

ASTERIA Warp 評価ガイド



目次

ステップ

0

事前準備

» [P.3](#)

- インストールの確認
- ユーザーの作成
- テストデータのコピー

ステップ

1

フローの作成

» [P.5](#)

- 新規プロジェクトの追加
- コンポーネントの配置
- プロパティの設定
- ストリームの定義
- マッピングの設定
- 実行結果の確認

ステップ

2

エラー処理の設定

» [P.19](#)

- エラー処理の概要
- メインフローのコピー
- エラー処理フローの設定
- 実行結果の確認

ステップ

3

デザイナーの便利機能

» [P.2](#)

- デバッグ実行
- 実行時間の計測(プロファイルモード)
- ドキュメントの出力(仕様書出力)

事前準備

本評価ガイドに沿ってフローの開発を進める前に
テストデータなどの環境の準備を行います。
手順にしたがって環境を確認してください。



インストールの確認

フローの開発・実行のため、下記の環境がインストールされていることを確認してください。

- ASTERIA Warp for Windows/ASTERIA Warp for Linux(ASTERIA Warpサーバー)
- ASTERIA Warp Designer for Windows(フローデザイナー)

インストールが完了していない場合には下記ドキュメントを参考にインストールを完了してください。

- Windows版ASTERIA Warp簡易インストールガイド
(<https://support.asteria.com/hc/ja/articles/360000391761>)
- インストールガイド
(<https://help.asteria.com/documentation/warp/ja/current/InstallGuide.pdf>)



ユーザーの作成

インストール後の初期設定などで開発用ユーザーアカウントを作成していない場合には
下記手順を参考に管理コンソール(FSMC)からユーザーアカウントを作成してください。

フローサービスマニュアル-はじめに-フローの基本-アカウントの作成

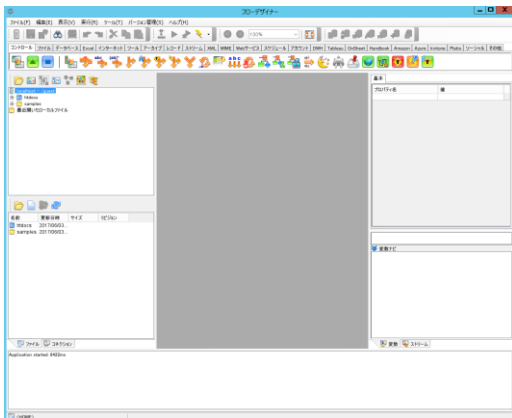
(<https://help.asteria.com/documentation/warp/ja/current/flow/designer/flowservice/flow.html#flow.mc.account>)



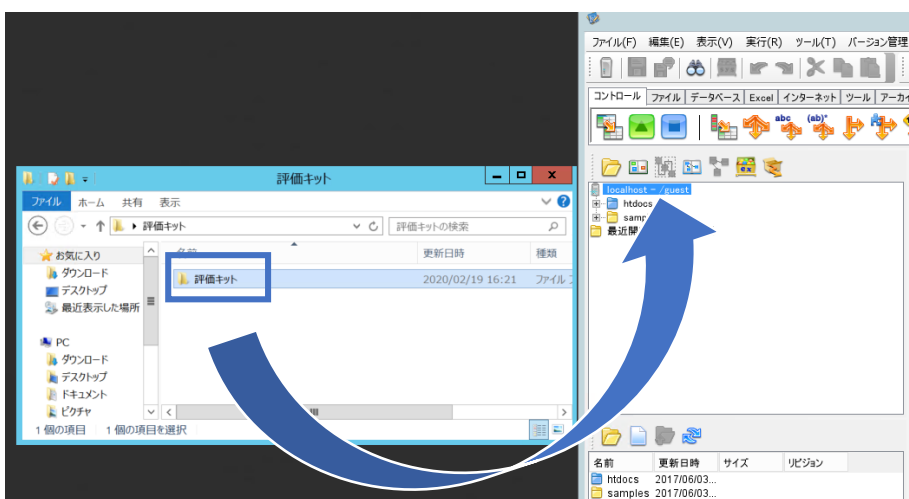
テストデータのコピー

本評価ガイドに添付されたテストデータを開発ユーザーのホームフォルダーにコピーします。

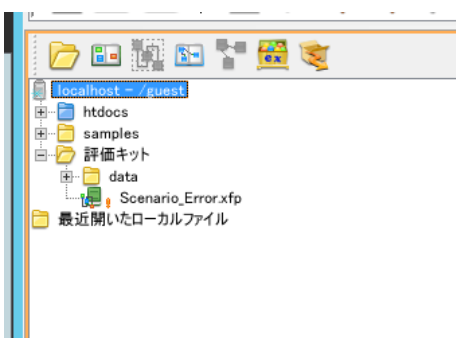
1. フローデザイナーを起動し、該当のユーザーアカウントでログインしてください。



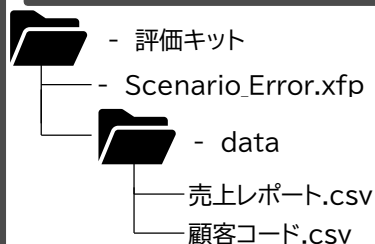
2. フローデザイナーをインストールした環境で「評価キット.zip」を解凍します。
3. 評価キットフォルダをツリーパインのサーバー名にドラッグ&ドロップで配置します。



4. 評価キットフォルダがツリーパイン上に表示されます。



評価キットのフォルダ構成



ステップ1 フローの作成

シナリオに沿って新規フローを開発します。

データの加工/変換を含めた
フローデザイナーの設定手順・操作感を確認してください。



作成するフローの概要



別システムから出力される「売上レポート」を読み込み、
フィールド(取引区分)の値に応じて返品情報のみを「返品一覧」に出力します。
出力する際、「売上レポート」のフィールド(顧客コード)を顧客名に変換します。

売上レポート.csv

	受注ID	売上日	取引区分	顧客コード	金額
1	1	2020/1/1	1	C0001	1000
2	2	2020/1/3	2	C0004	3000
3	3	2020/1/5	1	C0002	3000
4	4	2020/1/7	2	C0001	1000
5	5	2020/1/10	2	C0005	1000
6	6	2020/1/11	1	C0002	2000
7	7	2020/1/13	2	C0003	6000
8	8	2020/1/15	2	C0003	2000
9	9	2020/1/20	1	C0002	2000
10	10	2020/1/25	1	C0001	3000
11					

返品一覧.csv

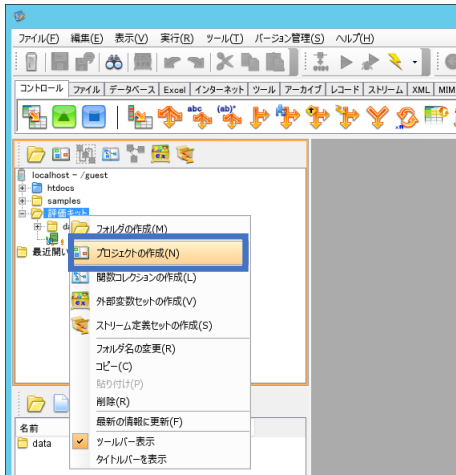
	受注ID	売上日	顧客名	金額
1	2	2020/1/3	顧客4	3000
2	4	2020/1/7	顧客1	1000
3	5	2020/1/10	顧客5	1000
4	7	2020/1/13	顧客3	6000
5	8	2020/1/15	顧客3	2000
6				



