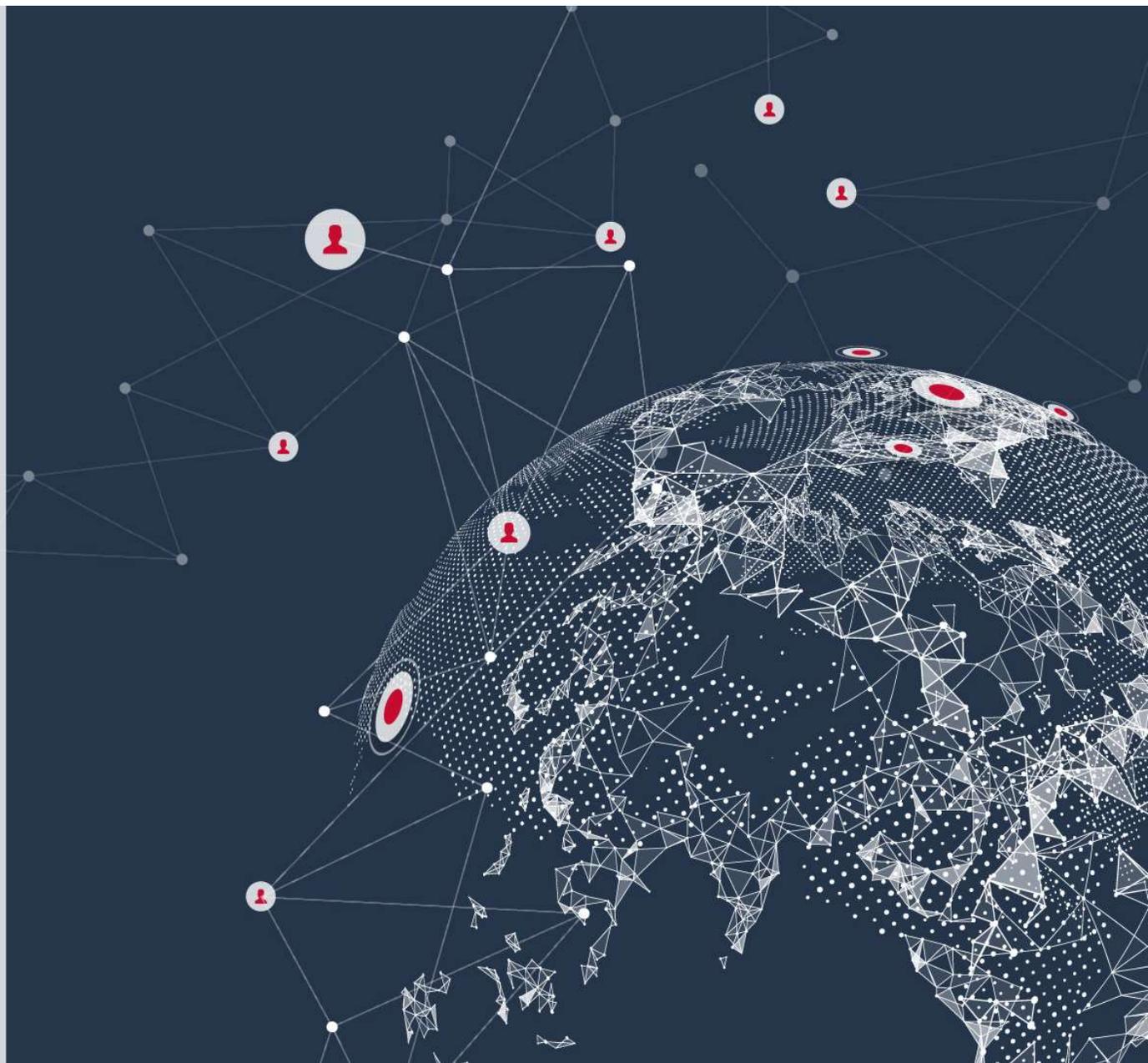


ABBYY®

FlexiCapture
文書定義應用

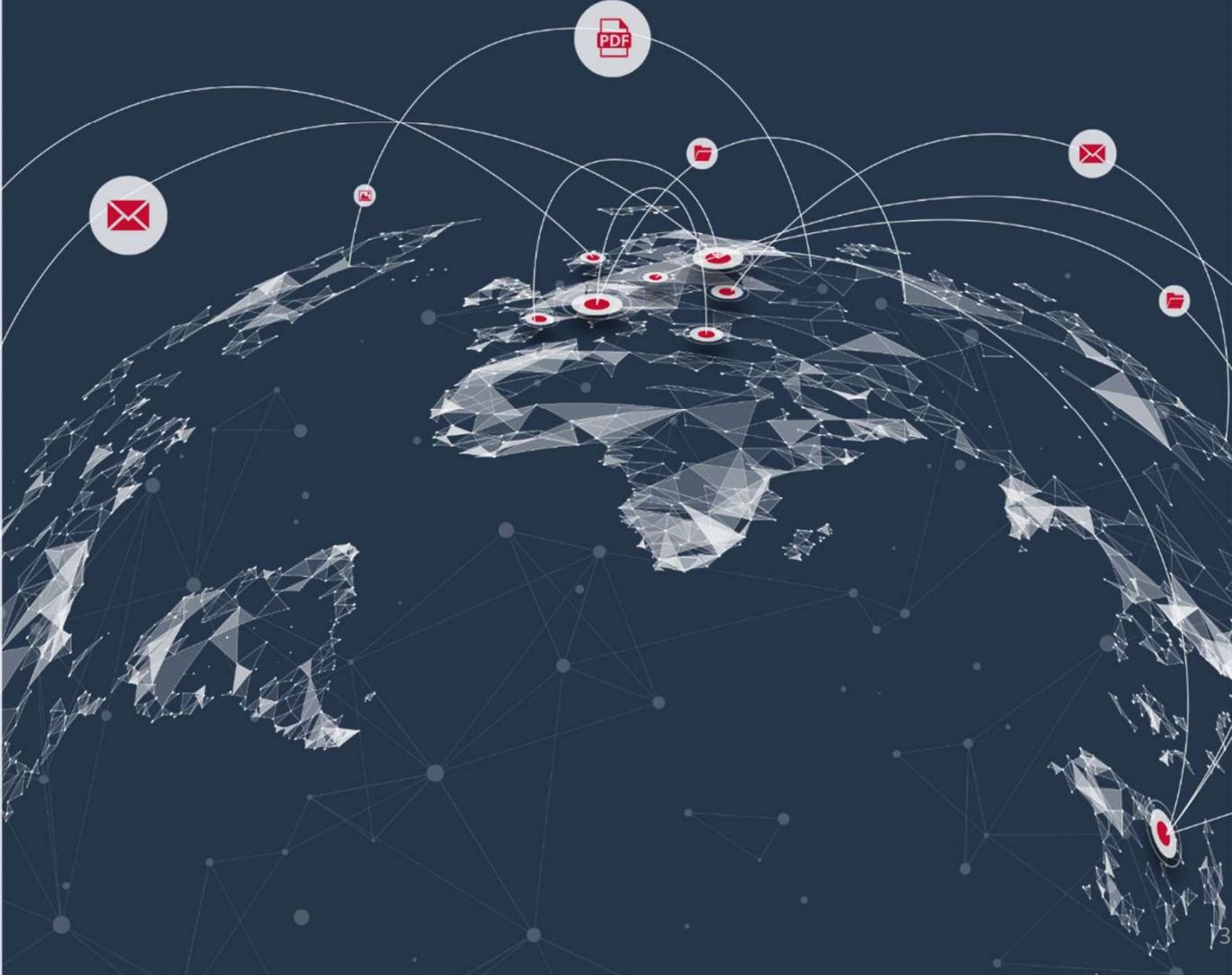


ABBYY FlexiCapture 文書定義応用



文書定義基礎では、ABBYY FlexiCapture 文書定義に関して最低限必要な設定方法を紹介しました。
本ドキュメントでは、より実践的な文書定義方法について、トピック毎に紹介していきます。

