



目次

- 1. 改訂情報
- 2. はじめに
 - 2.1. intra-mart Accel Platform への移行について
 - 2.2. IM-FormaDesigner への移行について
 - 2.3. 移行手順概要
- 3. ステップ 1
 - 3.1. 前提条件
 - 3.2. 制限事項
 - 3.3. 事前作業
 - 3.4. IM-VisualDesigner 移行手順
 - 3.5. 検証手順
 - 3.6. 運用手順
- 4. ステップ 2
 - 4.1. 作業内容概要
 - 4.2. IM-VisualDesigner のドキュメントから IM-FormaDesigner のアプリケーションのひな形を生成する手順
 - 4.3. IM-FormaDesigner アプリケーションの開発および検証する手順
- 5. ステップ 3
 - 5.1. 制限事項
 - 5.2. 作業内容概要
 - 5.3. IM-VisualDesigner → IM-FormaDesigner 切替手順
 - 5.4. 検証手順
- 6. 付録
 - 6.1. 移行支援ツールの内部仕様
 - 6.2. フォーム変換ルール
 - 6.3. IM-VisualDesigner とIM-FormaDesignerの機能比較

変更年月日	変更内容
2014-09-01	初版
2016-08-01	第2版 下記を変更・追加しました。 <ul style="list-style-type: none">■ 「はじめに」にトランザクションデータやファイルアップロードのデータの保存先に関する説明を追加しました。

本書では IM-VisualDesigner Ver.7.2（以下、IM-VisualDesigner）から IM-FormaDesigner for Accel Platform（以下、IM-FormaDesigner）への移行支援ツールを用いた移行手順および開発手順について説明します。

移行手順については下記の2つに大きく分かれます。

- IM-VisualDesigner を intra-mart Accel Platform へ移行します。
- intra-mart Accel Platform 上の IM-VisualDesigner から IM-FormaDesigner へ移行します。

intra-mart Accel Platform への移行について

本書の移行手順に従い移行支援ツールを用いることで、intra-mart WebPlatform 上で運用していた IM-VisualDesigner Ver.7.2の既存ドキュメントを intra-mart Accel Platform へ移行し、利用することが可能です。

intra-mart Accel Platform で帳票（ドキュメント）の追加を行うことはできません。新規の帳票については IM-FormaDesigner を用いて開発してください。

既存のドキュメントの運用に必要なメンテナンスを intra-mart Accel Platform 上で行うことは可能です。

移行の対象は以下の通りです。

- IM-VisualDesigner 標準ドキュメント
IM-VisualDesigner のドキュメントでワークフロー連携していないドキュメントです。
トランザクションデータを含めて intra-mart Accel Platform へ移行することができます。
- IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow)
IM-Workflow と連携しているIM-VisualDesigner のドキュメントです。
トランザクションデータを含めて intra-mart Accel Platform へ移行することができます。
- IM-VisualDesigner ワークフロー連携ドキュメント(BPW)
ドキュメントワークフロー（BPW）と連携しているIM-VisualDesigner のドキュメントです。
トランザクションデータを含めて intra-mart Accel Platform へ移行することができます。
- IM-VisualDesigner BPM連携ドキュメント
im-BPMと連携しているIM-VisualDesigner のドキュメントです。
intra-mart Accel Platform 上でim-BPMが非サポートのため、intra-mart Accel Platform 上で利用することはできません。

IM-FormaDesigner への移行について

移行の対象は以下の通りです。

- IM-VisualDesigner 標準ドキュメント
IM-VisualDesigner 標準ドキュメント はIM-FormaDesigner にトランザクションデータを含めて移行することができます。
- IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow)
IM-FormaDesigner にトランザクションデータを含めて移行することができます。
また、移行後は、IM-VisualDesigner で申請・処理された案件はIM-FormaDesigner の画面で閲覧できます。
移行支援ツールを利用する場合でも、過去に申請・処理された案件については IM-VisualDesigner で閲覧し、新規の申請のみ IM-FormaDesigner の画面から行うという移行パターンも可能です。
- IM-VisualDesigner ワークフロー連携ドキュメント(BPW)
IM-FormaDesigner はドキュメントワークフロー（BPW）との連携に対応していないため、IM-VisualDesigner で申請・処理された案件（トランザクションデータ）は IM-FormaDesigner に移行することができません。

移行前 => 移行後

移行前	=> 移行後
IM-VisualDesigner 標準ドキュメント	=> IM-FormaDesigner - 標準

移行前	=>	移行後
IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow)	=>	IM-FormaDesigner - IM-Workflow
IM-VisualDesigner ワークフロー連携ドキュメント(BPW)	=>	IM-FormaDesigner - IM-Workflow (注)

(注) 新規に IM-Workflow のフロー定義として定義しなおす必要があります。ドキュメントワークフロー(BPW)のプロセス定義からの IM-Workflow のフロー定義への変換機能は提供していません。

i コラム

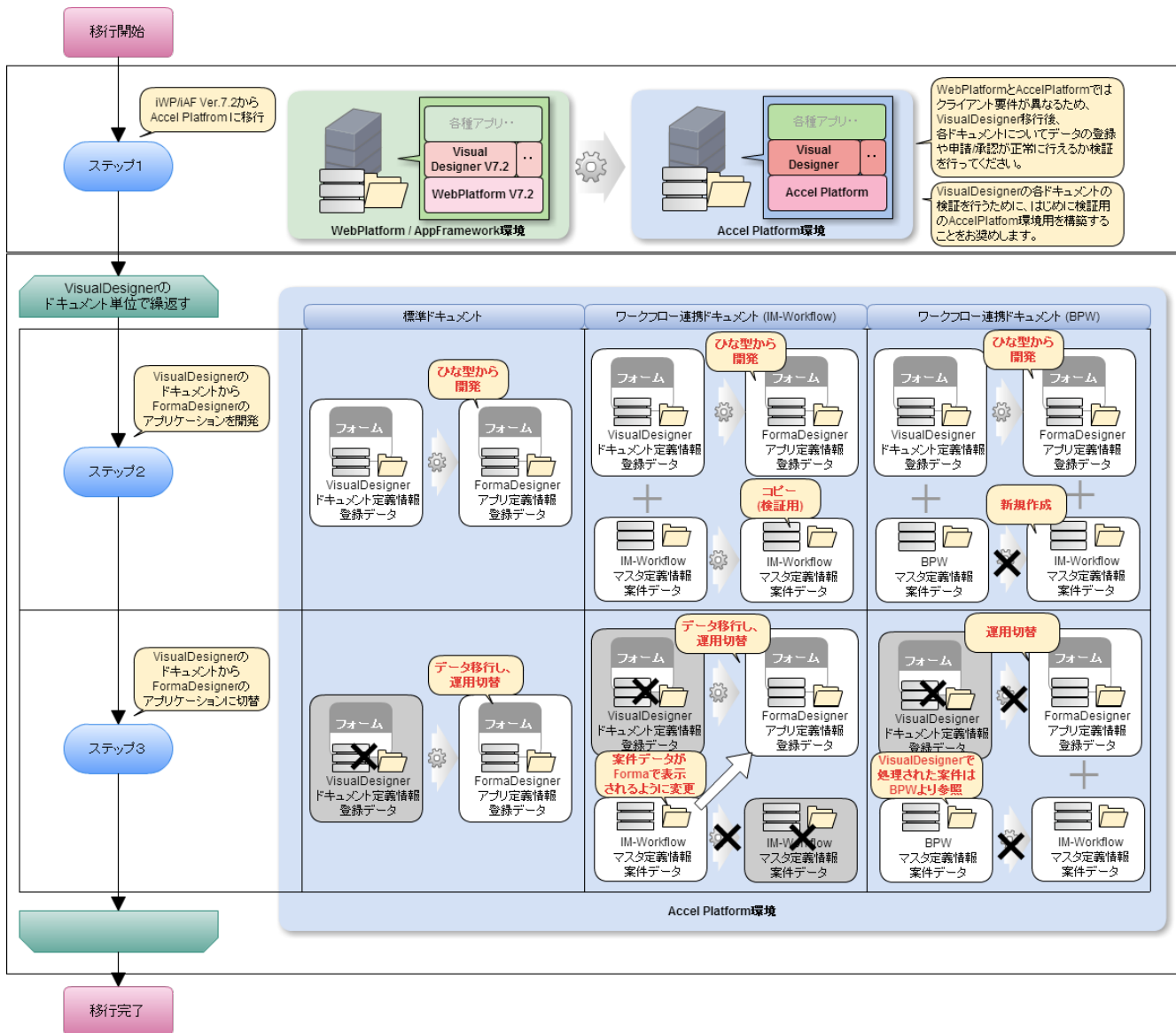
intra-mart Accel Platform 2016 Summer(8.0.14)、IM-FormaDesigner 2016 Summer(8.0.13)以降のバージョンに移行する場合には、本書の移行作業がすべて完了したらワークフローのトランザクションデータやファイルアップロードのデータの保存先を切り替えることができます。

手順の詳細については、以下のドキュメントを参照してください。

- 「IM-Workflow 管理者操作ガイド」- 「トランザクションデータ、添付ファイルの保存先を変更する」
- 「IM-FormaDesigner 作成者操作ガイド」- 「IM-FormaDesigner の高度な設定を行う」- 「ファイルアップロードのファイルの保存先 (transaction-file-location / fileupload)」

移行手順概要

移行手順は以下の3つのステップで構成されています。



- intra-mart WebPlatform / AppFramework 環境を intra-mart Accel Platform 環境に移行します。intra-mart WebPlatform / AppFramework 上で動作している IM-VisualDesigner の各ドキュメントが、intra-mart Accel Platform 上で動作することを検証の上、intra-mart Accel Platform 環境に移行を行います。
- 既存の帳票を IM-VisualDesigner から IM-FormaDesigner へ移行しない場合は、ステップ1で移行作業が完了します。

ステップ2

- 各IM-VisualDesigner のドキュメントを、IM-FormaDesigner のアプリケーションに変換し、アプリケーションを開発します。ステップ2の移行対象のドキュメントに関連した業務は、IM-FormaDesigner のアプリケーションの開発し、ステップ3を実行するまでは、移行対象のドキュメントで業務を継続します。



注意

他の帳票から連携する（リンク機能）目的にのみ利用されているドキュメントについては、ドキュメントに紐づくトランザクションデータが存在しないため、ステップ3は実行できません。

ステップ3

- ステップ2で開発した IM-FormaDesigner のアプリケーションに運用を切替えます。
※切替時に IM-VisualDesigner のドキュメントは利用されないように設定します。



コラム

ステップ1での intra-mart Accel Platformへの移行に関しては、IM-VisualDesigner 以外に導入しているアプリケーションも移行できるかどうか調査が必要です。そのため、はじめに「[移行ガイド](#)」を必ず確認してください。



コラム

ステップ2、3はシステムを停止せずアプリケーション単位で順次移行します。

intra-mart WebPlatform / AppFramework 環境を intra-mart Accel Platform 環境に移行します。

前提条件

- ステップ1では、intra-mart WebPlatform / AppFramework 上で動作している IM-VisualDesigner が intra-mart Accel Platform 上で動作するまでの手順を記載しています。
intra-mart Accel Platform 上で IM-VisualDesigner が正常に動作することを確認した上で次のステップ2に進んでください。
- バージョン
 - 移行元の環境が IM-VisualDesigner Ver.7.2パッチ5 の状態とします。



注意

IM-VisualDesigner 製品へのカスタマイズを行っている場合、移行に失敗する可能性があります。
上記検討にあたっては、コンサルティングサービスのご利用をお奨めいたします。

- データベース
 - 利用する intra-mart Accel Platform の前提条件に準じます。「[前提条件 - データベース](#)」を参照してください。
- オペレーティングシステム
 - 利用する intra-mart Accel Platform の前提条件に準じます。「[前提条件 - オペレーティングシステム](#)」を参照してください。
- タイムゾーン
 - 利用する intra-mart Accel Platform の前提条件に準じます。「[前提条件 - タイムゾーン](#)」を参照してください。

制限事項

- 移行にあたっての制限事項は「[リリースノート](#)」-「[制限事項](#)」の「移行」に記載されている通りです。
バージョンアップ時には、かならず全ての制限事項をご確認ください。
- IM-VisualDesigner BPM連携ドキュメント についてはim-BPMが intra-mart Accel Platform 上で非サポートのため、intra-mart Accel Platform へ移行することはできません。

事前作業

intra-mart WebPlatform / AppFramework環境、バージョンに対する事前作業については、「[移行ガイド](#)」-「[事前作業](#)」を参照してください。

IM-VisualDesigner を利用する際に独自のプログラムを用いていた場合（IM-Workflow のユーザプログラム、独自画面の呼び出しなど）は、ユーザモジュールとして開発を行い、intra-mart Accel Platform 対応、または移行（互換）向け対応が必要です。

詳細については、「[移行ガイド](#)」-「[ユーザモジュールへの対応](#)」を参照してください。

IM-VisualDesigner 移行手順

1. intra-mart WebPlatformから情報のエクスポート

intra-mart WebPlatformから情報のエクスポートについては、「[移行ガイド](#)」-「[1. intra-mart WebPlatformから情報のエクスポート](#)」を参照してください。

