

Kofax RPA

Document Transformation スタート ガイド

バージョン: 11.0.0

日付: 2019-12-17

The logo for KOFAX, consisting of the word "KOFAX" in a bold, blue, sans-serif font.

© 2019–2020 Kofax. All rights reserved.

Kofax is a trademark of Kofax, Inc., registered in the U.S. and/or other countries. All other trademarks are the property of their respective owners. No part of this publication may be reproduced, stored, or transmitted in any form without the prior written permission of Kofax.

目次

はじめに.....	4
関連ドキュメント.....	4
トレーニング.....	5
Kofax 製品のヘルプの入手.....	5
第 1 章: チュートリアル.....	7
事前のステップ.....	7
ロボットとタイプの作成.....	7
変換ロボットへの変数の追加.....	10
Management Console へのロボットのアップロード.....	11
Document Transformation ワークフローの作成.....	11
ドキュメントの有効性を確認するためのステップの追加.....	17
ドキュメントからのデータ抽出.....	18
人による検証のステップの設定.....	19
ロボットが無効なドキュメントと有効なドキュメントを処理できるように構成.....	33
変換されたドキュメントの検証.....	35

はじめに

このガイドでは、Document Transformation 機能を Kofax RPA 環境で使用方法について説明するチュートリアルを提供します。このチュートリアルでは、OCR、抽出、フィールドの書式設定、検証などの機能を取り上げます。

チュートリアルでは、Kofax RPA をダウンロードしてコンピュータに正常にインストールしていることを前提としています。Kofax RPA 製品グループから Management Console および Development Database を開始した後、Windows サービス パネルで「Document Transformation - Server Scheduler Service」が開始されていることを確認します。

Document Transformation 機能の詳細な説明については、「Kofax RPA のヘルプ」を参照してください。

関連ドキュメント

Kofax RPA のドキュメント セットには次の場所からアクセスできます。¹

https://docshield.kofax.com/Portal/Products/RPA/11.0.0_qrvv5i5e1a/RPA.htm

このガイドの他に、ドキュメント セットには次の項目が含まれています。

Kofax RPA Release Notes (Kofax RPA リリース ノート)

その他の Kofax RPA ドキュメントからは入手できない最新の詳細やその他の情報が含まれています。

Kofax RPA Installation Guide (Kofax RPA インストール ガイド)

Kofax RPA およびそのコンポーネントを開発環境にインストールする方法について説明します。

Kofax RPA Administrator's Guide (Kofax RPA 管理者ガイド)

Kofax RPA での管理タスクについて説明します。

Kofax RPA ヘルプ

Kofax RPA の使用方法について説明しています。ヘルプは、『Kofax RPA ユーザー ガイド』という PDF 形式のドキュメントとしても提供されています。

Kofax RPA Desktop Automation スタート ガイド

Kofax RPA Desktop Automation を使用してロボットを構築するプロセスを実行するためのチュートリアルを提供します。

¹ オンラインのドキュメント セットにアクセスするにはインターネットに接続する必要があります。インターネットに接続せずにアクセスする方法については、『Installation Guide』(インストール ガイド)を参照してください。

Kofax RPA Desktop Automation サービス設定ガイド

リモート コンピュータで Desktop Automation を使用するために必要な Desktop Automation サービスを設定する方法について説明します。

Kofax RPA Developer's Guide (Kofax RPA 開発者ガイド)

RoboServer 上でロボットを実行するために使用される API に関する情報が含まれています。

Kofax RPA Integration API documentation (Kofax RPA 統合 API 文書)

Kofax RPA へのプログラムでのアクセスを提供する Kofax RPA Java API および Kofax RPA .NET API についての情報が含まれています。Java API 文書は、オンラインおよびオフラインの Kofax RPA 文書から入手できますが、.NET API 文書はオフラインのみとなります。

注 Kofax RPA API は、元の製品名である「RoboSuite」に対する詳細な参照を含んでいません。RoboSuite の名前は後方互換性を確保するために残されています。API 文書の中では、RoboSuite という用語は Kofax RPA と同じ意味で使われています。

トレーニング

Kofax は、Kofax RPA ソリューションを最大限に活用するために、教室でのトレーニングとコンピュータでのトレーニングの両方を提供しています。利用可能なトレーニング オプションとスケジュールの詳細については、www.kofax.com の Kofax Web サイトをご覧ください。

Kofax 製品のヘルプの入手

[Kofax Knowledge Base \(Kofax ナレッジ ベース\)](#) リポジトリにある記事の内容は定期的に更新され、Kofax 製品の最新情報について参照できます。製品に関してご不明の点がある場合は、Knowledge Base (ナレッジ ベース) で情報を検索することをお勧めします。

Kofax Knowledge Base (Kofax ナレッジ ベース) を参照するには、[Kofax Web サイト](#) にアクセスして、ホームページでサポートを選択してください。

注 Kofax Knowledge Base (Kofax ナレッジ ベース) は Google Chrome、Mozilla Firefox または Microsoft Edge 向けに最適化されています。

Kofax Knowledge Base (Kofax ナレッジ ベース) では以下の内容を提供します:

- 強力な検索機能で必要な情報をすぐに見つけることができます。
Search (検索) ボックスに目的の語句を入力し、検索アイコンをクリックしてください。
- 製品情報、設定の詳細、リリース情報などのドキュメント。

Kofax Knowledge Base (Kofax ナレッジ ベース) のホームページをスクロールして、製品ファミリーを見つけます。目的の製品ファミリー名をクリックして、関連記事の一覧を表示します。一部の製品ファミリーの場合は、関連記事を表示するために Kofax Portal (Kofax ポータル) の有効なログイン情報を入力する必要があります。

- Kofax Customer Portal (Kofax カスタマー ポータル) へのアクセス (資格のあるカスタマー向け)
ページ上部にある **Customer Support** (カスタマー サポート) リンクをクリックしてから、**Log in to the Customer Portal** (カスタマー ポータルにログイン) をクリックします。
- Kofax Partner Portal (Kofax パートナー ポータル) へのアクセス (資格のあるカスタマー向け)
ページ上部にある **Partner Support** (パートナー サポート) リンクをクリックしてから、**Log in to the Partner Portal** (パートナー ポータルにログイン) をクリックします。
- Kofax サポート コミットメント、ライフサイクル ポリシー、電子フルフィルメントの詳細、セルフサービス ツールへのアクセス。
General Support (一般サポート) セクションまでスクロールして、**Support Details** (サポートの詳細) をクリックし適切なタブを選択します。

第 1 章

チュートリアル

このガイドは、製品に付属しているサンプル ロボットのような Document Transformation ロボットを作成する方法について説明するステップバイステップのチュートリアルです。*DocumentTransformationInvoiceExample* という名前のサンプル ロボットには、変換可能な .tif ファイルがすでに含まれています。

製品に付属している *DocumentTransformationInvoiceExample* ロボットは、ロボットとタイプを新規に作成せずにテストを実行することができます。**Design Studio** の **Examples** フォルダでロボットを開いて、タイプ *invoice* と *DocumentTransformationInput* を開き、このガイドに従ってロボットの構築方法を確認できます。

ロボットを最初から作成する場合は、このガイドの手順を使用して、必ずサンプルとは異なる一意の名前をロボットとタイプに割り当ててるようにしてください。

このチュートリアルでは最も一般的に使用される Document Transformation 関数のいくつかを、主要 4 パート構成で取り上げます。

1. 変換ロボット、Desktop Automation ロボットおよびタイプを作成します。
2. 「Desktop Automation ワークフローの呼び出し」ステップを使用して、Web オートメーション ロボットから Desktop Automation ロボットを呼び出します。
3. Document Transformation ワークフローを Desktop Automation ロボットで作成します。このステップには、[Document Transformation ブラウザ ウィンドウを開くステップ] の設定、.tif ファイルからのデータ抽出、および変換されたドキュメントの人による検証と人によらない検証の設定が含まれます。
4. 変換されたドキュメントを検証します。

事前のステップ

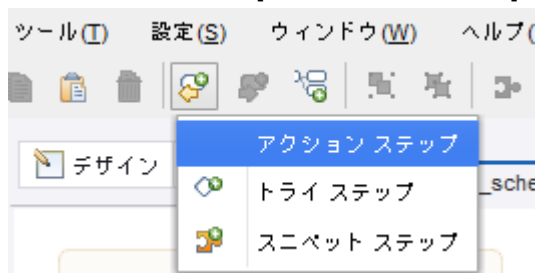
ロボットとタイプの作成

最初に、Design Studio で Web オートメーション ロボットを作成する必要があります。次に、変換ワークフローを構築するために必要な Desktop Automation ロボットを作成し、Web オートメーション ロ

ロボットから Desktop Automation ロボットを呼び出します。また、抽出されたデータを保存するためにタイプを作成する必要があります。


1. Web オートメーション ロボットを作成する

- a. **Design Studio** で、[ファイル] > [新しいロボット] をクリックします。
- b. ロボットに **DocumentTransformationInvoiceExample** という名前を付け、プロジェクトを選択し、[次へ] > [次へ] の順にクリックします。
デフォルトで [スマート再実行 (フル)] 実行モードが選択されます。
- c. [終了] をクリックします。
デフォルトでは、新しく作成されたロボットでは終了ステップが選択されます。
- d. 新規作成ロボットに [アクション ステップ] を挿入します。