新規プロジェクトの追加

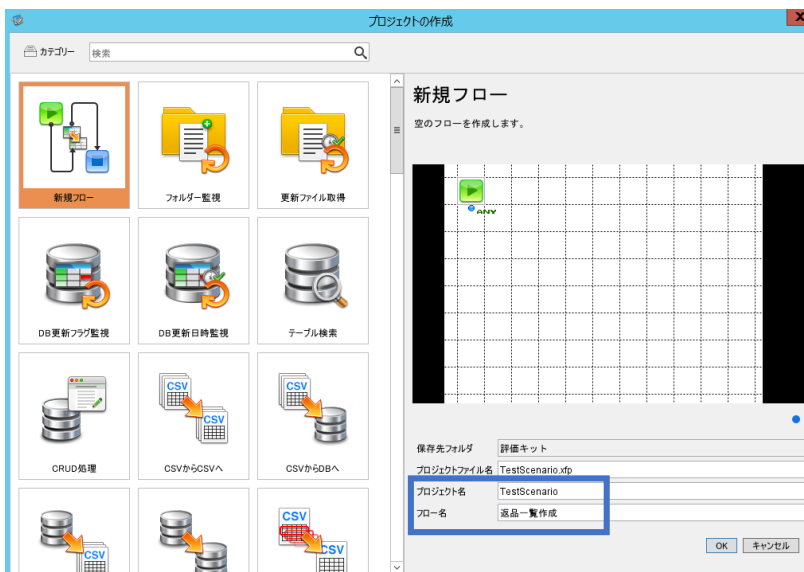
新規プロジェクトを追加します。

1. ツリーペイン上の評価キットフォルダを右クリックし、「プロジェクトの追加」を選択します。

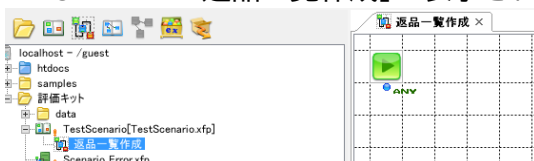


2. プロジェクトの作成ウィンドウで下記のようにプロジェクト名・フロー名を変更しOKボタンを押下、新プロジェクト・フローを追加します。

- プロジェクト名: TestScenario
- フロー名: 返品一覧作成



3. ツリーペイン上に新しいプロジェクト「TestScenario」が追加され、ワークスペースに新しいフロー「返品一覧作成」が表示されます。

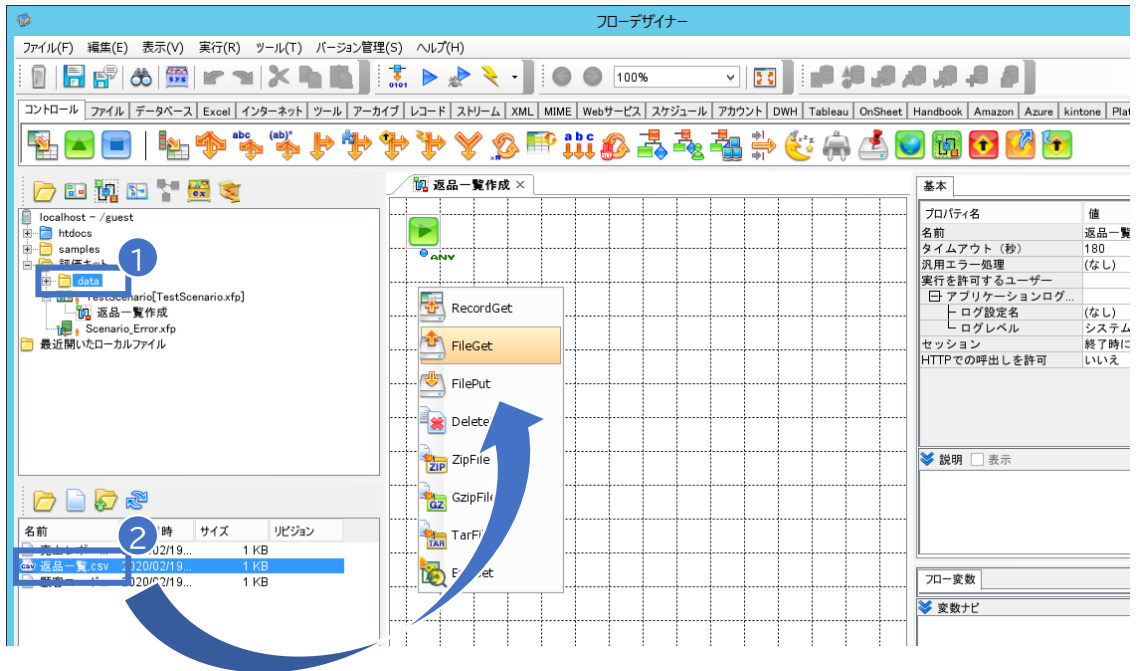




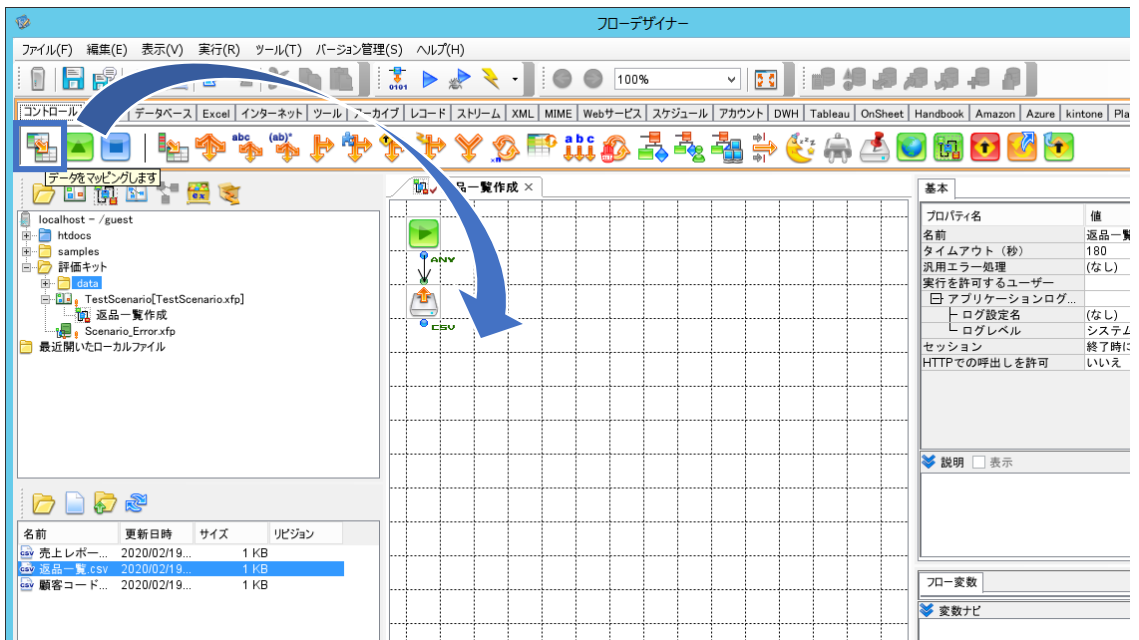
コンポーネントの配置

ワークスペースにコンポーネントを配置し、処理の流れを定義します。

1. ツリーペイン上「data」フォルダを選択後、ファイルペインに表示される「売上レポート.csv」をワークスペースにドラッグ&ドロップで配置し、表示されるコンポーネント一覧からFileGetコンポーネントを選択します。



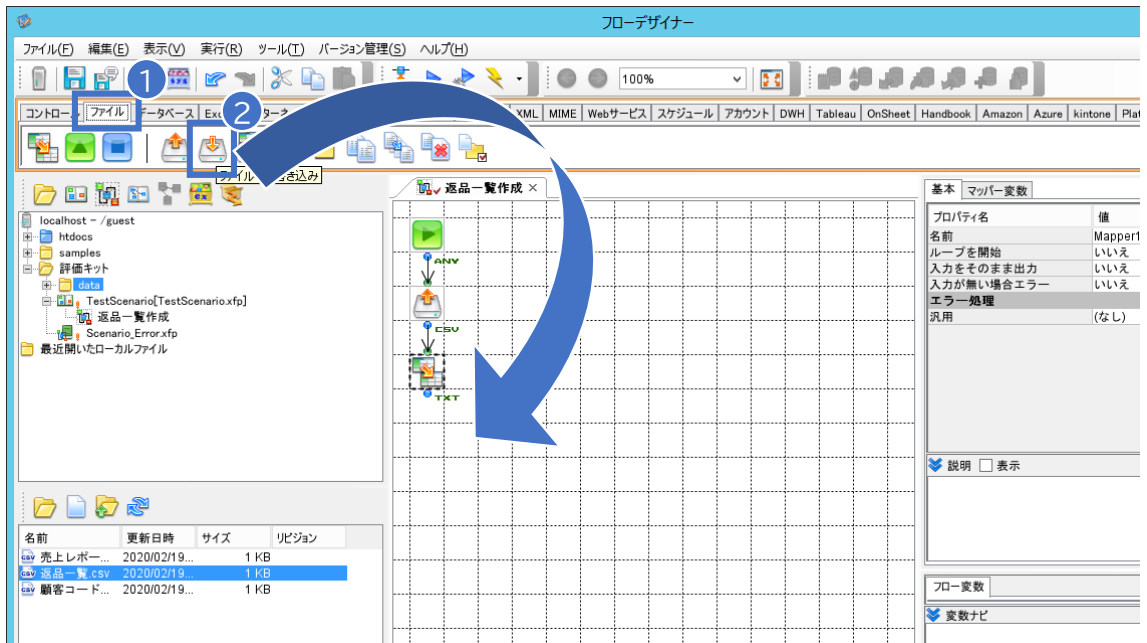
2. パレット上のMapperコンポーネントをドラッグ&ドロップでワークスペースに配置します。