2. データベースのコピー

データベースのコピーについては、「[移行ガイド](#)」-「[2. データベースのコピー](#)」を参照してください。

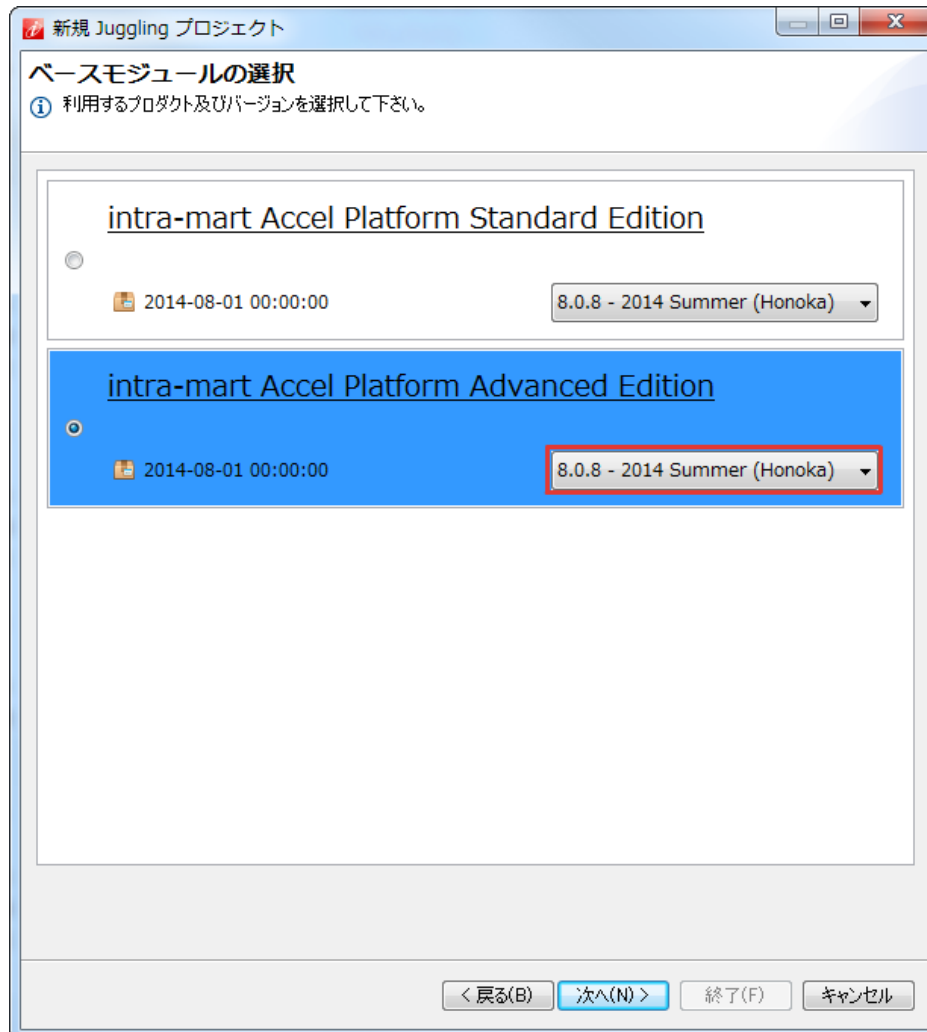
3. IM-Jugglingによるwarの作成

コラム

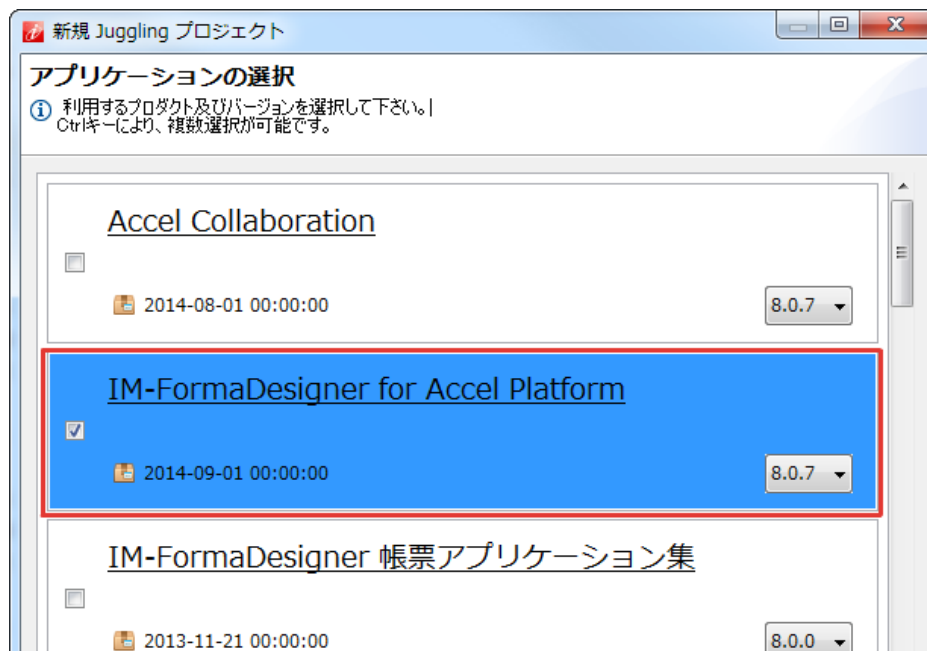
移行を目的としたIM-Jugglingによるwarの作成にあたっては、「移行ガイド」-「3. IM-Jugglingによるwarの作成」を参照してください。また、IM-FormaDesigner のセットアップも必要なため、「IM-FormaDesigner セットアップガイド」と合わせて参照してください。

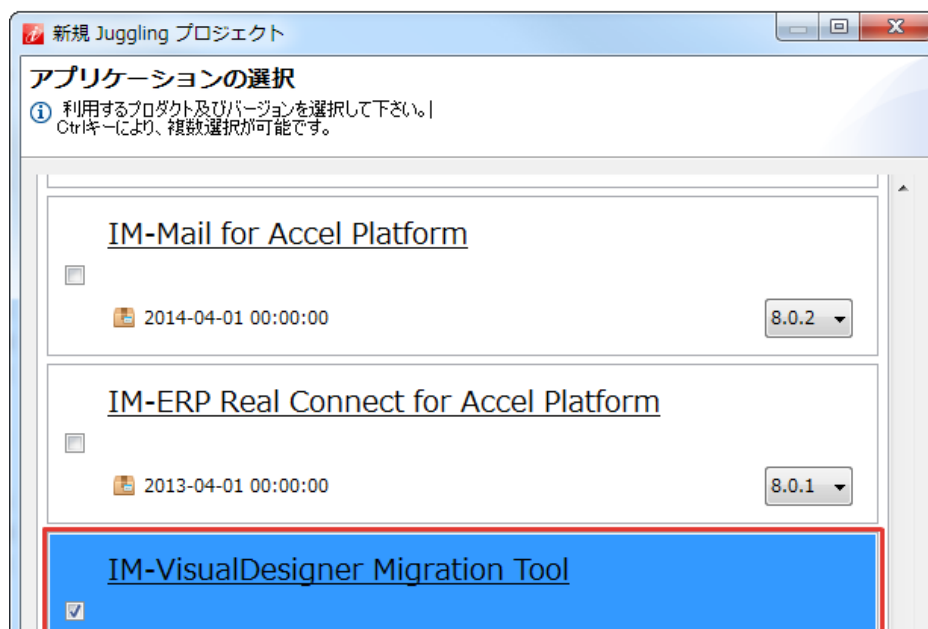
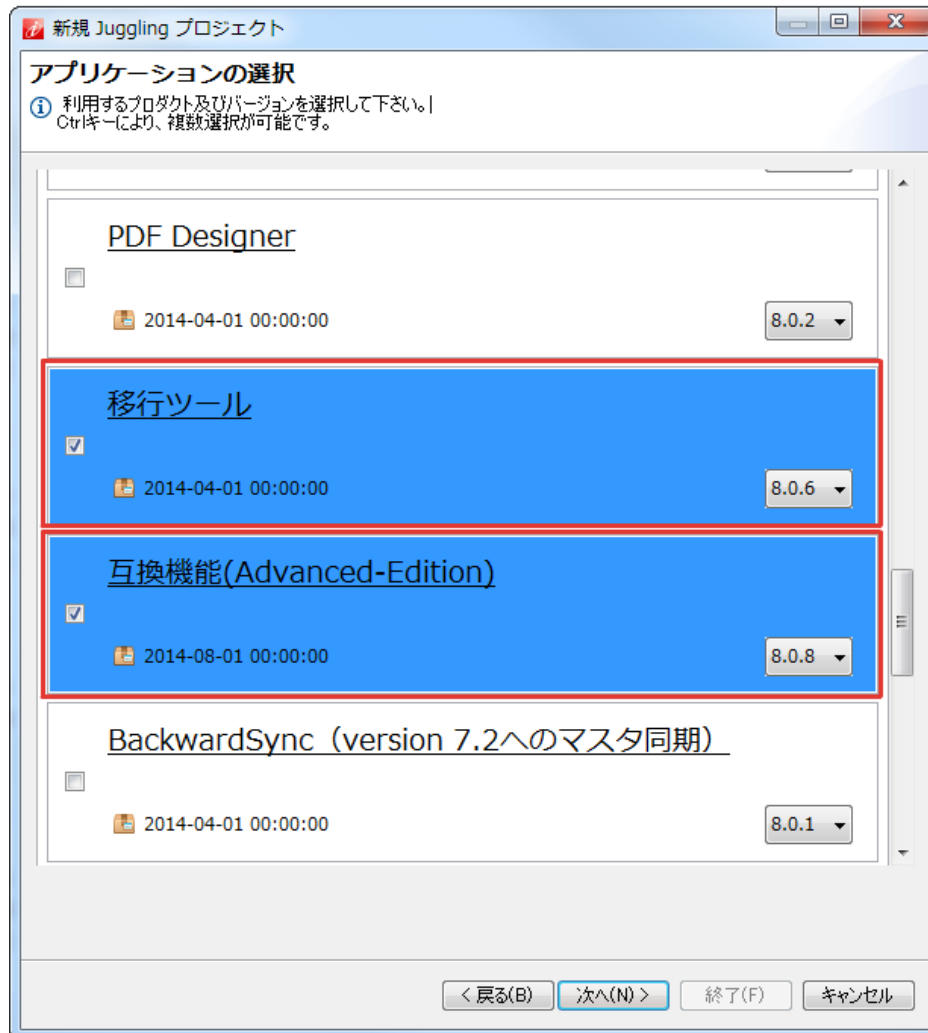
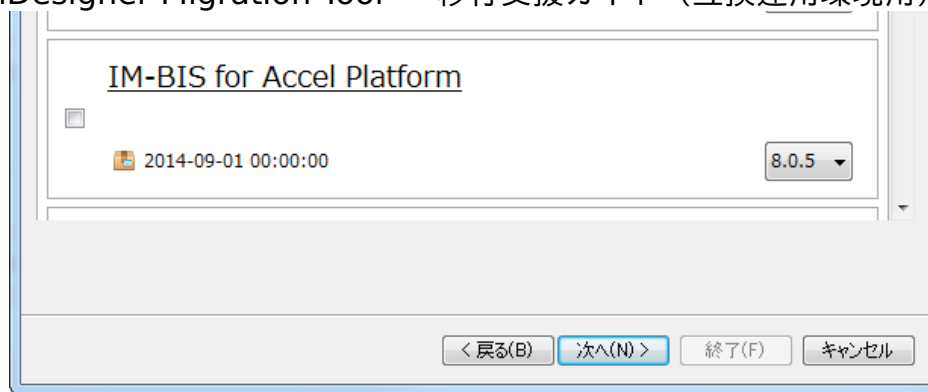
1. IM-Jugglingを起動し、Juggling プロジェクトを作成します。

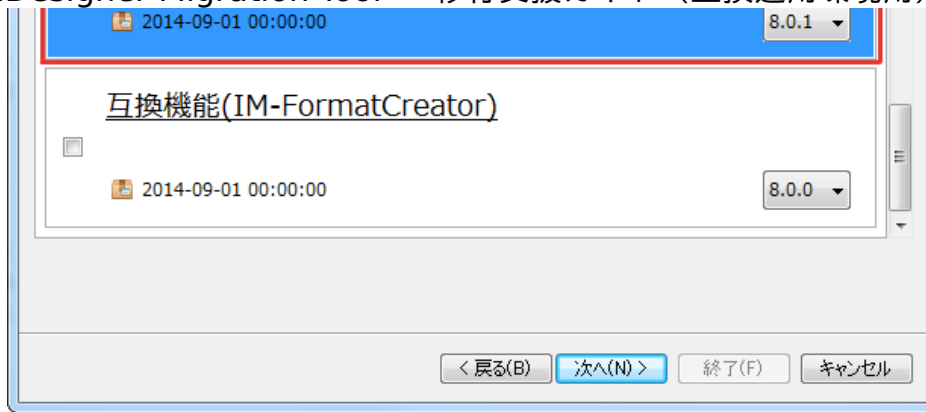
1. ベースモジュールの選択より、intra-mart Accel Platform Advanced Edition（8.0.8）を選択します。



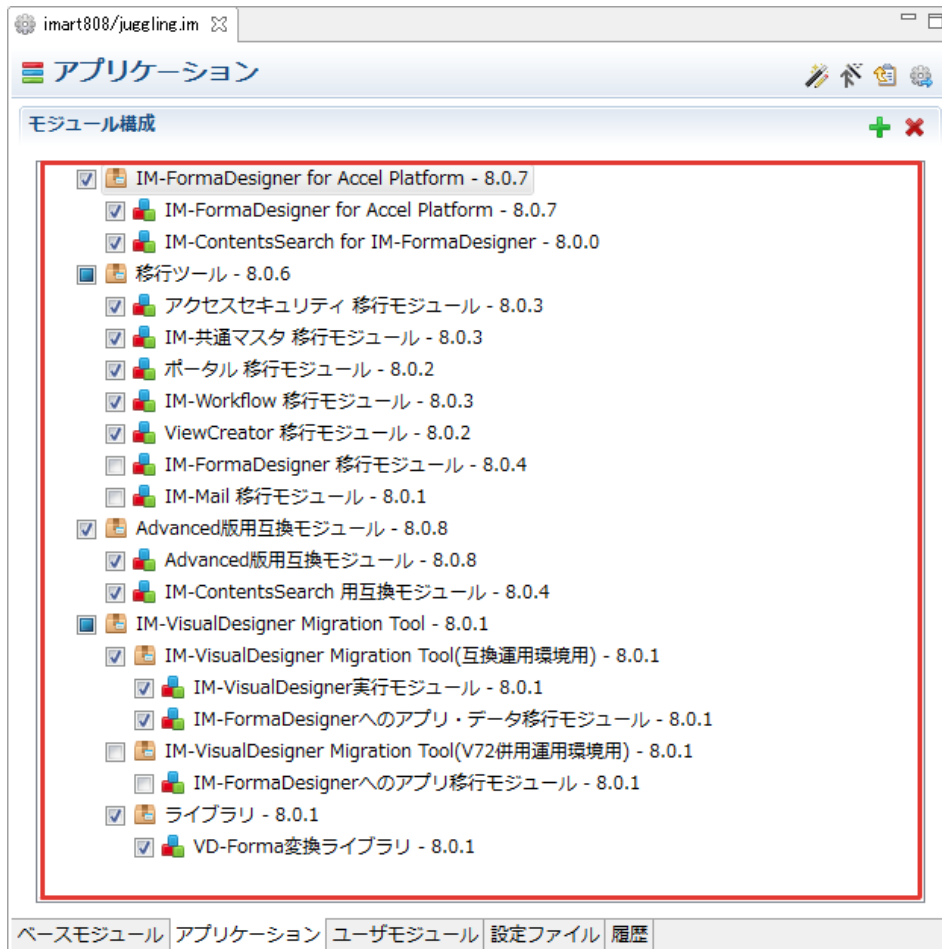
2. アプリケーションの選択より、IM-FormaDesigner for Accel Platform、移行ツール、互換機能(Advanced-Edition)、IM-VisualDesigner Migration Tool を選択します。







