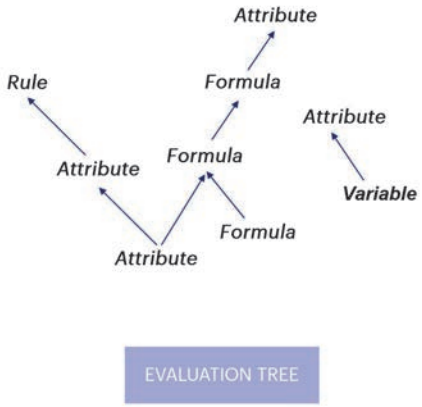
トランザクションにおける
ルールの実行

まとめ

GeneXus™

GeneXus

トランザクションにおけるルールのトリガーイベント



EVALUATION TREE

TRIGGERING EVENTS

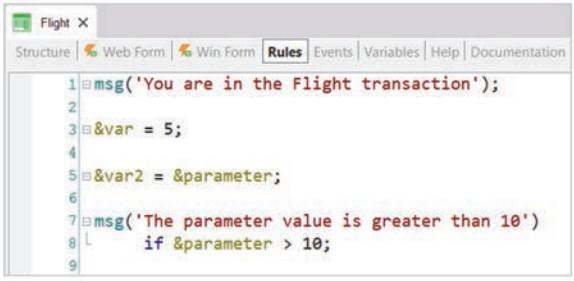
以前の章で、GeneXus がルールおよび式の実行順序を確立するために作成する評価ツリーと、評価ツリーで確立された順序が希望と異なる場合にルールの実行を条件付けるために利用できるトリガーイベントについて学習しました。

この章では、これまでに学んだことを復習し、定着させてから、まだ取り上げていなかった新しいトリガーイベントを見ていきます。

GeneXus

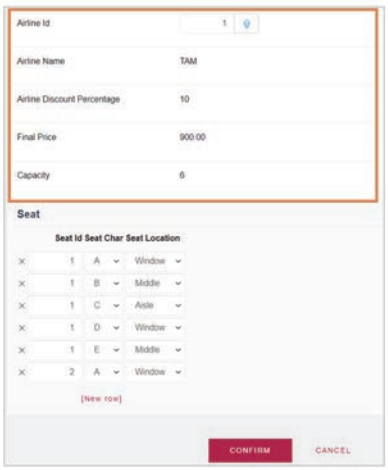
スタンドアロンルール

- トランザクションが実行されるとすぐに実行される
- 実行について条件は存在せず、実行するためにデータを受け取るまで待機する必要がない



最初にトリガーされるルールは、いわゆるスタンドアロンルールです。実行されるものに依存しない、または受け取ったパラメーターによって提供される必要な情報を既に持っているルールです。

トランザクションにおけるルールのトリガーイベント



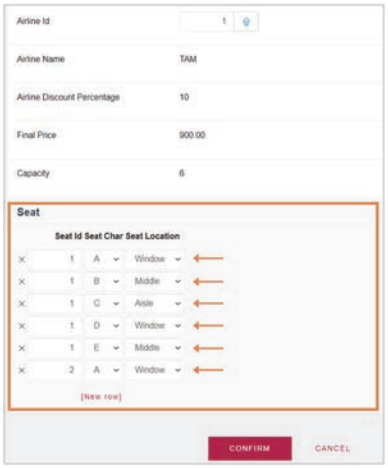
- トランザクションが実行されるとすぐに実行される
- 実行について条件は存在せず、実行するためにデータを受け取るまで待機する必要がない

関連付けられているデータを第 1 レベルから取得できるルールおよび式

スタンドアロンルールを実行した後、トランザクションの第 1 レベルに関連付けられている、トリガーイベントを条件としないルールおよび式が、実行に必要な値が利用可能になったときに評価ツリーに従って実行されます。