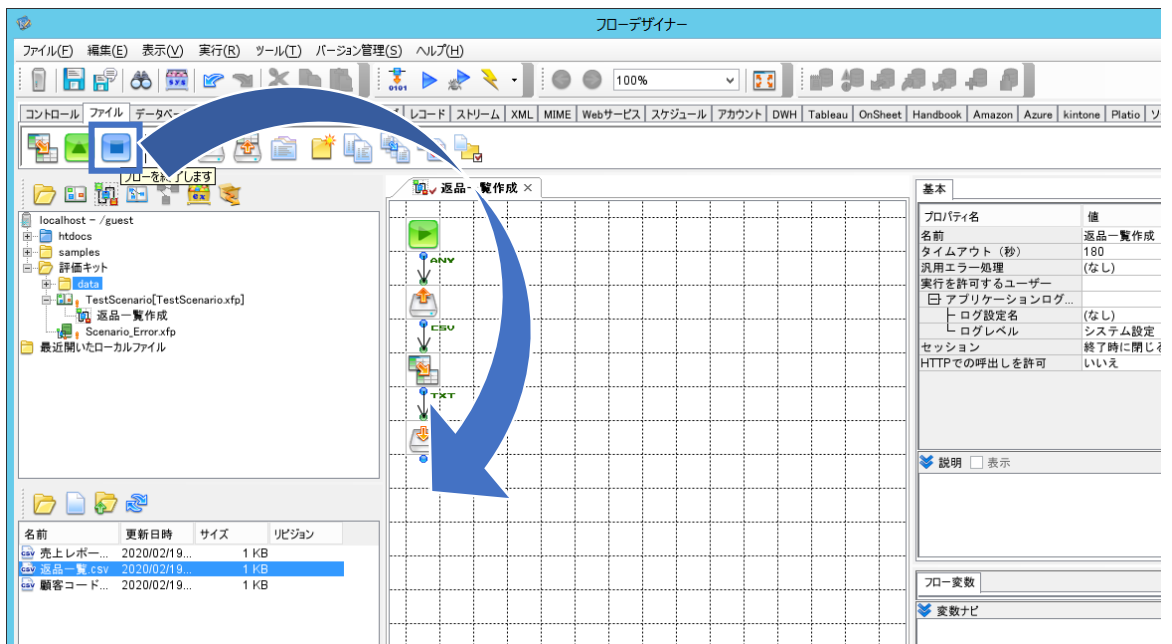


コンポーネントの配置

3. パレット上の「ファイル」タブを選択し、FilePutコンポーネントをドラッグ&ドロップでワークスペースに配置します。



4. パレット上のEndコンポーネントをドラッグ&ドロップでワークスペースに配置します。

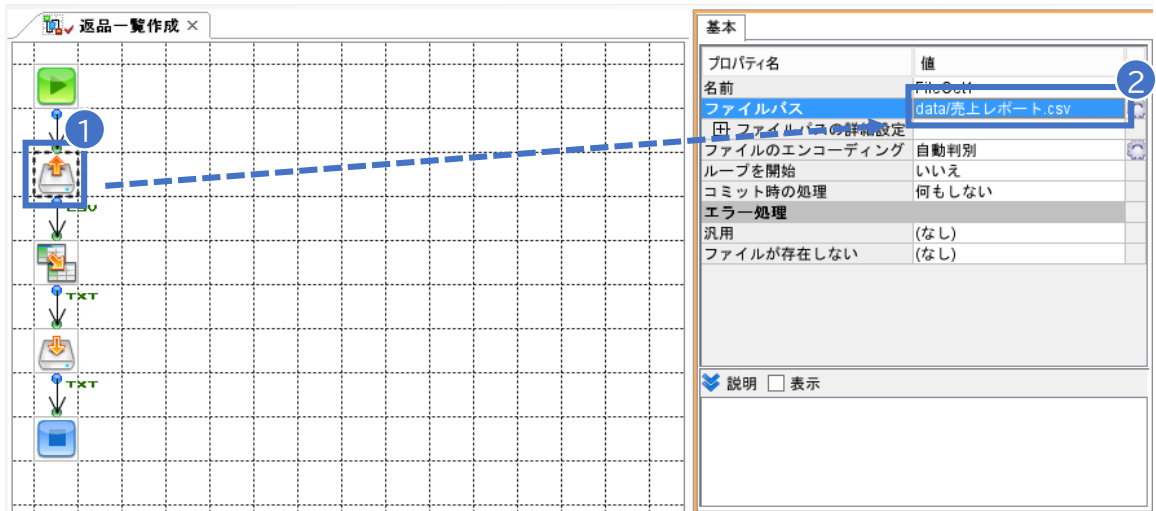




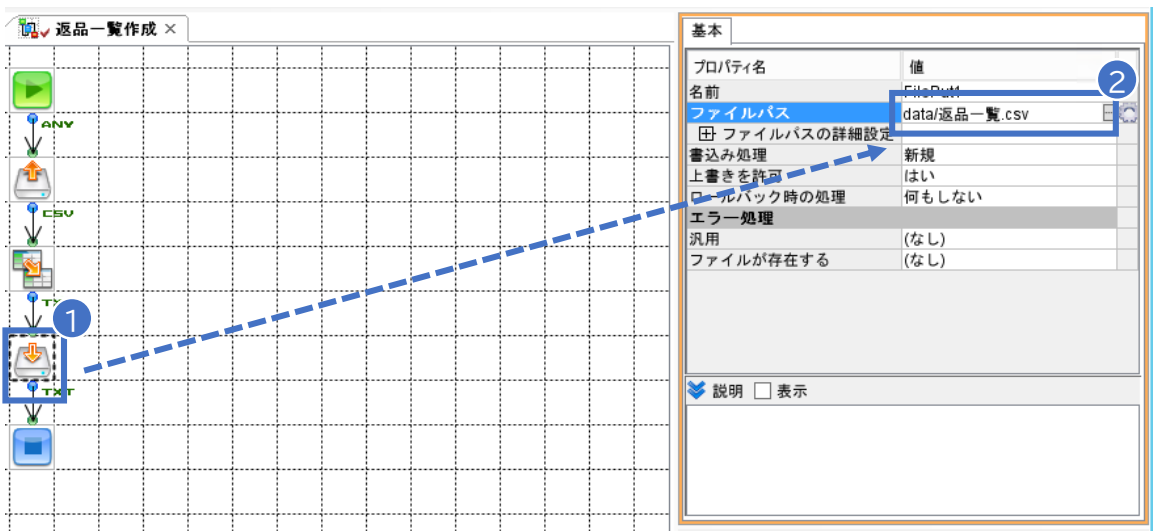
プロパティの設定

配置した各コンポーネントに対し、動作に必要な情報(プロパティ)を設定します。

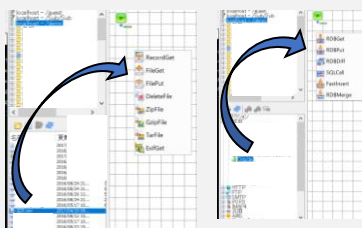
1. ワークスペース上に配置したFileGetコンポーネントを選択し、インスペクタ上の「ファイルパス」プロパティに値(data/売上レポート.csv)がセットされていることを確認します。



2. ワークスペース上に配置したFilePutコンポーネントを選択し、インスペクタ上の「ファイルパス」プロパティに値(data/返品一覧.csv)を入力します。



■一口メモ ショートカットでプロパティ設定



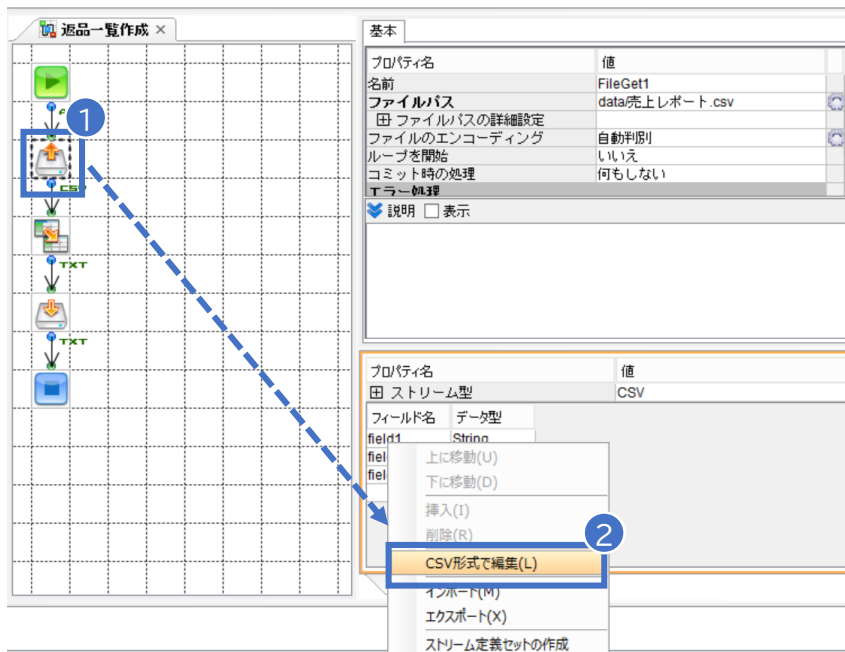
ファイルペインやコネクションペインからドラッグ&ドロップでワークスペースに配置したコンポーネントは各プロパティの設定を済ませた状態で配置されます。



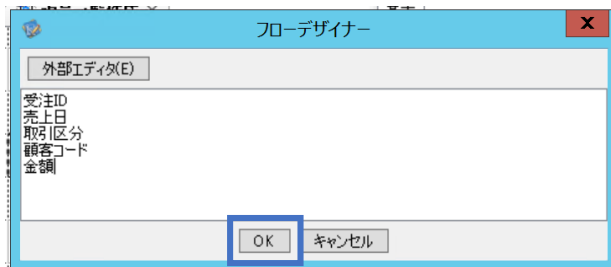
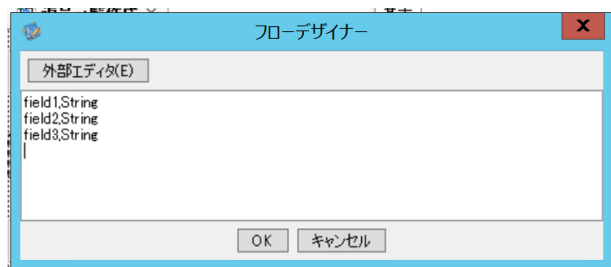
ストリームの定義

配置した各コンポーネントに対し、入出力するデータフォーマット(ストリーム定義)を定義します。

1. ワークスペース上に配置したFileGetコンポーネントを選択し、ファイルペイン上の各フィールド名の上で右クリックし、「CSV形式で編集」を選択します。



2. 開かれた編集画面でフィールド名を下記のように変更し、OKボタンを押下します。





ストリームの定義

- ワークスペース上に配置したMapperコンポーネントを選択し、ファイルペイン上の「ストリーム型」を“CSV”に変更します。

基本 マッパー変数

プロパティ名	値
名前	Mapper1
ループを開始	いいえ
入力をそのまま出力	いいえ
入力がない場合エラー	いいえ
エラー処理	
汎用	(なし)

説明 ☐ 表示

プロパティ名 値

ストリーム型	Text
出力エンコーディング	Text
	HTML
	ParameterList
	record
	CSV
	FixedLength
	MIME
	XML

変数 ストリーム

- 表示されたCSVストリームのプロパティ一覧で「出力エンコーディング」プロパティを“utf-8”から“Windows-31J”に変更します。

基本 マッパー変数

プロパティ名	値
名前	Mapper1
ループを開始	いいえ
入力をそのまま出力	いいえ
入力がない場合エラー	いいえ

説明 ☒ 表示

プロパティ名 値

ストリーム型	CSV
読み込み開始行	1
フィールド数	3
区切り文字	,
囲み文字	"
囲み処理の対象	すべて
空データの囲み処理	する
囲み文字のエスケープ	バックスラッシュ
空白を削除	はい
出力エンコーディング	utf-8
改行コード	utf-8
先頭行のフィールド名を出力	shift_jis
フィールド名	データ型
field1	String

出力エンコーディング

utf-8

Windows-31J



ストリームの定義

5. 「ストリーム型」プロパティの横の“ー”マークをクリックしストリームプロパティを折りたたんだ後、フィールド一覧を右クリックし、「CSV形式で編集」を選択します。

The screenshot shows the 'Stream Type' property expanded in the left pane. A blue arrow points from the minus sign next to 'Stream Type' to the 'CSV Form Edit' option in the right pane's context menu.

プロパティ名	値
名前	Mapper1
ループを開始	はい
入力をそのまま出力	はい
入力がない場合エラー	はい

プロパティ名	値
フィールド名	データ型
field1	String
field2	String
field3	String

6. 開かれた編集画面でフィールド名を下記のように変更し、OKボタンを押下します。

The screenshot shows the 'Flow Designer' dialog box with the 'External Data (E)' tab selected. The field names are being changed from 'field1', 'field2', and 'field3' to '受注ID', '売上日', and '顧客名' respectively. A blue arrow points from the 'CSV Form Edit' option in the previous screenshot to this dialog box.

