



Red Hat Process Automation Manager 7.8

プロセスデザイナー「Business Process Model
and Notation (BPMN2)」のリファレンスガイド

Red Hat Process Automation Manager 7.8 プロセスデザイナー「Business Process Model and Notation (BPMN2)」のリファレンスガイド

Red Hat Customer Content Services
brms-docs@redhat.com

法律上の通知

Copyright © 2020 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

概要

本書は、Red Hat Process Automation Manager 7.8 のレガシーおよび新規プロセスデザイナー向けの BPMN2 リファレンスガイドです。

目次

前書き	3
第1章 BUSINESS PROCESS MODELING AND NOTATION バージョン 2.0	4
第2章 BPMN2 向けの RED HAT PROCESS AUTOMATION MANAGER サポート	6
第3章 プロセスデザイナーでの BPMN2 イベント	11
3.1. 開始イベント	11
3.2. 中間イベント	14
3.3. 終了イベント	17
第4章 プロセスデザイナーでの BPMN2 タスク	20
第5章 プロセスデザイナー内の BPMN2 サブプロセス	24
第6章 プロセスデザイナーでの BPMN2 ゲートウェイ	28
第7章 プロセスデザイナーでの BPMN2 接続オブジェクト	31
第8章 プロセスデザイナーでの BPMN2 スイムレーン	32
第9章 プロセスデザイナーでの BPMN2 アーティファクト	33
付録A バージョン情報	34

前書き

本書では、Red Hat Process Automation Manager 7.8 のレガシーおよび新規プロセスデザイナーに含まれる BPMN2 要素について説明します。プロセスマネージャーは、Business Central でのビジネスまたはケース管理プロセスの設計時に、本書を参照して BPMN2 シンボルや、用途の詳細を確認できます。BPMN2 に関する詳細は、[Business Process Model and Notation Version 2.0](#) 仕様を参照してください。



注記

レガシーのプロセスデザイナーは廃止されました。このデザイナーは更新されず、次のリリースで無効になります。

レガシーのプロセスデザイナーで作成したプロセスを新規プロセスデザイナーにインポートした場合には、送受信および手動のタスクは、空のタスクに変換されて、グループ要素やデータオブジェクト要素は削除されます。

第1章 BUSINESS PROCESS MODELING AND NOTATION バージョン 2.0

Business Process Modeling and Notation バージョン 2.0 (BPMN2) 仕様は、ビジネスプロセスを描画表現するための標準や要素の実行セマンティクスの定義、XML 形式でのプロセス定義を行うときに使用する Object Management Group (OMG) 仕様です。

プロセスは、そのプロセス定義によって定義または決定されます。プロセスはナレッジベースに存在し、ID で識別されます。

表1.1 一般的なプロセスプロパティー

ラベル	説明
Name	プロセスの名前を入力します。
Documentation	プロセスについて記述します。このフィールドのテキストはプロセスドキュメントに含まれます (該当する場合)。
ID	このプロセスの識別子 (orderIdItems など) を入力します。
Package	Red Hat Process Automation Manager プロジェクトにおけるこのプロセスのパッケージの場所を入力します (例: org.acme)。
ProcessType	プロセスがパブリックかプライベートかを指定します (現在サポートされていません)。
Version	プロセスのアーティファクトバージョンを入力します。
Ad hoc	このプロセスがアドホックサブプロセスである場合は、このオプションを選択します。
Process Instance Description	プロセスの目的の説明を入力します。
Imports	クリックしてインポート ウィンドウを開き、プロセスに必要なデータタイプクラスを追加します。
Executable	このオプションを選択して、プロセスを Red Hat Process Automation Manager プロジェクトの実行可能な部分にします。
SLA Due Date	サービスレベルアグリーメント (SLA) の有効期限の日付を入力します。
Process Variables	プロセスのプロセス変数を追加します。プロセス変数は、特定のプロセスインスタンス内で表示されます。プロセス変数はプロセスの作成時に初期化され、プロセスの完了時に破棄されます。変数タグは、変数の動作をより詳細に制御します。たとえば、変数が required または readonly としてタグ付けされているかどうかなどを細かく制御します。変数タグの詳細は『 Business Central でのビジネスプロセスの設計 』を参照してください。

ラベル	説明
Metadata Attributes	メタデータ属性が存在する場合に何らかのアクションを実装するリスナーなど、カスタムイベントリスナーに使用するカスタムメタデータ属性の名前と値を追加します。
Global Variables	プロセスにグローバル変数を追加します。グローバル変数は、プロジェクトのすべてのプロセスインスタンスとアセットに表示されます。グローバル変数は通常、ビジネスルールおよび制約によって使用され、ルールまたは制約によって動的に作成されます。

プロセスは、モデル化した要素のセットを格納するコンテナの役割を果たします。プロセスには、フローオブジェクトおよびフローを使用してビジネスプロセスの実行ワークフローまたはパーツを指定する要素が含まれます。プロセスごとに、独自の BPMN2 のダイアグラムが割り当てられます。Red Hat Process Automation Manager には、BPMN2 ダイアグラムを作成するための新規プロセスデザイナーのほか、**.bpmn2** 拡張を使用して以前の BPMN2 ダイアグラムを開くレガシープロセスデザイナーが含まれます。新規プロセスデザイナーでは、レイアウトと機能セットが向上され、今後も開発が続けられる予定です。デフォルトでは、新規ダイアグラムは新規プロセスデザイナーで作成されます。

第2章 BPMN2 向けの RED HAT PROCESS AUTOMATION MANAGER サポート

Red Hat Process Automation Manager では、BPMN 2.0 標準を使用して、ビジネスプロセスのモデル化が可能です。Red Hat Process Automation Manager を使用してこれらのビジネスプロセスを実行、管理、監視することができます。包括的な BPMN 2.0 仕様には、コレオグラフィーやコラボレーションなどの項目の表現方法に関する詳細が含まれます。ただし、Red Hat Process Automation Manager は、実行可能なプロセスの指定に使用可能な仕様の一部のみを使用します。これには、BPMN2 仕様の共通の実行可能なサブクラスに定義されているほぼすべての要素および属性だけでなく、追加の要素や属性も含まれます。

以下の表は、BPMN2 要素がレガシーのプロセスデザイナーでサポートされているか、レガシーおよび新規プロセスデザイナーでサポートされているか、またはサポートされていないかを示すアイコン一覧を示しています。

表2.1 サポートの状態を示すアイコン

キー	説明
	レガシーおよび新規プロセスデザイナーでのサポート
	レガシーのプロセスデザイナーでのみサポート
	サポートなし

アイコンのない要素は、BPMN2 仕様には存在しません。

表2.2 BPMN2 の Catch イベント

要素名	始端	中間
なし		
メッセージ		
タイマー		










要素名	始端	中間
エラー		
エスカレーション		
取り消し		
補正		
条件		
リンク		
シグナル		
多重		
並列多重		

表2.3 BPMN2 送出および中断なしイベント

要素名	送出		中断なし	
	終端	中間	始端	中間

要素名	送出	中断なし	中断なし	中断なし
なし				
メッセージ				
タイマー				
エラー				
エスカレーション				
取り消し				
補正				
条件				
リンク				
シグナル				
中断				
多重				