3. アプリケーションより、IM-FormaDesigner for Accel Platform、移行ツール、Advanced版用互換モジュール、IM-VisualDesigner Migration Tool（IM-VisualDesigner Migration Tool（互換運用環境用）、ライブラリ）を選択します。



2. 移行対象機能の移行モジュール、テナントデータベースの接続設定、Storageディレクトリの設定、ログイングループIDの設定などについては、「[3. IM-Jugglingによるwarの作成](#)」を参照してください。

コラム

intra-mart Accel Platform の言語（ロケール）設定はintra-mart WebPlatform 7.2の言語（ロケール）設定と同じにする必要があります。

intra-mart WebPlatform 7.2を標準の言語設定（日本語ロケールのみ）で利用していた場合は、intra-mart Accel Platform の言語設定を変更し、英語/中国語を外してください（intra-mart Accel Platform の標準の言語設定が日本語/英語/中国語の3言語となっているためです）。

言語設定については、「[設定ファイルリファレンス](#)」-「[ロケールマスタ](#)」を参照してください。

1. IM-VisualDesigner の設定ファイルstraform.iniを移行します。
- 以下を intra-mart WebPlatform から Juggling プロジェクトに移動してください。

```
{Resource Serviceサーバルート}/pages/product/src/straform/straform.ini
↓↓↓↓↓↓
{Juggling プロジェクト}/jssp/product/src/straform/straform.ini
```

2. 互換モジュール の設定ファイルconf/parameter.xmlを編集します。

- 「ProjectNavigator」内の < (プロジェクト名) /conf/parameter.xml> ファイルをダブルクリックで開き、「ソース」タブを選択してください。
- intra-mart WebPlatform 7.2 をインストールした際の文字コードを記述します。

```
<param>
  <param-name>compatible.file.encoding</param-name>
  <param-value>Windows-31J</param-value>
</param>
```

3. ビルドウィザードより、war を作成します。



注意

「サンプル」は選択しないでください。

4. war を Web Application Server にデプロイします。

4. intra-mart Accel PlatformへのStorageの移行

1. 「[移行ガイド](#)」 - 「[4. intra-mart Accel PlatformへのStorageの移行](#)」に基づいて、Storageの移行を行ってください。
2. intra-mart Accel Platform（Web Application Server サーバ）を起動します。
warが展開され、Storageディレクトリが作成されるまでお待ちください。
3. 「[移行ガイド](#)」 - 「[4. intra-mart Accel PlatformへのStorageの移行](#)」に基づいて、Storage Serviceで利用していたファイルを移行します。
 - StraForm

```
[%STORAGE_72%]/straform/
↓↓↓↓↓↓
[%PUBLIC_STORAGE_80%]/straform/
```



コラム

- intra-mart WebPlatform / AppFramework のStorageのルートを [%STORAGE_72%] と表記しています。
- intra-mart Accel Platform のStorageのルートを [%PUBLIC_STORAGE_80%] と表記しています。 | (例) %STORAGE_ROOT%/public/storage/%LOGIN_GROUP_ID%/straform

5. システム管理者 によるテナント初期設定

1. 「[移行ガイド](#)」 - 「[5. システム管理者 によるテナント初期設定](#)」に基づいて、テナント環境セットアップ、各種ライセンスの登録などを行ってください。
2. IM-VisualDesigner のクエリー設定においてシステムデータソースを接続先として利用していた場合は、同一の接続IDでシェアードデータソースの設定を行ってください。
 - シェアードデータソースの設定方法については、「[移行ガイド](#)」 - 「[データソースマッピング設定](#)」を参照してください。

6. テナント初期設定後の各種メンテナンス

1. 移行先の環境に テナント管理者 でログインします。
 - ロール設定
 - IM-VisualDesigner で作成した画面の利用者に「VDユーザ」ロールを付与します。
 - IM-VisualDesigner で画面を作成するユーザ、マスタメンテナンスを行うユーザに「VD管理者」ロールを付与します。
 - メニュー設定
 - Ver7.2環境で登録した各ドキュメントのメニューURLから".jssp"を除去します。

メニューアイテムの編集

メニューアイテムID *

メニューアイテム名 *

日本語 *	<input type="text" value="サンプル一覧"/>
英語	<input type="text" value="サンプル一覧"/>
中国語	<input type="text" value="サンプル一覧"/>

URL *

呼び出し方法

引数

+ 行追加 - 選択行削除

キー	値
<input type="checkbox"/>	
<input type="checkbox"/>	sf_document_id doc_sample_taishoku

アイコン画像

ファイルパス

CSS Sprites

IFRAME表示

ポップアップ表示

説明

- 「IFRAME表示」のチェックを外します。

メニューアイテムの編集

メニューアイテムID *

メニューアイテム名 *

日本語 *	<input type="text" value="サンプル一覧"/>
英語	<input type="text" value="サンプル一覧"/>
中国語	<input type="text" value="サンプル一覧"/>

URL *

呼び出し方法

引数

+ 行追加 - 選択行削除

キー	値
<input type="checkbox"/>	
<input type="checkbox"/>	sf_document_id doc_sample_taishoku

アイコン画像

ファイルパス

CSS Sprites

IFRAME表示

ポップアップ表示

説明

- IM-VisualDesigner の画面を表示するメニューアイテムごとにアクセス権限の設定を行うことはできません。同一のアク

セス権限を設定したいメニューアイテムごとにそれらを包含するメニューグループを作成し、それに対する認可設定を行うことでアクセス権限の設定を行ってください。

- 認可設定
 - メニューグループに対して適切な権限設定を行います。

検証手順

1. IM-VisualDesigner 動作検証

- クライアント要件が intra-mart WebPlatform / AppFramework と intra-mart Accel Platform では異なるため、カスタムスクリプトが正常に動作することを確認します。

2. IM-VisualDesigner 修正

- 検証の手順内でもスクリプトエラー等が発生した場合、検証環境でカスタムスクリプトの修正およびエクスポートを行い、移行先の環境でインポート（更新）を行います。
 - 詳細手順
 1. 検証環境の Straform Designer を使って、対象のカスタムスクリプトを修正します。
 2. カスタムスクリプトを修正したフォームを開いた状態で、メニューから[ファイル]-[エクスポート]-[ZIP ファイル]を選択します。
 3. [保存]ボタンを押下し、zipファイルを出力します。
 4. 移行先にVD管理者でログインし、メニューから [IM-VisualDesigner] - [フォーム一覧] - [編集] から上記zipファイルをアップロードし、[更新] ボタンをクリックします。



コラム

IM-VisualDesigner Client のデプロイ機能は、intra-mart Accel Platform 環境に対しては行うことができません。

運用手順

IM-VisualDesigner の各設定において intra-mart Accel Platform から変更となった点、また制限された点について説明します。基本的な設定手順については、「[IM-VisualDesigner Ver.7.2 サーバモジュール 操作ガイド](#)」を参照してください。

1. 制限されている機能について

- アイテムへのアクション設定

リンク機能にてコンテンツ種別「スクリプト開発モデル」「JavaEE開発モデル」のページを設定する場合、対象となるページのURLにノーマルが設定されている必要があります。
- ドキュメントの設定

新規のドキュメント追加を行うことはできません。
- インポート・エクスポート

ドキュメントのインポートを行うことはできません。

2. 手順が変更となる機能について

- メニュー設定

IM-VisualDesigner の「メニュー設定」機能は廃止されました。テナント管理者機能の「メニュー設定」をご利用ください。

 - テナント管理者としてログインし、「メニュー設定」画面へアクセスします。
 - 新規にメニューアイテムを追加し、URLに"straform/normal/view/list_view"、キー"sf_document_id"に対してメニューを行うドキュメントIDを設定します。
 - メニュー設定機能の操作方法の詳細については「[テナント管理者操作ガイド](#)」-「[メニューを設定する](#)」を参照してください。

 コラム

メニューアイテムごとにアクセス権限の設定を行うことはできません。同一のアクセス権限を設定したいメニューアイテムごとにそれらを包含するメニューグループを作成し、それに対する認可設定を行うことでアクセス権限の設定を行ってください。

- フォーム設定

フォームを更新する際に、IM-VisualDesigner Client上のフォームデプロイ機能は利用できません。フォームのインポート機能によって更新してください。

- フォームのインポート機能の操作方法の詳細については「[IM-VisualDesigner Ver.7.2 サーバモジュール 操作ガイド](#)」を参照してください。

各IM-VisualDesigner のドキュメントを、IM-FormaDesigner のアプリケーションに変換し、アプリケーションを開発します。

ステップ2、3はシステムを停止せずアプリケーション単位で順次移行します。

作業内容概要

ステップ2では、運用中のIM-VisualDesigner のドキュメントをIM-FormaDesigner のアプリケーションに切り替えるための準備を行います。

IM-VisualDesigner のドキュメントからIM-FormaDesigner のアプリケーションのひな型を生成する移行支援ツールを使って準備を進めます。

具体的な手順については下記の通りです。

手順

- 1. IM-VisualDesigner のドキュメントからIM-FormaDesigner のアプリケーションのひな形を生成
- 2. IM-FormaDesigner アプリケーションの開発および検証
 - IM-VisualDesigner 標準ドキュメント の開発と検証
 - IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) の開発と検証
 - IM-VisualDesigner ワークフロー連携ドキュメント(BPW) の開発と検証

1. IM-VisualDesigner のドキュメントからIM-FormaDesigner のアプリケーションのひな形を生成

移行支援ツールを利用すると、選択したIM-VisualDesigner のドキュメントからIM-FormaDesigner のアプリケーションのひな形が作成されます。

- IM-VisualDesigner のフォーム上のラベルアイテムと入力系アイテムを抽出し、IM-FormaDesigner のフォームを生成します。
- IM-VisualDesigner のクエリ設定を利用していた場合、IM-VisualDesigner のクエリ設定からSQL等を抽出し、IM-FormaDesigner のデータソース定義を生成します。
- IM-FormaDesigner アプリケーションの検証用に、IM-VisualDesigner のそのドキュメントのトランザクションデータ（IM-VisualDesigner の画面を通して入力された業務データ）をIM-FormaDesigner アプリケーションのトランザクションデータとして移行します。
※検証用のため、移行するデータは100件に制限されます。
- IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) の場合、IM-FormaDesigner アプリケーションの検証用に、IM-VisualDesigner で使用しているIM-Workflow の定義情報（フロー定義/ルート定義/コンテンツ定義）をIM-FormaDesigner 用にコピーを作成します。
※IM-VisualDesigner ワークフロー連携ドキュメント(BPW) の場合は、IM-Workflow の定義情報（フロー定義/ルート定義/コンテンツ定義）のコピーは作成されません。新規に作成してください。

2. IM-FormaDesigner アプリケーションの開発および検証

生成されたひな形から、IM-FormaDesigner アプリケーションの開発し検証を行います。

アプリケーションの開発と検証作業は、IM-VisualDesigner の下記ドキュメント種別により異なります。以下その概要を説明します。

- IM-VisualDesigner 標準ドキュメント
- IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow)
- IM-VisualDesigner ワークフロー連携ドキュメント(BPW)

IM-VisualDesigner 標準ドキュメント の開発と検証

開発作業は主にフォームの開発です。

レイアウトの設定、画面アイテム「スクリプト」を用いたアクション設定を行います。

各フォームの開発後に下記の設定を行います。

- 画面遷移の設定
- 権限設定
- 一覧表示設定
- メニュー設定

i コラム

権限設定は、IM-VisualDesigner と IM-FormaDesigner で設定できる内容が異なります。
 詳細については、「[アプリケーション利用の権限を設定する](#)」を参照してください。

次に、検証を行います。検証の観点としては以下の通りです。

- 新規にデータを登録し、参照/更新できること
 ※特に各画面アイテムに設定した項目の動作確認（入力チェック/スクリプト等）
- IM-VisualDesigner から移行したデータを、参照/更新できること

IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) の開発と検証

開発作業は主にフォームの開発とWF連携設定です。

はじめに、レイアウトの設定、画面アイテム「スクリプト」を用いたアクション設定を行います。

各フォームの開発後に、下記 IM-FormaDesigner のWF連携設定を適宜行います。

- 承認画面設定
- 案件プロパティ設定
- 追記設定

次に、検証を行います。検証の観点としては以下の通りです。

- フロー定義に従って、申請/承認/参照を行い設定した内容が正しく動作すること
 ※各画面アイテムに設定した項目の動作確認（入力チェック/スクリプト等）や 追記設定などのWF連携設定等
- IM-VisualDesigner から移行したデータを、参照できること

i コラム

IM-VisualDesigner から検証用に移行したデータは、IM-Workflow の完了済み案件等から閲覧することはできません。
 IM-FormaDesigner の一覧表示機能を利用して、参照画面を表示することで、移行したデータが参照できるかどうか確認してください。