GeneXus

トランザクションにおけるルールのトリガーイベント



- トランザクションが実行されるとすぐに実行される
- 実行について条件は存在せず、実行するためにデータを受け取るまで待機する必要がない

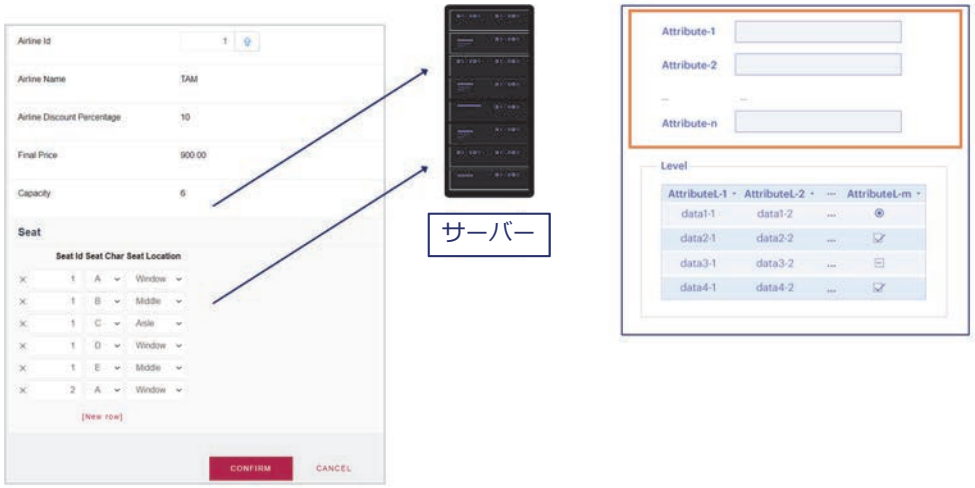
関連付けられているデータを第 1 レベルから取得できるルールおよび式

関連付けられているデータを第 2 レベルから取得できるルールおよび式

第 2 レベルでは、明細行ごとに、トランザクションのそのレベルに関連付けられているルールおよび式が第 1 レベルと同じ基準で実行されます。

GeneXus

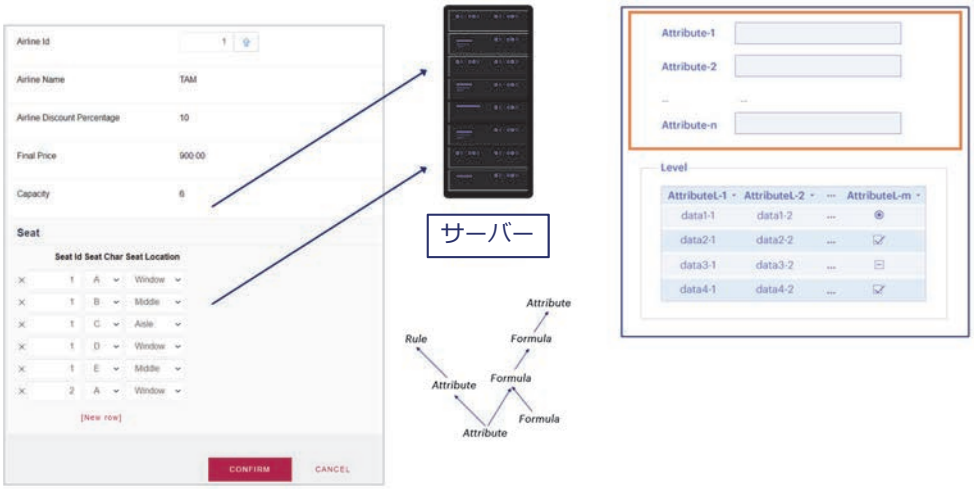
トランザクションにおけるルールのトリガーイベント



ユーザーが [実行] をクリックすると、データがブラウザーから Web サーバーに送信されます。これにより、ユーザーが各フィールドを 1 つずつ入力するかのようにフォームが再実行されます。ヘッダーから始まり、ほかに依存しないルールや、モードに依存するルール (Default ルールなど) がトリガーされます。