要素名	送出	中断なし
並列多重		

表2.4 BPMN2 要素

要素タイプ	要素	対応
タスク	ビジネスルール	
	スクリプト	
	ユーザータスク	
	サービスタスク	
複数のインスタンスサブプロセスを含むサブプロセス	埋め込み	
	アドホック	
	再利用可能	
	イベント	
ゲートウェイ	包含的	
	排他的	

要素タイプ	要素	対応
	並列	
	イベントベース	
	コンプレックス	
接続オブジェクト	シーケンスフロー	
	関連フロー	
スイムレーン	スイムレーン	
アーティファクト	グループ	
	テキストのアノテーション	

BPMN2 の背景およびアプリケーションに関する詳細は、[OMG Business Process Model and Notation \(BPMN\) Version 2.0](#) を参照してください。

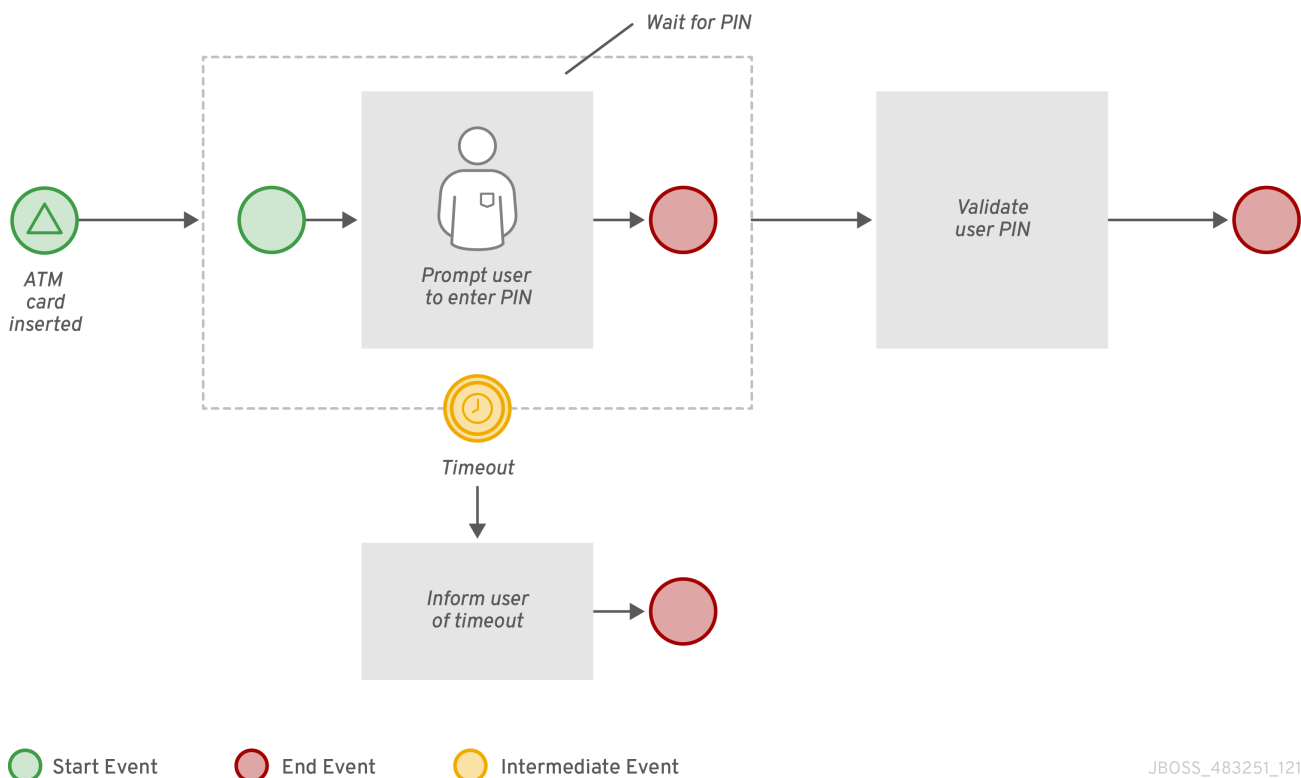
第3章 プロセスデザイナーでの BPMN2 イベント

イベントとは、ビジネスプロセスに発生する内容のことです。BPMN2 では、3 つのイベントカテゴリーをサポートします。

- 始端
- 終端
- 中間

開始イベントはイベントトリガーをキャッチし、終了イベントはイベントトリガーをスローします。中間イベントは、イベントトリガーをキャッチおよびスローできます。

以下のビジネスプロセスダイアグラムでは、イベントの例を紹介します。



この例では、以下のイベントが発生します。

- シグナルを受信すると、ATM カード挿入のシグナル開始イベントがトリガーされます。
- タイムアウトの中間イベントは、タイマートリガーをもとにした割り込みイベントです。つまり、タイマーイベントがトリガーされると、暗証番号待ちのサブプロセスがキャンセルされるという意味です。
- プロセスへの入力に応じて、Validate User Pin タスクに関連付けられた終了イベント、または Inform User of Timeout タスクに関連付けられた終了イベントが、プロセスを終了します。









3.1. 開始イベント

開始イベントを使用して、ビジネスプロセスの始端を示します。開始イベントには、受信シーケンスフローを割り当てることができず、外向きシーケンスフローだけを割り当てる必要があります。「なし (none)」開始イベントは、トップレベルプロセス、埋め込みサブプロセス、呼び出し可能サブプロセス、イベントサブプロセスで使用できます。

「なし (none)」開始イベントの例外を除けば、開始イベントはすべてキャッチイベントです。たとえば、シグナルの開始イベントは、参照のシグナル (イベントトリガー) を受信した場合にのみプロセスを開始します。イベントサブプロセスの開始イベントを割り込みまたは割り込みなしイベントに設定できます。イベントサブプロセスに対する割り込みありの開始イベントでは、包含プロセスまたは親プロセスの実行を停止または中断します。割り込みなしの開始イベントは、包含プロセスまたは親プロセスの実行を停止したり、中断したりしません。

表3.1 開始イベント

開始イベントタイプ	トップレベル	サブプロセス	
		割り込み	割り込みなし
なし			
条件			
補正			
エラー			
エスカレーション			

開始イベントタイプ	トップレベル	サブプロセス	
メッセージ			
シグナル			
タイマー			

なし

「なし」開始イベントは、トリガー条件のない開始イベントです。プロセスまたはサブプロセスには、「なし」開始イベントを最大1つ含めることができます。このイベントは、デフォルトでプロセスまたはサブプロセスの開始によりトリガーされ、外向きフローがすぐに実行されます。

サブプロセスで「なし」開始イベントを使用すると、親プロセスからサブプロセスに、プロセスフローの実行が移動し、「なし」開始イベントがトリガーされます。これは、トークン（プロセスフローの内の現在の場所）が親プロセスからサブプロセスのアクティビティに渡され、サブプロセスの「なし」開始イベントが独自のトークンを生成します。

条件

条件付きの開始イベントは、ブール型の条件定義を含む開始イベントです。条件が最初に **false** と評価され、次に **true** に評価された場合に実行がトリガーされます。プロセスの実行は、開始イベントがインスタンス化された後に条件が **true** と評価された場合にのみ開始されます。