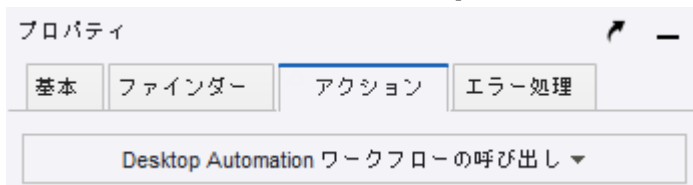
- e. ステップの名前を「**Transform Document**」に変更します。
- f. 変更を保存します。

2. Desktop Automation ロボットを作成する

- a. [ファイル] > [新しい Desktop Automation ロボット] をクリックします。
- b. ロボットに **DocumentTransformationInvoiceExampleDA** という名前を付け、[終了] をクリックします。
エディター ウィンドウの新しいタブに新しいロボットが表示されます。青いアイコンで識別される Web オートメーション ロボットとは対照的に、Desktop Automation ロボットは緑色のアイコンで識別されます： 
この時点で、最初に Web オートメーション ロボットから新しいロボットを呼び出す必要があるため、ワークフローは編集できません。

3. Desktop Automation ワークフローの呼び出しステップを追加する

- a. **DocumentTransformationInvoiceExample** Web オートメーション ロボットでタブを開きます。
- b. 挿入ステップで、[アクション] タブの [アクションを選択] をクリックし、[**Desktop Automation** ワークフローの呼び出し]



を選択します。

- c. [ワークフロー] ドロップダウン リストで、[**DocumentTransformationInvoiceExampleDA**] ロボットを選択します。
 - d. 変更を保存します。
- ### 4. 新しいタイプを作成

このロボットを使用して、ベンダーの ID、請求書番号、請求書日付、合計を抽出します。抽出したデータを格納するには、以下のタイプを作成します。

- a. [ファイル] > [新しいタイプ] をクリックします。
- b. タイプに **invoice** という名前を付け、プロジェクトを選択し、[終了] をクリックします。
- c. プラス記号をクリックして新しい属性をタイプに追加します。次の属性を追加して、そのタイプを指定します。

名前	属性の種類
VendorID	Short Text
InvoiceNumber	Short Text
InvoiceDate	Short Text
Total	Number

- d. **DocumentTransformationInput** という名前のタイプを別に作成します。
- e. 次の属性を追加して、そのタイプを指定します。

名前	属性の種類
doc	Binary

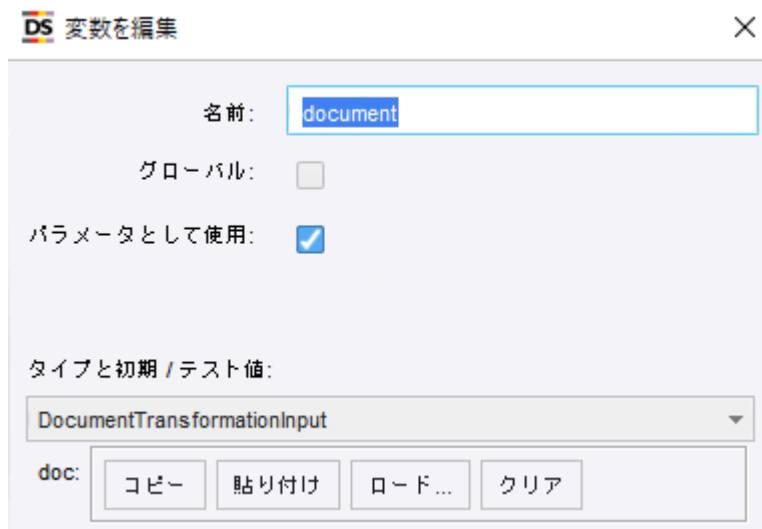
- f. 変更を保存します。

終了したら次のセクションに進みます。

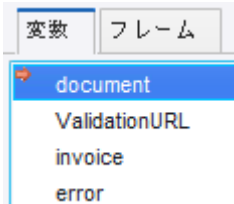
変換ロボットへの変数の追加

この手順では、コンプレックス タイプの変数とシンプル タイプの変数を追加します。複雑な変数には複数の名前が付いた値を含めることができます。これは請求書のデータのようなさまざまなデータの抽出に有用です。シンプル変数には、テキスト文字列のような単一の値のみを含めることができます。

1. Design Studio の [変数] フィールドで右クリックし、[コンプレックス タイプの変数を追加] > [DocumentTransformationInput] の順にクリックします。
 - a. 新しいダイアログ ボックスで「DocumentTransformationInput」変数の名前を「document」に変更し、[パラメータとして使用] を選択します。

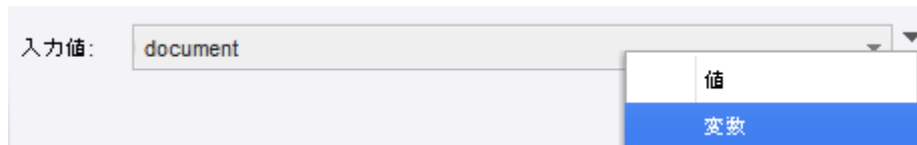


- b. [ロード] をクリックし、変換する領収書ドキュメントを開きます。サンプル プロジェクトの Kofax RPA [Examples] フォルダに移動し、ファイル **InvoiceExampleDocument.tif** を選択します。この領収書は、組み込みの Invoice VAT Transformation プロジェクトで読み取ります。
2. [変数] フィールドで右クリックし、[コンプレックス タイプの変数を追加] > [invoice] の順にクリックします。
 3. [変数] フィールドで右クリックし、[シンプル タイプの変数を追加] > [Short Text (シンプル)] の順にクリックします。新しい変数に「ValidationURL」という名前を付けます。
 4. [変数] フィールドで右クリックし、[シンプル タイプの変数を追加] > [Long Text (シンプル)] の順にクリックします。新しい変数に [Error] という名前を付けます。
 5. 変更を保存します。



6. **Transform Document** ステップの [アクション] タブを開き、次のように変数を構成します。


- a. [入力値] セクションで、プラス記号をクリックします。
ドロップダウン リストで、[変数] をクリックして「**document**」を選択します。



- b. [OK] をクリックします。
- c. [出力マッピング] セクションで、「**invoice**」、「**ValidationURL**」、「**error**」変数を追加します。

終了したら次のセクションに進みます。


Management Console へのロボットのアップロード

1. [Projects] ツリービューで、**DocumentTransformationInvoiceExample** ロボットを右クリックし、[アップロード] をクリックしてロボットを Management Console にアップロードします。
2. Management Console の URL を開き、[リポジトリ] > [ロボット] タブを開きます。
このチュートリアルでは、Management Console の URL に「`http://127.0.0.1:50080`」を使用します。
3. 「**DocumentTransformationInvoiceExample**」ロボットを選択します。
4. 行の右端までスクロールし、[REST] の下にある  をクリックします。
5. REST URL をコピーします。これは、Document Transformation の構成に必要になります。

終了したら次のセクションに進みます。

Document Transformation ワークフローの作成


この手順では、変数を設定して、Document Transformation ステップを設定し、変換されたドキュメントを待機するためのガードを追加します。

1. ツールバーの [実行の準備]  をクリックして、Web オートメーション ロボットの実行を許可します。

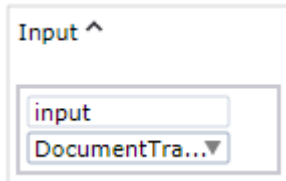
重要 Web オートメーション ロボット実行を許可する場合、または実行権を他の Web オートメーション ロボットに切り替える場合は、[実行の準備] 機能を使用する必要があります。

ロボットに実行権がある場合、ロボット アイコンに赤い点が表示され、ロボット タブが強調表示されます。Desktop Automation ロボットに実行権がある場合、Web オートメーションロボットが呼び出しているタブも強調表示されます。

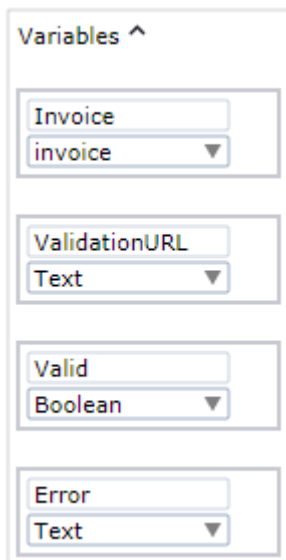


2. 実行が許可されたら、Desktop Automation ワークフローを開いて編集できます。これを行うには、ツールバーの [DA ロボットにステップ イントゥー]  をクリックします。
DocumentTransformationInvoiceExampleDA ロボットのタブが開き、エディターがアクティブになります。

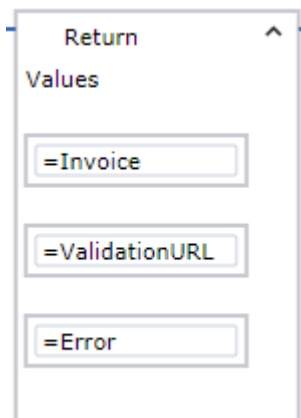
3. ワークフローで、次のステップに従います。
- a. [入力値] ボックスが次の方法で設定されていることを確認します。



