にアクセスすることなく、順序立てて実行されます。このクラスは継承できません。

シーケンシャル・ワークフローで 使えるアクティビティ

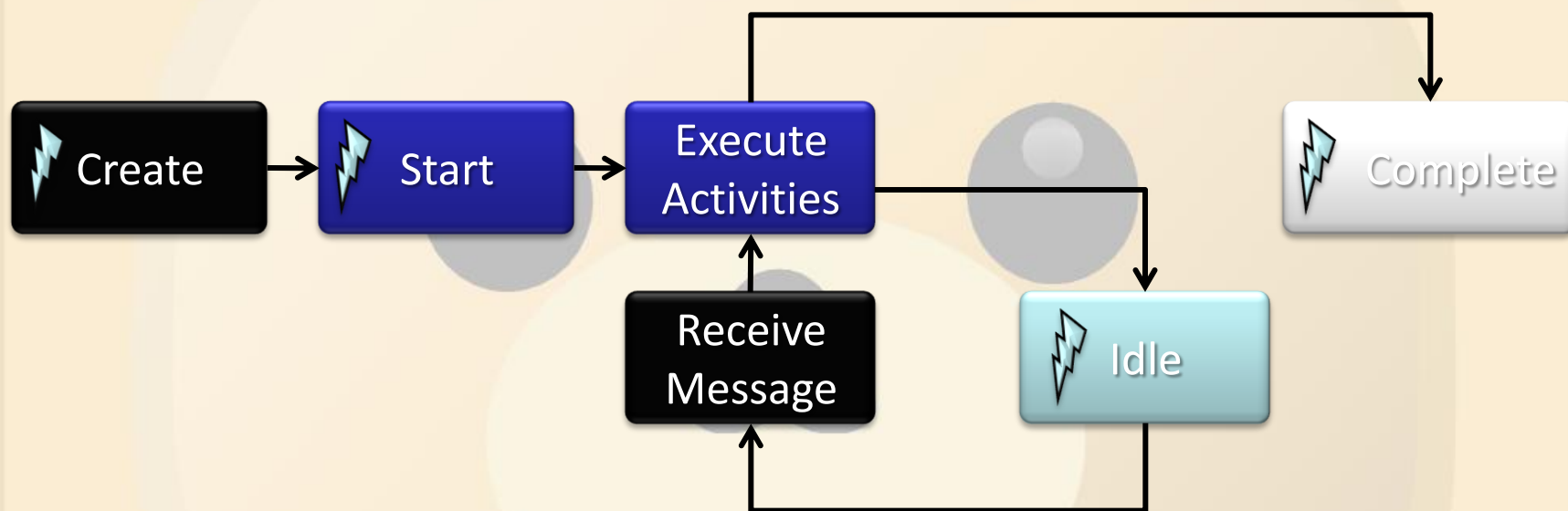
クラス	説明
TerminateActivity	異常状況が発生したとき、実行中のワークフロー インスタンスを終了します。このクラスは継承できません。
ThrowActivity	例外をスローする動作をモデル化します。このクラスは継承できません。
TransactionScopeActivity	ワークフローのうち、トランザクションとして一括処理する範囲。この Activity の実行が始まると新しい Transaction が起動され、Activity を正常に閉じるとトランザクションはコミットされます。このクラスは継承できません。
WebServiceFaultActivity	ワークフローから Web サービス クライアントへのエラーの送信を有効にします。このクラスは継承できません。
WebServiceInputActivity	ワークフローで Web サービスからのデータを受信できるようにします。このクラスは継承できません。
WebServiceOutputActivity	ワークフロー内部から Web サービスにデータを送信できるようにします。このクラスは継承できません。
WhileActivity	特定の条件が true である限り、1 つの子アクティビティを繰り返し実行します。

ステートマシン・ワークフロー

- ステートマシン・ワークフローは、一連の状態、遷移、および動作で構成されます。
- 1つの状態を開始状態とし、その後は、イベントや動作により、別の状態へと遷移していきます。(ただし、必ず別の状態へ遷移する必要は有りません)
- また、ステートマシン・ワークフローには、ワークフローの終わりを特定する最終状態を指定できます。