データタイプ・ オートコレクトオ プション

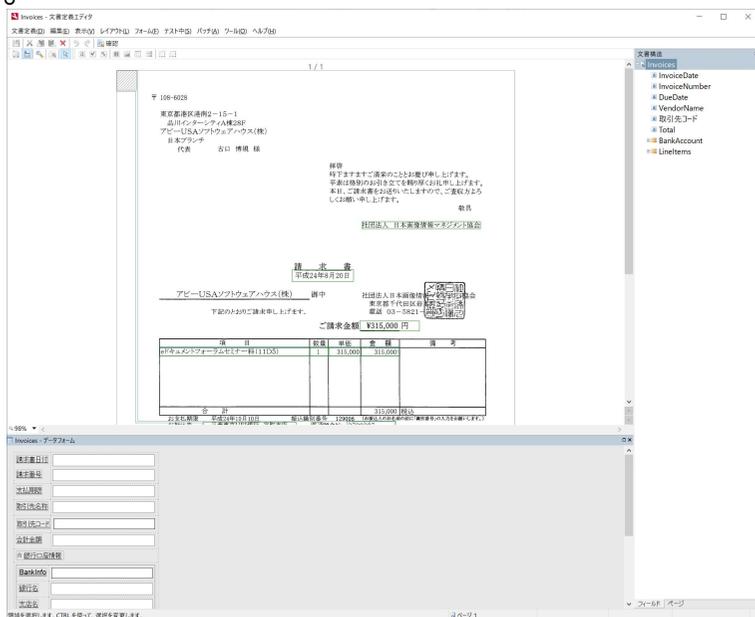


データタイプ

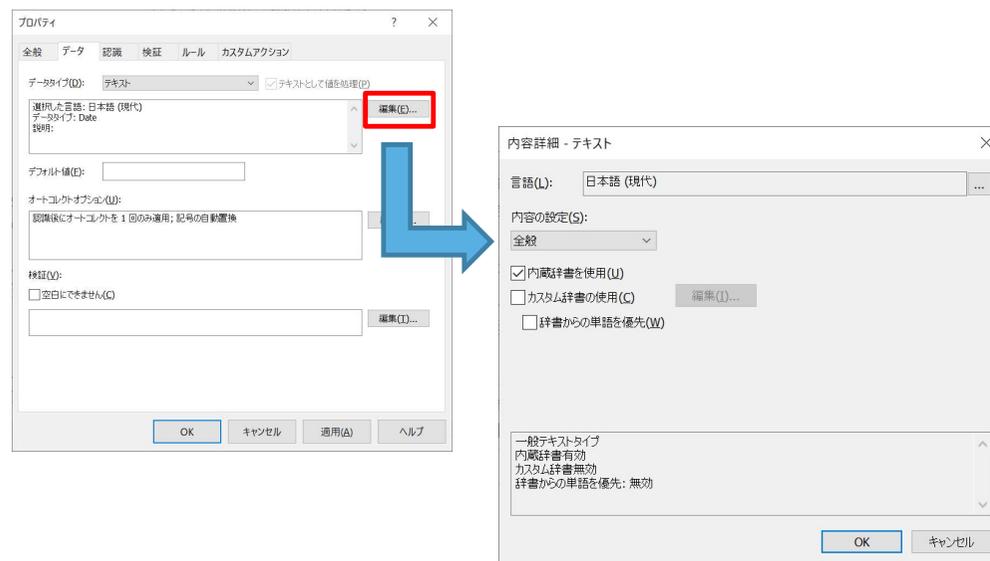


データタイプでは、フィールド毎の認識設定、特に認識対象文字種の設定を行います。
以下、ステップバイステップの設定方法を紹介します。

1. 文書定義エディタを開く。
プロジェクト > 文書定義を選択し、文書定義ダイアログより文書定義を選択、編集ボタンクリックにより、文書定義エディタを開く。



2. 対象フィールドのプロパティ > データタブを開く。
適切な「データタイプ」を選択し、編集ボタンをクリックする事で詳細設定が可能。



データタイプ

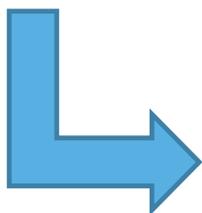
3. データタイプ: 特殊

以下のように、文字列の出現規則が決まっている場合、規則に応じて特殊データタイプの設定を行う。(*1)

*1: どのようなデータタイプ設定が最適かは、実サンプル画像を用いてテスト、確認する。

請求書
平成24年8月20日

【請求書日付の例】



内容詳細 - テキスト

言語(L): 日本語 (現代)

内容の設定(S):

全般
全般
特殊

カスタム辞書の使用(C) 編集(I)...

辞書からの単語を優先(W)

一般テキストタイプ
内蔵辞書有効
カスタム辞書無効
辞書からの単語を優先: 無効

OK キャンセル

4. 特殊データタイプ設定画面にて、新規ボタンをクリック

内容詳細 - テキスト

言語(L): 日本語 (現代)

内容の設定(S):

特殊

選択した言語のデータタイプ(I):

- AccountNumber
- Date
- InvoiceNo
- Price
- Qty

新規(W)...

編集(E)...