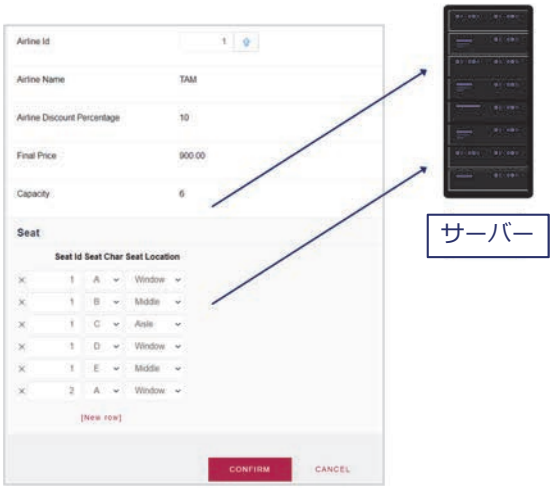
GeneXus

トランザクションにおけるルールのトリガーイベント

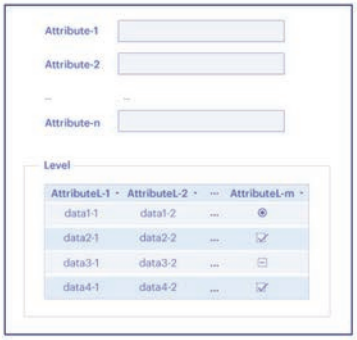


各フィールドが実行され、検出された依存関係に基づいて、対応するルールがトリガーされます。

トランザクションにおけるルールのトリガーイベント



サーバー

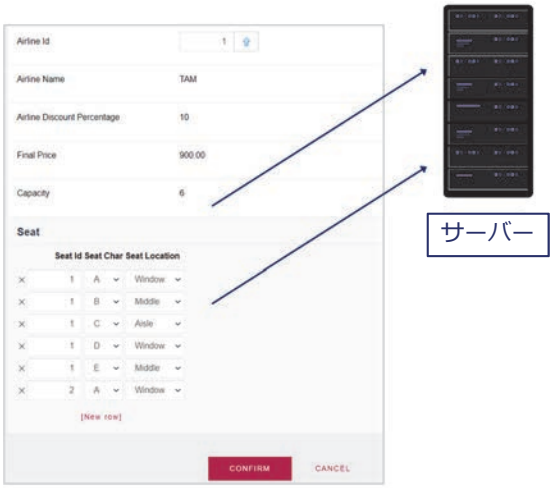


検証段階:

- 参照整合性チェック
- 並行性制御メカニズム
- 許容範囲内の値

ルールおよび式の実行が完了すると、GeneXus は**検証**段階に入ります。検証段階では、すべての参照整合性チェックが検証され、関連データが変更されていないことが並行性制御メカニズムによってチェックされ、項目属性に入力された値が許容範囲内にあるかどうかがチェックされます。