プロセスには、複数の条件開始イベントを含めることができます。

補正

補正開始イベントは、補正中間イベントのターゲットアクティビティとして、サブプロセスを使用した場合に、補正イベントのサブプロセスを開始するのに使用します。

エラー

プロセスまたはサブプロセスには、複数のエラー開始イベントを含めることができます。特定の **ErrorRef** プロパティを含むエラーオブジェクトを受信した場合に、この開始イベントがトリガーされます。エラーオブジェクトは、エラーの終了イベントで生成可能です。これは、プロセスの終端が不正であることを示します。エラーの開始イベントが含まれるプロセスインスタンスは、該当するエラーオブジェクトの受信後に実行が開始されます。エラー開始イベントは、エラーオブジェクトの受信直後に実行されて、外向きフローが実行されます。

エスカレーション

エスカレーション開始イベントは、特定のエスカレーションコードを含むエスカレーションによりトリガーされる開始イベントです。プロセスには、複数のエスカレーション開始イベントを含めることができます。エスカレーション開始イベントが含まれるプロセスインスタンスは、定義されているエスカレーションオブジェクトを受信すると、実行を開始します。プロセスがインスタンス化され、直後にエスカレーション開始イベントが実行され、外向きフローが実行されます。

メッセージ

プロセスまたはイベントのサブプロセスには、複数の「メッセージ」開始イベントを含めることができます。これらのイベントは、通常特定のメッセージによりトリガーされます。「メッセージ」開始イベントが含まれるプロセスインスタンスの実行は、該当のメッセージを受信した後に、このイベントからのみ開始されます。メッセージの受信後に、プロセスはインスタンス化され、「メッセージ」開始イベントが即座に実行されます (外向きフローが実行されます)。

メッセージは、要素なしなど、任意の数のプロセスおよびプロセス要素により消費可能であるため、1つのメッセージで複数の「メッセージ」開始イベントをトリガーできるので、複数のプロセスがインスタンス化されます。

シグナル

シグナル開始イベントは、特定のシグナルコードを含むシグナルによりトリガーされます。プロセスには、複数のシグナル開始イベントを含めることができます。シグナル開始イベントは、インスタンスが該当のシグナルを受信した後にのみ、プロセスインスタンス内で実行を開始します。その後に、シグナル開始イベントが実行され、外向きフローが実行されます。

タイマー

タイマー開始イベントは、タイミングのメカニズムを含む開始イベントです。プロセスには、複数のタイマー開始イベントを含めることができます。タイマー開始イベントは、タイミングのメカニズムが適用された後に、プロセスの開始時にトリガーされます。

サブプロセスでタイマー開始イベントを使用すると、プロセスフローの実行が親プロセスからサブプロセスに移動し、タイマー開始イベントがトリガーされます。親サブプロセスアクティビティからトークンを取得し、サブプロセスのタイマー開始イベントがトリガーされ、タイマーがトリガーされるまで待機します。タイミングの定義で指定した時間が経過したら、外向きフローが実行されます。

3.2. 中間イベント

中間イベントは、ビジネスプロセスのフローを駆動します。中間イベントは、ビジネスプロセスの実行中にイベントをキャッチまたはスローするときに使用します。中間イベントは、開始イベントと終了イベントの間に配置され、サブプロセス、人間のタスクなどのアクティビティの境界にキャッチイベントとして使用することもできます。境界キャッチイベントは、割り込みまたは割り込みなしとして設定できます。割り込みありの境界キャッチイベントは、バインドされているアクティビティを取り消しますが、割り込みなしのイベントは取り消しません。







中間イベントは、プロセス実行時に発生する特定の状況処理します。このような状況が中間イベントのトリガーになります。プロセスには、外向きフローを1つ含む中間イベントを、アクティビティの境界に配置できます。

アクティビティの実行時にイベントが発生した場合には、このイベントにより、外向きフローへの実行がトリガーされます。1つのアクティビティに、複数の境界中間イベントが含まれる可能性があります。境界中間イベントで、アクティビティから必要な動作によって、以下のいずれかの中間イベントタイプを使用できる点に注意してください。

- 割り込みあり: アクティビティの実行は中断され、中間イベントの実行がトリガーされます。
- 割り込みなし: 中間イベントがトリガーされ、アクティビティの実行が続行されます。

表3.2 中間イベント

中間イベントタイプ	キャッチ	境界		送出
		割り込み	割り込みなし	
メッセージ				
タイマー				
エラー				
シグナル				
条件				

中間イベントタイプ	キャッチ	境界	送出
補正			
エスカレーション			

メッセージ

メッセージの中間イベントは、メッセージオブジェクトを管理可能にする中間イベントです。以下のイベントのいずれかを使用します。

- スローメッセージの中間イベントでは、定義したプロパティーをもとにメッセージオブジェクトを作成します。
- キャッチメッセージの中間イベントは、定義したプロパティーを使用してメッセージオブジェクトがないかリッスンします。

タイマー

タイマー中間イベントでは、ワークフローの実行を遅延させたり、定期的にトリガーしたりできます。このイベントは、指定した期間が経過したら1回または複数回、トリガーできるタイマーを表します。タイマー中間イベントがトリガーされたら、タイマー条件(定義した時間)がチェックされ、外向きフローが実行されます。タイマー中間イベントがプロセスワークフローに配置されている場合には、内向きフローが1つと、外向きフローが1つ含まれます。内向きフローがイベントに移動すると、これが実行されます。タイマー中間イベントがアクティビティー境界に配置されている場合には、アクティビティーの実行と同時に、この実行がトリガーされます。

包含のプロセスインスタンスを完了するか、中断するなど、タイマー要素がキャンセルされると、タイマーがキャンセルされます。

条件

条件の中間イベントは、ブール型の条件がトリガーとして含まれる中間イベントです。このイベントは、条件で **true** と判断され、外向きフローが実行された場合に、さらにワークフロー実行をトリガーします。

このイベントは、**Expression** プロパティーを定義する必要があります。条件の中間イベントがプロセスワークフローに配置されている場合は、内向きフロー1つ、外向きフロー1つが含まれ、内向きフローがイベントに移動したときに、実行が開始されます。条件の中間イベントがアクティビティー境界

に配置されている場合には、アクティビティーの実行時に、この実行がトリガーされます。イベントが割り込みなしの場合には、条件が **true** の場合は継続して、このイベントがトリガーされます。

シグナル

シグナルの中間イベントでは、シグナルオブジェクトを生成または消費できます。以下のオプションのいずれかを使用してください。

- スローシグナルの中間イベントは、定義したプロパティーをもとにシグナルオブジェクトを生成します。
- キャッチシグナルの中間イベントは、定義したプロパティーを使用してシグナルオブジェクトがないかリッスンします。

エラー

エラーの中間イベントは、アクティビティー境界でのみ使用可能な中間イベントです。このイベントでは、プロセスが、該当するアクティビティー内のエラー終了イベントに反応できるようになります。このアクティビティーは、アトミックであってはいけません。アクティビティーが、エラー終了イベントで完了し、対応の **ErrorCode** プロパティーでエラーオブジェクトを生成した場合には、エラーの中間イベントがこのエラーオブジェクトをキャッチして、実行が外向きフローに進みます。