単純なステートマシン・ワークフロー

- 最も単純なステートマシン・ワークフロー

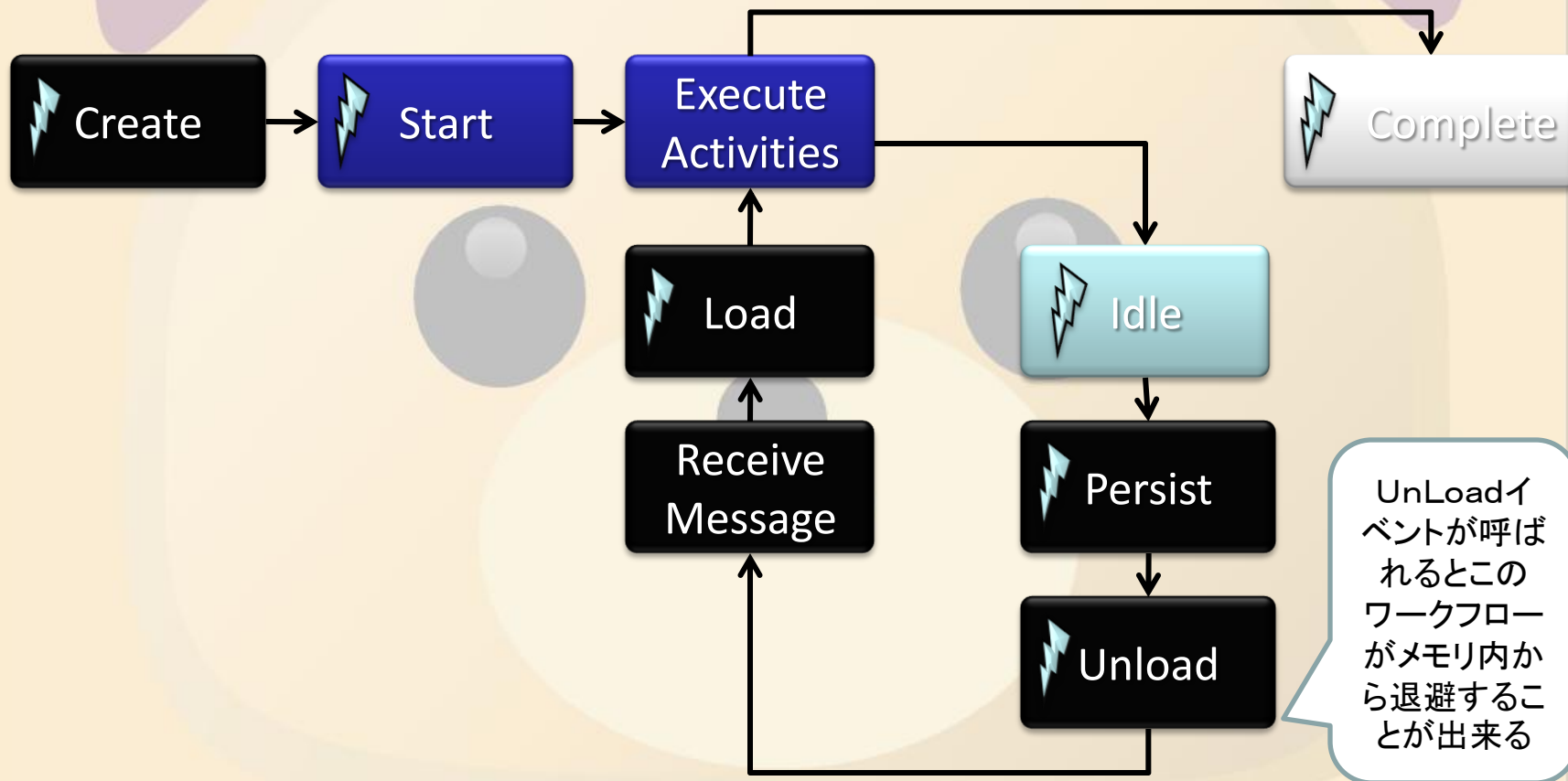


出典:Microsoft社 MSDNライブラリ



持続性のあるステートマシン・ワークフロー

- 持続性のあるステートマシン・ワークフロー



出典:Microsoft社 MSDNライブラリ

ステートマシン・ワークフローで 使えるアクティビティ

クラス	説明
CallExternalMethodActivity	ローカル サービスでのメソッドの呼び出しに使用されるワークフロー通信アクティビティを定義します。このアクティビティは、ローカル サービスでワークフローからホストにデータを送信するために使用されます。
<i>CompensateActivity</i>	ICompensatableActivity インターフェイスを実装する、処理が終了した子 Activity に対して、補正処理を起動します。このクラスは継承できません。
CodeActivity	ここにワークフロー内で実行するロジックを記述します。
CompensatableSequenceActivity	SequenceActivity アクティビティの補正可能バージョンを定義します。このクラスは継承できません。
CompensatableTransactionScopeActivity	TransactionScopeActivity アクティビティの補正可能バージョンを定義します。このクラスは継承できません。
ConditionedActivityGroup	制約に基づく実行コンテキストの定義を子アクティビティのセットに提供します。
DelayActivity	タイマを設置し、非同期的にタイマの期限切れを待機するロジックを提供します。このクラスは継承できません。
EventDrivenActivity	実行がイベントによって初期化される Activity をラップします。このクラスは継承できません。これは、イベントの処理に使用される CompositeActivity です。通常、イベントは、遅延タイマの期限切れに反応してホストまたはランタイムによって発生させることができます。EventDrivenActivity は、SequenceActivity から継承されるので、これは、最初のアクティビティを IEventActivity にする必要のあるという追加の制約が加えられたシーケンスです。
EventHandlingScopeActivity	子アクティビティの実行と共にイベント処理を有効にします。このクラスは継承できません。



ステートマシン・ワークフローで 使えるアクティビティ

クラス	説明
<code>FaultHandlerActivity</code>	<code>FaultType</code> プロパティで指定された型の <code>Exception</code> を操作するアクティビティ クラスを表します。このクラスは継承できません。
<code>HandleExternalEventActivity</code>	ローカル サービスによって発生するイベントの処理に使用されるワークフロー通信アクティビティを定義します。
<code>IfElseActivity</code>	条件に従って、 <code>IfElseBranchActivity</code> 型の 2 つ以上のアクティビティの中から 1 つのアクティビティを実行します。このクラスは継承できません。