トランザクションにおけるルールのトリガーイベント



→ **on BeforeValidate イベント**

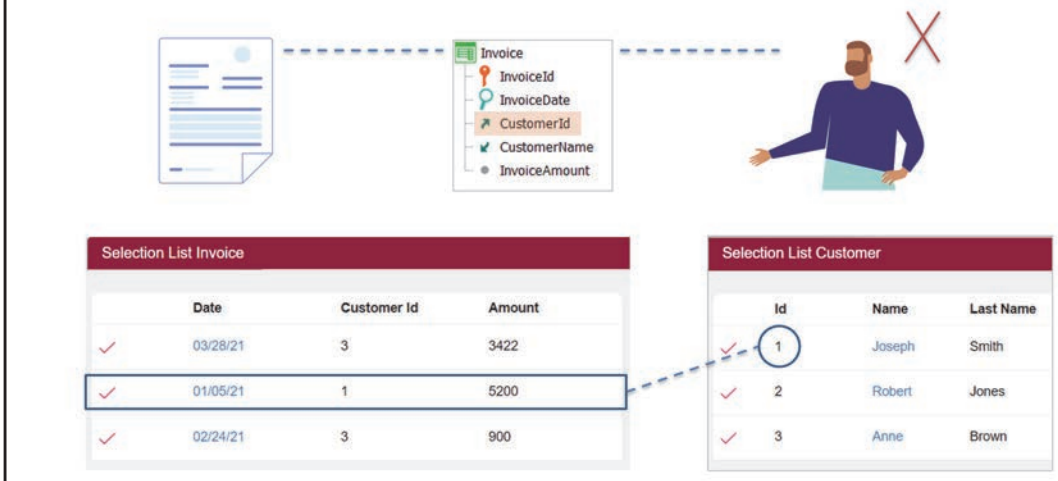
検証段階:

- 参照整合性チェック
- 並行性制御メカニズム
- 許容範囲内の値

→ **on AfterValidate イベント**

GeneXus では、検証を実行する直前に **on BeforeValidate** トリガーイベント、直後に **on AfterValidate** トリガーイベントを使用してルールを実行するように条件付けできます。

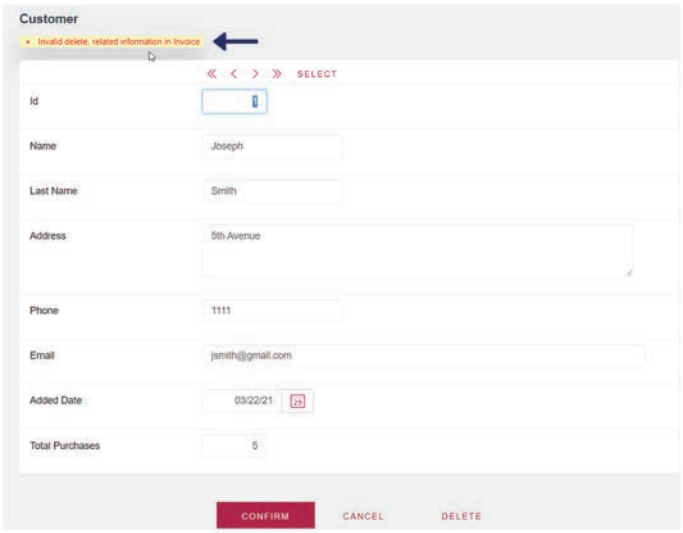
on BeforeValidate イベント



「on BeforeValidate」の使用例を見てみましょう。

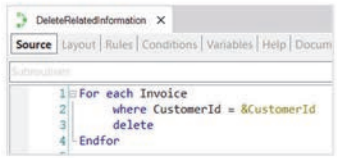
ある顧客をシステムから削除するとします。この顧客には、関連付けられている請求書があります。

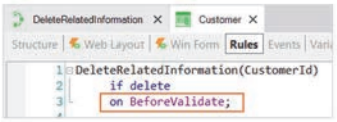
on BeforeValidate イベント

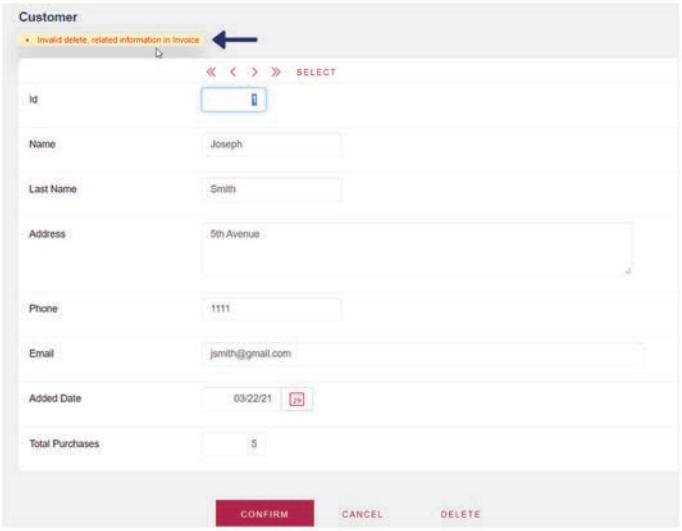


削除を実行しようとすると、エラーメッセージが表示されます。これは、この顧客に関連付けられているレコードが Invoice テーブルにあるために、削除を実行できないことを示しています。