補正

補正中間イベントは、トランザクションサブプロセスのアクティビティーに接続されている境界イベントです。補正終了イベントまたはキャンセル終了イベントで、このイベントを終了できます。補正中間イベントは、補正アクティビティーに接続されているフローと関連付ける必要があります。

境界補正の仲介イベントに関連付けられているアクティビティーは、トランザクションサブプロセスが補正終了イベントで終了した場合に実行されます。この実行は、対応のフローで続行されます。

エスカレーション

エスカレーション中間イベントは、エスカレーションオブジェクトを生成または消費できる中間イベントです。イベント要素が実行すべきアクションに合わせて、以下のオプションのいずれかを使用する必要があります。

- スローエスカレーションの中間イベントは、定義したプロパティーをもとにエスカレーションオブジェクトを生成します。
- キャッチエスカレーションの中間イベントは、定義したプロパティーを使用してエスカレーションオブジェクトがないかリッスンします。

3.3. 終了イベント

終了イベントは、ビジネスプロセスの終了に使用します。終了イベントには、外向きのシーケンスフローが何も含まれない場合があります。また、ビジネスプロセスに複数の終了イベントが存在する場合があります。「なし」終了イベントおよび中断終了イベント以外の終了イベントはすべてスローイベントです。

終了イベントは、ビジネスプロセスの完了を示します。終了イベントは、特定のワークフローを終了するノードです。このイベントには、1つまたは複数の内向きシーケンスフローがあり、外向きフローはありません。

プロセスには最低でも1つの終了イベントが含まれている必要があります。

ランタイム中は、終了イベントでプロセスワークフローを終了します。終了イベントは、そのイベントに到達したワークフローのみ終了できます。終了イベントタイプによってはプロセスインスタンス内の全ワークフローを終了できます。

表3.3 終了イベント

終了イベント	アイコン
なし	
メッセージ	
シグナル	
エラー	
補正	
エスカレーション	
中断	

なし

「なし」終了イベントは、他に特別な動作がプロセスの終端に関連付けられていないことを示します。

メッセージ

フローがメッセージの終了イベントに入ると、このフローは終了し、終了イベントがプロパティーに定義されているようにメッセージを生成します。

シグナル

スローシグナルの終了イベントは、プロセスまたはサブプロセスフローの終了に使用します。実行フローがこの要素に入ると、実行フローが終了し、**SignalRef** プロパティーで特定されたシグナルを生成します。

エラー

スローエラーの終了イベントは、内向きワークフローを完了します。つまり、内向きのトークンを消費し、エラーオブジェクトを生成します。プロセスまたはサブプロセスで他に実行されているワークフローは、影響を受けません。

補正

補正終了イベントは、トランザクションのサブプロセスを終了し、サブプロセスアクティビティーに接続されている補正中間イベントで定義した補正をトリガーするのに使用します。

エスカレーション

エスカレーション終了イベントは、内向きワークフローを終了します。これは、内向きのトークンを消費して、プロパティーに定義されているようにエスカレーションシグナルを生成し、エスカレーションプロセスをトリガーします。

中断

中断終了イベントは、指定したプロセスインスタンス内の全実行フローを終了します。実行中のアクティビティーはキャンセルされます。サブプロセスインスタンスは、中断終了イベントに到達した場合は中断されます。

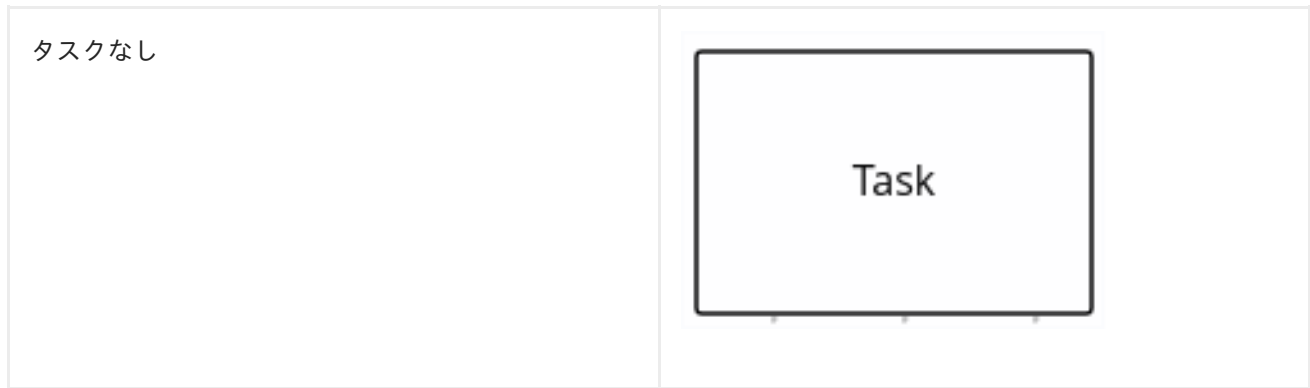
第4章 プロセスデザイナーでの BPMN2 タスク

タスクは、プロセスモデルに定義されている自動アクティビティーで、プロセスフロー内で最小の作業単位です。BPMN 2 仕様に定義されているタスクタイプで、Red Hat Process Automation Manager のプロセスデザイナーパレットで利用できるのは以下のとおりです。

- ビジネスルールタスク
- スクリプトタスク
- ユーザータスク
- サービスタスク
- タスクなし

表4.1 タスク

ビジネスルールタスク	 Task
スクリプトタスク	 Task
ユーザータスク	 Task
サービスタスク	 Service Task

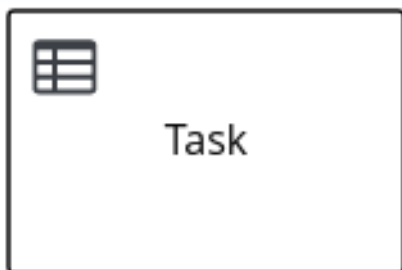


さらに、BPMN2 仕様では、カスタムタスクの作成が可能になります。Red Hat Process Automation Manager には、以下の事前定義済みのカスタムタスクが含まれます。

- REST サービスタスク: リモートの RESTful サービスの呼び出しに使用します。
- メールサービスタスク: メールの送信に使用します。
- ログサービスタスク: メッセージのログ記録に使用します。
- Java サービスタスク: Java コードの呼び出しに使用します。
- WebService サービスタスク: リモートの WebService コールの呼び出しに使用します。
- DecisionTask タスク: DMN ダイアグラム実行に使用します。