- b. この変数には、検証後の変換されたドキュメントが含まれます。
- c. [変数] ボックスで、次の変数を追加し、各タイプを指定します。



- d. ワークフローで、**Return** ステップを展開し、以下の変数を追加します。
1. **Invoice** -必ず大文字で始めるようにしてください。
 2. **ValidationURL**
 3. **Error**



4. [Document Transformation ステップ] 追加するには、リターン ステップの前のフローポイント (小さな円) を右クリックして、**Document Transformation** をクリックします。
このステップでは、データを InvoiceExampleDocument.tif から抽出します。

ヒント このセクションの後半にある Document Transformation ステップを実行すると、文書の変換が失敗したというエラーと変換プロジェクトを参照するメッセージが表示される場合があります。これは、Document Transformation トレーニング ファイルを保存するフォルダがないことを意味する場合があります。

この問題を解決するには、「Invoice VAT」プロジェクトの [Project Builder] 内で [プロジェクト設定] を開き、[一般] タブで [OK] をクリックします。フォルダが存在しない場合、作成することを求める通知を受け取ります。続行するには、[はい] をクリックします。

詳細については、『Kofax RPA Installation Guide』(Kofax RPA インストール ガイド) の「Set up Online Learning」(オンライン学習の設定) セクションを参照してください。


5. プロパティを以下のように設定します。

- a. [ドキュメント ソース] で、[バイナリ変数] 選択し、[バイナリ] フィールドをクリックして、**input.doc** を入力します。変数を評価できるように、左側に等号が表示されていることを確認します。

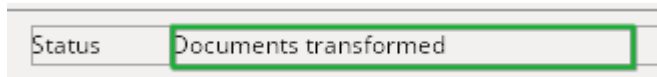
- b. [検証 URL] を選択して、シンクライアント サービスの URL を指定します。このプロパティは、処理されたドキュメントを検証の目的で送信するために必要です。
- c. [コールバック URL] を選択し、[Management Console へのロボットのアップロード](#) でコピーした REST URL を貼り付けます。

シンクライアント サービスは、ドキュメントの検証後にこの URL を呼び出します。検証が完了すると、Management Console はこの URL を使用して、ロボットを開始します。

注 REST URL は、デフォルトでは 127.0.0.1 で始まります。Management Console が Document Transformation Service とは別のコンピュータで実行されている場合は、Management Console にアクセスするためのサービスの適切な URL を指定してください。

- d. [ステップ オーバー]  をクリックしてこのステップを実行します。ステータスが「Waiting for transformed documents.」に変更されます。「Documents transformed.」に変更されるまで待機します。

6. 「Documents transformed」をクリックして、変換されたドキュメントを待機するためのガードを追加します。

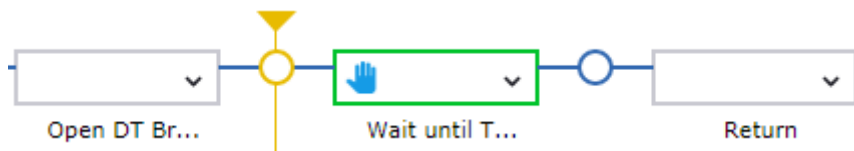


- a. ガードが正しいテキストを検索できるようにするため、XML ツリーの以下のラベルを展開します。

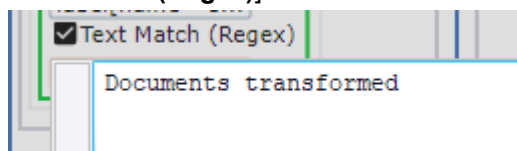
```
<label name="status_message" der_x="494" der_y="56" der_width="230" der_height="20">Documents transformed</label>
```

テキスト「Documents transformed」を右クリックし、[ガード]>[ロケーションが見つかった場合]の順にクリックします。

ガードステップがワークフローに挿入されます。ステップ名を「Wait until Transformed」に変更します。



- b. **Wait until Transformed** ステップで、[ロケーションが見つかった場合] ガードを展開し、[テキスト マッチ (Regex)] を選択して「Documents transformed」と入力します。



- c. [ステップ オーバー] をクリックしてこのステップを実行します。
d. 変更を保存します。

終了したら次のセクションに進みます。

ドキュメントの有効性を確認するためのステップの追加

変換したドキュメントが有効かどうかを確認するには、次のステップを Document Transformation ワークフローに追加します。ドキュメントが有効でないようであれば、人による検証のために送信できます (人による検証のステップの設定 を参照)。

1. XML ツリー ビューで **document** 要素を見つけて右クリックし、[ここから値を抽出] > [属性] > [valid Into] > [Valid: Boolean] の順にクリックします。

```

[-] <folder [-] valid="false" der_x="0" der_y="84" der_width="0" d
  <xvalues [+]/>
  [+] <document [-] class="Invoice" classificationUnconfident="
    indexInFolder="0" name="document" nextValidationInsta
    timeExtraction="2361" timeOCR="2089" valid="false" val
    der_height="0">...</document>
  
```

新しいステップがワークフローに挿入されます。ステップ名を「**Extract Document.valid**」に変更します。

2. [ステップ オーバー] をクリックしてこのステップを実行します。



3. 変更を保存します。

終了したら次のセクションに進みます。

ドキュメントからのデータ抽出

この手順では、データを InvoiceExampleDocument.tif から変数 VendorID、InvoiceNumber、InvoiceDate、Total に抽出します。最後に、[値返却] ステップをロボットに追加すると、抽出された値が返されます。

1. XML ツリー ビューで、**application/folder/document** の下にある **fields** 要素を展開します。

```
<dts ⊕ >
  ⊖ <application ⊕ >
    ⊕ <toolbar ⊕ >...</toolbar>
    ⊖ <folder ⊕ >
      <xvalues ⊕ />
      ⊖ <document ⊕ >
        ⊕ <fields ⊖ der_x="0" der_y="84" der_width="0" der_height="0">...</fields>
        ⊕ <xvalues ⊕ >...</xvalues>
        ⊕ <pages ⊕ >...</pages>
```

2. 値を抽出するには
 - a. VendorID の値と alternative を確認するには、対応する 2 番目のフィールドを展開します。
 - b. field 要素を右クリックし、[ここから値を抽出] > [属性] > [validatedText Into] > [Invoice: invoice] > [VendorID: Text] の順にクリックします。

```
⊖ <field ⊖ name="VendorID" page="0" confidence="0.613246" errorD
  formattingFailed="false" valid="true" validatedText="198000" verifie
  198000
  ⊕ <alternative ⊕ >198000</alternative>
```

注 テキストの 198000 をクリックしないようにしてください。

新しい [値を抽出] ステップがワークフローに挿入されます。ステップ名を「**Extract Vendor ID**」に変更します。

- c. このアクションをフィールド 17 の InvoiceNumber とフィールド 18 の InvoiceDate に対して繰り返します。

新しい [値を抽出] ステップがワークフローに挿入されます。各ステップ名を「**Extract Invoice Number**」および「**Extract Invoice Date**」に変更します。

- d. 最後から 3 番目のフィールド TotalAmount を右クリックし、[ここから値を抽出] > [属性] > [doubleValue Into] > [Invoice: invoice] > [Total: Number] をクリックします。

変換プロジェクトはこのフィールドに数値フォーマッタを持っているため、フィールドは数値に変換されます。

[値を抽出] ステップがもう 1 つワークフローに挿入されます。ステップ名を「**Extract Total Number**」に変更します。

3. [実行を開始] ▶ をクリックし、ワークフローを最後まで実行します。

[値を抽出] ステップが実行されると、抽出された値を確認することができます。右側の [ワークフロー状態] ペインで、[変数] 分岐を展開します。

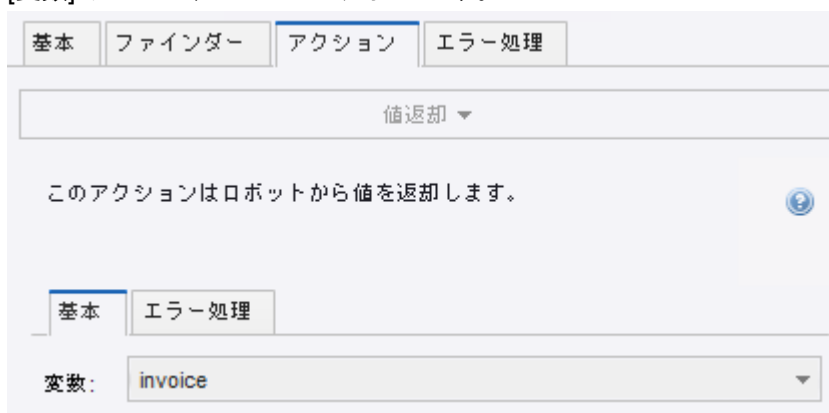
4. 変更を保存し、Desktop Automation ロボットからステップアウトします。

Desktop Automation ロボットからステップアウトして、Web オートメーション ロボットでの作業に切り替えるには、Desktop Automation ワークフロー全体を実行し、ツールバーの [ステップアウト] をクリックします。Web オートメーション ロボットで、「Desktop Automation ワークフローの呼び出し」ステップは実行されていると表示されます。

これで、Document Transformation ロボットは領収書の詳細のすべてに対応できるようになりました。Design Studio の [変数] セクションで **invoice** 変数を確認します。