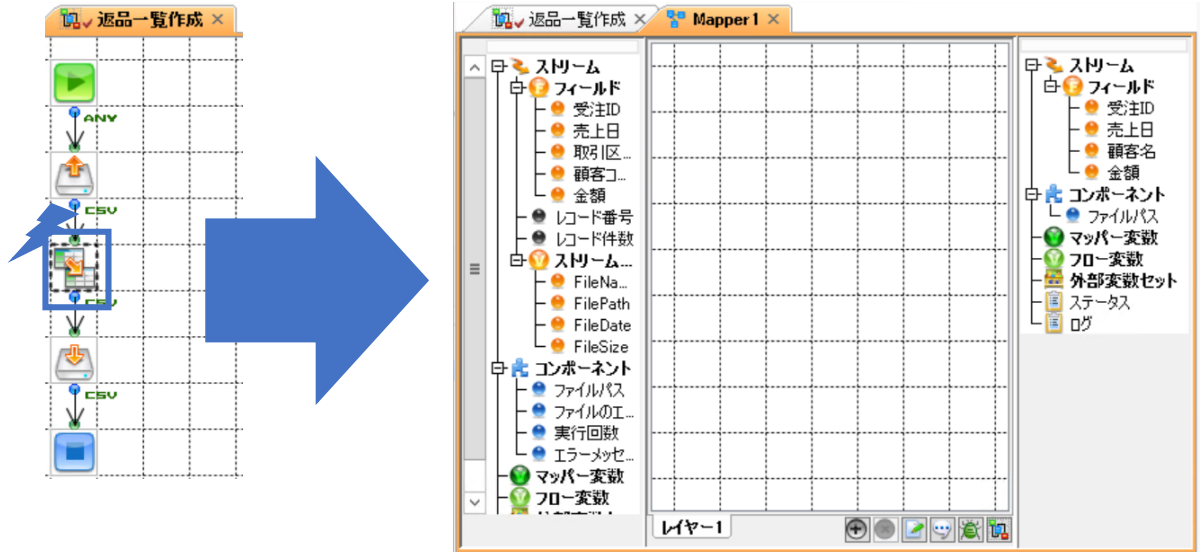
外部エディタ(E)
受注ID
売上日
顧客名
金額



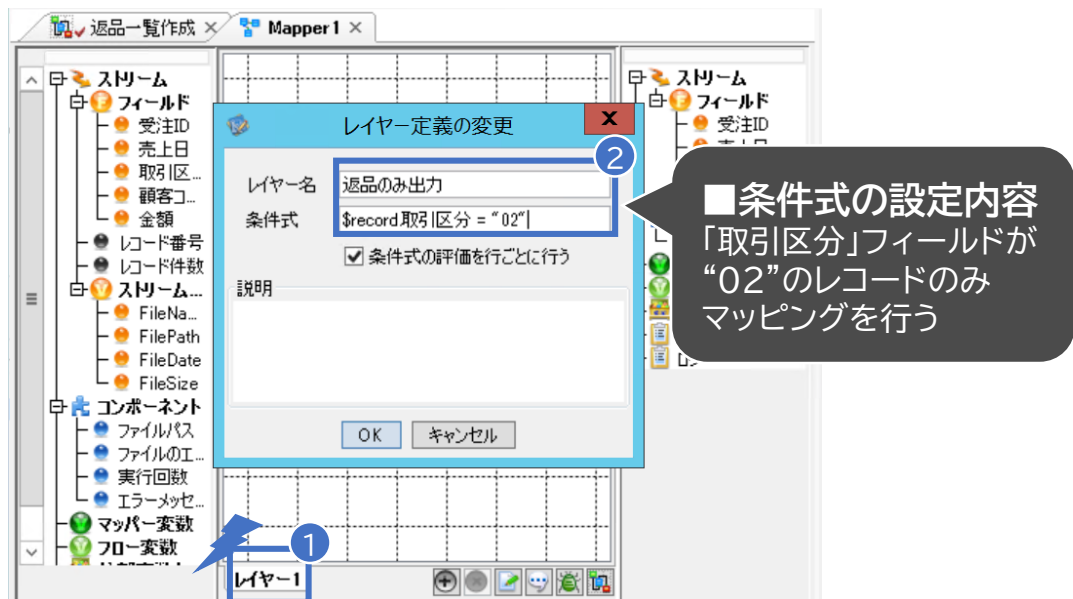
マッピングの設定

出力したいフォーマットに合わせてマッピングを定義します。このとき、「マッパー関数」を利用したデータの加工/変換、「条件付きレイヤー」を利用したレコードの絞り込みを行います。

1. ワークスペース上に配置したMapperコンポーネントをダブルクリックしマッパー画面を開きます。



2. レイヤー名をダブルクリックし開かれたレイヤー定義ダイアログで下記のように設定します。



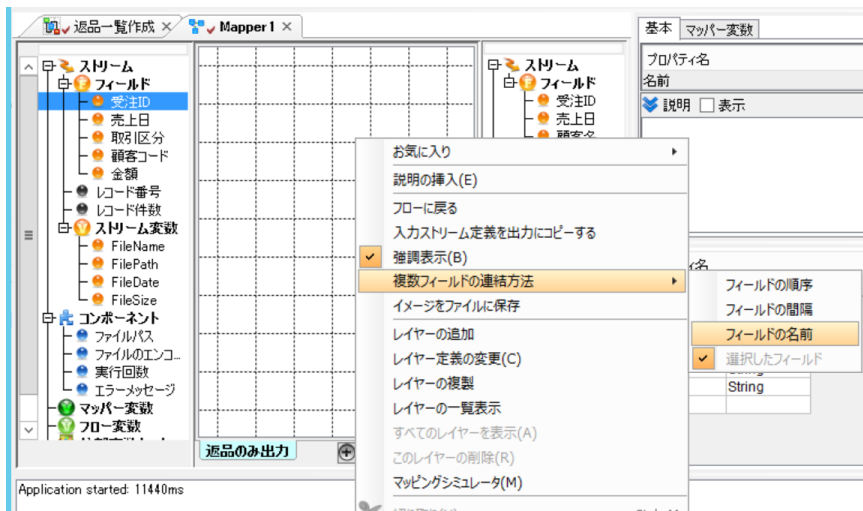
■一口メモ 条件付きレイヤー

レイヤーは多数のマッピングのリンクを見やすくするために使いますが、ある特定の条件のときだけマッピングするように定義することもできます。条件を定義したレイヤー、「条件付きレイヤー」はレイヤーの追加時に条件式を指定します。条件を設定したレイヤーは、マッピングウィンドウ下部のレイヤー名タブの色が変わります。条件の追加時には入力補完も利用でき便利です。

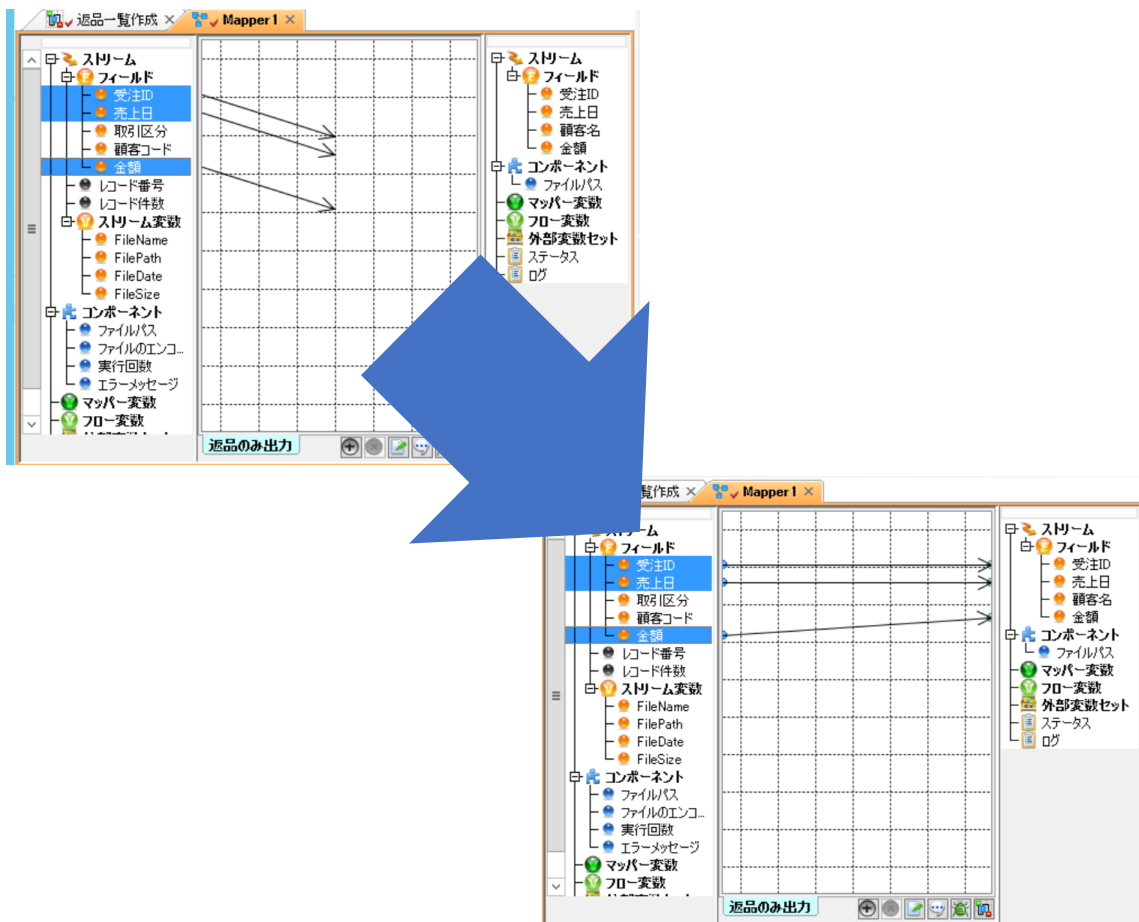


マッピングの設定

3. マッパー画面の右クリックし、「複数フィールドの連結方法」で「フィールドの名前」を選択します。



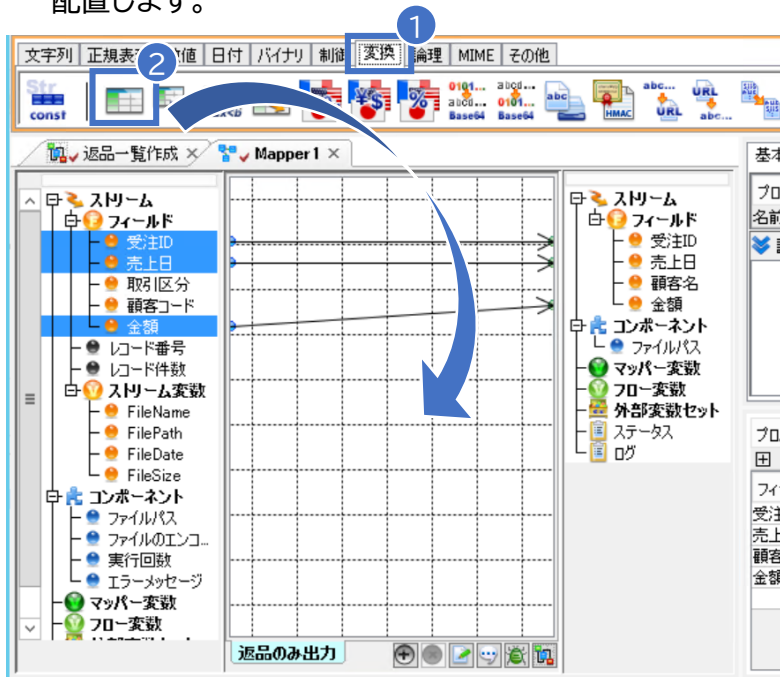
4. データ変換の必要のない「受注ID」「売上日」「金額」をctrlボタンを押下しながら複数選択し、そのままドラッグ&ドロップでマッピングします。