ビジネスルールタスク

ビジネスルールタスクは、DMN モデルまたはルールフローグループを使用して、意思決定を行う方法を定義します。

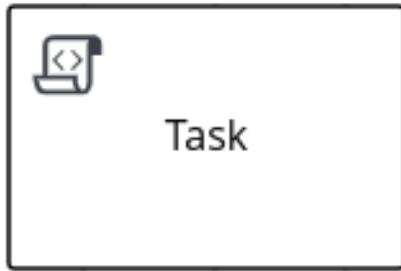


プロセスが DMN モデルで定義したビジネスルールタスクに到達したら、プロセスエンジンが、入力された内容を使用して DMN モデルを実行します。

プロセスがルールフローグループで定義したビジネスルールタスクに到達したら、プロセスエンジンは、定義済みのルールフローグループでルールを実行開始します。ルールフローグループにアクティブなルールがない場合には、実行は次の要素に移動します。ルールフローグループの実行中は、アクティブなルールフローグループに所属するアクティベーションは、他のルールで変更されるので、新たにアジェンダに追加できます。

スクリプトタスク

スクリプトタスクは、プロセス実行中に実行されるスクリプトを表します。



関連付けられたスクリプトは、プロセス変数やグローバル変数にアクセスできます。スクリプトタスクを使用する前に以下の一覧をレビューしてください。

- プロセスでは詳細にわたる実装の内容は回避してください。スクリプトタスクは、変数の操作に使用できますが、より複雑な操作をモデル化する場合にサービスタスクの使用を検討してください。
- 即座にスクリプトを実行するようにしてください。すぐに実行しない場合には、非同期サービスタスクを使用してください。
- スクリプトタスクを使用して外部のサービスの問い合わせを回避してください。サービスタスクを使用して、外部サービスとの通信をモデル化します。
- スクリプトで例外がスローされないようにしてください。ランタイムの例外はスクリプト内などで、キャッチし、管理するか、プロセス内で処理できるシグナルまたはエラーに変換する必要があります。

実行中にスクリプトタスクに到達したら、スクリプトが実行され、外向きフローに移動します。

ユーザータスク

ユーザータスクは、システムで自動的に実行できないプロセスワークフローに含まれるタスクなので、ユーザー(人間)、つまり、アクターの介入が必要です。



実行時に、ユーザータスク要素は、1つ以上のアクターのタスク一覧に表示されるタスクとしてインスタンス化されます。ユーザータスク要素で **Groups** 属性が定義されている場合に、このユーザータスク要素は、グループに所属する全ユーザー一覧に表示されます。このグループに所属するメンバーは誰でもタスクをクレームできます。

タスクがクレームされると、他のユーザーのタスク一覧からこのタスクは消失します。

ユーザータスクは、ドメイン固有のタスクとして実装され、カスタムタスクのベースとして機能します。

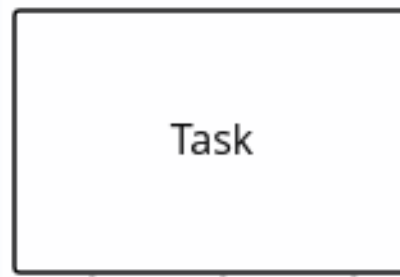
サービスタスク

サービスタスクは、人間の介入を必要としないタスクです。これらは、外部のソフトウェアサービスによって自動的に完了します。



タスクなし

アクティベーション時に完了するタスクはありません。これは概念モデルのみです。「なし」タスクは、ITシステムによって実際に実行されることはありません。



第5章 プロセスデザイナー内の BPMN2 サブプロセス

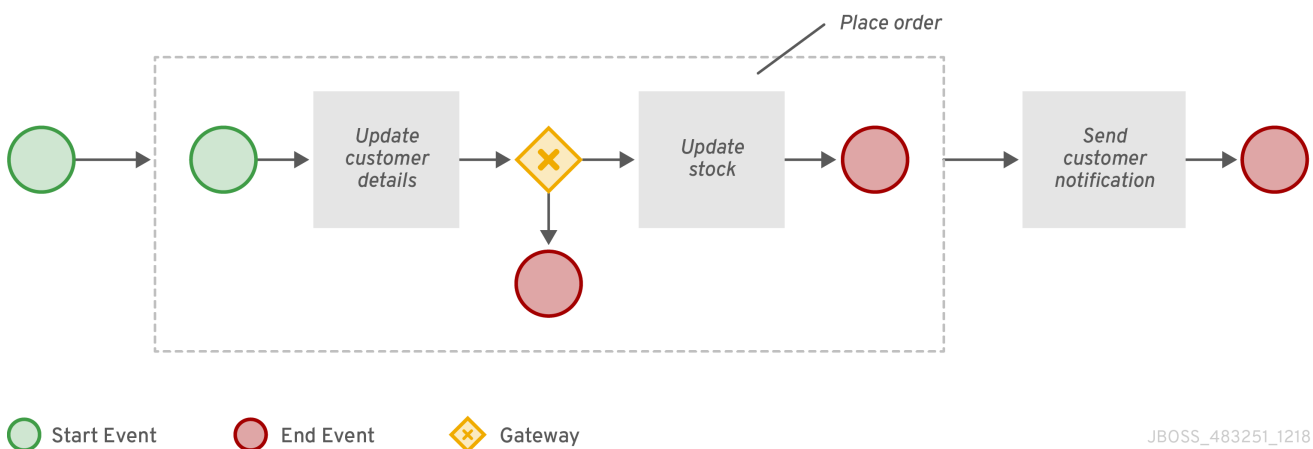
サブプロセスは、複数のノードが含まれるアクティビティです。サブプロセスにメインのプロセスの一部を埋め込むことができます。また、サブプロセスには変数定義を追加できます。これらの変数は、サブプロセス内の全ノードにアクセスできます。

サブプロセスには、内向きの接続と外向きの接続を1つずつ含める必要があります。サブプロセス内の中断終了イベントは、サブプロセスインスタンスを終了しますが、親プロセスインスタンスを自動的に終了することはありません。サブプロセスからアクティブな要素がなくなると、サブプロセスが終了します。

Red Hat Process Automation Manager では、以下のサブプロセスのタイプがサポートされます。

- 埋め込みのサブプロセス。親プロセス実行の一部で、親プロセスのデータを共有します。
- アドホックサブプロセス。厳密な要素実行の順番はありません。
- 再利用可能なサブプロセス。親プロセスから独立しています。
- イベントサブプロセス。開始イベントまたはタイマーでのみトリガーされます。
- マルチインスタンスのサブプロセス

以下の例では、発注のサブプロセスは、その注文を受けるのに十分な在庫があるかを確認し、注文できた場合に在庫情報を更新します。注文の可否により、メインのプロセス経路で、顧客に通知が行きます。



埋め込みサブプロセス


埋め込みサブプロセスは、プロセスの一部をカプセル化します。このサブプロセスには、開始イベントと、最低でも1つの終了イベントが含まれている必要があります。この要素を使用して、このコンテナ内の全要素にアクセスできるローカルのサブプロセス変数を定義できます。

アドホックサブプロセス

アドホックサブプロセスまたはプロセスには、埋め込みの内部アクティビティが複数含まれ、通常のプロセスルーティングに比べて、より柔軟な順番で実行することを目的としています。通常のプロセスとは違い、アドホックサブプロセスには、開始イベントから終了イベントまでといった、完全な体系化された BPMN2 ダイアグラムの説明は含まれません。代わりに、アクティビティ、シーケンスフロー、ゲートウェイ、中間イベントのみが含まれます。また、アドホックサブプロセスには、データオブジェクトやデータの関連付けも含めることができます。アドホックサブプロセス内のアクティビティでは、内向きおよび外向きのシーケンスフローを含める必要はありませんが、その中に含まれて