削除(D)

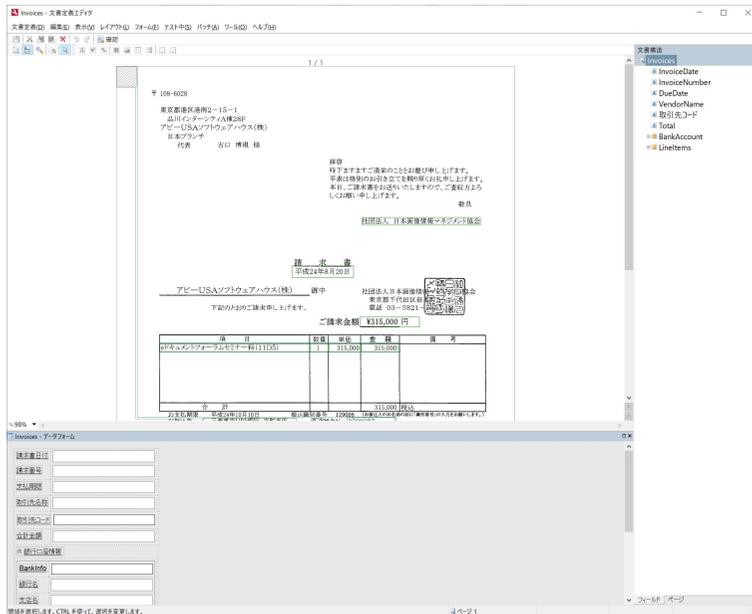
データタイプ: AccountNumber
説明:

OK キャンセル

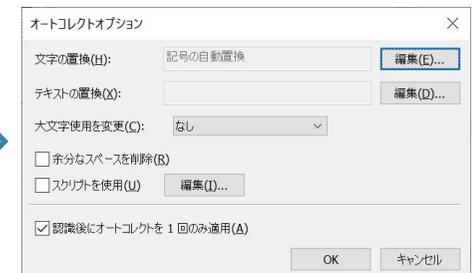
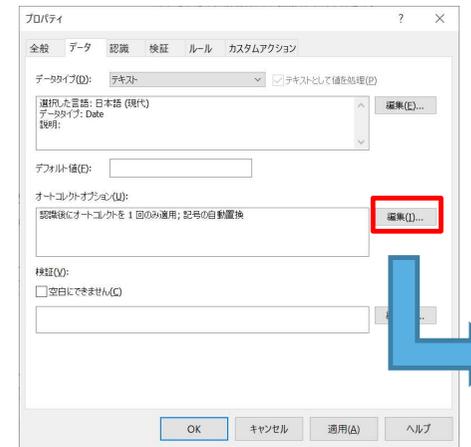
オートコレクトオプション

オートコレクトオプションでは、認識後の文字列置換・削除設定を行います。事前に結果文字列の規則性が分かっている場合、オートコレクトオプションによって認識精度を向上させることが可能です。以下、ステップバイステップにて設定方法を紹介します。

1. 文書定義エディタを開く。
プロジェクト > 文書定義を選択し、文書定義ダイアログより文書定義を選択、編集ボタンクリックにより、文書定義エディタを開く。

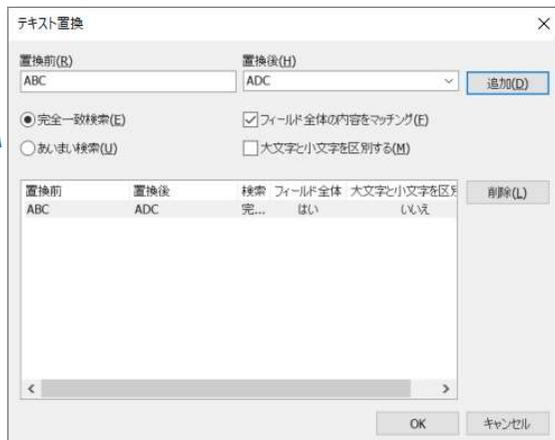
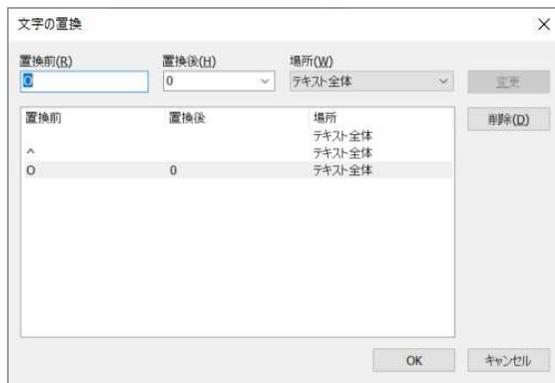