5. Design Studio で、別の [アクション ステップ] を Web オートメーション ロボットに追加し、[値返却] ステップにします。ツールバーで、[選択したステップの後にステップを挿入] > [アクション ステップ] の順にクリックし、[アクション] タブで [値返却] の順に選択します。

[変数] リストで、**invoice** を選択します。



この方法で、ロボット デバッガに invoice パラメータを表示することができます。

6. 変更を保存します。

これで、Management Console で実行される有効なロボットが作成できました。このロボットは、Document Transformation クライアントによって呼び出されます。

重要 変換が完了する前に、ドキュメントのバックアップ コピーがあることを確認してください。変換が終了した場合、またはエラーが発生した場合には、ドキュメントおよびデータは格納されません。

終了したら次のセクションに進みます。

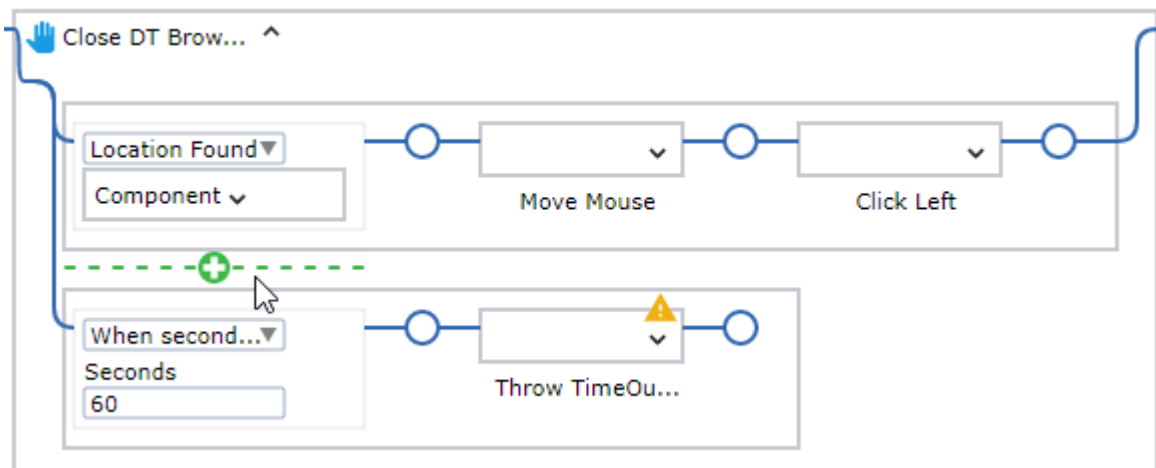
人による検証のステップの設定

この手順では、人による検証のためにドキュメントを送信するシナリオを処理するガードを設定し、人による検証のワークフローに特別な条件分岐を追加します。

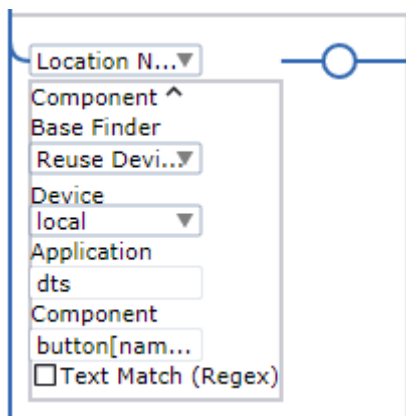
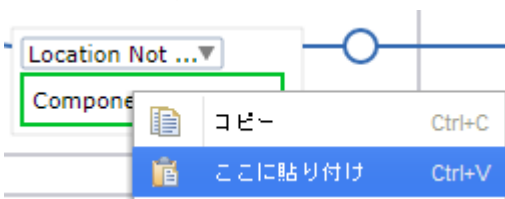
Transform Document ステップを再度開く必要があります。ステップを再度開く前にロボットを更新し、前回のロボットの実行で開いた Document Transformation ブラウザ ウィンドウが、現在は閉じていることを確認します。

ヒント: または、クリック ステップを追加してロボットの実行終了後にブラウザ ウィンドウを閉じるようにすることもできます。これを実行するには、Desktop Automation ロボットでブラウザの [閉じる] ボタンを右クリックし、[クリック] > [左] の順にクリックします。新しいステップがワークフローに追加されます。

新しいステップを展開し、中間にある緑のプラス アイコンをクリックして新しいガードを追加します。



新しいガードを [ロケーションが見つからなかった場合] に設定し、[ロケーションが見つかった場合] ガードから [コンポーネント] を [ロケーションが見つからなかった場合] ガードの [コンポーネント] ボックスにコピーします。



作成したステップは、Document Transformation ブラウザが開いていた場合にはすぐに閉じ、そうでない場合には実行を続行します。ステップの名前を「Close DT Browser」に変更します。



完了したら、以下の手順に進んで人による検証を構成します。この手順では、検証されたドキュメントが Document Transformation クライアントから検証コールバックを経由して戻されたケースを扱います。

1. **Document Transformation** ステップの後にあるフロー ポイントをダブルクリックします。

Wait until Transformed ステップの「ロケーションが見つかった場合」ガードが、変換されたドキュメントを待機します。

Document Transformation クライアントは、人による検証の後にコールバック URL

`http://127.0.0.1:50080/rest/run/Default project/Examples/Transform/`

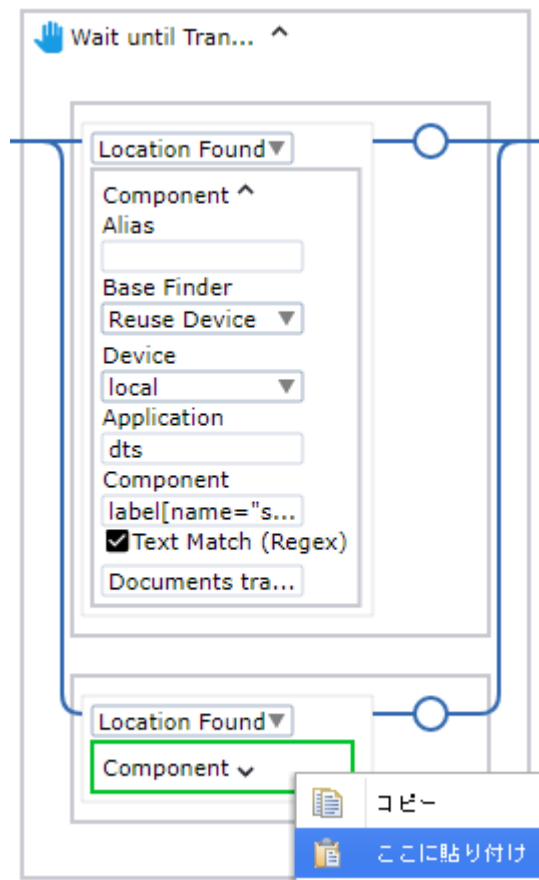
`DocumentTransformationInvoiceExample.robot` を呼び出すと、変換されたドキュメントを .zip ファイルとしてロボットの **document.doc** 変数に渡します。 .zip ファイルは次のようになります:

 project	2019/03/29 3:20	ファイルフォルダー	
 config.xml	2019/03/29 3:20	XMLドキュメント	1 KB
 document.tif	1980/01/01 0:00	TIF ファイル	42 KB
 document.xdc	2019/03/29 3:20	Kofax Transformation XDocument File	37 KB
 Folder.xfd	2019/03/29 3:20	XFD ファイル	1 KB
 tree.dat	2019/03/29 3:20	DAT ファイル	97 KB

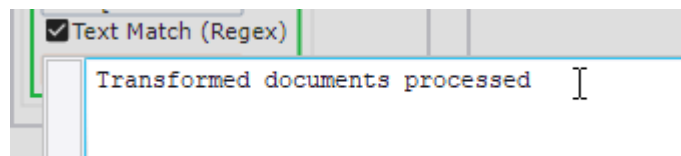
Document Transformation ブラウザで .zip ファイルを開くと、ステータスが「Transformed document processed」に変更されます。

2. 変換されたドキュメントが処理されるまで待機するようにガードを設定するには、**Wait until Transformed** ステップで、追加の [ロケーションが見つかった場合] ガードを作成し、最初の [ロ

ケースが見つかった場合] ガードの [コンポーネント] ボックスをコピーして 2 番目のガードの [コンポーネント] ボックスに貼り付けます。



2 番目のガードの [テキスト マッチ (Regex)] を「Transformed documents processed」に変更します。

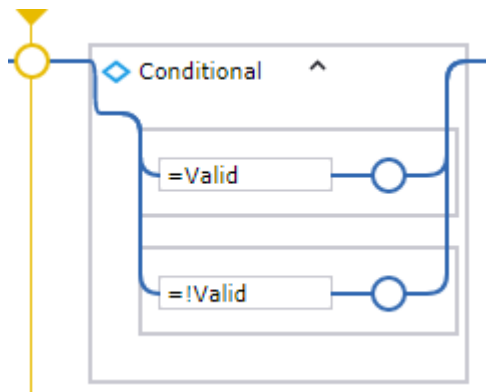


2 つのガードが設定され、ワークフローでは人による検証と、人を必要としない検証の 2 つのシナリオを処理できるようになりました。

ヒント このステップでは、オプションで追加のガードを設定して、変換中に発生する可能性のあるエラーを処理できます。詳細については、[エラー処理の設定](#)を参照してください。