IM-VisualDesigner ワークフロー連携ドキュメント(BPW) の開発と検証

開発作業は主にフォームの開発、IM-Workflow の定義情報作成（フロー定義/ルート定義）とWF連携設定です。

はじめに、レイアウトの設定、画面アイテム「スクリプト」を用いたアクション設定を行います。

次に IM-Workflow の定義情報作成（フロー定義/ルート定義）を、既に運用しているBPWのフローをベースに、新規に作成します。

最後に、IM-FormaDesigner からコンテンツ連携情報を作成し、下記WF連携設定を適宜行います。

- 承認画面設定
- 案件プロパティ設定
- 追記設定

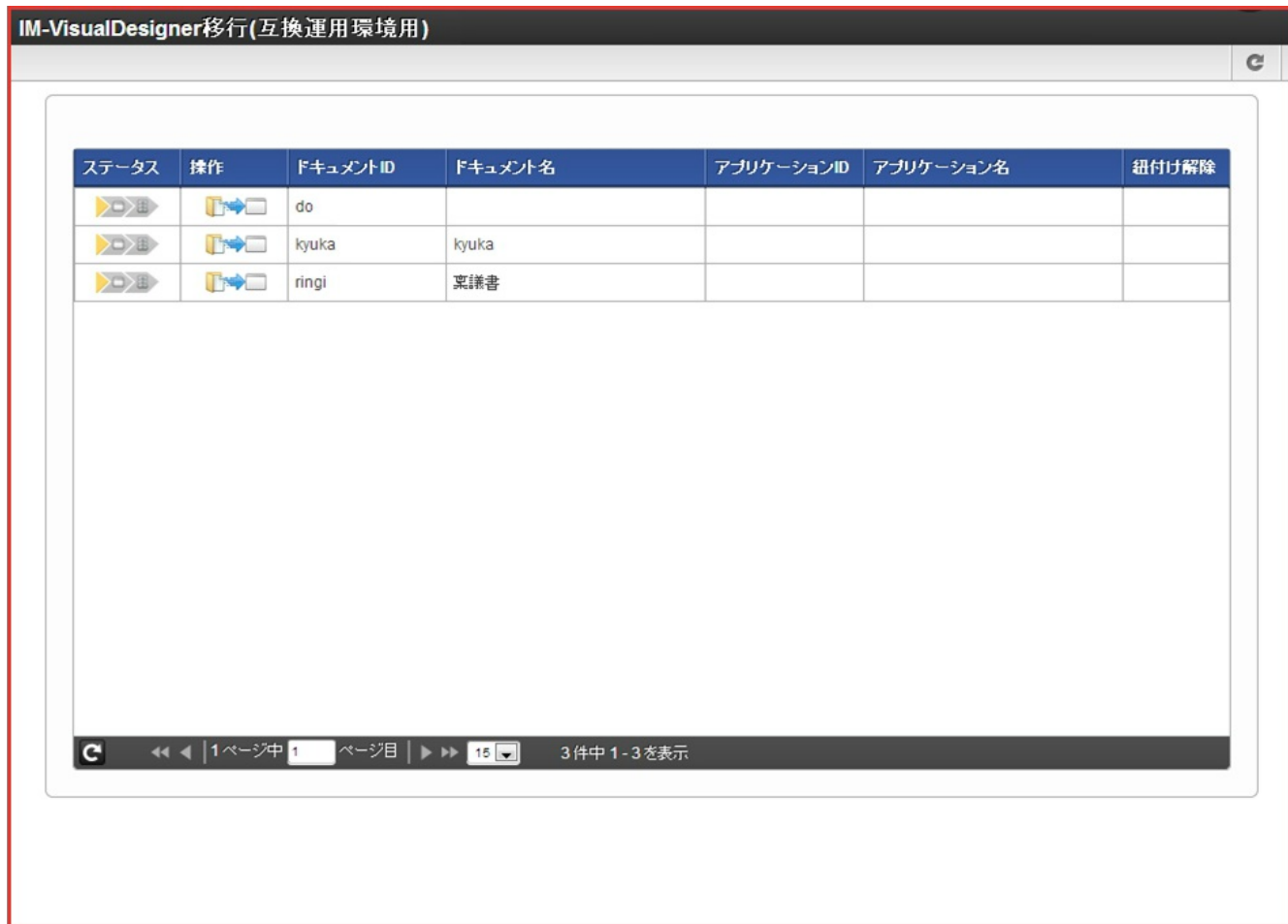
次に、検証を行います。検証の観点としては以下の通りです。

- フロー定義に従って、申請/承認/参照を行い設定した内容が正しく動作すること
 ※各画面アイテムに設定した項目の動作確認（入力チェック/スクリプト等）や 追記設定などのWF連携設定等

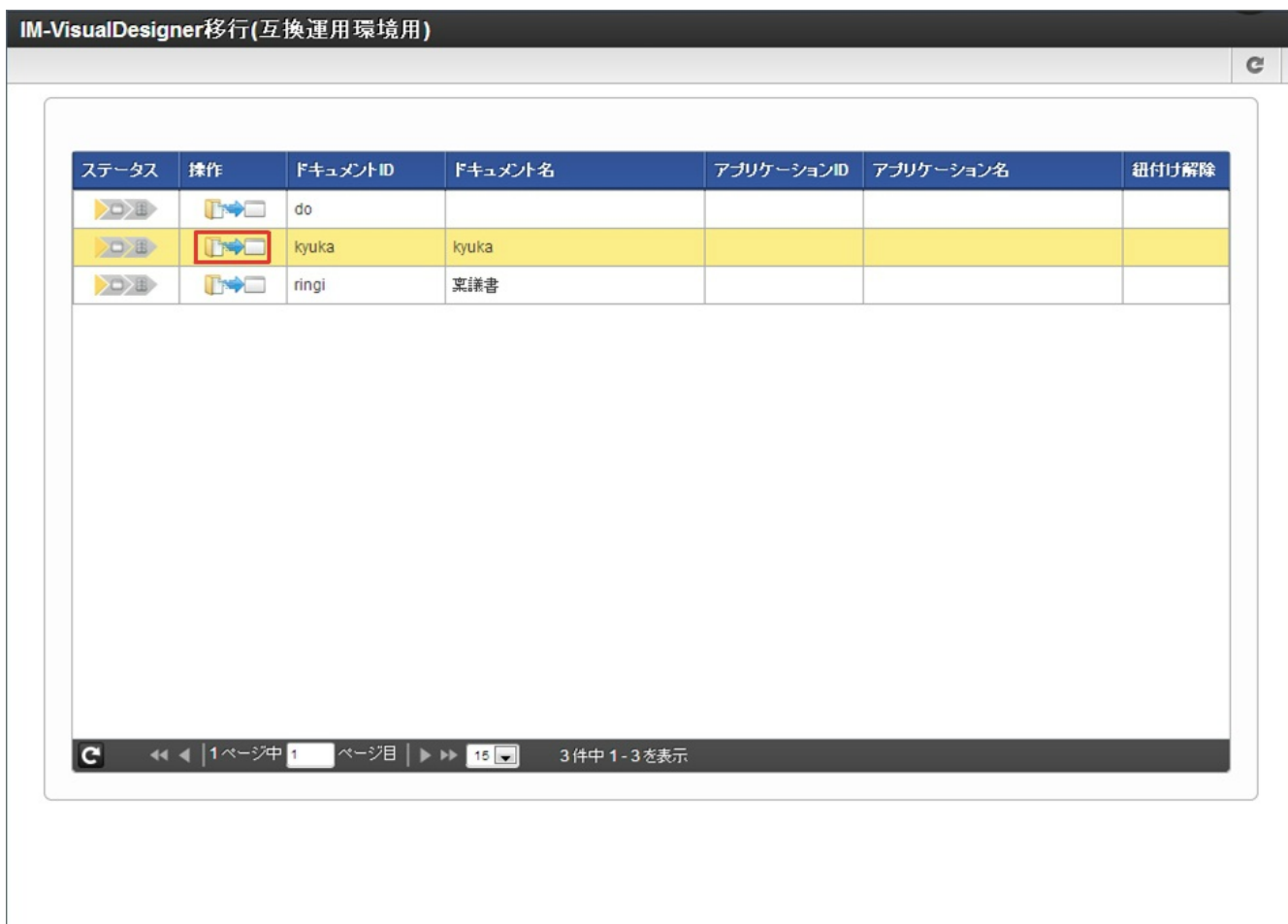
IM-VisualDesigner のドキュメントから IM-FormaDesigner のアプリケーションのひな形を生成する手順

1. IM-VisualDesigner ドキュメント→IM-FormaDesigner アプリケーション 変換

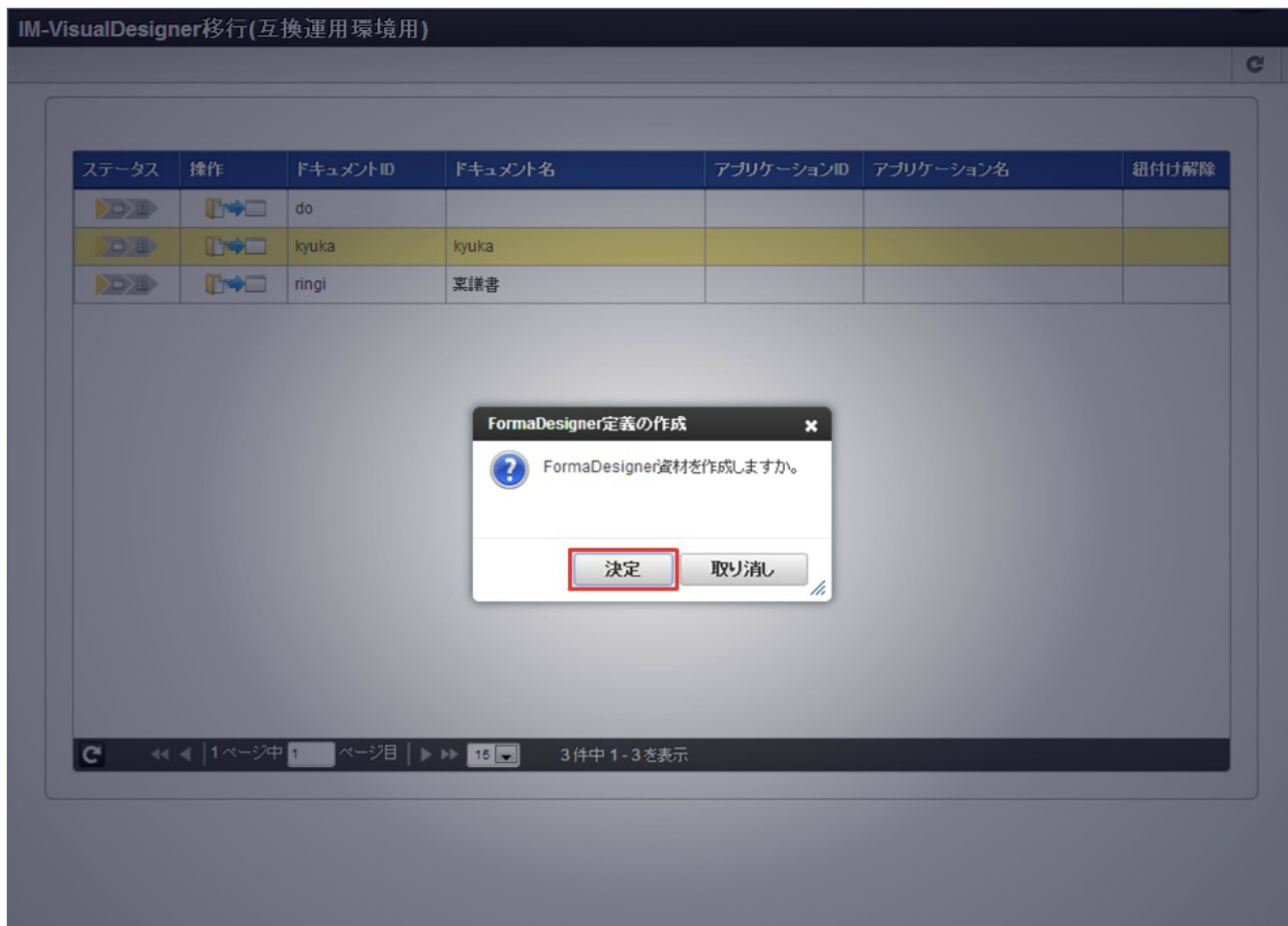
1. VD管理者でログインし、「IM-VisualDesigner（互換運用環境用）」画面を開きます。



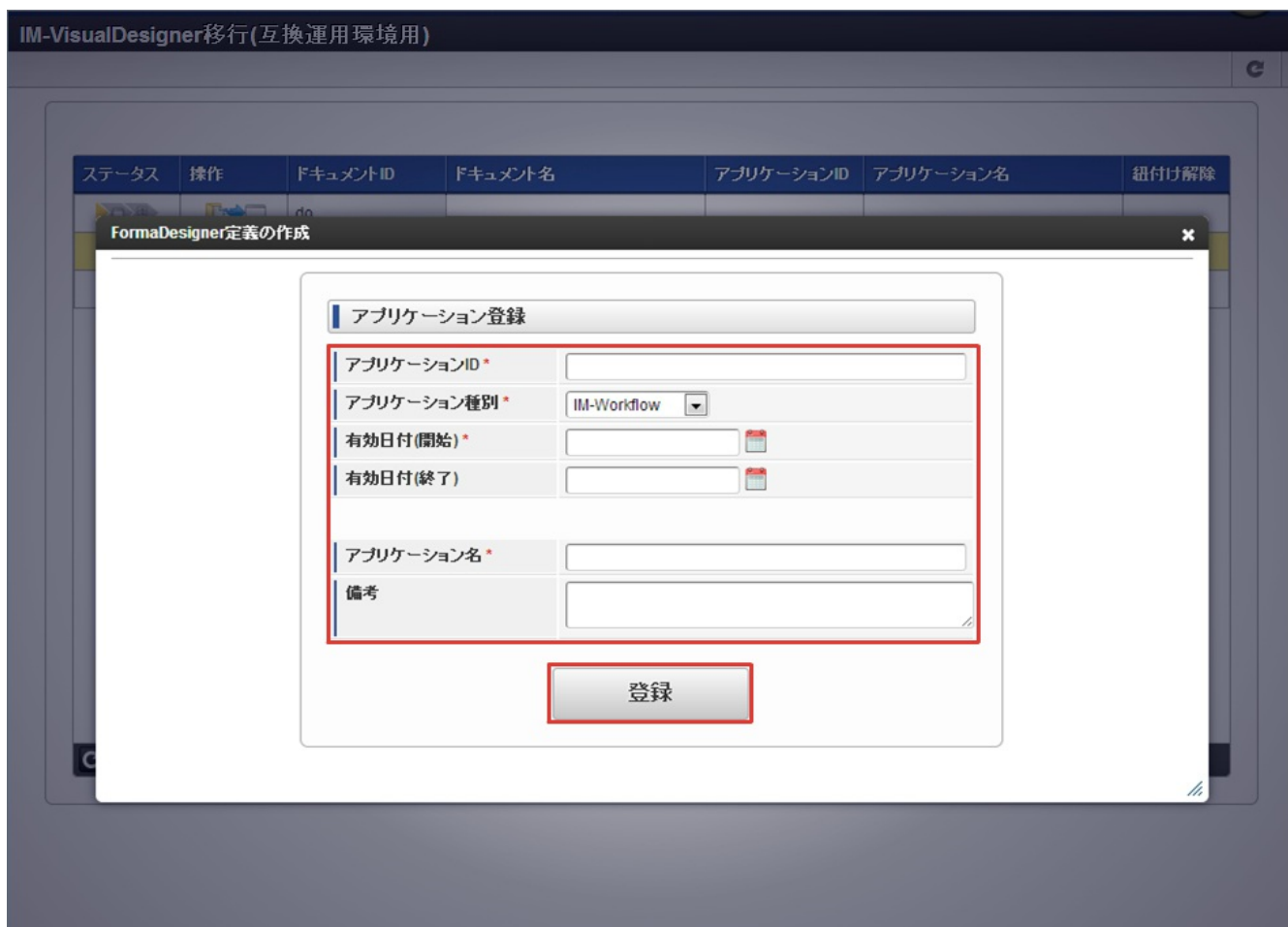
2. 変換対象のドキュメントの操作アイコンをクリックします。



3. 「FormaDesigner資材を作成しますか。」の問いに「決定」をクリックします。



4. アプリケーション情報を入力し、「登録」をクリックします。



i コラム

アプリケーション登録時に入力データを保管するためのテーブルを下記の命名規則のテーブル名で生成しています。

- ヘッダーテーブル
 - imfr_ut_[アプリケーションID]
- 明細行テーブル
 - imfr_ut_[アプリケーションID]_[IM-VisualDesigner のテーブルアイテムのID]