いるアクティビティー間のシーケンスフローを指定できます。このサブプロセスを使用する場合には、シーケンスフローで、通常のプロセスと同じように順序の制約が課されます。意味をもたせるには、中間イベントに外向きのシーケンスフローを設定して、アドホックプロセスがアクティブな間は複数回トリガーできるようにします。

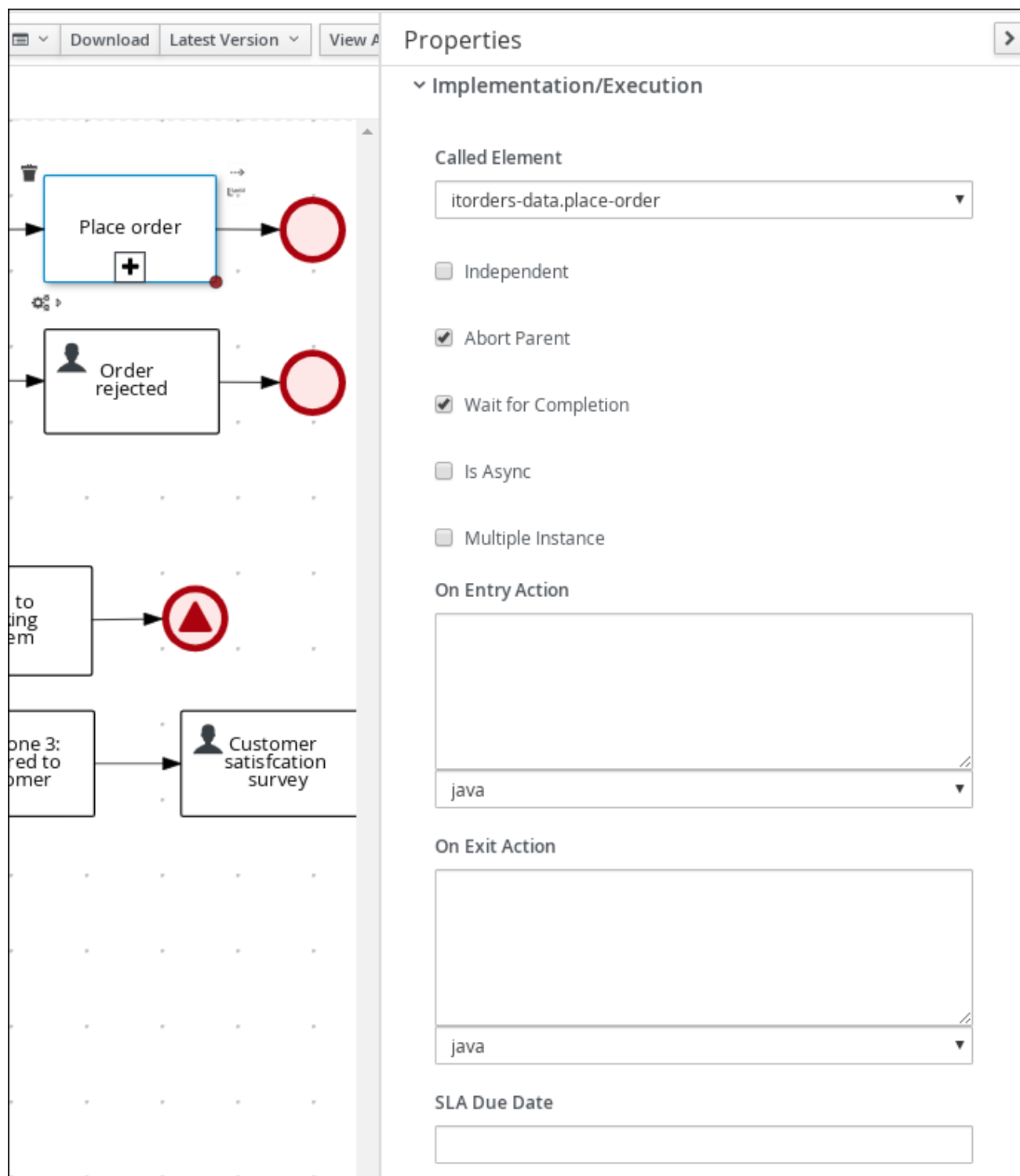
再利用可能なサブプロセス

再利用可能なサブプロセスは、親プロセス内で縮小表示されます。再利用可能なサブプロセスを設定するには、再利用可能なサブプロセスを選択し、 をクリックして **Implementation/Execution** を展開します。以下のプロパティーを設定します。

- **呼び出された要素:** アクティビティーにより呼び出してインスタンス化するサブプロセスの ID。
- **独立:** 選択されている場合には、サブプロセスが独立プロセスとして開始されます。選択されていない場合には、親プロセスが中断されると、アクティブなサブプロセスが取り消されます。
- **親の強制終了:** この項目が選択されていて、呼び出したプロセスインスタンスの実行中にエラーが発生した場合には、再利用可能な従属サブプロセスで、親プロセスを中断できます。たとえば、サブプロセスを呼び出そうとしてエラーが発生した時や、サブプロセスインスタンスを中断する時などです。このプロパティーは、**Independent** プロパティーが選択されている時しか、表示されません。以下のルールが適用されます。
 - 再利用可能なサブプロセスが独立している場合に、**親の強制終了** は使用できません。
 - 再利用可能なサブプロセスが独立していない場合 (従属している場合) は、**親の強制終了** を使用できます。
- **完了するまで待機:** 選択されている場合には、呼び出されたサブプロセスインスタンスが終了されるまで、指定の **終了時アクション** は実行されません。親プロセスの実行は、**終了時アクション** が完了するまで継続されます。このプロパティーはデフォルトで選択されています (**true** に設定されています)。
- **非同期:** タスクを非同期で呼び出して、すぐに実行できないようにする場合に選択します。
- **複数インスタンス:** サブプロセス要素を指定の回数実行する場合に選択します。選択されている場合には、以下のオプションを使用できます。
 - **MI 実行モード:** 複数インスタンスを並行して実行するか、順次実行するかを指定します。**Sequential** に設定されている場合には、以前のインスタンスが完了するまで新規インスタンスは作成されません。
 - **MI コレクション入力:** 新規インスタンスを作成する要素コレクションを表現する変数を選択します。サブプロセスは、コレクションのサイズと同じ回数だけ、インスタンス化されません。
 - **MI データ入力:** コレクションで、選択された要素が含まれる変数名を指定します。この変数は、コレクション内の要素にアクセスする時に使用します。
 - **MI コレクション出力:** マルチインスタンスノードの出力を収集する要素コレクションを表現する任意の変数。
 - **MI データ出力:** **MI コレクション出力** プロパティーで選択した出力コレクションに追加する変数名を指定します。
 - **MI 完了条件 (mvel):** 指定した複数のインスタンスノードを完了できるかどうかを確認するために、完了済みのインスタンスを評価する MVEL 式。**true** と評価された場合には、残りのインスタンスはすべて取り消されます。