on BeforeValidate イベント



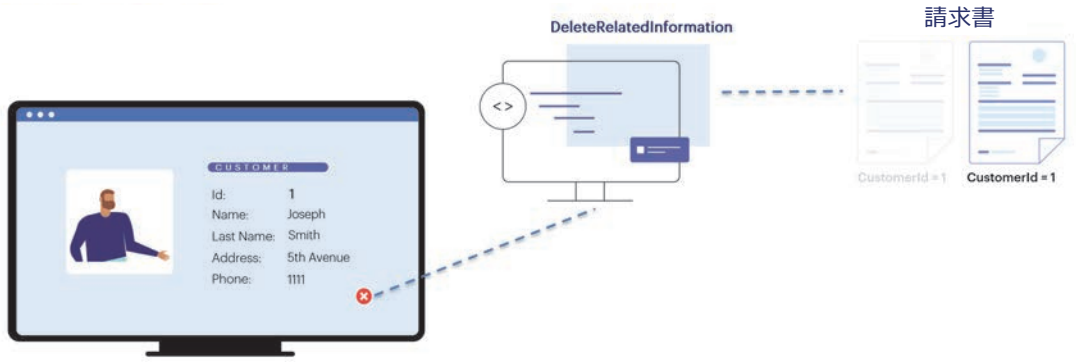




これを解決するためには、この顧客に関連する情報を対応するテーブルから削除するプロシージャーを呼び出します。そうすることで、削除を実行する際にデータの整合性違反が発生しないようにすることができます。

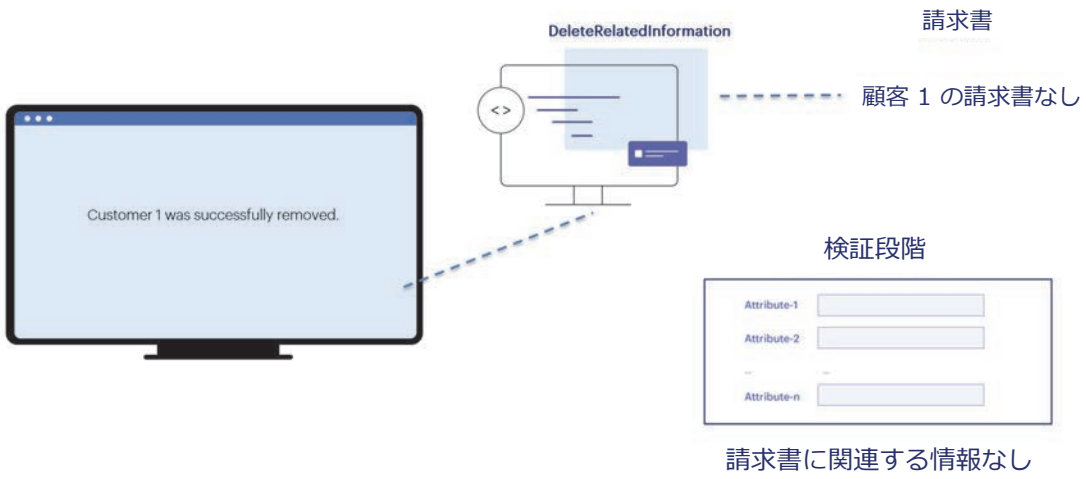
このプロシージャーは、GeneXus が参照整合性チェックを開始する前に実行する必要があります。先ほど説明したように、参照整合性チェックは検証段階で行われるため、on BeforeValidate イベントでルールが実行されるように条件を設定します。