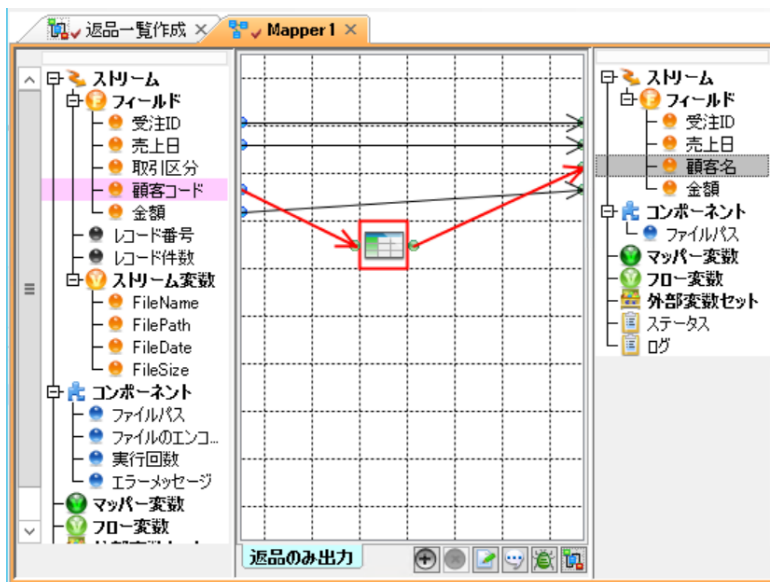


マッピングの定義

5. パレット上の「変換」タブを選択し、Table関数をドラッグ&ドロップでマッパー画面の上に配置します。



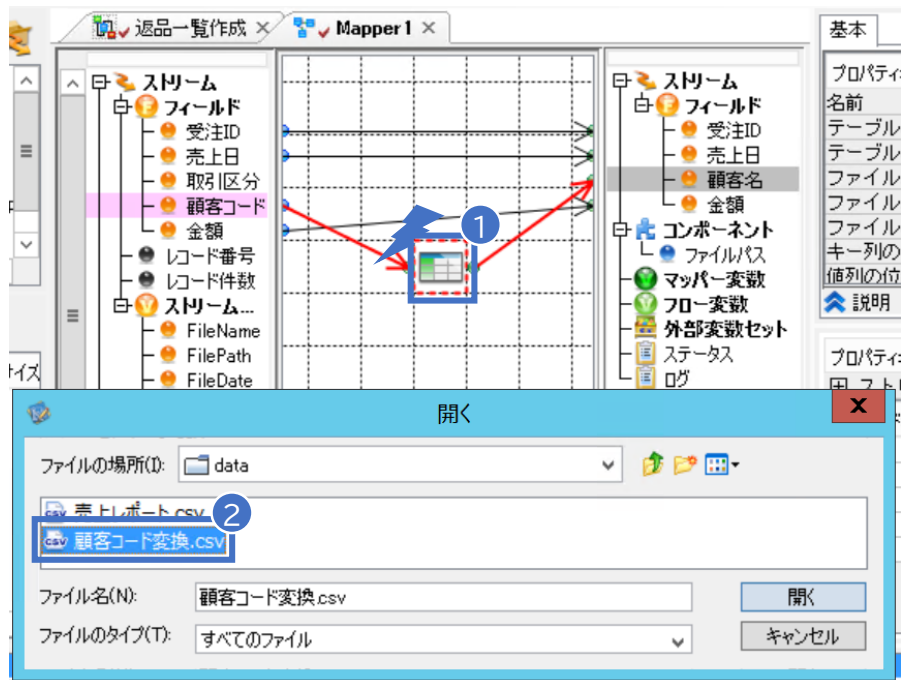
6. 「顧客コード」からTable関数、Table関数から「顧客名」へマッピングします。





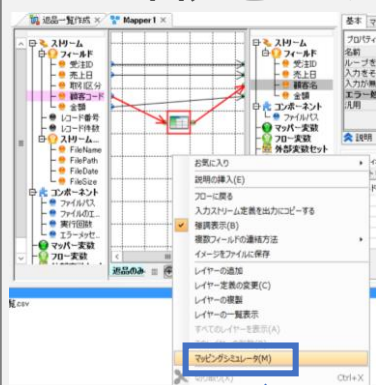
マッピングの定義

7. マッパー画面上のTable関数ダブルクリックし開かれたファイル選択画面で「data/顧客コード変換.csv」を選択する。



フローの設定はこれで完了です。実行して結果を確認してみましょう。

■一口メモ マッピングシミュレーター



マッパー画面の右クリックメニューで呼び出せる「マッピングシミュレーター」とは、選択したレイヤーでのマッピングを実行して結果を表示する機能です。シミュレーションできるのは、マッパーコンポーネントと関数コレクションの関数です。

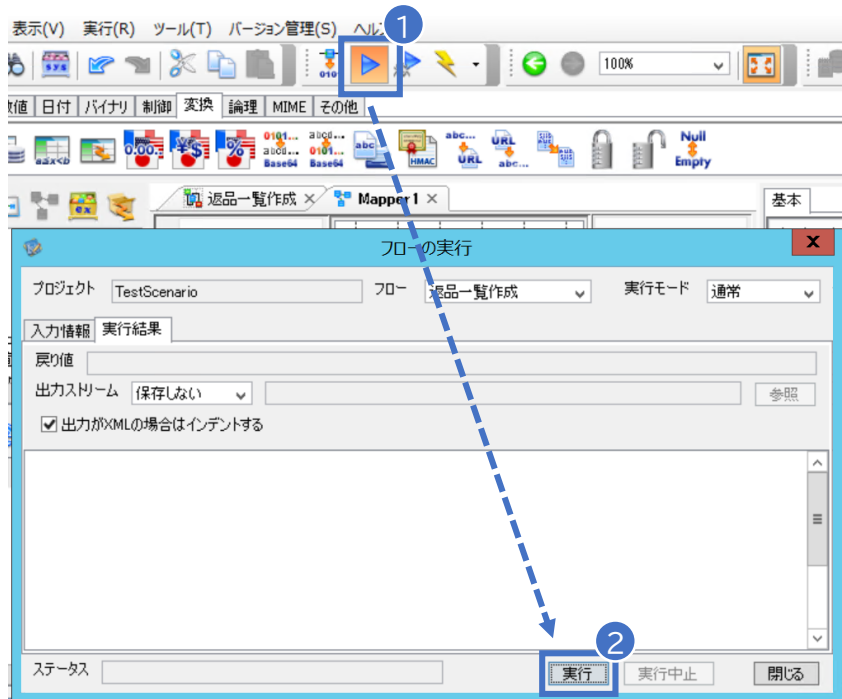
シミュレーターウィンドウで入力フィールドに値を設定して実行すると、関連づけられたマッパー関数の実行結果が出力フィールドに表示され、実行状況を確認することができます。



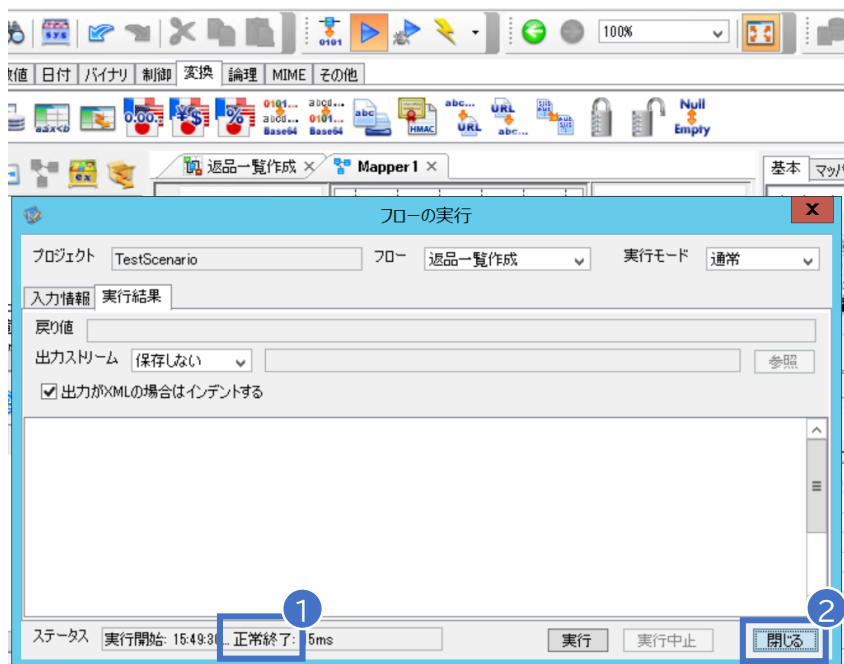
実行結果の確認

作成したフローを実行し、正常に動作するか、結果データは出力されているかを確認します。

1. フローデザイナー画面上部の実行ボタンをクリック、実行コンソールを呼び出し、実行ボタンを押下します。



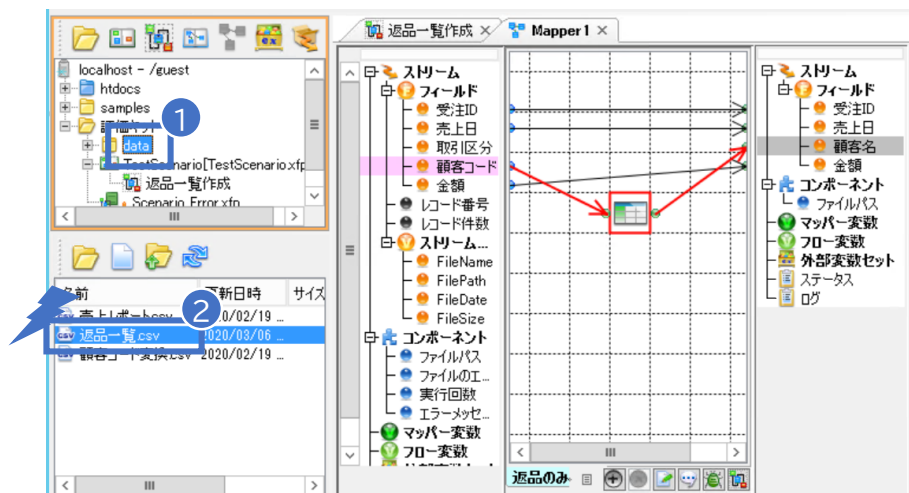
2. ステータスが“正常終了”していることを確認し、実行コンソールを閉じます。





実行結果の確認

- ツリーペイン上の「data」フォルダ(ファイルの出力先)を選択し、ファイルペインに表示された結果ファイル(返品一覧.csv)をダブルクリックします。



- 結果ファイル(返品一覧.csv)が開かれ、結果データが確認出来ます。

	A	B	C	D	E
1	2	2020/1/3	顧客4	3000	
2	4	2020/1/7	顧客1	1000	
3	5	2020/1/10	顧客5	1000	
4	7	2020/1/13	顧客3	6000	
5	8	2020/1/15	顧客3	2000	
6					
7					