テーブル名が30文字を超えた場合、テーブル生成またはテーブル設定の処理に失敗するため、IM-FormaDesigner のアプリケーションのひな形生成処理が失敗します。

テーブル名が30文字を超えないような文字数のアプリケーションIDを入力してください。

また、ステップ3のトランザクションデータの移行処理が上記の命名規則のテーブル名を想定しているため、ステップ3を実行する場合はアプリケーション作成後にテーブル名を変更しないでください。

5. 正常に完了していることを確認します。

IM-VisualDesigner移行(互換運用環境用)

i VisualDesignerからFormaDesignerへの変換に成功しました。

アプリケーション情報		
アプリケーションID	アプリケーション名	処理結果
apid_sample	apnm_sample	アプリケーションの登録に成功しました。

テーブル設定情報		
オブジェクトID	テーブル名	処理結果
imfr_ut_apid_sample	imfr_form	テーブル生成と設定に成功しました。

フォーム情報		
フォームID	フォーム名	処理結果
517tew77mnbuwer	kyuka	フォームの変換に成功しました。

IM-Workflow定義のコピー情報(フロー定義)		
フローID	フロー名	処理結果
517tew77nkiwb		517t8wqhrf7hs2uフロー定義のコピーに成功しました。

IM-Workflow定義のコピー情報(コンテンツ定義)		
コンテンツID	コンテンツ名	処理結果
517tew77nfgwa		kyukaコンテンツ定義のコピーに成功しました。

IM-Workflow定義のコピー情報(ユーザプログラム)		
ユーザプログラムパス	ユーザプログラム種別	処理結果

IM-Workflow連携情報		
コンテンツID	コンテンツ名	処理結果
517tew77nfgwa		WF連携設定の登録に成功しました。

VisualDesignerドキュメント - FormaDesignerアプリケーションの紐付け情報	
処理結果	
VisualDesignerドキュメント - FormaDesignerアプリケーションの紐付け情報の更新に成功しました。	

トランザクションデータ移行	
処理結果	
トランザクションデータの移行に成功しました。	

! 注意

他の帳票から連携する（リンク機能）目的にのみ利用されているドキュメントについては、ドキュメントに紐づくトランザクションデータが存在しないため、「トランザクションデータの移行」においてエラーが発生します。

ただ、「トランザクションデータの移行」のエラーが、フォーム・クエリ情報の変換に影響することはありません。

! 注意

Oracleデータベースをご利用の場合、23文字以上のフィールドIDが存在するフォームを変換すると、22文字以上のフィールドIDは一部の情報がシステム側で自動採番された値に更新されます。カラム名も同様に交換されます。

Oracleデータベースではカラム名の長さに関して30バイトの制限があるため、製品間でのカラム接頭辞の差を考慮し変換処理を実施しております。

- **IM-VisualDesigner**
 - vd_ud_[フィールドID]
- **IM-FormaDesigner**
 - imfr_ud_[フィールドID]

i コラム

ドキュメント設定の使用フォーム一覧 に設定されているフォームのみ IM-FormaDesigner アプリケーション のフォームに変換されます。

i コラム

成功後、IM-FormaDesigner のアプリケーションを再作成したい等の理由で、再度「IM-VisualDesigner → IM-FormaDesigner 変換」を行う場合は、以下の手順を行ってください。

1. 「紐付け解除」アイコンをクリックします。

ステータス	操作	ドキュメントID	ドキュメント名	アプリケーションID	アプリケーション名	紐付け解除
		do				
		kyuka	kyuka	aosmpleid	aosmplenm	
		ringi	東証書			

2. 「Formaアプリ作成管理」のアプリ一覧画面より、以前の変換時に作成された該当アプリケーションを削除します。
3. メニューから[ワークフロー管理者]→[マスタ定義]→[コンテンツ定義]より、以前の変換時に作成された該当コンテンツを削除します。
4. メニューから[ワークフロー管理者]→[マスタ定義]→[ルート定義]より、以前の変換時に作成された該当ルートを削除します。
5. メニューから[ワークフロー管理者]→[マスタ定義]→[フロー定義]より、以前の変換時に作成された該当フローを削除します。
6. 再度、操作アイコンをクリックして、「IM-VisualDesigner → IM-FormaDesigner 変換」を行います。
※手順6で入力チェック以外のエラーが発生した場合は、上記2～5の手順で削除対象が漏れている可能性がありますので、再度、手順2からやり直してください。

6. 変換された IM-FormaDesigner のフォーム修正を行います。詳細は次の「[IM-FormaDesigner アプリケーションの開発および検証する手順](#)」へ。

i コラム

上記手順では、IM-VisualDesigner のラベルアイテムや入力系アイテムを IM-FormaDesigner の画面アイテムに変換しただけの状態のため、フォームの位置調整や装飾、スクリプトやアクション処理は、IM-FormaDesigner を使って再度設定する必要があります。

開発および検証手順

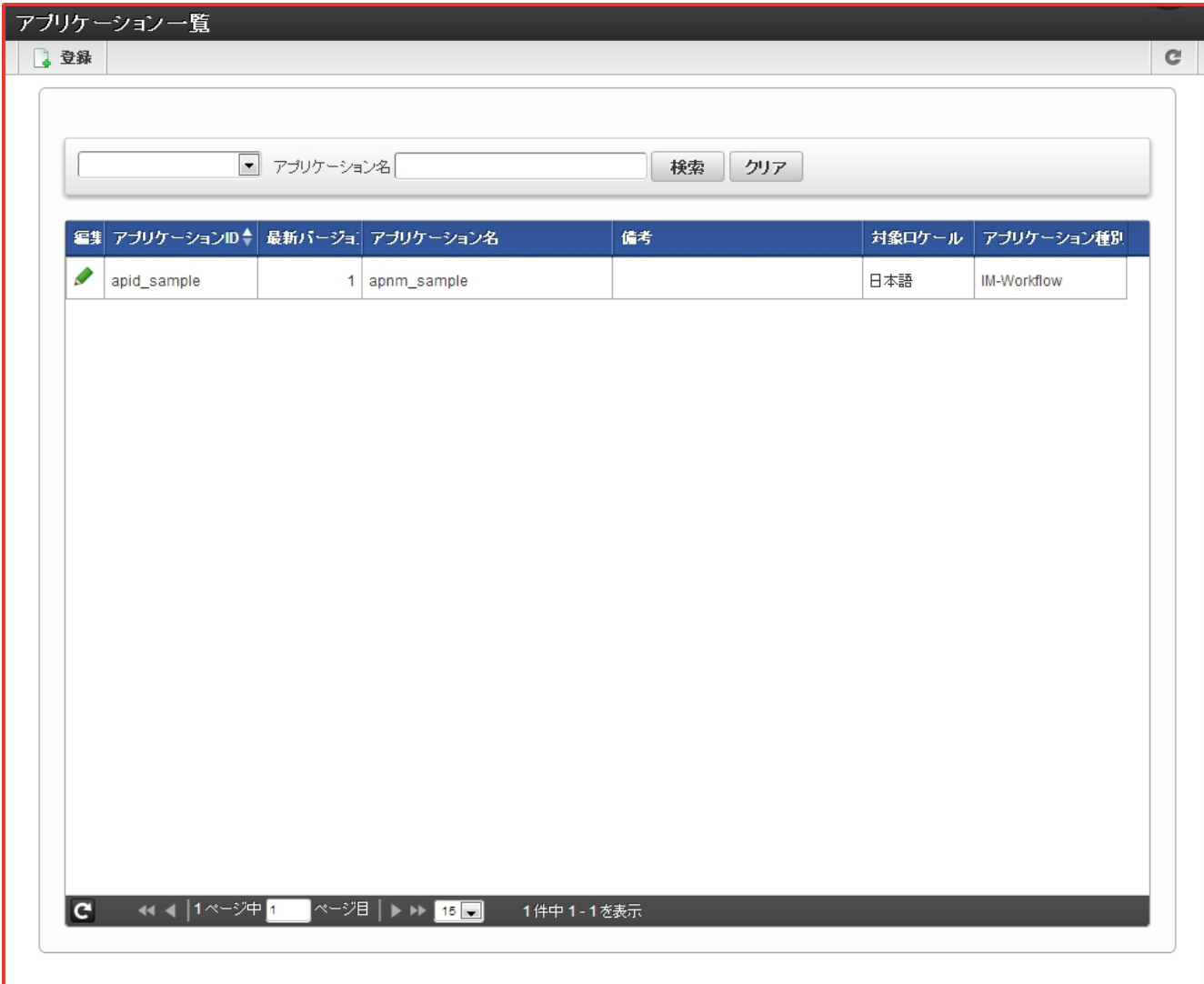
- 1. IM-FormaDesigner フォーム修正
- 2. 各種設定
- 3. IM-FormaDesigner 動作検証


コラム

IM-FormaDesigner の操作については、「IM-FormaDesigner 作成者操作ガイド」を参照してください。

1. IM-FormaDesigner フォーム修正

1. Forma管理者でログインし、「アプリケーション一覧」画面を開きます。



編集	アプリケーションID	最新バージョン	アプリケーション名	備考	対象ロケール	アプリケーション種別
	apid_sample	1	apnm_sample		日本語	IM-Workflow

2. 修正対象のアプリケーションの「編集」アイコンをクリックします。

フォーム設定


←

アプリケーションID: apid_sample
アプリケーション名: apnm_sample

アプリケーション情報 フォーム設定 テーブル設定 WF連携設定

登録

アプリケーション履歴一覧

編集	アプリケーション履歴番号	有効日付(開始)	有効日付(終了)	備考	更新日
	1	2013/09/24	2999/12/31		2013/09/24

1 ページ中 1 ページ目 1 件中 1-1 を表示

3. 「フォーム一覧」タブより、修正対象フォームの「フォーム編集」アイコンをクリックします。

フォーム一覧

←

アプリケーションID: apid_sample
アプリケーション名: apnm_sample
アプリケーション履歴番号: 1

アプリケーション履歴情報 フォーム一覧 フォーム遷移一覧 ユーザプログラム一覧

登録

編集	フォーム編集	プレビュー	フォーム名	フォーム備考
			apnm_sampleフォーム	

4. IM-VisualDesigner フォームのラベルアイテムと入力系アイテムが IM-FormaDesigner の画面アイテムとして存在することを確認します。

フォーム編集

更新 画像アップロード ラベル一覧 フィールド一覧 グリッド 枠線 再利用 H ヘッダーとフッター ツールキット アイテムコピー 日本語

休暇届

申請日

所属組織

所属 ユーザ名

氏名

以上のとおり休暇を取得します。

〈注意〉 1. 休暇届は、休暇の前日までに提出して下さい。
2. 育児、介護の休暇は別の用紙で届けてください。

期間

開始日

から

終了日

まで

休暇日数

日間

休暇残日数

休暇残業日数

日

休暇区分

休暇区分

事由

事由

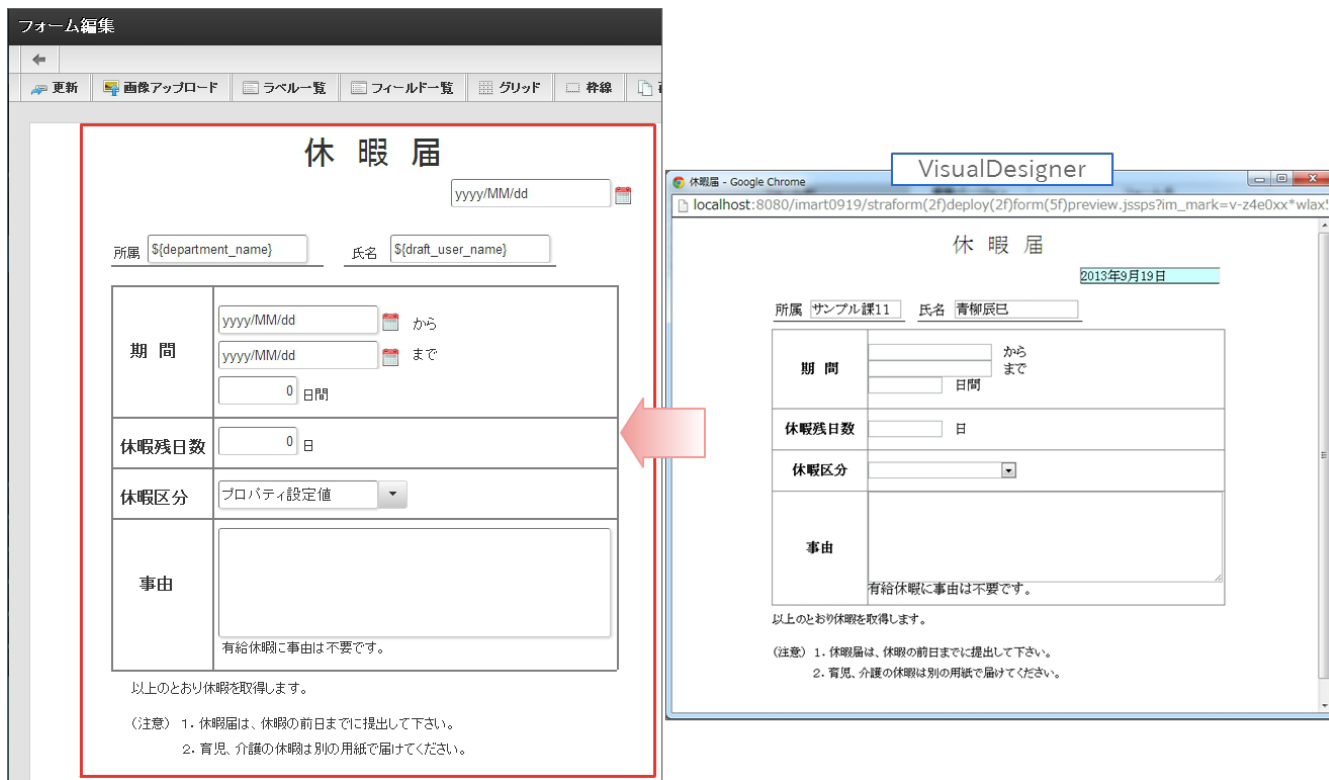
有給休暇に事由は不要です。

i コラム

ユーザ名

IM-FormaDesigner のテキストボックスの左側はラベルアイテムではありません。IM-FormaDesigner の場合、入力系の画面アイテムにラベルと同等のプロパティが付いており、移行時に IM-VisualDesigner の入力系アイテムのアイテム名がそのプロパティに付与されます。

5. IM-VisualDesigner フォームをもとに、IM-FormaDesigner フォームの各アイテムの位置調整や装飾を修正します。以下は、サンプルとして「休暇届」のフォームの修正を行っています。



6. IM-VisualDesigner のフォームで各アイテムに設定していた内容を適宜 IM-FormaDesigner の各アイテムのプロパティに設定します。
IM-FormaDesigner の各アイテムのプロパティの設定方法については、「[IM-FormaDesigner デザインヘルプ](#)」を参照してください。