on BeforeValidate イベント



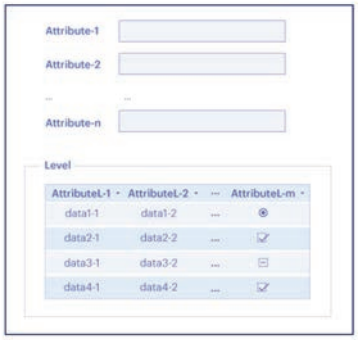
こうすると、ID が 1 の顧客を削除しようとした場合、検証チェックが開始される前に、その顧客に関連付けられている請求書を削除するプロシージャーが呼び出されます。

on BeforeValidate イベント



そのため、検証が実行されると、関連付けられているレコードは検出されず、顧客は削除されます。

on BeforeValidate / on AfterValidate



on BeforeValidate イベント

ヘッダーの検証

on AfterValidate イベント

(各行)

on BeforeValidate イベント

明細行の検証

on AfterValidate イベント

on BeforeValidate トリガーイベントは、処理しているインスタンスの情報 (ヘッダーまたは明細行) の検証が行われる直前に発生します。つまり、「ヘッダーの検証」または「明細行の検証」アクションの直前に行われます。ここでは、どのトリガーイベントにも条件付けられておらず、そのレベルに関連しているすべてのルールが既にトリガーされていることに留意してください。また、**on AfterValidate** トリガーイベントによって、インスタンスのデータを検証した直後、対応するテーブルに物理的に保存する前にルールを実行するかどうかを設定できます。

on BeforeValidate イベントまたは on AfterValidate イベントに条件付けられたすべてのルールは、このタイミングでトリガーされます。これらはどのような順序で実行されるのでしょうか。ルールは、それらの間に依存関係がないかぎり、記述されている順序で実行されます。

GeneXus

トランザクションにおけるルールのトリガーイベント

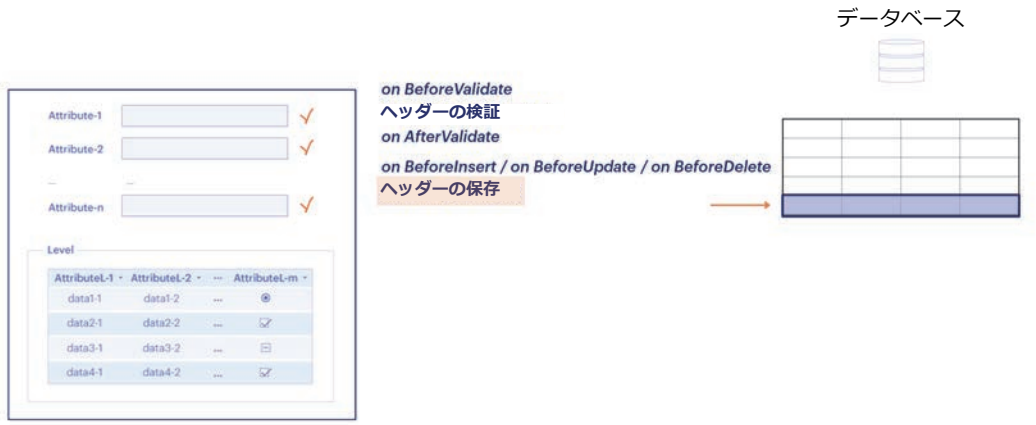
on BeforeValidate
ヘッダーの検証
on AfterValidate
on BeforeInsert / on BeforeUpdate / on BeforeDelete