2. 対象フィールドのプロパティ > データタブを開く。
オートコレクトオプションにて編集ボタンをクリックし、オートコレクトオプション設定ダイアログを開く。

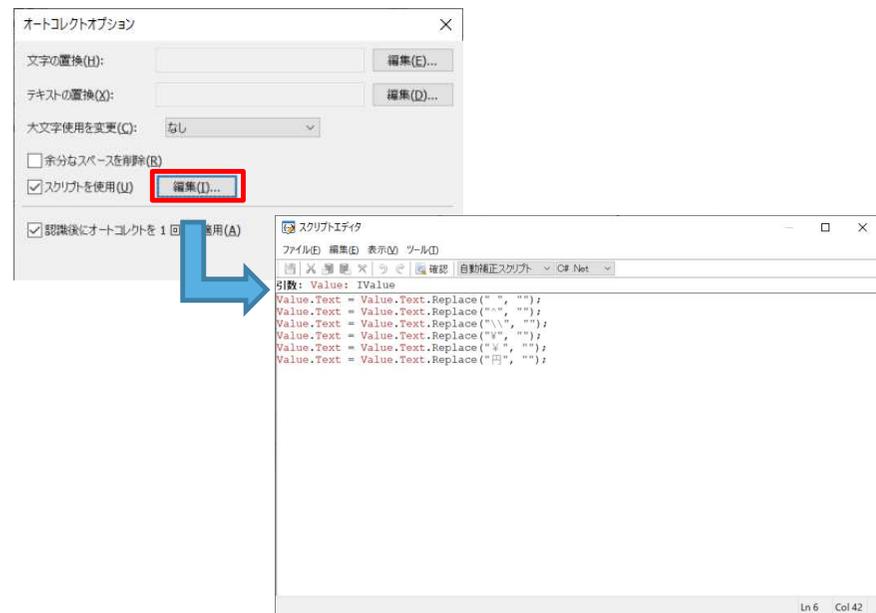


オートコレクトオプション

3. 置換: 文字の置換、テキスト置換設定が可能。



4. スクリプトでの処理も可能。
処理順は、スクリプト > 文字 / テキストの置換、となる。

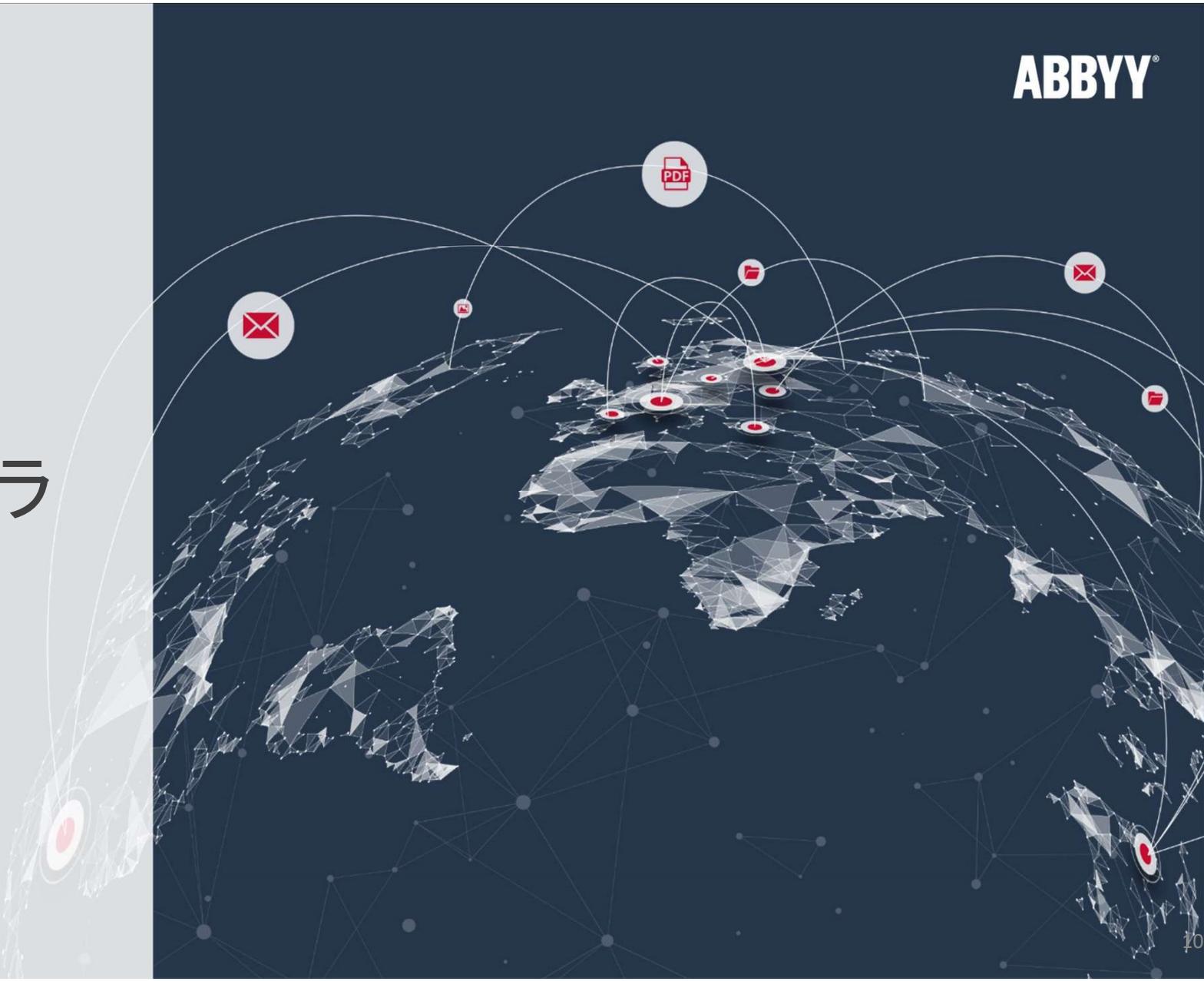


オートコレクトオプション

3. 「認識後にオートコレクトオプションを1回のみ適用」にチェックする事で、検証者による修正時は適用外とすることが可能。



イベントハンドラ



イベントハンドラ

FlexiCaptureでは、バッチ、文書、およびフィールドに影響するイベントのハンドラ(スクリプト)を作成できます。

イベントハンドラ詳細については、

https://help.abbyy.com/ja-jp/flexicapture/12/standalone_administrator/events

をご参照ください。

イベントハンドラは、バッチ毎に設定するものと、文書毎に設定するものとに分かれ、設定箇所も異なります。

1. バッチ毎

プロジェクト／バッチタイプのプロパティ > イベントハンドラより設定



2. 文書毎

文書定義のプロパティ > イベントハンドラより設定



イベントハンドラ

イベントハンドラの内、代表的な2つについて、以下に紹介いたします。

1. 文書状態変更後

文書の状態は以下の通り決められており、文書状態フラグ(のセット)として状態を保持しています。

本イベントハンドラは、文書状態が変化するたびに発生します。文書の状態の変化を記述するフラグのセットは、入力パラメータとしてイベントに渡されます。

文書状態を決定するフラグ

- Preprocessed – 予備処理済み
- Analyzed – 分析済み
- Recognized – 認識済み
- HasErrors – エラーあり
- Assembled – アセンブル済み
- Verified – 検証済み
- Exported – エクスポート済み
- HasLayout – レイアウトあり

※例えば、認識後アセンブリまで完了時、すなわち文書確定時、(ワンタイムで)処理を実施したい場合、「Assembled – アセンブル済み」フラグ条件によって処理を記述します(右図)。

C#でのスクリプト例

```

// Document の使用例
string text = Document.Field("Document Section 1¥¥Value").Text;

// ChangedStates の使用例
// Status: 2: recognized – 認識されたか
if( ChangedStates.Has((int)TStateType.ST_Recognized)==true )
{
    // 認識後の処理を記述
}

// Status: 4: Assembled – アセンブル済み (アセンブリが発生して成功しないと呼ばれない)
}
else if( ChangedStates.Has((int)TStateType.ST_Assembled)==true )
{
    // アセンブリが成功した際の処理を記述
}
    
```