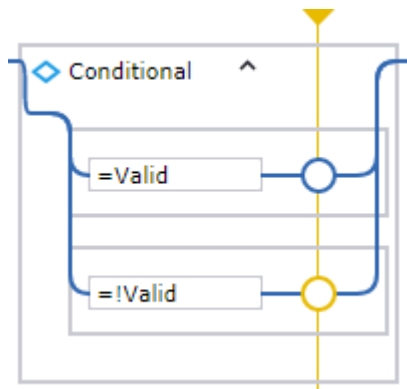
3. このポイントに対して実行する **Extract Document.valid** ステップの後のフローポイントをダブルクリックします。

4. 人による検証のために特別な分岐を追加します。フローポイントを右クリックし、[条件ステップ]をクリックします。新しいステップがワークフローに挿入されます。
 - a. 次の2つの条件を **Conditional** ステップに追加します。

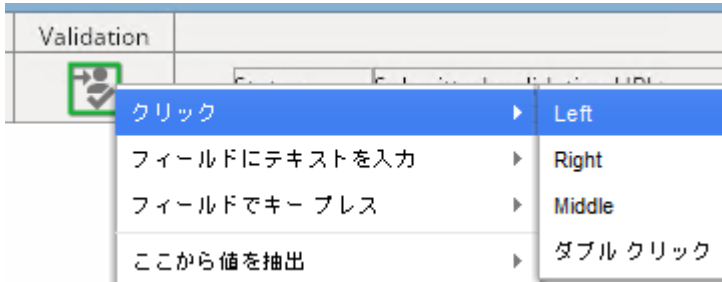



=!Valid の感嘆符は「not」を表します (「有効でない」)。

- b. [ステップ イントゥー] をクリックし、**Conditional** ステップに入ります。
この時点でロボットのドキュメントは有効ではないため、2 番目の条件 (分岐) が実行されま
す。



5. 次に、ドキュメントを Document Transformation クライアントに送信する必要があります。[レコーダービュー]で、検証ボタンを右クリックし、[クリック]>[左]の順にクリックします。

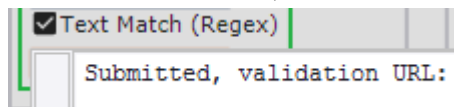



Conditional ステップの 2 番目の分岐に新しいステップが挿入されます。ステップ名を「**Send to Validation**」に変更します。[ステップ オーバー]  をクリックしてステップを実行します。

- a. ステータスが「**Submitted, validation URL:**」に変更されたらクリックし、[ガード]>[ロケーションが見つかった場合]の順にクリックします。

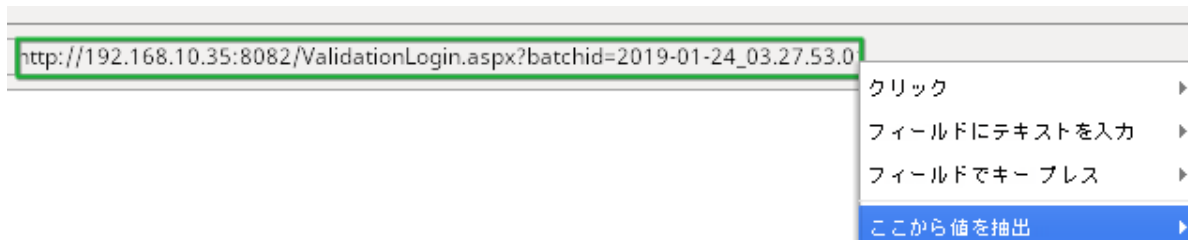
Conditional ステップの 2 番目の分岐に新しいステップが挿入されます。ステップ名を「**Wait for URL**」に変更します。

- b. **Wait for URL** ステップで、[コンポーネント] ボックスを展開し、[テキスト マッチ (Regex)] を選択し、「**Submitted, validation URL:**」と入力します。

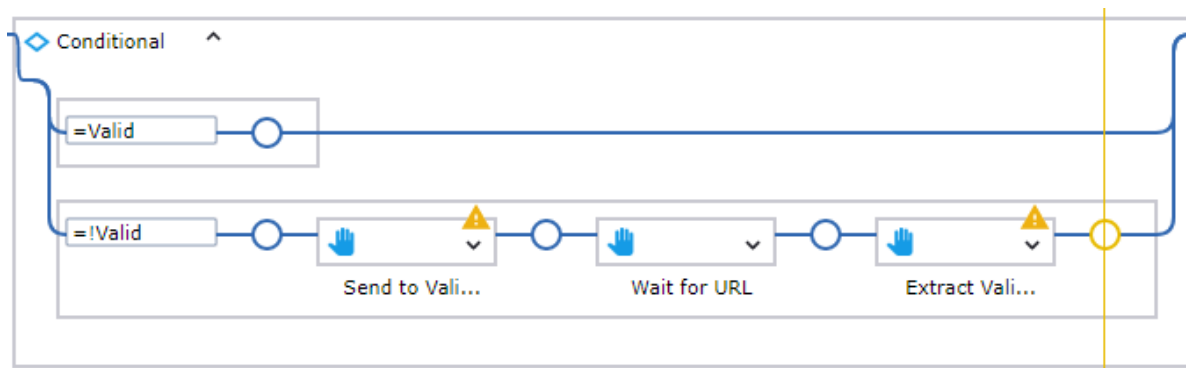


- c. [ステップ オーバー]  をクリックします。

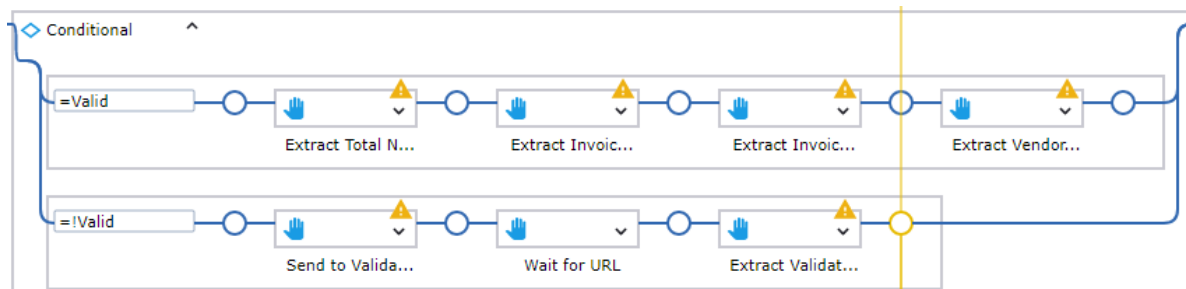
6. 以下の URL をクリックし、[ここから値を抽出] > [テキスト イントゥー] > [ValidationURL: Text] の順にクリックします。



「値を抽出」ステップが **Conditional** ステップ内に挿入されます。ステップ名を「**Extract Validation URL**」に変更し、実行します。



7. **Conditional** ステップの後の 4 つの [値を抽出] ステップを選択し、**Conditional** ステップの =Valid 分岐にカット アンド ペーストします。



8. 変更を保存します。

これで変換ワークフローは終了です。[実行を開始] をクリックし、ワークフローを最後まで実行します。これでワークフローには 3 つの可能なパスがあることになります。

- 新しいドキュメントは抽出と分類の処理のために Document Transformation サービスに送信され、「有効」が戻る。ドキュメントには人による検証は不要である。
- 新しいドキュメントは抽出と分類の処理のために Document Transformation サービスに送信され、「無効」が戻る。ドキュメントには人による検証が必要である。
- 人による検証後にドキュメントの .zip パッケージが Document Transformation クライアントから送信され、「有効」が戻る。検証 URL が取得される。

完了したら、Desktop Automation ロボットを終了して [ロボットが無効なドキュメントと有効なドキュメントを処理できるように構成](#) に進みます。

エラー処理の設定

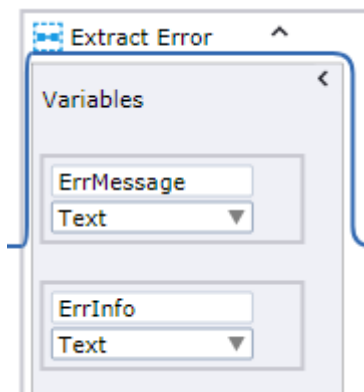
オプションでエラー処理を設定して、変換プロセス中に問題が発生した場合にエラーが報告され、エラーに関する情報が変数に保存されるようにできます。後で、エラー情報を含む [電子メールを人に送信](#) するように設定できます。

次の手順は、**Wait until Transformed** ステップでエラー処理のガードを設定する方法を説明します。インストールで提供される *DocumentTransformationInvoiceExample* ロボットには、**Close DT Browser** および **Extract Document.valid** ステップに「数秒が経過したとき」ガードが追加されています。この2つのステップは同様な方法で設定されています。

1. **Wait until Transformed** ステップで、2番目の [ロケーションが見つかった場合] ガードの下に別の [ロケーションが見つかった場合] ガードを作成し、2番目のガードから [コンポーネント] ボックスをコピーして、3番目のガードの [コンポーネント] ボックスに貼り付けます。
2. コピーした [コンポーネント] ボックスを展開し、次の変更を行います。
 - [コンポーネント] フィールドで、値を `label[name="status_title"]` に変更します。
 - [テキスト マッチ (Regex)] を **Error** に変更します。