<code>InvokeWebServiceActivity</code>	プロキシ クラスを介して Web サービスを呼び出し、指定されたとおりにパラメータの引き渡しと受け取りを行います。このクラスは継承できません。
<code>InvokeWorkflowActivity</code>	ワークフローを非同期的に実行します。このクラスは継承できません。
<code>ParallelActivity</code>	一連の子アクティビティを同時に実行します。このクラスは継承できません。
<code>PolicyActivity</code>	1 つのステップまたはアクティビティとしてワークフローの実行の一部で実行される <code>Rule</code> クラスのインスタンスのコレクションを表します。
<code>ReplicatorActivity</code>	子アクティビティの複数のインスタンスを実行します。
<code>SequenceActivity</code>	1 つの定義済み順序に従って、子アクティビティを実行します。
<code>SetStateActivity</code>	ステート マシン ワークフローの <code>StateActivity</code> への遷移を提供します。このクラスは継承できません。



ステートマシン・ワークフローで 使えるアクティビティ

クラス	説明
<i>StateActivity</i>	StateMachineWorkflowActivity の状態を表します。
<i>StateFinalizationActivity</i>	ステート マシン ワークフローの別の状態に遷移する前に、格納されているアクティビティを実行するアクティビティを表します。このクラスは継承できません。
<i>StateInitializationActivity</i>	StateActivity が実行を開始したときに実行されるアクティビティグループのコンテナとして機能します。このクラスは継承できません。
SuspendActivity	実行中のワークフロー インスタンスを中断します。このクラスは継承できません。
SynchronizationScopeActivity	ワークフローのうち、共有変数へのアクセスを制御する必要がある範囲を表します。SynchronizationScopeActivity の複数のインスタンスが同じ変数にアクセスする場合、それぞれのアクティビティが同時にアクセスすることなく、順序立てて実行されます。このクラスは継承できません。
TerminateActivity	異常状況が発生したとき、実行中のワークフロー インスタンスを終了します。このクラスは継承できません。
ThrowActivity	例外をスローする動作をモデル化します。このクラスは継承できません。
TransactionScopeActivity	ワークフローのうち、トランザクションとして一括処理する範囲。この Activity の実行が始まると新しい Transaction が起動され、Activity を正常に閉じるとトランザクションはコミットされます。このクラスは継承できません。
WebServiceFaultActivity	ワークフローから Web サービス クライアントへのエラーの送信を有効にします。このクラスは継承できません。