結果ファイルは対象のファイル拡張子(今回の場合は".csv")に紐づいたアプリケーションで開かれます。

実行結果の確認が出来ました。

変換前のファイルと比較すると出力されるレコードが絞り込まれ、値が変換されています。

	A	B	C	D	E
1	1	2020/1/1	1 C0001	1000	
2	2	2020/1/3	2 C0004	3000	
3	3	2020/1/5	1 C0002	3000	
4	4	2020/1/7	2 C0001	1000	
5	5	2020/1/10	2 C0005	1000	
6	6	2020/1/11	1 C0002	2000	
7	7	2020/1/13	2 C0003	6000	
8	8	2020/1/15	2 C0003	2000	
9	9	2020/1/20	1 C0002	2000	
10	10	2020/1/25	1 C0001	3000	
11					
12					

	A	B	C	D	E
1	2	2020/1/3	顧客4	3000	
2	4	2020/1/7	顧客1	1000	
3	5	2020/1/10	顧客5	1000	
4	7	2020/1/13	顧客3	6000	
5	8	2020/1/15	顧客3	2000	
6					
7					

売上レポート.csv
(データファイル)

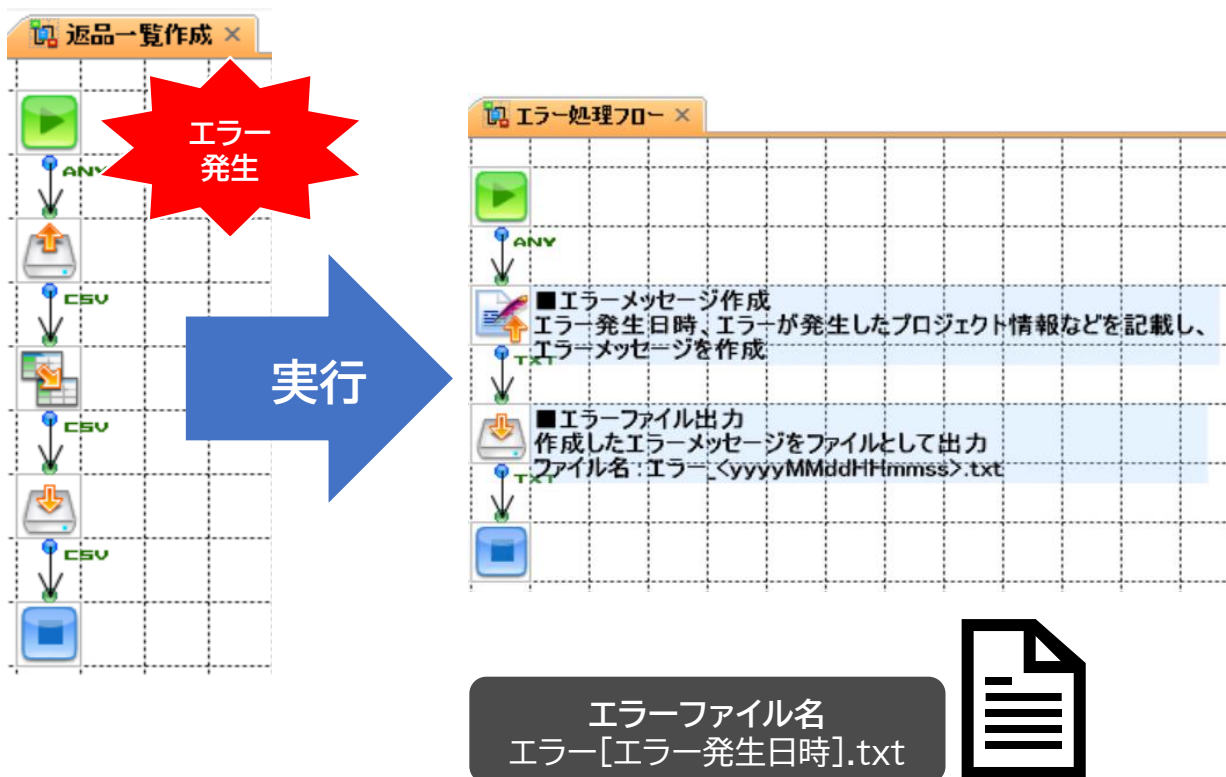
返品一覧.csv
(結果ファイル)

ステップ2 エラー処理の設定

エラーが発生した場合に備えて
ステップ1で作成したフローに対して「エラー処理フロー」を設定します。
ASTERIA Warpではどのようにエラーハンドリングを行うのか
確認してください。

エラー処理の概要

エラーが発生すると「エラーの発生日時」や「エラーの発生したプロジェクト名」「エラーの発生したフロー名」、「エラーメッセージ」などを記録したメッセージファイルを出力します。



■一口メモ エラー処理フロー

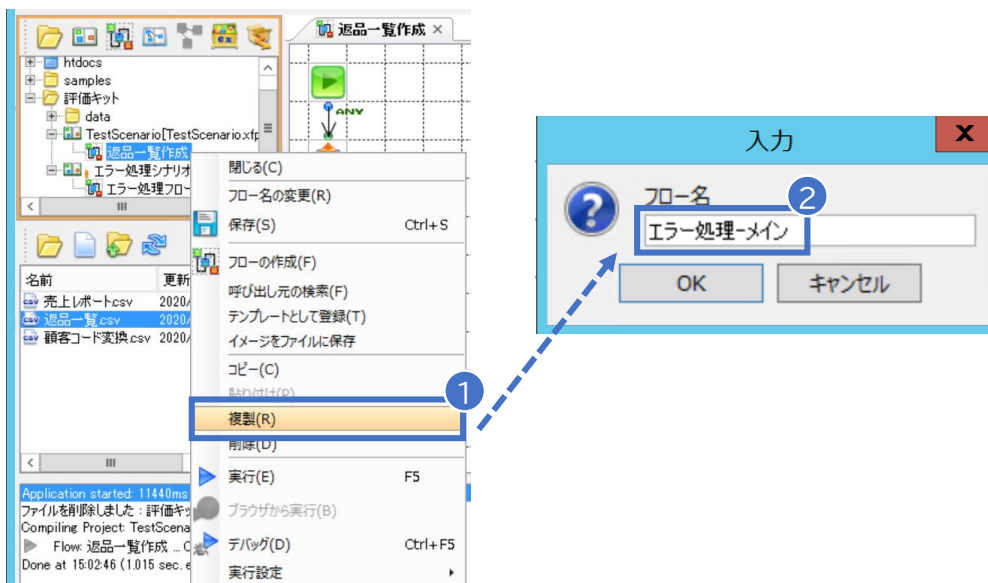
エラー処理プロパティの設定で特定のフローを選択した場合、フローまたはコンポーネントでエラーが発生した場合にそのフローが呼び出されます。このように呼び出されるフローを「エラー処理フロー」といい、呼び出し元フローのセッション、コネクション、トランザクション化状態を引き継ぎます。



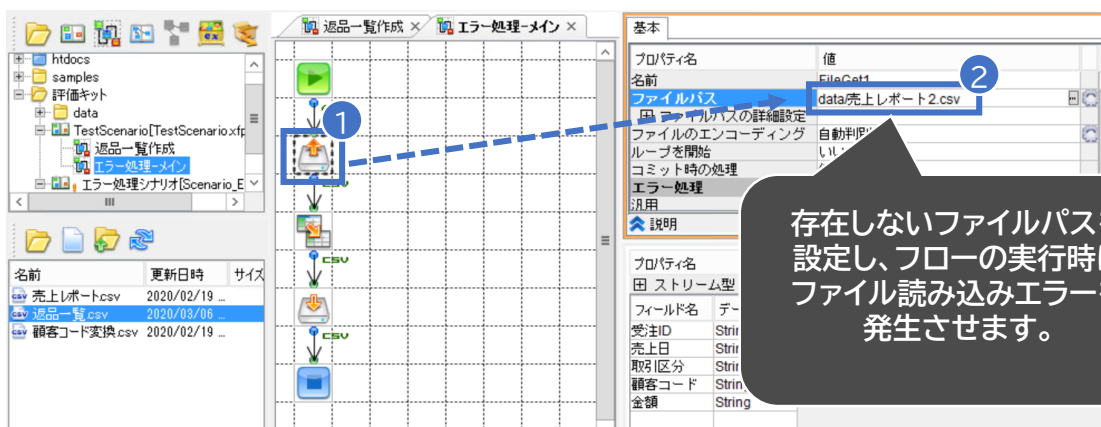
メインフローのコピー

ステップ1で作成したフローをコピーし、エラーが発生するように設定を変更します。

1. ツリーパイン上の返品一覧作成を右クリック、「複製」を選択し、コピー先のフロー名を「エラー処理-メイン」と設定します。



2. 追加された「エラー処理-メイン」フローのFileGetコンポーネントを選択し、インスペクタ上の「ファイルパス」プロパティを存在しないファイルパス(data/売上レポート2.csv)に変更します。