このガードにより、ワークフローは別のシナリオを処理します (エラーが報告された場合)。他の2つのシナリオについては、[人による検証のステップの設定](#) で説明されています。

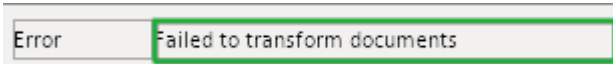
3. [ロケーションが見つかった場合] ガードの横にあるフロー ポイントを右クリックし、[グループ ステップ] を追加します。名前を「**Extract Error**」に変更します。
ステップを展開し、[変数] ボックスを展開して、次のようにテキスト タイプの2つの変数を追加します。



これらの変数には、エラーが発生した場合にエラー メッセージ名と情報が格納されます。

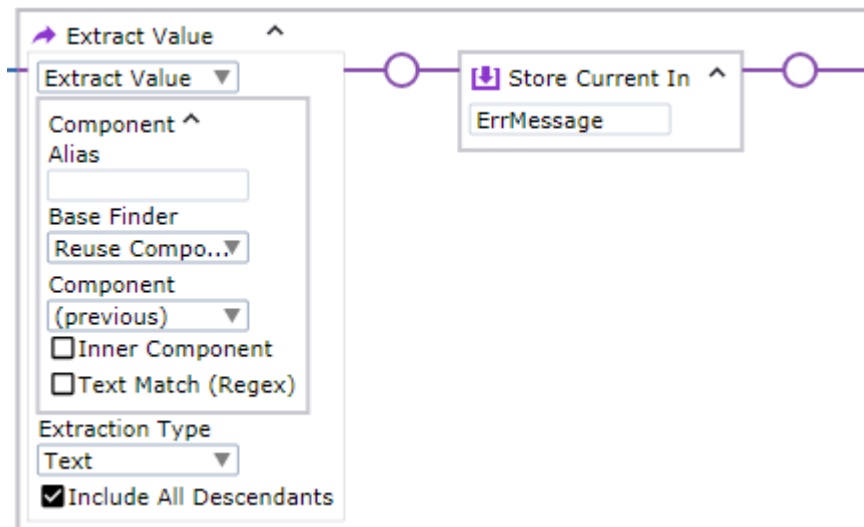
4. このポイントに対して実行する [変数] ボックスの横にあるフロー ポイントをダブルクリックします。

5. [レコーダー ビュー] で、[Failed to transform documents] ボタンを右クリックし、[ガード] > [ロケーションが見つかった場合] をクリックします。



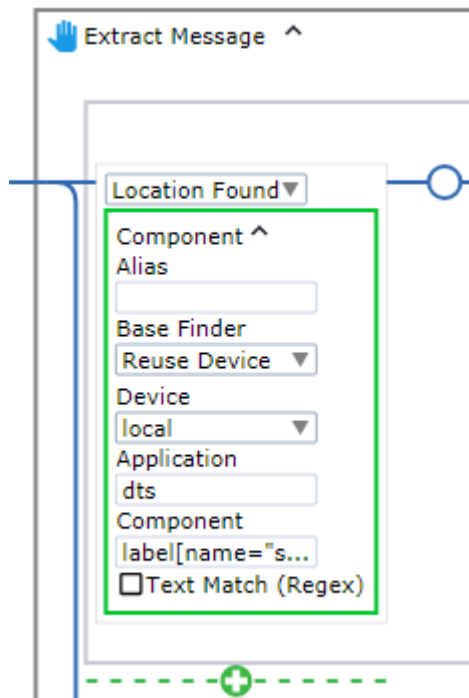
新しいガードがワークフローに挿入されます。これは、エラー メッセージ名を抽出するガードです。名前を「**Extract Message**」に変更します。

- [ロケーションが見つかった場合] ボックスの横のフローポイントを右クリックし、[値の抽出 ステップ] を追加して、次のように設定します。

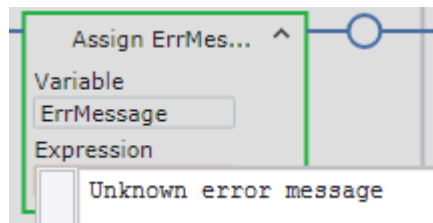


このステップでは、エラーが報告された場合、エラー名が変数 **ErrorMessage** に保存されます。

- [ロケーションが見つかった場合] ボックスの下プラス記号をクリックして、新しいガードを追加します。

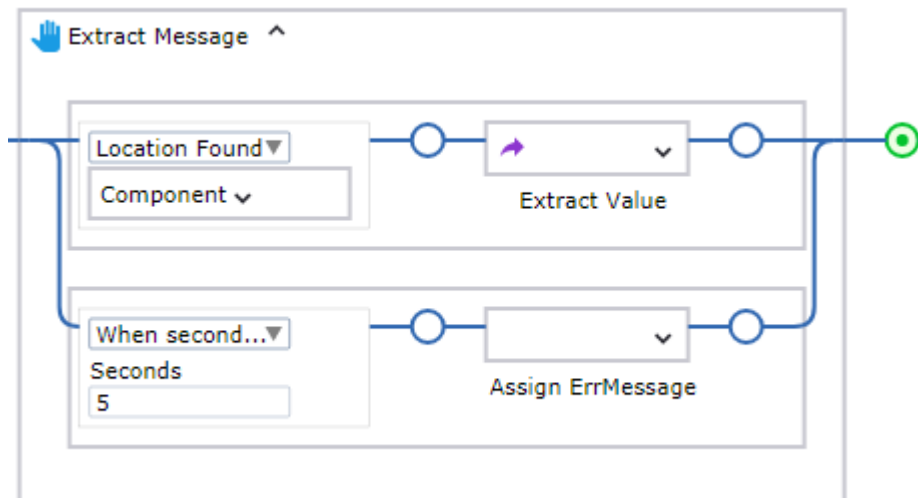


- デフォルトでは、「数秒が経過したとき」タイプのガードが挿入されます。5秒に設定します。
- [数秒が経過したとき] ガードの横にあるフローポイントを右クリックし、[割り当てステップ]を追加し、名前を「**Assign ErrorMessage**」に変更します。
割り当てステップを次のように設定します。



この2つのステップで不明なエラーが報告された場合、該当するエクスペッションが変数 **ErrorMessage** に保存されます。

6. このポイントに対して実行する **Extract Message** ステップの横にあるフロー ポイントをダブルクリックします。



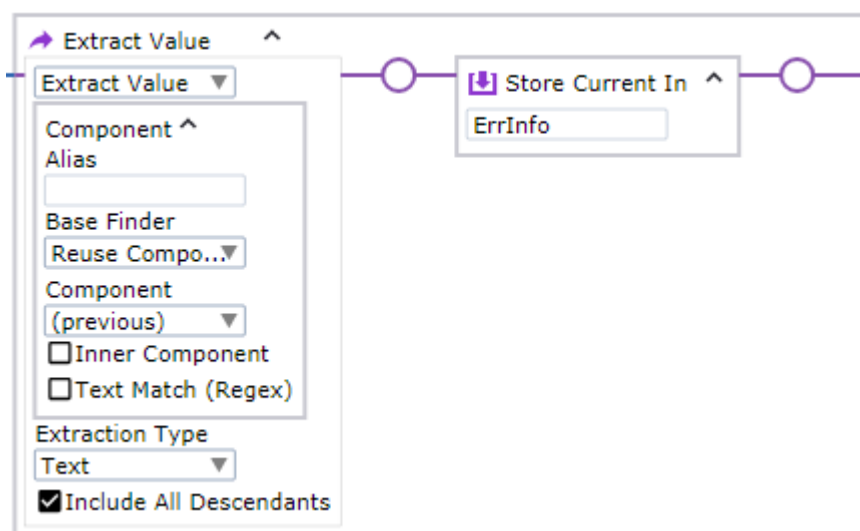
7. [レコーダー ビュー] で、エラーの説明が示されているボタンを右クリックし、[ガード] > [ロケーションが見つかった場合] をクリックします。

Error: connect ECONNREFUSED 127.0.0.1:50081

Document Transformation Service がターゲット コンピュータで実行されていない場合に表示される接続拒否エラーです。

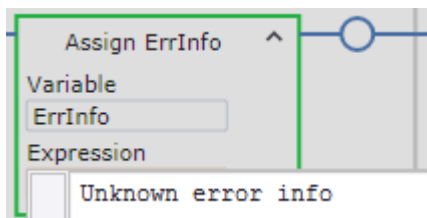
新しいガードがワークフローに挿入されます。エラーに関する情報を抽出するガードです。名前を「**Extract Info**」に変更します。

- [ロケーションが見つかった場合] ボックスの横のフローポイントを右クリックし、[値の抽出ステップ] を追加して、次のように設定します。



このステップでは、エラー メッセージ情報が変数 **ErrInfo** に保存されます。

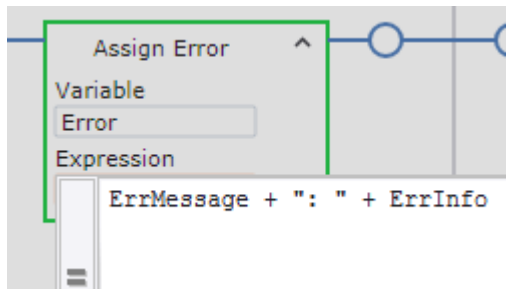
- [ロケーションが見つかった場合] ボックスの下のプラス記号をクリックして、新しいガードを追加します。
「数秒が経過したとき」を 5 秒に設定します。
- [数秒が経過したとき] ガードの横にあるフローポイントを右クリックし、[割り当て ステップ] を追加し、名前を「**Assign ErrInfo**」に変更します。
割り当てステップを次のように設定します。