検証でエラーが発生しなかった場合、レコードの種類に応じて次の処理が行われます。

- 挿入に対応するレコード: **on BeforeInsert** トリガーイベントを持つトランザクションの第 1 レベルに関連付けられているルールが実行されます。
- 更新に対応するレコード: **on BeforeUpdate** トリガーイベントを持つトランザクションの第 1 レベルに関連付けられているルールが実行されます。
- 削除に対応するレコード: **on BeforeDelete** トリガーイベントを持つトランザクションの第 1 レベルに関連付けられているルールが実行されます。

GeneXus

トランザクションにおけるルールのトリガーイベント



ヘッダーのデータは、対応するテーブルに記録されます。

GeneXus

トランザクションにおけるルールのトリガーイベント

そして、この第 1 レベルに関連付けられた、**on AfterInsert** イベント (レコードが挿入に対応する場合)、**on AfterUpdate** イベント (レコードが更新に対応する場合)、および **on AfterDelete** イベント (レコードが削除に対応する場合) を持つルールが実行されます。

トランザクションにおけるルールのトリガーイベント

トランザクションが 2 レベルのトランザクションである場合は、ヘッダーを記録した後、関連するトリガーイベントを持たない第 2 レベルの項目属性のルールおよび式が実行され、明細行ごとに次が行われます。

- 検証。第 2 レベルの項目属性を含む **on BeforeValidate** および **on AfterValidate** イベントに条件付けられているすべてのルールがトリガーされます。

GeneXus

トランザクションにおけるルールのトリガーイベント

- また、各明細行について、第 2 レベルの項目属性を含むイベント **on BeforeInsert** (または **on BeforeUpdate** か **on BeforeDelete**) および **on AfterInsert** (または **on AfterUpdate** か **on AfterDelete**) に条件付けられているすべてのルールがトリガーされます。