イベントハンドラ

2. 文書ルールチェック後

フィールド内容を修正する等、認識結果が変更される毎に「文書ルール」がチェックされ、その直後「文書ルールチェック後」イベントが発生します。

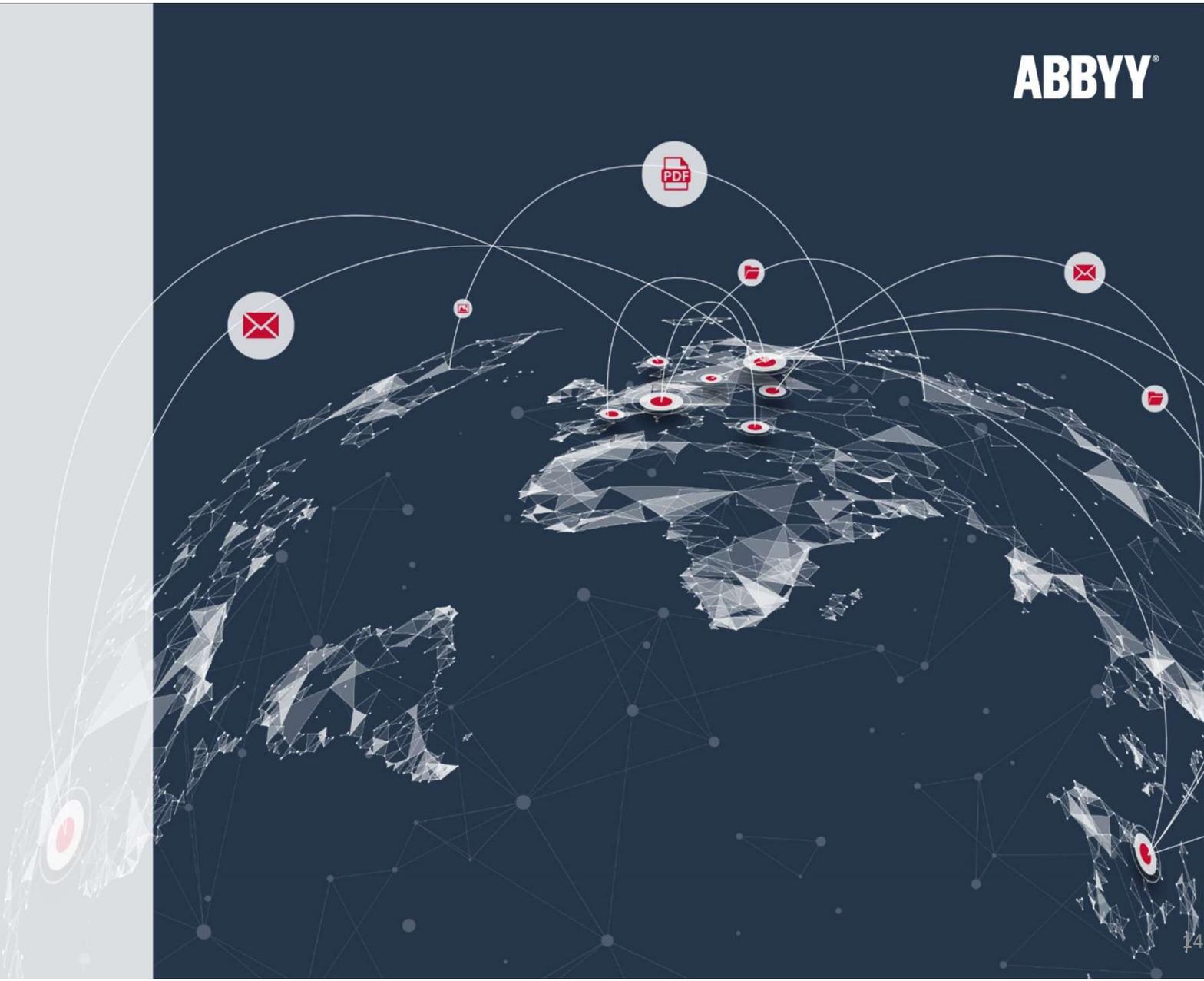
読み込みのフィールドとは別にオリジナルで作成したフィールド、表などがある場合、読み込みのフィールドを修正すると、オリジナルの変更がすぐに反映されるようにするため、「文書ルールチェック後」処理を設定します。

C#でのスクリプト例



```
// Document の使用例
string text = Document.Field("Document Section 1¥¥Value").Text;
```

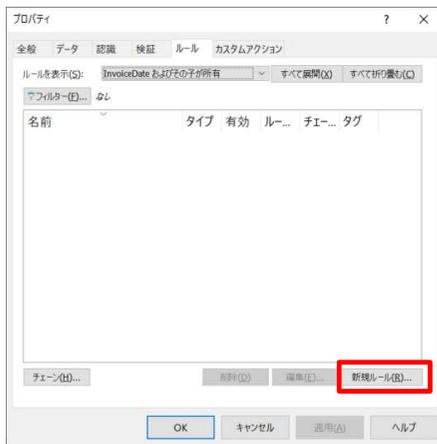
検証ルール



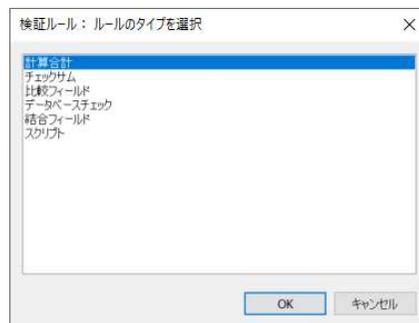
検証ルール

検証ルールによって、認識結果が間違っていないかチェックしたり、認識結果を編集することが可能です。検証ルールには、フィールド内／間で有効な物の他、文書構造チェックのようにセクションや文書単位で有効な物があります。全てのルールは、文書・フィールドの内容が変更される毎に呼び出されます。すなわち、認識直後のみではなく、検証時ドキュメントを修正・変更する度に呼び出されます。