この 2 つのステップで不明なエラーが報告された場合、該当するエクスペッションが変数 **ErrInfo** に保存されます。

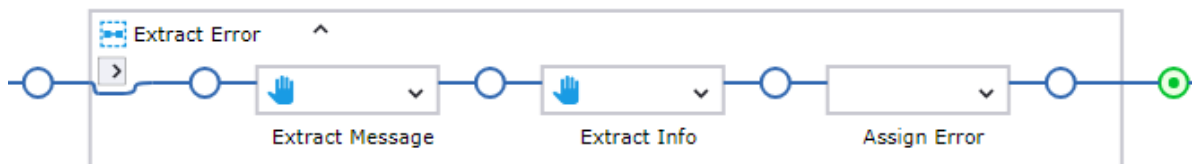
8. このポイントに対して実行する **Extract Info** ステップの横にあるフローポイントをダブルクリックします。

9. [Extract Info] ステップの横にあるフロー ポイントを右クリックし、[割り当て ステップ] を追加し、名前を「Assign Error」に変更します。
- [変数] フィールドに「Error」と入力します。Document Transformation ワークフローの作成で指定した変数の1つです。
 - [エクスペッション] フィールドに、次のエクスペッションを入力します。ErrorMessage + ": " + ErrInfo
左の灰色のバーをクリックすると等号が表示され、エクスペッションを評価できます。

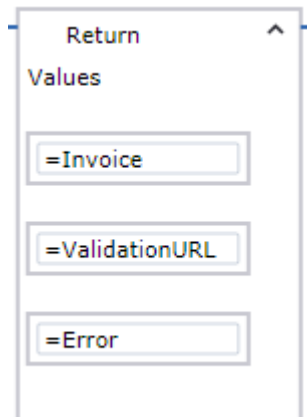


このステップでは、「Error」変数はエラー メッセージ名とエラー情報の両方を返します。

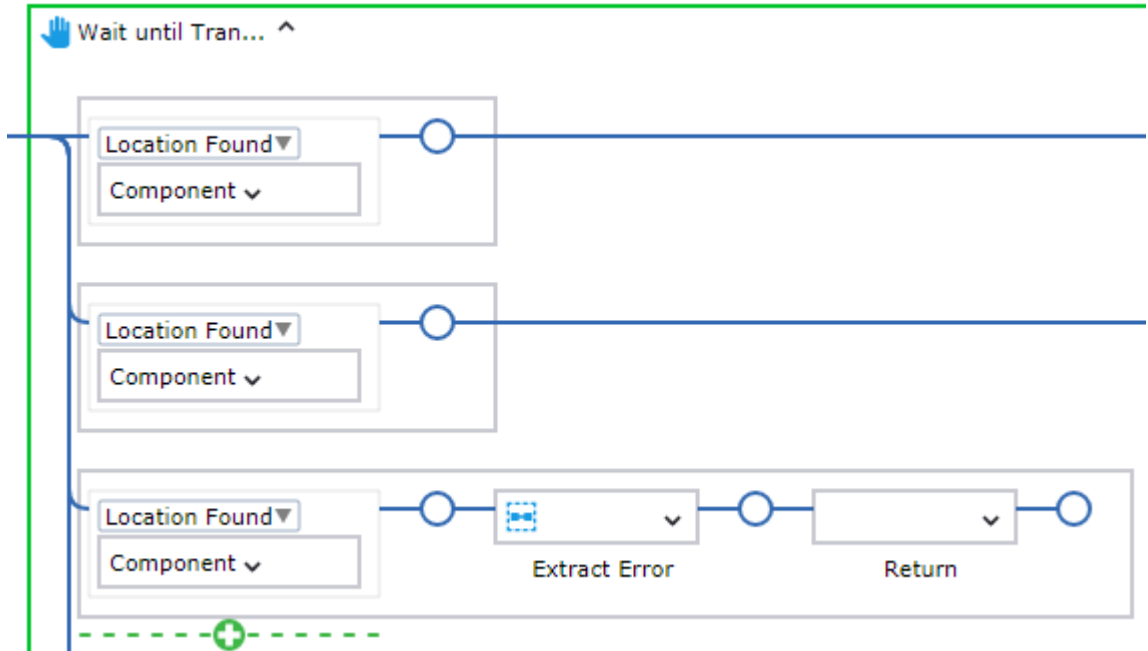
Extract Errorグループは、次のようになります。



10. グループの横にあるフロー ポイントを右クリックし、[リターン ステップ] を追加します。次のように設定します。



11. 次に、3番目の [ロケーションが見つかった場合] ガードの下にあるプラス記号をクリックして、別のガードを追加します。



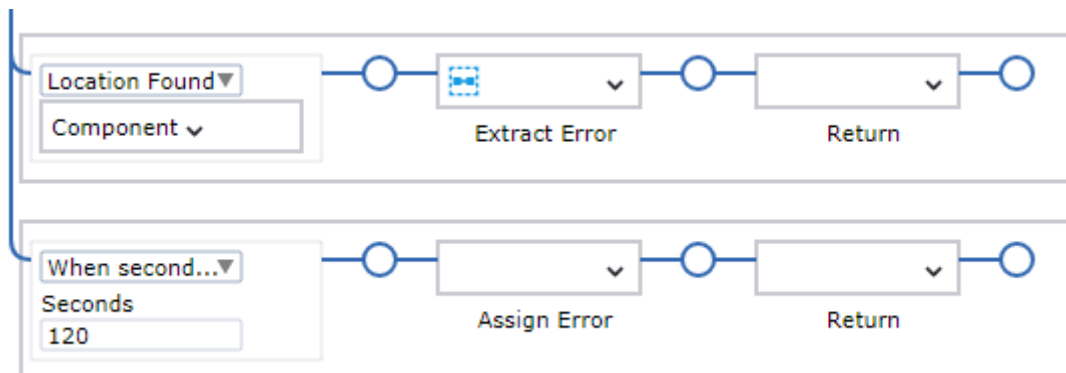
- 「数秒が経過したとき」を **120** 秒に設定します。
- このガードの横に、[割り当て ステップ] を追加し、名前を「**Assign Error**」に変更します。
- [変数] フィールドに「**Error**」と入力します。
- [エクスペッション] フィールドに、次のテキストを入力します。

Waiting for transformation timed out

- **Assign Error** ステップの後、ステップ 10 と同じ [リターン ステップ] を追加します。ステップをコピーして貼り付けることができます。


これらのステップを使用して、変換プロセスに時間がかかる場合、上記で指定したエクスペッションが戻り変数「**Error**」に格納されます。

Wait until Transformed ステップのエラー処理部分は、次のようになります。



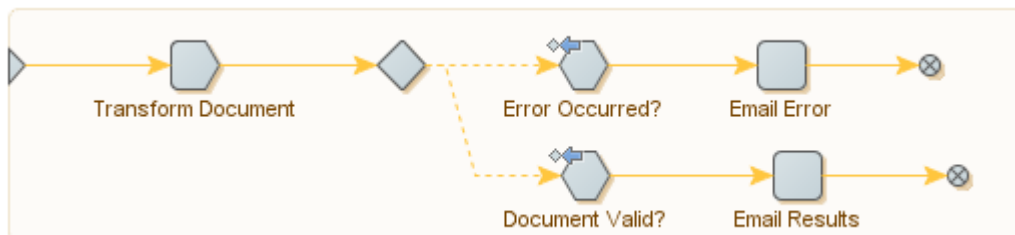
ロボットが無効なドキュメントと有効なドキュメントを処理できるように構成

この手順では、Design Studio のロボットに分岐を設定し、3つのシナリオ (ドキュメントが有効な場合と無効な場合、または変換中にエラーが発生した場合) を処理できるようにします。ドキュメントが有効な場合、ロボットは Transform Document ステップで抽出した値を電子メールで人に送信します。変換されたドキュメントが無効な場合、人による検証のために送信されます。エラーが発生した場合、エラー情報を含む電子メールが人に送信されます。

1. **Design Studio** で Transform Document ステップの後に [トライ ステップ] を追加します。
Transform Document ステップをクリックし、ツールバーで [選択したステップの後にステップを挿入]  > [トライ ステップ] の順にクリックします。
2. 最初の分岐に [アクション ステップ] を追加し、[値判定] ステップにします。
 [値返却] ステップを右クリックし、[ステップを前に挿入] > [アクション ステップ] の順にクリックし、[アクション] タブ上で [テスト値] を選択します。
 - a. [条件] セクションで「**error==""**」と入力します。直線型の引用符を使用します。
 - b. [If] および [Do] を [条件が満たされています] および [[エラー処理] 時に指定されたとおり] にそれぞれ設定します。
 - c. [エラー処理] タブの [Then] リストで [次の代替手段を試行] を選択します。
 - d. ロボットで、[テスト値] ステップの名前を「**Error Occurred?**」に変更します。
3. [Error Occurred?] ステップの後に [アクション ステップ] を追加し、[メール送信] ステップにします。
 [アクション] タブで、[出力] > [メール送信] を選択します。
4. [メール送信] ステップを次のように設定します。
 - a. [送信元アドレス] フィールドと [宛先アドレス] フィールドに、送信者と受信者の電子メール アドレスをそれぞれ入力します。
 - b. [メッセージ] フィールドに [エクスペッション] を設定し、以下のエクスペッションを追加します。
"Something went wrong during transformation or subsequent processing.\n" + error
 - c. [メッセージ タイプ] を [テキスト] に設定します。
 - d. [件名] フィールドに、次のようなテキストを入力します。"Transformation has failed."
 - e. [サーバー] タブで、SMTP サーバー設定を行います。
 - f. ロボットで、ステップの名前を「**Email Error**」に変更します。