コラム

IM-VisualDesigner で各フォームに対して設定していたクエリーについては、IM-FormaDesigner のデータソース定義に移行されます。データソース定義と各アイテムとの紐付け設定を適宜行ってください。

注意

IM-VisualDesigner のアクション設定に該当する機能は IM-FormaDesigner では提供されていません。IM-VisualDesigner でアクション設定を利用していた場合は、同様のイベント・アクション処理を記述したJavaScriptのソースコードを用意し、画面アイテム「スクリプト」のプロパティに設定してください。

7. IM-FormaDesigner フォーム修正が完了したら、次の各種設定を行います。

2. 各種設定

- データソース定義設定
 - IM-VisualDesigner のスキーマ名表示設定を有効にした状態で作成したクエリーについては、データソース定義の移行の際に移行されたSQLの中にスキーマ情報が含まれますので、適宜SQLを修正してください。
- 承認画面設定
 - アプリケーション種別が「IM-Workflow」の場合、IM-FormaDesigner 側で承認画面の設定適宜行います。
 1. 「Formaアプリ作成管理」のアプリ一覧画面より、対象アプリケーションの編集アイコンをクリックします。
 2. WF連携設定画面の「WF連携設定」タブを選択し、対象コンテンツの編集アイコンをクリックします。
 3. コンテンツ連携情報画面の「承認画面設定」タブを選択し、承認画面を登録します。

コラム

手順の詳細については「[IM-FormaDesigner 作成者操作ガイド](#)」-「[WF連携設定を行う](#)」の「承認画面設定登録」画面の機能と各部の説明を参照してください。

- 案件プロパティ設定
 - アプリケーション種別が「IM-Workflow」、かつ IM-VisualDesigner で案件プロパティ設定を行っている場合、IM-FormaDesigner 側で再設定が必要です。

1. 「Formaアプリ作成管理」のアプリ一覧画面より、対象アプリケーションの編集アイコンをクリックします。
2. WF連携設定画面の「WF連携設定」タブを選択し、対象コンテンツの編集アイコンをクリックします。
3. コンテンツ連携情報画面の「案件プロパティ設定」タブを選択し、IM-VisualDesigner で設定していた案件プロパティを登録します。

i コラム

手順の詳細については「[IM-FormaDesigner 作成者操作ガイド](#)」-「[WF連携設定を行う](#)」の「[「案件プロパティ登録」画面の機能と各部の説明](#)」を参照してください。

追記設定

- アプリケーション種別が「IM-Workflow」、かつ IM-VisualDesigner で追記設定を行っている場合、IM-FormaDesigner 側で再設定が必要です。
 1. 「Formaアプリ作成管理」のアプリ一覧画面より、対象アプリケーションの編集アイコンをクリックします。
 2. WF連携設定画面の「WF連携設定」タブを選択し、対象コンテンツの編集アイコンをクリックします。
 3. コンテンツ連携情報画面の「追記設定」タブを選択し、IM-VisualDesigner で設定していた追記設定を登録します。

i コラム

手順の詳細については「[IM-FormaDesigner 作成者操作ガイド](#)」-「[WF連携設定を行う](#)」の「[「追記設定登録」画面の機能と各部の説明](#)」を参照してください。

! 注意

「承認画面設定」「案件プロパティ設定」「追記設定」の各設定（WF連携設定）の注意事項

- コピーされたフロー定義、コンテンツ定義、ルート定義以外を用いてWF連携設定を行わないでください。
- コピーされたルート定義内の各ノードに割り振られているノードIDを変更しないでください。

IM-Workflow のユーザプログラム修正

- コンテンツ定義に独自のユーザプログラムを設定している場合、移行支援ツールの実行時に下記の命名規則でパスが変更されます。

IM-VisualDesigner	=> IM-FormaDesigner
スクリプト開発モデル	=> JavaScript のファイル名の末尾に"_forma"を付与
JavaEE開発モデル	=> Java のクラス名の末尾に"4Forma"を付与

- 変更前のファイルをコピーし、変更後のパスにリネームしてください。ファイルの内容については IM-VisualDesigner に依存した処理を削除し IM-FormaDesigner のコンテンツ定義としても動作する内容に修正してください。

3. IM-FormaDesigner 動作検証

- 過去に申請された案件（未完了案件、完了案件、過去案件）が IM-FormaDesigner のアプリケーションで開くことができるかを確認します。
 - IM-VisualDesigner から検証用に移行したデータは、IM-Workflow の処理済案件等から閲覧することはできないため、IM-FormaDesigner の一覧表示機能から参照画面を表示して確認を行います。

1. forma-config.xmlの設定

<CONTEXT_PATH>/WEB-INF/forma-config.xml 内の以下の設定値を false から true に変更します。変更後は再起動が必要です。

```
<!-- imw連携情報（設定変更可能） -->
<imw-setting>
  <!-- アプリケーション種別がim-workflowの場合、メニュー設定タブを表示する場合true/非表示の場合false -->
  <!-- true/falseを設定してください。 -->
  <show_menu_tab>false</show_menu_tab>
  <!-- アプリケーション種別がim-workflowの場合、一覧表示設定タブを表示する場合true/非表示の場合false -->
  <!-- true/falseを設定してください。 -->
  <show_list_item_disp_tab>false</show_list_item_disp_tab>
  <!-- アプリケーション種別がim-workflowの場合、権限設定タブを表示する場合true/非表示の場合false -->
  <!-- true/falseを設定してください。 -->
  <show_auth_setting_tab>false</show_auth_setting_tab>
</imw-setting>
```

2. 作成したアプリケーションの編集

- [権限設定]タブより、確認を行うユーザに対して権限設定を行います。
- [メニュー設定]タブより、一覧画面のメニューを登録します（確認する画面をメニューに登録します）。
- [一覧表示設定]タブより、一覧画面に表示するデータの列を選択します。

3. 一覧画面

- 一覧から確認したいレコードのリンクをクリックして IM-VisualDesigner のレコードを IM-FormaDesigner の画面で確認します。



コラム

forma-config.xmlの設定を行うと、フォーム設定画面に[権限設定]タブ、[メニュー設定]タブ、[一覧表示設定]タブが表示されます。

ステップ2で開発した IM-FormaDesigner のアプリケーションに運用を切替えます。

制限事項

- 移行元が IM-VisualDesigner ワークフロー連携ドキュメント(BPW) である場合、ステップ3は実行しないでください。IM-FormaDesigner がドキュメントワークフロー（BPW）との連携に対応していないため、IM-VisualDesigner で申請・処理された案件（トランザクションデータ）を IM-FormaDesigner に移行できないためです。
- 移行元が他の帳票から連携する（リンク機能）目的にのみ利用されているドキュメントである場合、ステップ3は実行しないでください。ドキュメントに紐づくトランザクションデータが存在しないためです。

作業内容概要

- 1. IM-FormaDesigner アプリケーションへの切替作業
 - IM-VisualDesigner 標準ドキュメント の場合の作業手順
 - IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) の場合の作業手順
 - IM-VisualDesigner ワークフロー連携ドキュメント(BPW) の場合の作業手順
- 2. 切替後の確認作業
 - IM-VisualDesigner 標準ドキュメント の場合の確認観点
 - IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) の場合の確認観点
 - IM-VisualDesigner ワークフロー連携ドキュメント(BPW) の場合の確認観点

1. IM-FormaDesigner アプリケーションへの切替作業

各ドキュメントの種別に応じて、運用中の IM-VisualDesigner のドキュメントを、ステップ2で開発した IM-FormaDesigner のアプリケーションに切替を行います。

注意

移行対象の IM-VisualDesigner のドキュメントに対して一度切替を実行すると、元に戻すことはできませんので注意してください。

IM-VisualDesigner 標準ドキュメント の場合の作業手順

- 移行対象の IM-VisualDesigner のドキュメントの一覧表示メニューを無効にします。
- 移行支援ツールを利用して、移行対象のトランザクションデータの移行を行います。
- 移行対象の IM-FormaDesigner のアプリケーションの一覧表示メニューを有効にします。

IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) の場合の作業手順

- PublicStorage のバックアップを取ります。
※移行支援ツールで IM-Workflow のPublicStorageに保存されている案件データを操作しますが、システムエラーにより失敗した場合に復旧できるように必ずバックアップを取ってください。
- 移行支援ツールを利用して、移行対象のトランザクションデータの移行を行います。
- ステップ2で、検証用に作成された IM-Workflow のフロー定義/コンテンツ定義/ルート定義を削除します。

コラム

過去に申請・処理された案件を IM-FormaDesigner へ移行しない場合はステップ3を実施せず、IM-VisualDesigner の新規の申請を凍結するように設定してください。

フロー定義の新規の申請を凍結するには、最新バージョンのバージョン期間の終了日を設定します。

1. 移行対象の IM-VisualDesigner のドキュメントで使用しているドキュメントワークフロー(BPW)のプロセス定義の終了日を設定します。
2. ステップ2で、検証用に作成された IM-Workflow のフロー定義/コンテンツ定義/ルート定義について、利用開始日を設定し、処理対象者を適宜設定します。

2. 切替後の確認作業

各ドキュメントの種別に応じて、切替後に以下の確認を行います。

IM-VisualDesigner 標準ドキュメント の場合の確認観点

1. 移行されたトランザクションデータが IM-FormaDesigner のアプリケーションより閲覧できることを確認します。

IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) の場合の確認観点

1. IM-Workflow の処理済一覧より、IM-VisualDesigner で処理された案件が IM-FormaDesigner のアプリケーションで閲覧できることを確認します。
2. IM-Workflow の申請一覧画面から移行対象のフローを選択した場合に、IM-FormaDesigner のアプリケーションが表示されることを確認します。

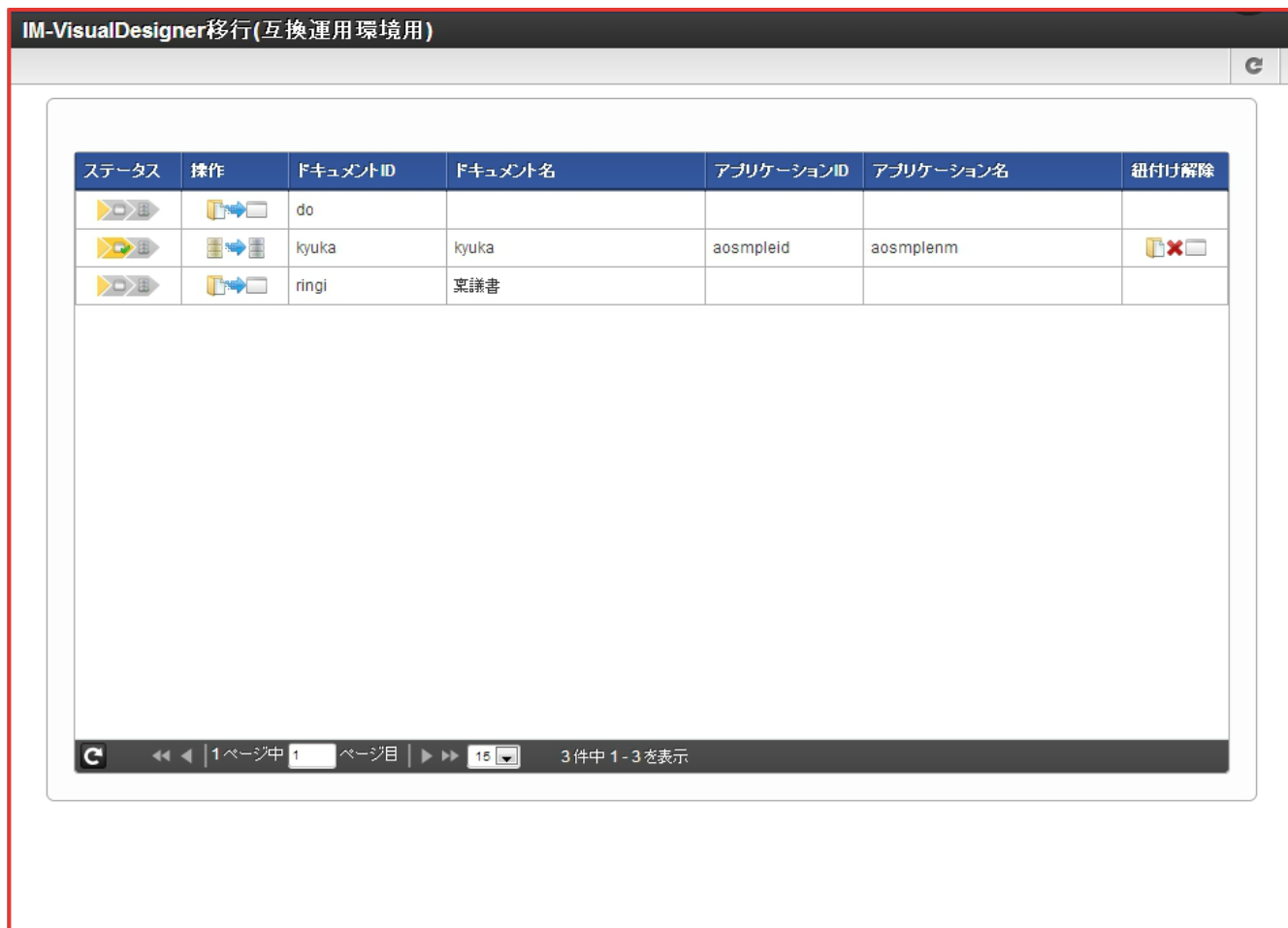
IM-VisualDesigner ワークフロー連携ドキュメント(BPW) の場合の確認観点