- **開始時アクション:** タスクの開始時のアクションを指定する Java または MVEL スクリプト。
- **終了時アクション:** タスクの終了時のアクションを指定する Java または MVEL スクリプト。
- **SLA 期日:** サービスレベルアグリーメント (SLA) の有効期限の日付。

図5.1 再利用可能なサブプロセスのプロパティ



The screenshot displays the Properties panel for a reusable sub-process in the Red Hat Process Automation Manager 7.8 Process Designer. The left pane shows a BPMN diagram with several tasks and events. The right pane, titled 'Properties', shows the 'Implementation/Execution' tab with the following settings:

- Called Element:** itorders-data.place-order
- Independent:** ☐
- Abort Parent:** ☒
- Wait for Completion:** ☒
- Is Async:** ☐
- Multiple Instance:** ☐
- On Entry Action:** java
- On Exit Action:** java
- SLA Due Date:** (empty field)

イベントサブプロセス

イベントサブプロセスは、開始イベントがトリガーされるとアクティブになります。親プロセスのコンテキストを中断するか、並行して実行できます。

外向きまたは内向きの接続では、イベントまたはタイマーがサブプロセスをトリガーできます。サブプロセスは、通常のコントロールフローの一部ではありません。自己完結型ではありますが、バインドされているプロセスのコンテキストで実行されます。

プロセスフロー内のイベントサブプロセスを使用して、主なプロセスフロー外で発生するイベントを処理します。たとえば、飛行機の予約時には、以下の2つのイベントが発生する可能性があります。

- 予約の取り消し (割り込み)
- 予約ステータスの確認 (割り込みなし)

イベントのサブプロセスを使用して、これらのイベント両方をモデル化します。

マルチインスタンスサブプロセス

マルチインスタンスサブプロセスは、実行がトリガーされると、複数回インスタンス化されます。インスタンスが順次作成されます。新しいサブプロセスインスタンスは、前のインスタンスが完了した後にのみ作成されます。

マルチインスタンスサブプロセスには、内向きの接続1つと、外向きの接続1つが含まれます。

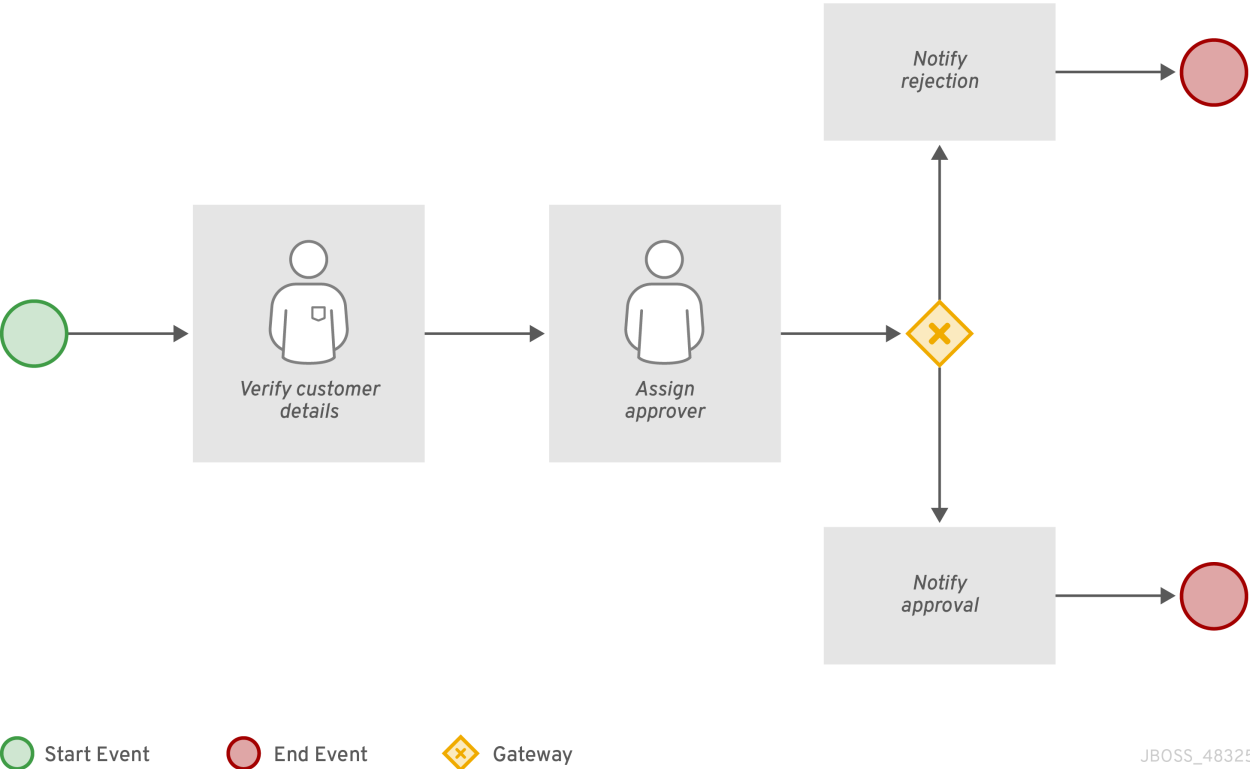
第6章 プロセスデザイナーでの BPMN2 ゲートウェイ

ゲートウェイは、ゲートメカニズムと呼ばれる条件セットを使用して、ワークフロー内にブランチを作成するか、ワークフロー内のブランチを同期するのに使用します。BPMN2 は、2 種類のゲートウェイをサポートします。

- 収束ゲートウェイ。複数のフローを1つにマージします。
- 分岐ゲートウェイ。1つのフローを複数のフローに分割します。

1つのゲートウェイに、複数の内向きと外向きフローを割り当てることはできません。



以下のビジネスプロセスダイアグラムで、XOR ゲートウェイは、条件が True と評価された内向きフローのみを評価します。



この例では、顧客の詳細がユーザーにより検証され、ユーザーが承認できるようにプロセスが割り当てられます。承認されると、承認通知がユーザーに送信されます。要求イベントが却下された場合には、却下通知がユーザーに送信されます。

表6.1 ゲートウェイ要素

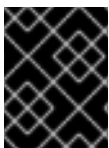
要素タイプ	アイコン
排他的論理和 (XOR)	

要素タイプ	アイコン
包含的	
並列	
イベント	

排他的

排他的な分岐ゲートウェイでは、最初の内向きフローで条件が True と評価されたもののみが選択されます。収束ゲートウェイでは、トリガーされた内向きフローごとに、次のノードがトリガーされます。

このゲートウェイは、1つだけ外向きフローをトリガーします。フローの条件が True と評価されタナカで、優先順位が **最も低い** 数字が選択されます。



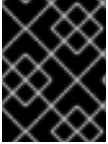
重要

実行時に、最低でも1つの外向きフローが True と評価されるようにしてください。そうでないと、プロセスインスタンスは、ランタイムの例外で中断されます。

収束ゲートウェイでは、ワークフローブランチが、ゲートウェイに到達すると同時に外向きフローに進むことができます。内向きフローの1つがゲートウェイをトリガーすると、ワークフローはゲートウェイの外向きフローに進みます。複数の内向きフローからトリガーされた場合には、トリガーごとに次のノードをトリガーします。