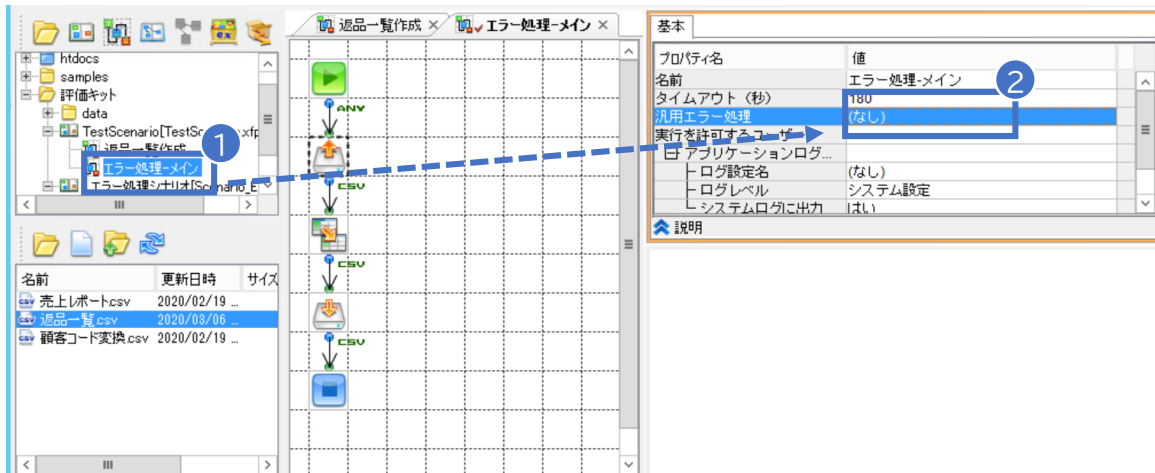


エラー処理フローの設定

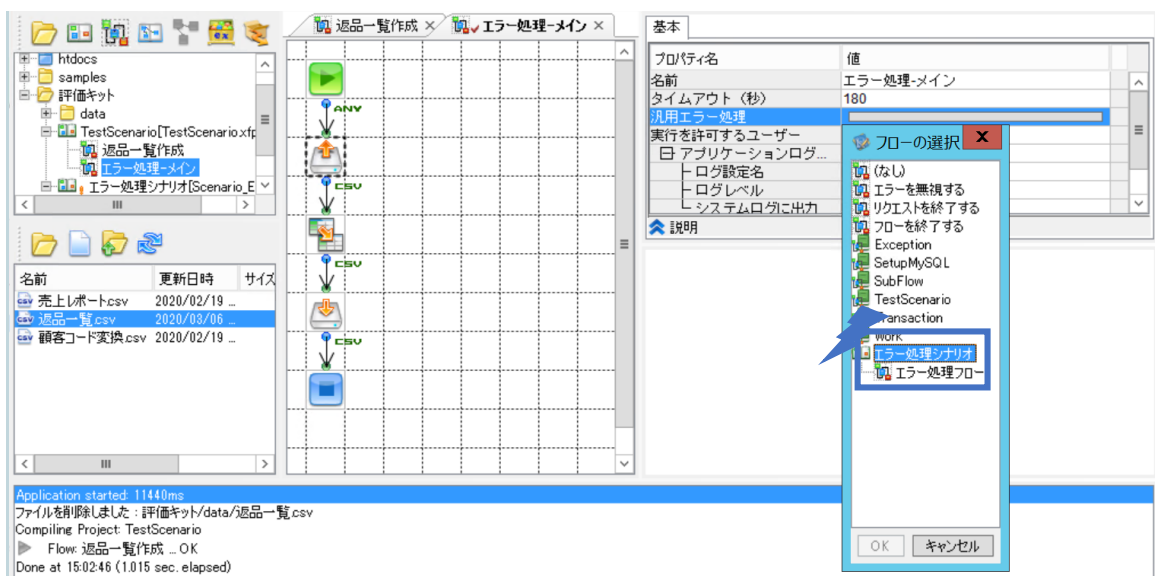
エラー発生時の動作(エラーハンドリング)の指定に、任意のフローを設定します。

今回は評価キットとしてあらかじめ作成してあるフローをエラー処理として設定しましょう。

1. ツリーペイン上の「エラー処理-メイン」を選択し、インスペクタ上の「汎用エラー処理」プロパティの値欄をクリックします。



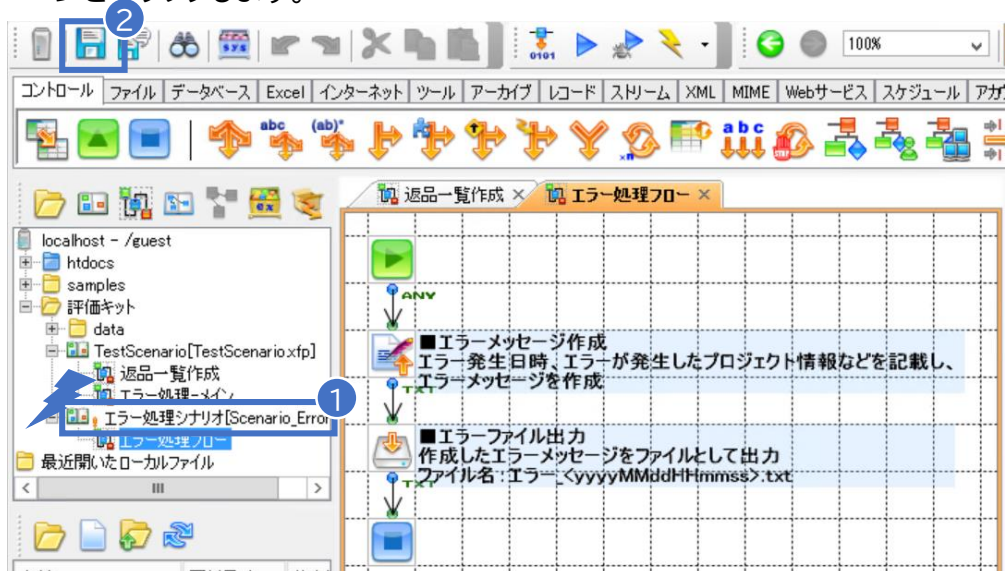
2. 開かれたプロジェクト一覧で「エラー処理シナリオ」プロジェクトをダブルクリックし、配下の「エラー処理フロー」フローを選択する。



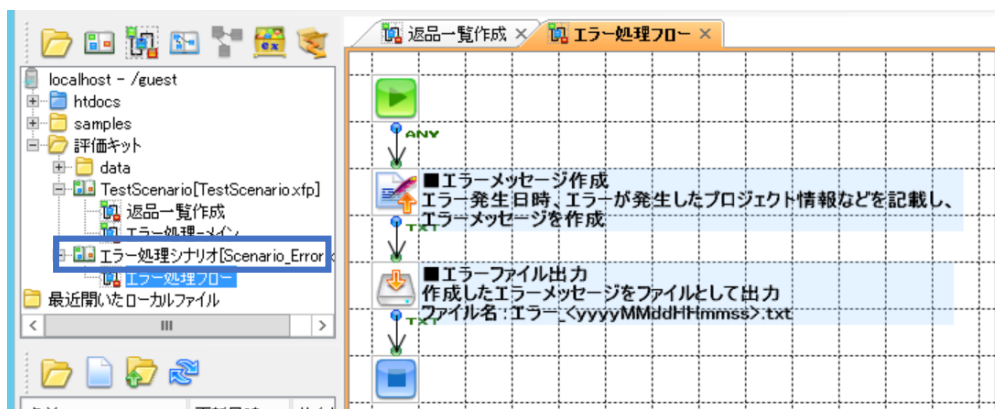


エラー処理フローの設定

- ツリーペイン上に表示されているプロジェクトファイル「Scenario_Error.xfp」をダブルクリックで開き、「エラー処理フロー」フローがオープンされたことを確認の上で保存ボタンをクリックします。

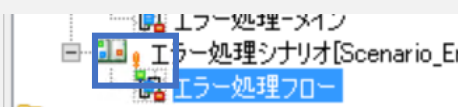


- 対象のフローが保存・コンパイルされ、ツリーペイン上のプロジェクト名から「！」マークが消えることを確認する。

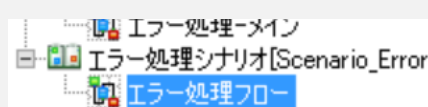


■一口メモ コンパイル

作成したフローを実行するには、サーバーへ登録を行う必要があります。
登録する方法としては、フローデザイナー上での「保存」「実行」「コンパイル」またはコマンドラインなどのインターフェースからの登録があります。
サーバーへ登録されたプロジェクトはツリーペイン上から「！」マークが消えます。