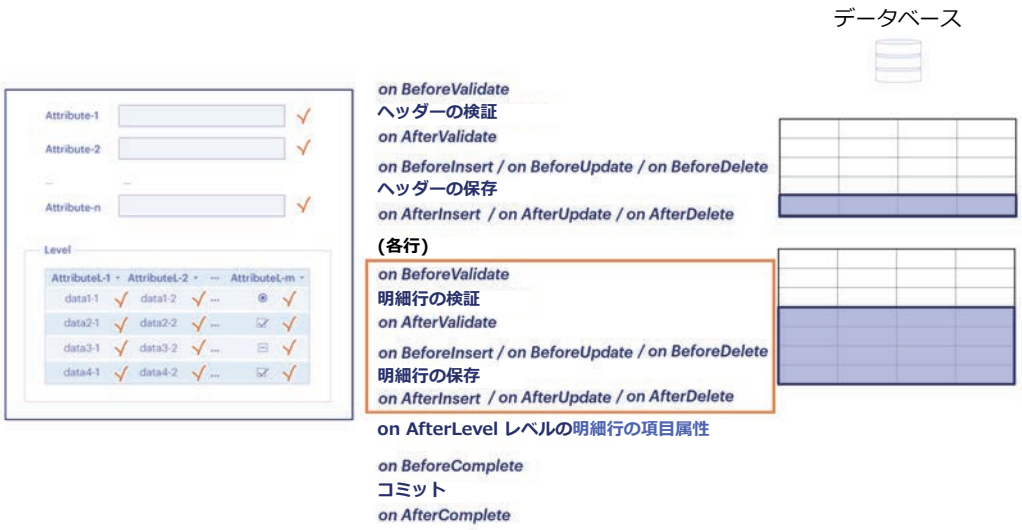
GeneXus

トランザクションにおけるルールのトリガーイベント

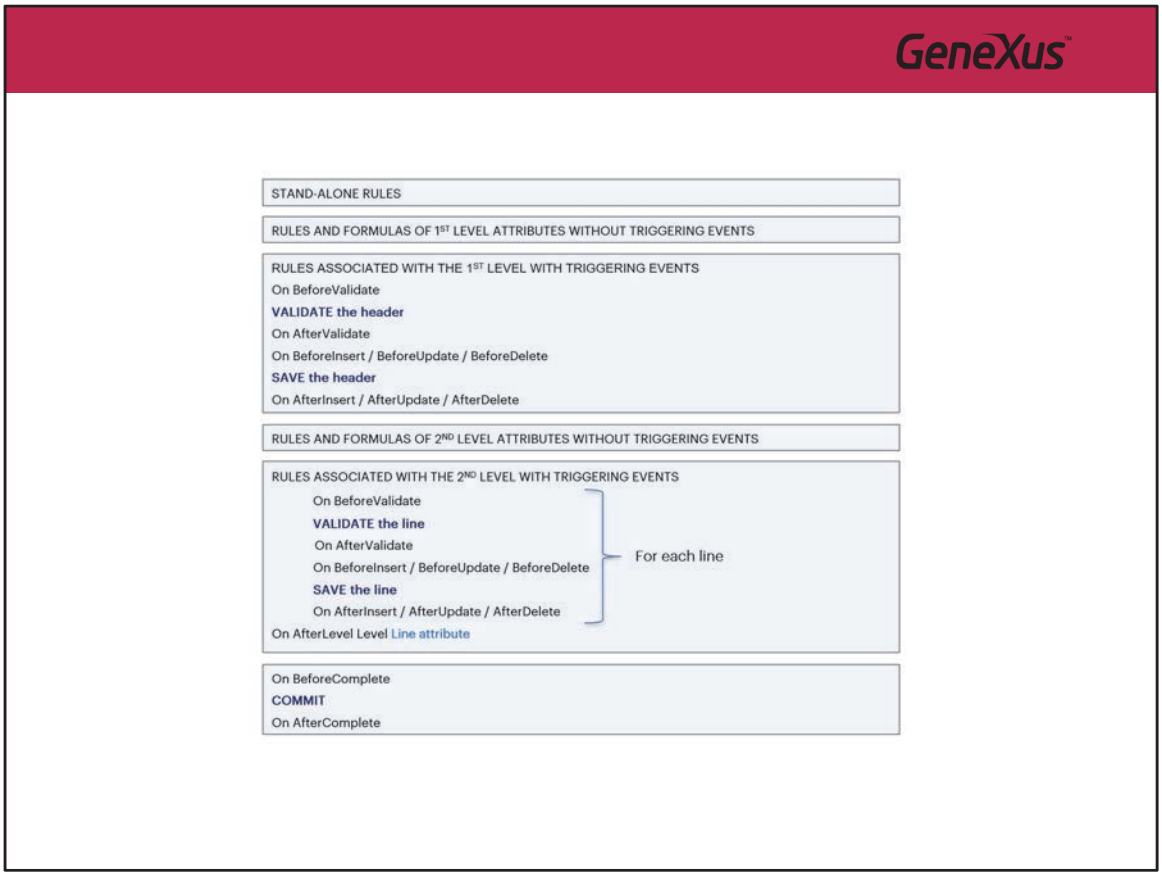
次に、特定のレベルを実行した直後に何かを実行または評価する **on AfterLevel** イベントを持つルールがトリガーされます。

さらにグリッド (別のレベル) がある場合、最初のグリッドに対して行われたのと同じことが繰り返され、そのレベルの on AfterLevel イベントも実行されます。これが最後のグリッドに到達するまで行われます。

トランザクションにおけるルールのトリガーイベント



すべての処理が行われたら、**コミット**が実行されます。これにより、ヘッダーのデータおよびトランザクションのすべての明細行がデータベースに統合されます。コミットの直前にルールを実行する **on BeforeComplete** イベントと、コミットの直後にルールを実行する **on AfterComplete** イベントがあります。



トランザクション内のルールが実行される順序、ルールに割り当てることができるトリガーイベント、トリガーイベントがトリガーされるタイミング、トリガーイベントの前および後に実行されるアクションについて理解しておくことは非常に重要です。これらの知識は、トランザクションの動作を正しくプログラムするために不可欠です。