包含的

包含的な分岐ゲートウェイでは、内向きフローが選択され、さらに True と評価された外向きフローすべてが選択されます。優先順位の数値が低い接続は、数値が高い接続よりも先にトリガーされます。優先順位は評価されますが、BPMN2 仕様では優先順位の順番は保証されません。ワークフローで **priority** 属性に依存しないようにしてください。



重要

実行時に、最低でも1つの外向きフローが True と評価されるようにしてください。そうでないと、プロセスインスタンスは、ランタイムの例外で中断されます。

包含的な収束ゲートウェイでは、包含的な分岐ゲートウェイでこれまでに作成された内向きフローすべてがマージされます。これは、包含ゲートウェイブランチの同期エントリーポイントとして機能します。

並列

並列ゲートウェイを使用して、並列フローを同期し、作成します。並列の分岐ゲートウェイでは、内向きフローが選択されると同時に、外向きフローもすべて選択されます。収束並列ゲートウェイでは、ゲートウェイは、内向きのフローがすべて到達するまで待機してからでないと、外向きフローをトリガーしません。

イベント

イベントベースのゲートウェイは分岐のみで、データをもとにした排他的ゲートウェイ (プロセスデータに反応) とは対照的に、発生する可能性のあるイベントに反応できます。発生したイベントをもとに、外向きフローに移動します。1度に行うことができる外向きフローは1つとなっています。ゲートウェイは、イベントベースのゲートウェイに接続されている中間イベントが発生した場合にのみ、プロセスがインスタンス化される、開始イベントとして機能する可能性があります。

第7章 プロセスデザイナーでの BPMN2 接続オブジェクト

接続オブジェクトは、BPMN2 要素 2 つの間の関連性を作成します。接続オブジェクトが転送された場合には、関連付けは順番に行われ、プロセスのインスタンス内で、要素の 1 つが他の要素よりも先に即座に実行されることを指定します。接続オブジェクトは、関連付けられているプロセス要素の上、下、右、左側で開始、終了できます。OMG BPMN2 仕様では、プロセスの動作を簡単に理解して従うことができるように、接続オブジェクトを独断で配置できます。

BPMN2 は主に、2 種類の接続オブジェクトをサポートします。

- シーケンスフロー: プロセスの要素を接続し、インスタンス内でこれらの要素を実行する順番を定義します。
- 関連付けフロー: 実行セマンティクスなしでプロセスの要素を接続します。関連付けフローは転送できません。できる場合は、単方向となっています。



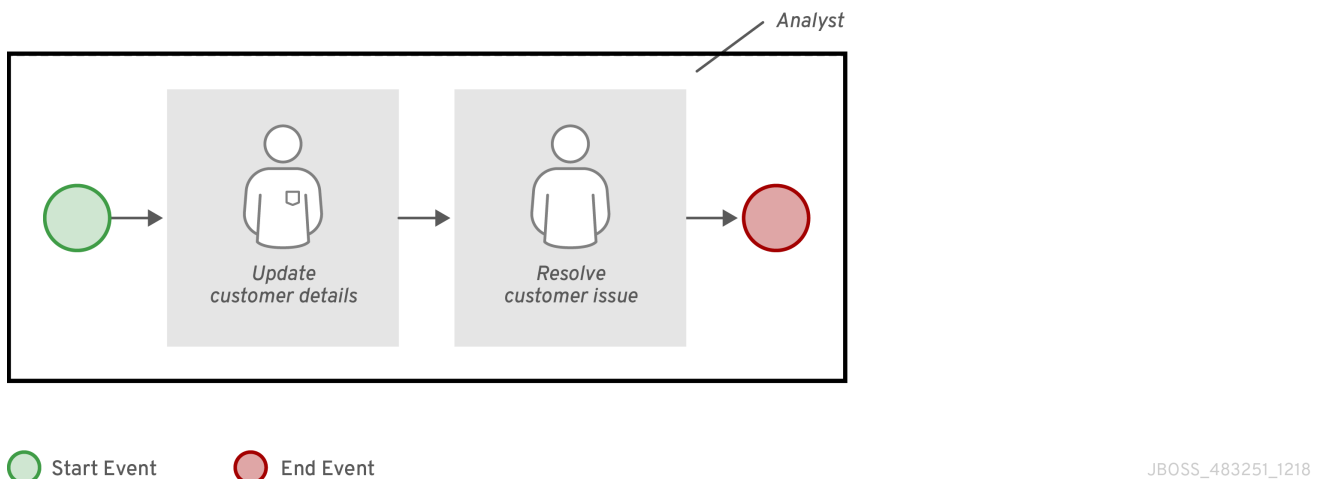
注記

新しいプロセスデザイナーは、転送されない関連付けフローのみをサポートします。レガシーのデザイナーは、一方向と単方向のフローをサポートしています。

第8章 プロセスデザイナーでの BPMN2 スイムレーン

スイムレーンは、1つのグループまたはユーザーに関連のあるタスクを視覚的にグループ化するプロセス要素です。スイムレーンとユーザータスクを組み合わせ使用し、人間による複数のタスクを同じアクターに割り当てます。ランタイム時に、スイムレーンは、同じプロセスインスタンス内で、タスクを自動的にクレームするか、そのレーンで別のタスクを完了したユーザーに割り当てます。スイムレーンで最初のタスクが作成され、タスクにアクター ID が指定されている場合には、アクター ID がそのスイムレーンの他の全タスクにも割り当てられます。レーンとは、プロセス内のサブパーティションのことで、プロセス要素をグループ化し、共通のパラメーターを定義できます。

以下の例では、アナリストには 2 つのユーザータスクが割り当てられています。



Update Customer Details と Resolve Customer Issue のタスクの **Group** フィールドには **analyst** の値が割り当てられています。プロセスが開始されて、Update Customer Details タスクがアナリストユーザーにより要求されるか、開始されるか、完了されると、Resolve Customer Issue タスクが要求されて、最初のタスクを完了したユーザーに割り当てられます。ただし、Update Customer Details タスクにアナリストグループが割り当てられていて、2 番目のタスクにユーザーやグループが割り当てられていない場合に、プロセスは最初のタスク完了後に停止します。

第9章 プロセスデザイナーでの BPMN2 アーティファクト

アーティファクトは、プロセスに関する追加情報を渡すのに使用します。アーティファクトは、BPMN2 ダイアグラムに描写されているオブジェクトで、プロセスワークフローの一部でないものを指します。アーティファクトには、内向きフローオブジェクトも、外向きフローオブジェクトもありません。アーティファクトの目的は、ダイアグラムを理解するのに必要な追加情報を提供することです。アーティファクトの表には、レガシーのプロセスデザイナーでサポートされているアーティファクトが一覧で表示されています。

表9.1 アーティファクト

アーティファクトのタイプ	説明
グループ	プロセス全体で重要なタスクまたはプロセスを整理します。新しいプロセスデザイナーでは、グループアーティファクトはサポートされません。
テキストのアノテーション	BPMN2 ダイアグラムの文字情報を追加で提供します。

付録A バージョン情報

本書の最終更新日: 2020 年 9 月 8 日 (木)