検証ルール設定方法



フィールドのプロパティ



【GUIで設定可能なルール】

計算合計: 複数のフィールドの値の合計値を計算し、他のフィールドに記入する。

チェックサム: 複数のフィールドの値の合計値を計算し、事前に定められた値または他のフィールドの値と異ならないかを確認。

比較フィールド: 複数のフィールドの値を比較。

データベースチェック: データベースから取り出した同等の値と、取得した値を比較。

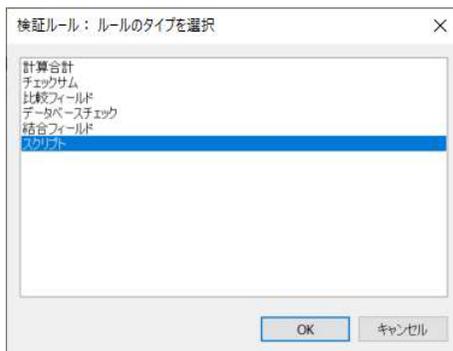
結合フィールド: 複数のフィールドの値を結合する。複数フィールドの値を一つに統合する際、任意のセパレーターを追加することが可能。

【スクリプトルール】

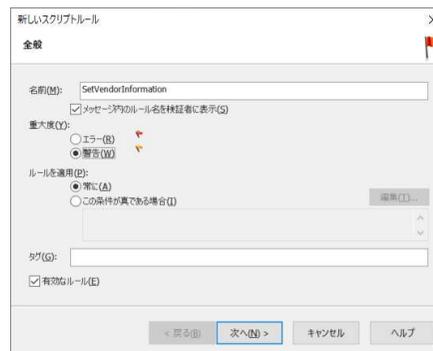
スクリプト: 特定データ(フィールド)に適用するスクリプトベースのルール。

検証ルール

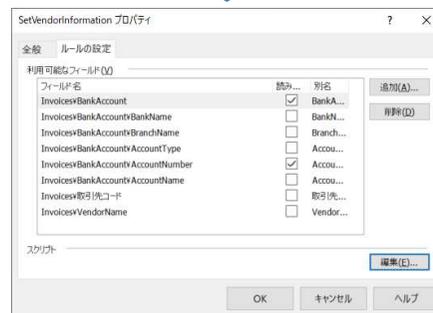
以下、スクリプトルールの例となります。



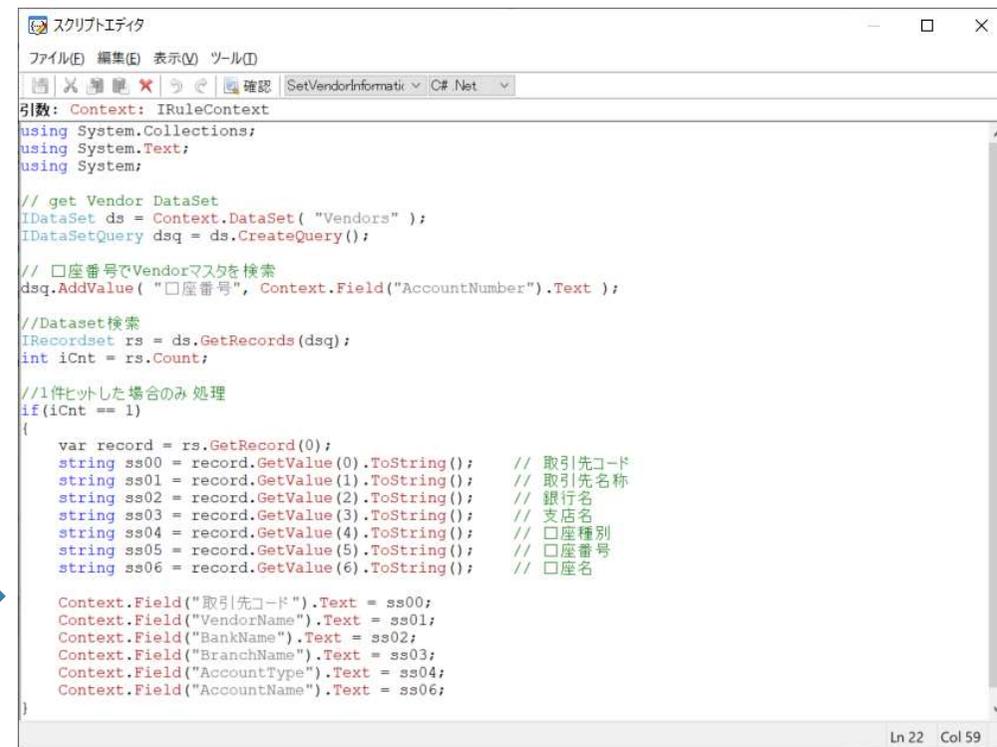
ルールタイプ: スクリプトを選択



ルール名、その他を設定



ルールに関係するフィールドを追加、選択



OCR結果: 口座番号をキーにVendor DataSetより関連ベンダ情報を取得、該当フィールドにセットするスクリプト

Thank you!!