ステートマシン・ワークフローで 使えるアクティビティ

クラス	説明
WebServiceInputActivity	ワークフローで Web サービスからのデータを受信できるようにします。このクラスは継承できません。
WebServiceOutputActivity	ワークフロー内部から Web サービスにデータを送信できるようにします。このクラスは継承できません。
WhileActivity	特定の条件が true である限り、1 つの子アクティビティを繰り返し実行します。

WFは.NET Framework 3.5で どう変わったのか

- WFは2つのアクティビティが追加されただけです。
- しかし、この2つのアクティビティの追加は非常に大きな意味があります。

.NET Framework 3.5で追加されたアクティビティ

クラス	説明
ReceiveActivity	Windows Communication Foundation (WCF) サービス コントラクトによって定義された操作を実装するサービス側アクティビティ。
SendActivity	Windows Communication Foundation (WCF) サービス操作の同期呼び出しをモデル化するクライアント側アクティビティ。

ReceiveActivityとは

- WCF サービス コントラクトによって定義された操作を実装するサービス側アクティビティ。
- ReceiveActivity アクティビティは、サービス コントラクトで定義されている 1つの操作のみを実装します。
- つまり、これをワークフローに追加、設定するだけでワークフロー内でWCFからの呼び出しを待つことができます。
- ReceiveActivityが無かった.NET Framework 3.0の時代はWCFをラッピングしたカスタムアクティビティを作成し、ワークフローを作成する必要がありました。これには非常にコストがかかっていました。
- しかし、WCFの作成手順は変わっていないので、そのコストは見込んでおいてください。

The screenshot shows a workflow diagram with a 'codeActivity1' activity connected to an 'ExpenseReportReviewed' event. Below the diagram is the Properties window for 'receiveActivity1'.

プロパティ	
receiveActivity1 System.Workflow.Ac	
Handlers	
OperationValidat	
アクティビティ	
(Name)	receiveActivity1
CanCreateInstanc	False
ContextToken	
Description	
Enabled	True
FaultMessage	
ServiceOperation	ExpenseContracts.
パラメータ	
expenseReportR	

[ハンドラの生成、バインド可能なプロパティの具格、選択したプロパティのバインド\(BL\)、ReceiveActivityの表示、キャンセル、ハンドラの表示、エラー、ハンドラの表示、同じ操作の動作を表示します、同じコントラクトの動作を表示します](#)

Description
アクティビティの説明を指定してください。

SendActivityとは

- WCFサービス操作の同期呼び出しをモデル化するクライアント側アクティビティ。
- SendActivityアクティビティは、サービスコントラクトで定義されている 1つの操作のみを実装します。
- つまり、これをワークフローに追加、設定するだけでワークフロー内からWCFの呼び出すことができます。
- SendActivityが無かった.NET Framework 3.0の時代はWCFをラッピングしたカスタムアクティビティを作成し、ワークフローを作成する必要がありました。これには非常にコストがかかっていました。
- しかし、WCFの作成手順は変わっていないので、そのコストは見込んでおいてください。



Tech Ed 2008 North Americaでの WFの扱いは

- 非常に注目されています。
- WF+WCFは.NET デベロッパーだけでなく、ITプロの方も関心が高いです。
- なぜならば、
 - SharePoint Servicesにも搭載される。
 - BizTalk 2006 R3にも搭載される。
 - ...など

是非試してみてください。

- サンプルや資料が多く提供されています。

参考URL

Windows Vista 技術資料一覧

<http://www.microsoft.com/japan/msdn/windowsvista/techarts.aspx>

Windows Workflow Foundation と Windows Communication Foundation の統合

<http://www.microsoft.com/japan/msdn/net/wf/bb266709.aspx>

Windows Workflow Foundation integration with Windows Communication Foundation

<http://msdn.microsoft.com/en-us/library/cc626077.aspx>

ワークフロー サービスと永続性サービスの作成

<http://msdn.microsoft.com/ja-jp/library/bb412181.aspx>

Workflow Services Samples (WF)

<http://msdn.microsoft.com/en-us/library/bb943473.aspx>

Creating Workflow Services and Durable Services

<http://msdn.microsoft.com/en-us/library/bb412181.aspx>