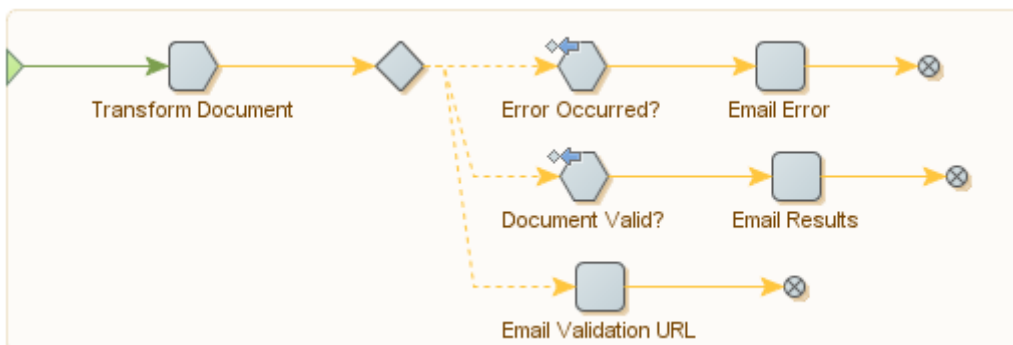


5. 2 番目の分岐に [アクション ステップ] を追加し、[テスト値] ステップにします。
 - a. [条件] セクションで、「**ValidationURL=""**」と入力します。直線型の引用符を使用します。
 - b. [If] および [Do] を [条件が満たされていません] および [[エラー処理] 時に指定されたとおり] にそれぞれ設定します。
 - c. [エラー処理] タブの [Then] リストで [次の代替手段を試行] を選択します。
 - d. ロボットで、[値判定] ステップの名前を「**Document Valid?**」に変更します。
6. [値返却] ステップをクリックして、[メール送信] ステップにします。
7. 2 番目の [メール送信] ステップを次のように設定します。
 - a. [送信元アドレス] フィールドと [宛先アドレス] フィールドに、送信者と受信者の電子メールアドレスをそれぞれ入力します。
 - b. このステップの [メッセージ] フィールドに [エクスペッション] を設定し、以下のエクスペッションを追加します。
"VendorID=" + invoice.VendorID + "\n" + "InvoiceNumber=" + invoice.InvoiceNumber + "\n" + "InvoiceDate=" + invoice.InvoiceDate + "\n" + "Total=" + invoice.Total
 - c. [メッセージ タイプ] を [テキスト] に設定します。
 - d. [件名] フィールドに、次のようなテキストを入力します。"This is the result of the robot execution."
 - e. [サーバー] タブで、SMTP サーバーを設定し、変更を保存します。
 - f. ロボットで、ステップの名前を「**Email Results**」に変更します。



8. 3 番目の分岐に [アクション ステップ] を追加し、[メール送信] ステップにします。
9. 2 番目の [メール送信] ステップを次のように設定します。
 - a. [送信元アドレス] フィールドと [宛先アドレス] フィールドに、送信者と受信者の電子メールアドレスをそれぞれ入力します。
 - b. このステップの [メッセージ] フィールドに [エクスペッション] を設定し、以下のエクスペッションを追加します。
>><a href="<<+ValidationURL+>>">Click me!<<
 - c. [メッセージ タイプ] を [HTML] に設定します。
 - d. [件名] フィールドに、次のようなテキストを入力します。"Please validate this invoice."
 - e. [サーバー] タブで、SMTP サーバー設定を行います。
 - f. ロボットで、ステップの名前を「**Email Validation URL**」に変更し、変更内容を保存します。

これでロボットが使用できるようになり、以下ようになります。



注 [メール送信] ステップはデザイン モードでは実行できないため、デバッグ モードで実行する必要があります。このモードに切り替えるには、ツールバーで [デバッグ] をクリックし、[シングル ステップ] をクリックします。

デザイン モードとデバッグ モードを同時に実行することもできません。

ドキュメントが有効な場合、ロボットは変換されたドキュメントからの次の値を電子メールで送信します: VendorID、Number、Date、Total。変換されたドキュメントが無効な場合、検証 URL を含む電子メールが人に送信され、人はその URL をクリックしてドキュメントを検証できます。変換プロセス中にエラーが発生した場合、エラー情報を含む電子メールが人に送信されます。

ロボットを Management Console にアップロードします。これでロボットを Design Studio でテストできるようになりました。

検証に必要なドキュメントはすべて以下の URL から利用可能です。

<http://127.0.0.1:50082/Validation.aspx>

この URL の **127.0.0.1** は Thin Client サーバーの IP アドレスであり、インストール場所に応じて異なります。

人による検証の手順については、次のセクションを参照してください。

変換されたドキュメントの検証

このトピックでは、変換された請求書を検証するために実行できる基本的なステップについて説明します。

ブラウザで検証 URL <http://127.0.0.1:50082/Validation.aspx> を開いた後、変換されたドキュメントを含むバッチを開きます。左側の [分類結果] ペインに、Document Transformation 結果が表示されます。

ヒント ドキュメント検証プロセスを容易にし、加速するには、可能な場合はいつでもキーボードを使用することをお勧めします。たとえば、バッチを開く場合は Ctrl + O をバッチを、サスペンド (保存して閉じておく) する場合は Ctrl + S を押します。ショートカットを使用するには、キーを同時ではなく 1 つずつ連続して押します。

1. [ID Number] フィールドの横にあるフィールドが選択され、ここで ID を確認する必要があります。

ID Number	<input type="text" value="198000"/>	<input type="text"/>
Company Code	<input type="text" value="3"/>	

正しい ID が抽出されたことを確認するには、Enter を押します。

ID Number	<input type="text" value="198000"/>	<input type="text" value="Confirmed"/>
Company Code	<input type="text" value="3"/>	

2. 次の無効である可能性があるフィールド [Invoice Number] (請求書番号) が選択されます。

- Invoice -

Invoice Number	<input type="text" value="67-90943"/>
----------------	---------------------------------------

ドキュメントの右上隅の数字を確認できます。赤くマークされ、検証が必要であることが通知されます。

Date: 06.11.2010
Invoice No: 67-90943

フィールドを検証し、正しい請求書番号が抽出されていることを確認するには、Enter を押します。

3. 次の無効である可能性があるフィールド [Invoice Date] (請求書日付) が表示されます。

Invoice Date
04.11.2010

ドキュメントの右上隅に、実際の日付が 06.11.2011 と表示されます。抽出された日付は、トレーニングの目的で意図的に無効になっています。

日付を修正するには、次のうちいずれかの方法に従います。

- 自動フィールド入力を使用します。まず、Backspace キーを使用して、正しくない日付をクリアします。ドキュメントのゼロで始まるすべての数字を表示するには、「0」と入力します。下矢印キーを押し、「06.11.2010」を選択します。

Invoice Date
06.11.2010
06.11.2010
04.11.2010
0000

ドキュメントの請求書日付は緑でマークされています。

Date: 06.11.2010
Invoice No: 67-90943

Enter を押して、新しい日付を確認します。

ヒント ドキュメントの検証時に、オンライン ラーニングを有効にして、同じようなドキュメントでのフィールド認識の確率を上げることができます。この機能は、請求書などのサンプルドキュメントのレイアウトの記憶に基づいています。自動フィールド入力を使用、ドキュメントに正しい値(正しい請求書日付など)を手動で入力または選択することにより、ナレッジベースに貢献します。その結果、次回同様のドキュメントを表示する際に、請求書日付は自動的に正しいものになります。
この機能は、[プロジェクト設定]の[一般]タブで有効になっています。

- ドキュメントで、請求書日付をクリック、または投げ縄をかけて選択し、Enter を押します。

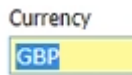
Date: 06.11.2010
Invoice No: 67-90943

4. 次の無効である可能性があるフィールド [PO Number] (注文書番号) が表示されます。

PO Number
4500017282

抽出された数字が正しいことを確認します。注文書番号が正しいことを確認すると、無効とマーク付けされていた [Document Type] (ドキュメント タイプ) フィールドの値が自動的に確認されます。

5. [Currency] (通貨) フィールドが選択されます。



抽出された通貨が正しいことを確認します。

すべての値を確認すると、バッチが有効であるという通知を受け取ります。これで通知を閉じることができます。抽出された値は、**ロボットが無効なドキュメントと有効なドキュメントを処理できるように構成**で設定したように、電子メールで人に送信されます。

これでチュートリアルが完了しました。