1. IM-Workflow の申請一覧画面から移行対象のフローを選択した場合に、IM-FormaDesigner のアプリケーションが表示されることを確認します。

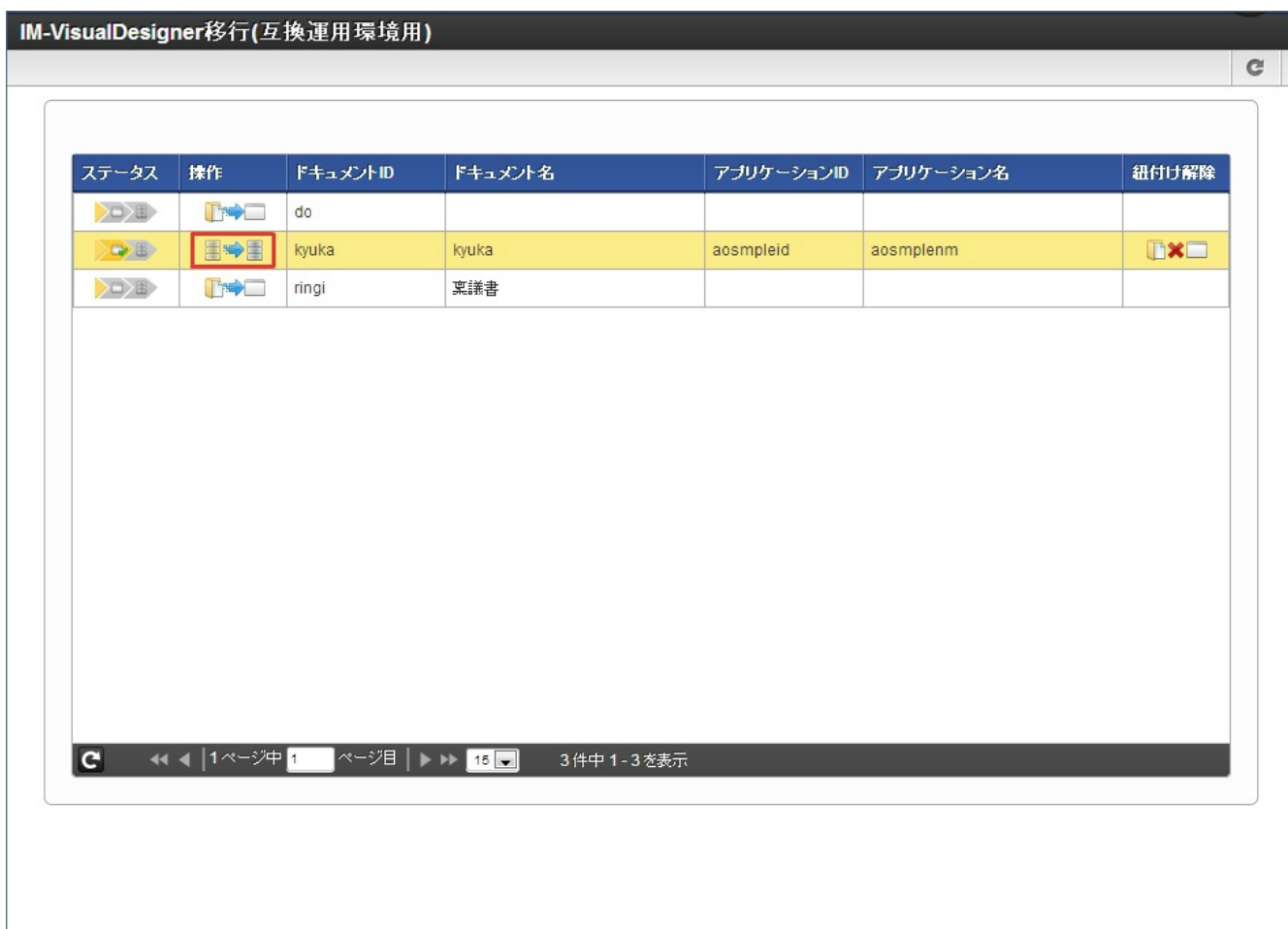
IM-VisualDesigner → IM-FormaDesigner 切替手順

1. IM-VisualDesigner ドキュメント→IM-FormaDesigner アプリケーション 切替

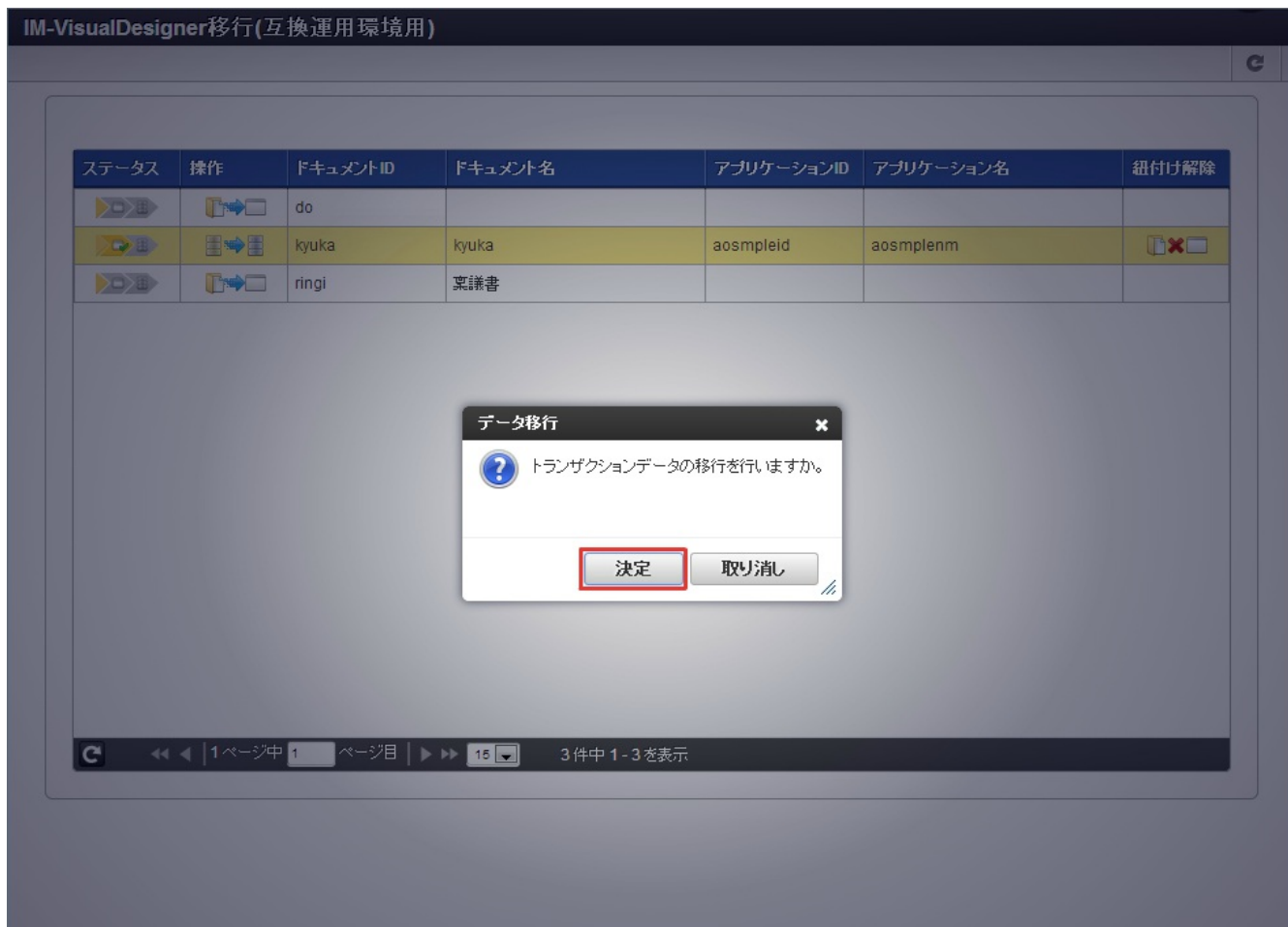
1. intra-mart WebPlatform 環境のフルバックアップ（データベース、storage、treasure等）を行います。
2. VD管理者でログインし、「IM-VisualDesigner移行（互換運用環境用）」画面を開きます。



3. ステップ2で対応した切替対象のドキュメントの操作アイコンをクリックします。



4. 「トランザクションデータの移行を行いますか。」の問いに「決定」をクリックします。

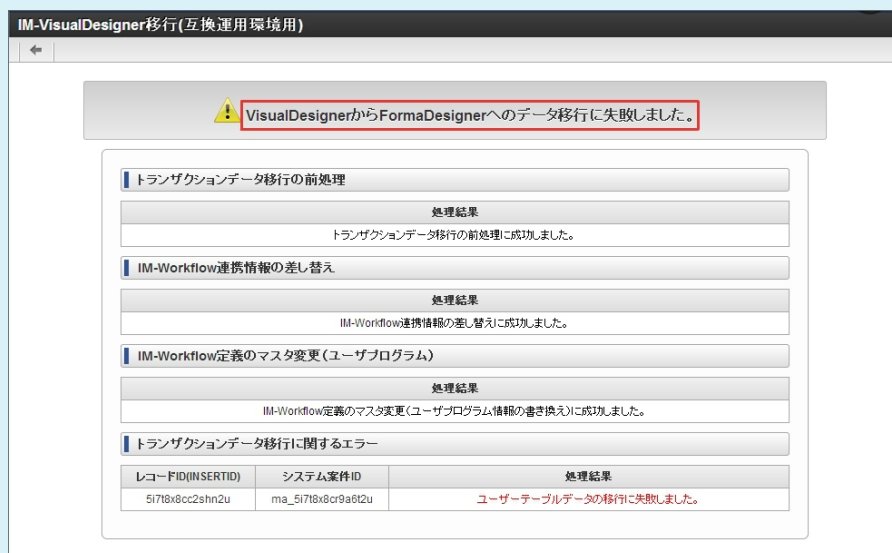


5. 正常に完了していることを確認します。



コラム

エラーが発生した場合は、エラーデータの修正を行い、データ移行が正常に完了するまで上記の操作をリトライします。



検証手順

1. 切替検証

- 過去に申請されたものも含めて全ての案件が、IM-FormaDesigner のアプリケーションで正常に表示されることを確認します。

移行支援ツールの内部仕様

ステップ1 IM-VisualDesigner Ver.7.2 から intra-mart Accel Platform への移行

コラム

トランザクションデータのうち、PublicStorageにコピーした添付ファイルの情報ファイルの文字コードを intra-mart WebPlatform 7.2 をインストールした際の文字コードから intra-mart Accel Platform に設定されている文字コードへ変換します。

ステップ2 IM-VisualDesigner のドキュメントから IM-FormaDesigner のアプリケーションのひな形生成

i コラム

入力された情報を基に IM-FormaDesigner のアプリケーションを生成します。IM-VisualDesigner のドキュメント情報からフォーム・テーブルに関する情報を抽出し、IM-FormaDesigner 用にデータの変換を行い、生成したアプリケーションに設定します。

- IM-FormaDesigner 用に生成されるテーブルのカラム名・フィールドID
Oracleデータベースをご利用の場合、23文字以上のフィールドIDが存在するフォームを変換すると、22文字以上のフィールドIDは一部の情報がシステム側で自動採番された値に更新されます。カラム名も同様に変換されます。Oracleデータベースではカラム名の長さに関して30バイトの制限があるため、製品間でのカラム接頭辞の差を考慮し変換処理を実施しております。
- **IM-VisualDesigner**
 - vd_ud_[フィールドID]
- **IM-FormaDesigner**
 - imfr_ud_[フィールドID]
- IM-FormaDesigner 用に生成されるテーブルのテーブル名
 - ヘッダーテーブル
 - imfr_ut_[アプリケーションID]
 - 明細行テーブル
 - imfr_ut_[アプリケーションID]_[IM-VisualDesigner のテーブルアイテムのID]
- IM-FormaDesigner に移行される情報

IM-VisualDesigner	=> IM-FormaDesigner
フォーム	=> フォーム（注1）
テーブル設定	=> テーブル設定（注2）
DB参照機能（クエリー）	=> データソース定義（注3）
IM-Workflow 連携設定 （ドキュメントID - コンテンツID）	=> WF連携設定 （アプリケーションID - コピーされたコンテンツID）

（注1） フォームの変換については、「[フォーム変換ルール](#)」を参照してください。

（注2） テーブル設定には IM-FormaDesigner 用に生成されたるテーブルの情報が設定されます。

（注3） データソース定義については、DB参照機能（クエリー）設定のSQLと抽出条件が移行されます。

- IM-Workflow に移行される情報

IM-VisualDesigner	=> IM-FormaDesigner
ドキュメントに紐付くコンテンツ定義	=> コピーされたコンテンツ定義
コンテンツ定義に紐付くフロー定義	=> コピーされたフロー定義
フロー定義に紐付くルート定義	=> コピーされたルート定義
案件プロパティ設定、追記設定	=>（注4）

（注4） 案件プロパティ設定、追記設定は IM-FormaDesigner 側に移行されないため、別途設定が必要です。設定手順については「[IM-FormaDesigner アプリケーションの開発および検証する手順](#)」を参照してください。

（注5） 移行元が IM-VisualDesigner ワークフロー連携ドキュメント(IM-Workflow) でない場合、IM-Workflow の定義情報/連携情報は作成されません。

ステップ3 IM-VisualDesigner → IM-FormaDesigner 切替

i コラム

運用中の IM-VisualDesigner のドキュメントを、ステップ2で開発した IM-FormaDesigner のアプリケーションへの切替を行います。

IM-VisualDesigner の画面で入力されたトランザクションデータと IM-Workflow の案件データを IM-FormaDesigner へ移行します。

1. IM-FormaDesigner のアプリケーションへの切替

IM-Workflow 連携しているドキュメントの場合、データの移行前に IM-Workflow の定義情報/案件情報に対して IM-VisualDesigner のドキュメントから IM-FormaDesigner のアプリケーションへの切替を行います。

具体的には、コンテンツ定義の画面のパスとユーザプログラムのパスを IM-FormaDesigner の情報へ差し替えます。

- 【画面パスの変換】

画面定義に設定されている IM-VisualDesigner のパスを IM-FormaDesigner のパスへ変換します。

```

一時保存画面
straform/imw/view/apply_view → forma/imw/view/apply_view

再申請画面
straform/imw/view/retry_view → forma/imw/view/retry_view

処理画面
straform/imw/view/approve_view → forma/imw/view/approve_view

処理詳細画面
straform/imw/view/reference_view → forma/imw/view/reference_view

参照画面
straform/imw/view/reference_view → forma/imw/view/reference_view

未申請画面
straform/imw/view/apply_view → forma/imw/view/apply_view

申請画面
straform/imw/view/apply_view → forma/imw/view/apply_view

確認画面
straform/imw/view/approve_view → forma/imw/view/approve_view

確認詳細画面
straform/imw/view/reference_view → forma/imw/view/reference_view

過去案件詳細画面
straform/imw/view/reference_view → forma/imw/view/reference_view

```

- 【ユーザプログラム定義の変換】

```

承認アクション処理
straform/imw/process/approve_process → forma/imw/process/approve_process

案件プロパティ設定処理
straform/imw/process/edit_matter_prop_process → forma/imw/process/edit_matter_prop_process

申請アクション処理
straform/imw/process/apply_process → forma/imw/process/apply_process

```

- 【独自のユーザプログラム・画面パスの変換】

独自のユーザプログラム・画面パスについては、下記の変換ルールでパスの差し替えを行います。

IM-VisualDesigner	=> IM-FormaDesigner
スクリプト開発モデル	=> JavaScript のファイル名の末尾に"_forma"を付与
JavaEE開発モデル	=> Java のファイル名の末尾に"4Forma"を付与

- 【IM-FormaDesigner WF連携設定のコンテンツIDの変換】

IM-Workflow のコンテンツ定義と IM-FormaDesigner のアプリケーションの紐づき設定を変更します。

ひな形生成時にコピーしたコンテンツ定義から IM-VisualDesigner のドキュメントが紐づいていたコンテンツ定義へ紐づきを変更します。

2. データ移行

データ移行については、1レコードずつトランザクションを分離します。

- 【入力データ管理テーブル情報】

IM-VisualDesigner ドキュメントの画面上から入力されたデータを IM-FormaDesigner のアプリケーション用のテーブルへ移行します。

IM-VisualDesigner	=> IM-FormaDesigner
ヘッダテーブル	=> ヘッダテーブル
明細行テーブル	=> 明細テーブル

- 【添付ファイル情報】

IM-VisualDesigner ドキュメントの画面上からアップロードされた添付ファイルを IM-FormaDesigner の管理テーブルとPublicStorageへ移行します。

IM-VisualDesigner	=> IM-FormaDesigner
PublicStorageファイル Storage Service/Straform/attach/%グループ ID%/commit/%ドキュメントID%/登録ID%/	=> PublicStorageファイル forma/fileupload/アプリケーション ID/commit/ファイルID => 管理テーブル (imfr_t_fileupload)

- 【案件テーブルのデータ移行】

IM-Workflow の案件と画面から入力された情報を紐づけるためのレコードを IM-VisualDesigner の管理テーブル ("vd_t_imw_matter") から IM-FormaDesigner の管理テーブル ("imfr_t_imw_matter") へ移行します。

- IM-Workflow の案件のシステム案件ID - IM-VisualDesigner ドキュメントのインサートID
- IM-Workflow の案件のシステム案件ID - IM-FormaDesigner アプリケーションの登録データID (insertId)

- 【IM-Workflow 案件情報の変更】

コンテンツ定義の設定に対して実施した画面のパスとユーザプログラムのパスの変更を IM-Workflow の展開された各案件の情報に対しても実施します。

フォーム変換ルール

ステップ2の IM-VisualDesigner → IM-FormaDesigner 変換により、フォームやフォーム内の各アイテムがどのように変換されるかの一覧です。

- フォーム名
IM-VisualDesigner のフォーム名が IM-FormaDesigner のフォーム名に変換されます。
- 画面遷移
IM-VisualDesigner 上で設定されていた画面遷移は移行されません。
ただし、IM-VisualDesigner の「登録画面」については、IM-FormaDesigner の「メインフォーム」として設定されます。
- レイアウト
レイアウトは移行されません。
移行後は縦一列にアイテムが整列します。
- ラベル
画面アイテム「ラベル」に変換されます。

- 画面アイテム「ラベル」

移行されるプロパティ

名前	=> アイテム名
文字列	=> ラベル

- テキストフィールド
data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「文字列」
-----------	----------------

変換される画面アイテム

データ型「数値」 => 画面アイテム「数値」

データ型「日付」 => 画面アイテム「日付」

■ 画面アイテム「文字列」

移行されるプロパティ

名前 => フィールド識別ID

表示名 => アイテム名、フィールド識別名、ラベル

文字列 => フィールド初期値

■ 画面アイテム「数値」

移行されるプロパティ

名前 => フィールド識別ID

表示名 => アイテム名、フィールド識別名、（前）ラベル

文字列 => フィールド初期値

■ 画面アイテム「日付」

移行されるプロパティ

名前 => フィールド識別ID

表示名 => アイテム名、フィールド識別名、ラベル

■ テキストエリア

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」 => 画面アイテム「複数行文字列」

データ型「数値」 => 画面アイテム「数値」

データ型「日付」 => 画面アイテム「日付」

■ 画面アイテム「複数行文字列」

移行されるプロパティ

名前 => フィールド識別ID

表示名 => アイテム名、フィールド識別名、ラベル

文字列 => フィールド初期値

■ 画面アイテム「数値」

移行されるプロパティ

名前 => フィールド識別ID

表示名 => アイテム名、フィールド識別名、（前）ラベル

文字列 => フィールド初期値

■ 画面アイテム「日付」

移行されるプロパティ

名前 => フィールド識別ID

表示名 => アイテム名、フィールド識別名、ラベル

■ ボタン

attach_file の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

attach_file = true => 画面アイテム「ファイルアップロード」

attach_file = false => 移行されない

- 画面アイテム「ファイルアップロード」

移行されるプロパティ

名前	=> アイテム識別ID
文字列（登録・申請時）	=> ラベル

- チェックボックス

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「チェックボックス」
データ型「数値」	=> 画面アイテム「数値」
データ型「日付」	=> 画面アイテム「日付」

- 画面アイテム「チェックボックス」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル

- 画面アイテム「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、（前）ラベル
文字列	=> フィールド初期値

- 画面アイテム「日付」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル

- ラジオボタン

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「ラジオボタン」
データ型「数値」	=> 画面アイテム「数値」
データ型「日付」	=> 画面アイテム「日付」

- 画面アイテム「ラジオボタン」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル
項目（項目）	=> 項目定義（表示値）
項目（送信項目）	=> 項目定義（送信値）

※ アイテムにクエリ設定を行っている場合、その設定情報は移行されません（クエリは移行されます）。
クエリ設定を行う場合は、IM-FormaDesigner 側で「外部連携」の「データソース設定値」の設定を行ってください。

- 画面アイテム「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、（前）ラベル
文字列	=> フィールド初期値

- 画面アイテム「日付」

移行されるプロパティ	
名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル

- コンボボックス

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム	
データ型「文字列」	=> 画面アイテム「セレクトボックス」
データ型「数値」	=> 画面アイテム「数値」
データ型「日付」	=> 画面アイテム「日付」

- 画面アイテム「セレクトボックス」

移行されるプロパティ	
名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル
項目（項目）	=> 項目定義（表示値）
項目（送信項目）	=> 項目定義（送信値）

※ アイテムにクエリ設定を行っている場合、その設定情報は移行されません（クエリは移行されます）。
クエリ設定を行う場合は、IM-FormaDesigner 側で「外部連携」の「データソース設定値」の設定を行ってください。

- 画面アイテム「数値」

移行されるプロパティ	
名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、（前）ラベル
文字列	=> フィールド初期値

- 画面アイテム「日付」

移行されるプロパティ	
名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル

- リスト

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム	
データ型「文字列」	=> 画面アイテム「リストボックス」
データ型「数値」	=> 画面アイテム「数値」
データ型「日付」	=> 画面アイテム「日付」

- 画面アイテム「リストボックス」

移行されるプロパティ	
名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル
項目（項目）	=> 項目定義（表示値）
項目（送信項目）	=> 項目定義（送信値）

※ アイテムにクエリ設定を行っている場合、その設定情報は移行されません（クエリは移行されます）。
クエリ設定を行う場合は、IM-FormaDesigner 側で「外部連携」の「データソース設定値」の設定を行ってください。

- 画面アイテム「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、（前）ラベル
文字列	=> フィールド初期値

- 画面アイテム「日付」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、ラベル

- レイヤ

移行されません。

- テーブルアイテム

「テーブル行を追加」アクションの有無によって、変換される画面アイテム、およびテーブル内の子アイテムの変換アイテムが異なります。

変換される画面アイテム

「テーブル行を追加」有り	=> 画面アイテム「明細テーブル」
「テーブル行を追加」無し	=> 移行されない

- 「テーブル行を追加」有りのテーブルに紐づく子アイテムの場合

- ラベル

画面アイテム「ラベル」に変換されます。

※ テーブルとしては変換されず、テーブル外部に通常の画面アイテムとして変換されます。

- 画面アイテム「ラベル」

移行されるプロパティ

名前	=> アイテム名
文字列	=> ラベル

- テキストフィールド

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「明細テーブル」の列定義「文字列」
データ型「数値」	=> 画面アイテム「明細テーブル」の列定義「数値」
データ型「日付」	=> 画面アイテム「明細テーブル」の列定義「日付」

- 画面アイテム「明細テーブル」の列定義「文字列」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「日付」

移行されるプロパティ

名前	=> フィールド識別ID
----	--------------

移行されるプロパティ

表示名	=> アイテム名、フィールド識別名、列名
-----	----------------------

- テキストエリア
data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「明細テーブル」の列定義「文字列」
データ型「数値」	=> 画面アイテム「明細テーブル」の列定義「数値」
データ型「日付」	=> 画面アイテム「明細テーブル」の列定義「日付」

- 画面アイテム「明細テーブル」の列定義「文字列」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「日付」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名

- ボタン
attach_file の値によって、変換される画面アイテムが異なります。
※ テーブルとしては変換されず、テーブル外部に通常の画面アイテムとして変換されます。

変換される画面アイテム

attach_file = true	=> 画面アイテム「ファイルアップロード」
attach_file = false	=> 移行されない

- 画面アイテム「ファイルアップロード」

移行されるプロパティ

名前	=> アイテム識別ID
文字列（登録・申請時）	=> ラベル

- チェックボックス
data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「明細テーブル」の列定義「文字列」
データ型「数値」	=> 画面アイテム「明細テーブル」の列定義「数値」
データ型「日付」	=> 画面アイテム「明細テーブル」の列定義「日付」

- 画面アイテム「明細テーブル」の列定義「文字列」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名

- 画面アイテム「明細テーブル」の列定義「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「日付」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名

- ラジオボタン

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「明細テーブル」の列定義「ラジオボタン」
データ型「数値」	=> 画面アイテム「明細テーブル」の列定義「数値」
データ型「日付」	=> 画面アイテム「明細テーブル」の列定義「日付」

- 画面アイテム「明細テーブル」の列定義「ラジオボタン」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
項目（項目）	=> 項目定義（表示値）
項目（送信項目）	=> 項目定義（送信値）

※ アイテムにクエリ設定を行っている場合、その設定情報は移行されません（クエリは移行されます）。クエリ設定を行う場合は、IM-FormaDesigner 側で「外部連携」の「データソース設定値」の設定を行ってください。

- 画面アイテム「明細テーブル」の列定義「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「日付」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名

- コンボボックス

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「明細テーブル」の列定義「セレクトボックス」
データ型「数値」	=> 画面アイテム「明細テーブル」の列定義「数値」
データ型「日付」	=> 画面アイテム「明細テーブル」の列定義「日付」

- 画面アイテム「明細テーブル」の列定義「セレクトボックス」

移行されるプロパティ

名前	=> フィールド識別ID
----	--------------

移行されるプロパティ

表示名	=> アイテム名、フィールド識別名、列名
項目（項目）	=> 項目定義（表示値）
項目（送信項目）	=> 項目定義（送信値）

※ アイテムにクエリ設定を行っている場合、その設定情報は移行されません（クエリは移行されます）。クエリ設定を行う場合は、IM-FormaDesigner 側で「外部連携」の「データソース設定値」の設定を行ってください。

- 画面アイテム「明細テーブル」の列定義「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「日付」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名

- リスト

data_type の値によって、変換される画面アイテムが異なります。

変換される画面アイテム

データ型「文字列」	=> 画面アイテム「明細テーブル」の列定義「セレクトボックス」
データ型「数値」	=> 画面アイテム「明細テーブル」の列定義「数値」
データ型「日付」	=> 画面アイテム「明細テーブル」の列定義「日付」

- 画面アイテム「明細テーブル」の列定義「セレクトボックス」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
項目（項目）	=> 項目定義（表示値）
項目（送信項目）	=> 項目定義（送信値）

※ アイテムにクエリ設定を行っている場合、その設定情報は移行されません（クエリは移行されます）。クエリ設定を行う場合は、IM-FormaDesigner 側で「外部連携」の「データソース設定値」の設定を行ってください。

- 画面アイテム「明細テーブル」の列定義「数値」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名
文字列	=> フィールド初期値

- 画面アイテム「明細テーブル」の列定義「日付」

移行されるプロパティ

名前	=> フィールド識別ID
表示名	=> アイテム名、フィールド識別名、列名

- レイヤ

移行されません。

- 直線

移行されません。

- 矩形
 - 移行されません。
 - 「テーブル行を追加」無しのテーブルに紐づく子アイテムの場合
 - テーブルに紐付かない通常のアイテムとして変換されます。変換ルールは各アイテムの変換ルールを参照してください。
- 直線
 - 移行されません。
- 矩形
 - 移行されません。
- イメージ
 - 移行されません。

i コラム

IM-VisualDesigner 側のボックス系アイテム（チェックボックス、ラジオボタン、コンボボックス、リスト）のプロパティ「送信項目」に2バイト文字を利用していた場合は、IM-FormaDesigner では扱えません。

IM-VisualDesigner とIM-FormaDesignerの機能比較

IM-VisualDesigner と IM-FormaDesignerの機能比較一覧です。

IM-VisualDesigner Ver.7.2	IM-FormaDesigner for Accel Platform
<p>【データソース定義機能】 - 検索クエリの作成方法</p> <p>テーブルから任意のカラムを選択して検索クエリを作成します。</p>	<p>ユーザがSQLを記述し、検索クエリを作成します。 SQLの記述についてはViewCreatorを用いることで容易化できません。</p>
<p>【テーブルアイテム】 - テーブル内に配置できるアイテムの種類</p> <p>すべてのフィールドアイテムに対応しています。</p>	<p>一部のフィールドアイテムのみ対応しており、以下のフィールドアイテムはテーブル内に配置できません。</p> <ul style="list-style-type: none"> ・ 複数行文字列 ・ リストボックス ・ チェックボックス
<p>【ボックス系アイテム】 - 対応するデータ型</p> <p>文字列、数値、日付に対応しています。</p>	<p>文字列のみ対応しています（2バイト文字を含む文字列には対応していません）。</p>
<p>【関数機能】 - 利用方法</p> <p>指定のコードにより、テキストフィールドの初期値として関数を利用します。</p>	<p>画面アイテム「関数」を利用します。</p>
<p>【イベント・アクション設定機能】 - 設定方法</p> <p>イベントの種類とアクションの種類を設定します。</p>	<p>ユーザが画面アイテム「スクリプト」を用いてクライアントJavaScriptのコードを記述します。</p>
<p>【イベント・アクション設定機能】 - リンク機能</p> <p>別画面をポップアップで表示できます。 呼び出し先の画面は下記4つの中から選択可能です。</p> <ul style="list-style-type: none"> ・ スクリプト開発モデル ・ JavaEE開発モデル ・ IM-VisualDesigner で作成したドキュメント ・ 外部URL 	<p>別画面をポップアップで表示できます。 呼び出し先の画面は「IM-FormaDesigner で作成した画面」のみが利用可能です。</p>