コンパイル未実施の
プロジェクト表示



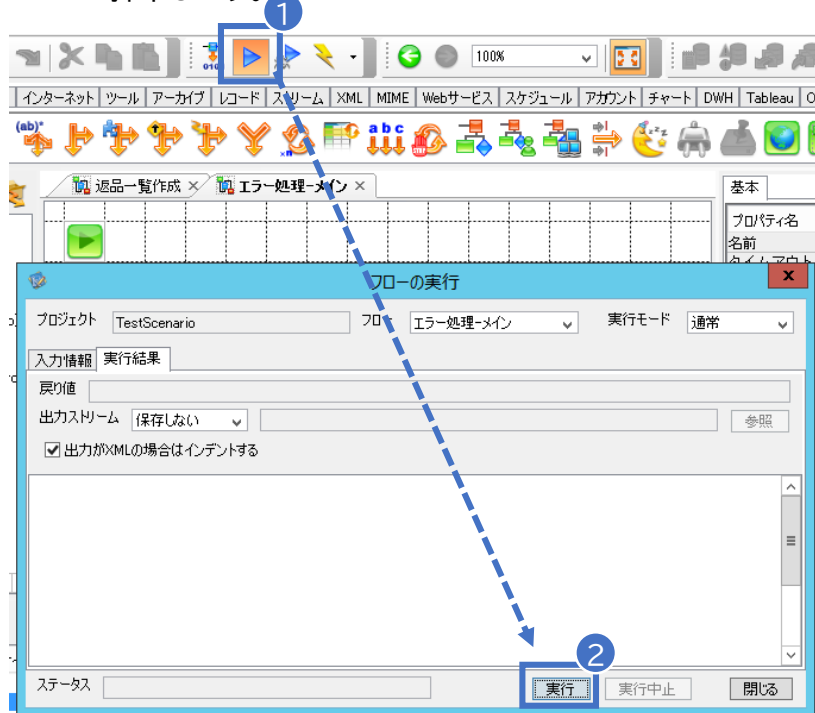
コンパイル実施済の
プロジェクト表示



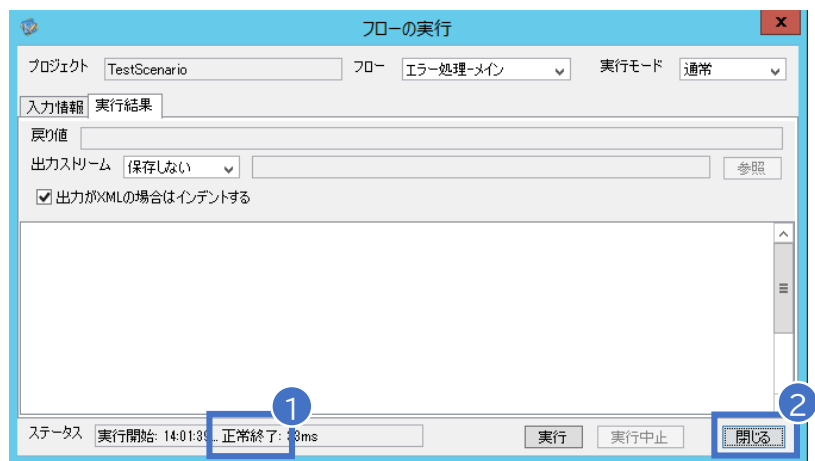
実行結果の確認

エラーの発生と、それをハンドリングするエラー処理フローを設定したフローを実行し、メッセージファイルが出力されるか確認しましょう。

1. フローデザイナー画面上部の実行ボタンをクリック、実行コンソールを呼び出し、実行ボタンを押下します。



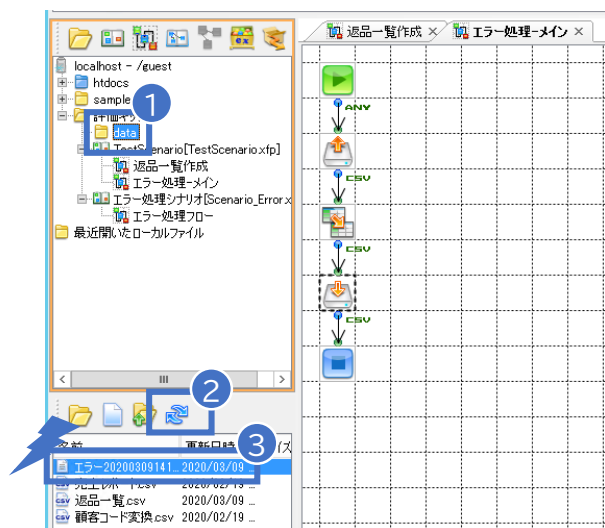
2. ステータスが“正常終了”していることを確認し、実行コンソールを閉じます。



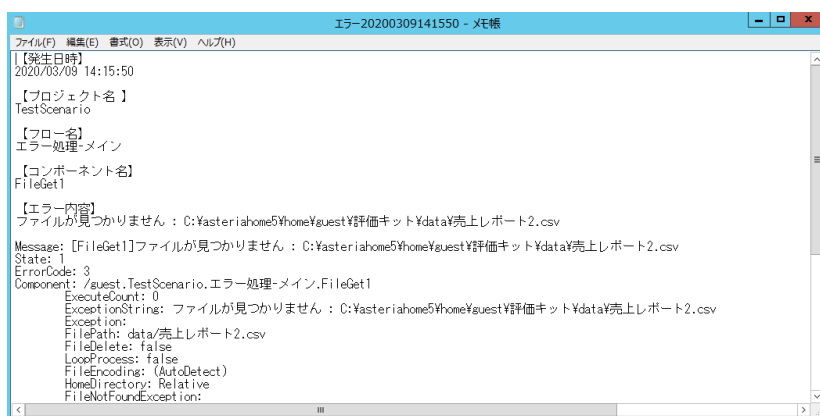


実行結果の確認

3. ツリーペイン上の「data」フォルダ(ファイルの出力先)を選択し、ファイルペインに表示された更新ボタンを押下、表示されたエラーファイル(エラー-yyyyyMMddHHmmss.txt)をダブルクリックします。



4. エラーファイルがオープンし、エラーメッセージがファイルに出力されていることを確認できます。



実行結果の確認が出来ました。

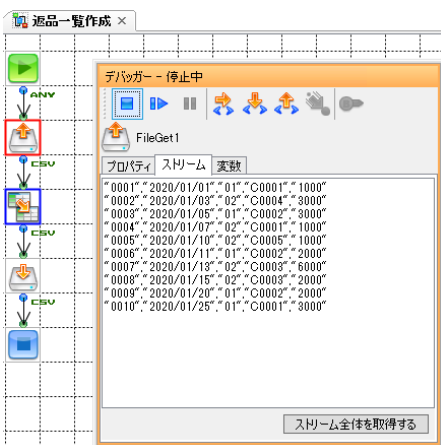
エラーは発生していますが、異常終了せずメッセージがファイルに出力されている(ハンドリングされている)ことを確認出来ます。

ステップ3 デザイナーの便利機能

フローデザイナーには開発時に役立つ便利機能が
まだまだたくさんあります。
その一部を体験してみましょう。

開発時に役立つ機能

デバッグ実行



フローの途中で処理を止め、加工/変換の様子を確認出来ます。
ASTERIA Warpでは「ステップイン」「ステップアウト」「ステップオーバー」に対応しています。

プロファイル実行

コンポーネント	開始	終了	実行	最小	最大	合計	平均	標準偏差	最大	合計
ベンチマーク CSV→固定値	Start1	2020-03-...	0	1	0	0	0	0	0	0
ベンチマーク CSV→固定値	Record5...	2020-03-...	0	10,000	0	2	2	9,999	0	162
ベンチマーク CSV→固定値	Branch5...	2020-03-...	0	639	0	2	2	420	0	988
ベンチマーク CSV→固定値	Mapper2...	2020-03-...	1	639	0	3	420	0	0	20
ベンチマーク CSV→固定値	Mapper2...	2020-03-...	1	639	0	3	420	0	0	842
ベンチマーク CSV→固定値	Mapper4...	2020-03-...	0	639	0	1	9	0	0	9
ベンチマーク CSV→固定値	Mapper3...	2020-03-...	14	639	1	6	431	0	0	84
ベンチマーク CSV→固定値	End2...	2020-03-...	0	639	0	1	1	0	0	81
ベンチマーク CSV→固定値	Mapper2...	2020-03-...	0	9,361	0	2	81	0	0	12,187
ベンチマーク CSV→固定値	FilePut1...	2020-03-...	11	9,361	1	34	6,338	0	0	2
ベンチマーク CSV→固定値	End1...	2020-03-...	0	9,361	0	1	2	0	0	2

各コンポーネント/フローの実行時間が計測できます。
どのコンポーネントがボトルネックになっているのか判定できるのでパフォーマンス改善に役立ちます。

仕様書出力



ユーザーが作成したプロジェクトとそれに関連する関数コレクション、外部変数セット、ストリーム定義セットを仕様書として活用できるようにそれぞれの情報をHTML形式で出力できます。

■一口メモ ASTERIA Warp Developer Network

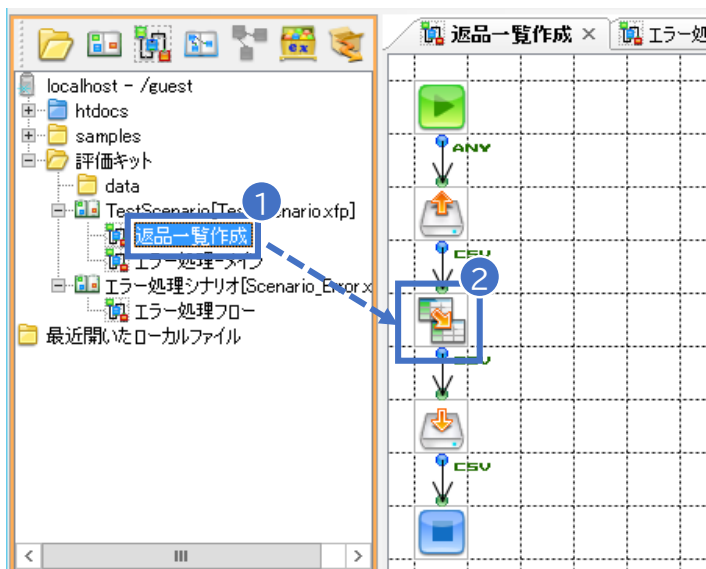
ASTERIA Warpのマニュアルや便利な使い方、逆引きリファレンスやFAQといった各種技術情報・ドキュメントが揃っているのが「ASTERIA Warp Developer Network」(ADN)です。このサイトでは新機能紹介や新アダプターの紹介など日々更新されていますし、Tipsやサンプルなども用意されているので、ぜひ一度ご覧ください。



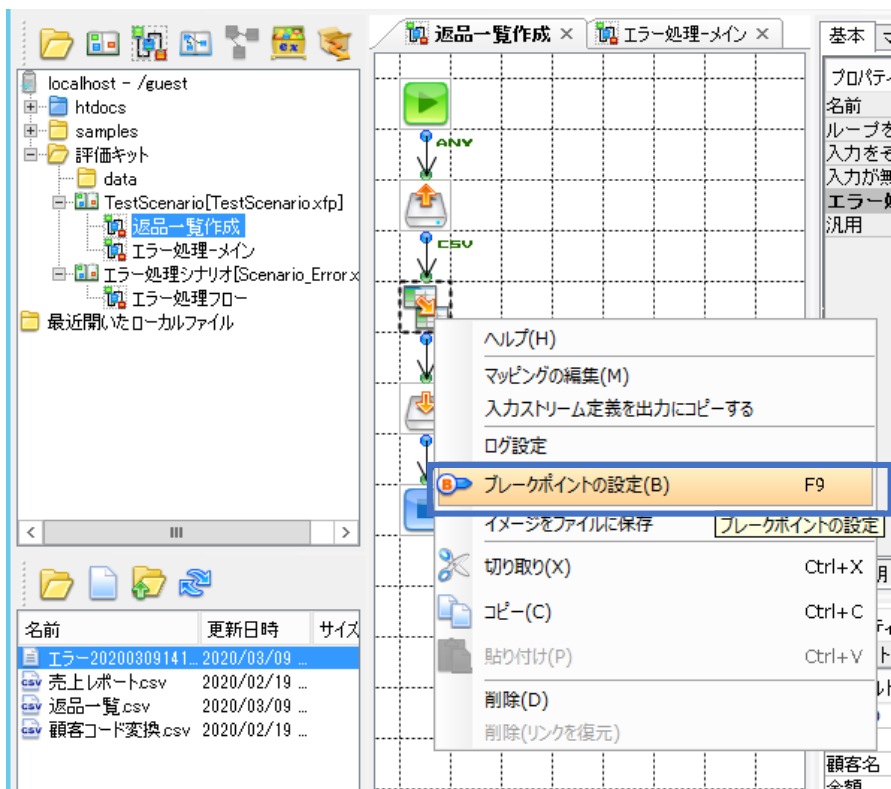
デバッグ実行

開発したフローを「デバッグ実行」で実行し、データが変換されるステップを確認しましょう。

1. ツリーペイン上の「返品一覧作成」を選択すると、ワークスペース上に「返品一覧作成」が開かれるので、Mapperコンポーネントを選択します。



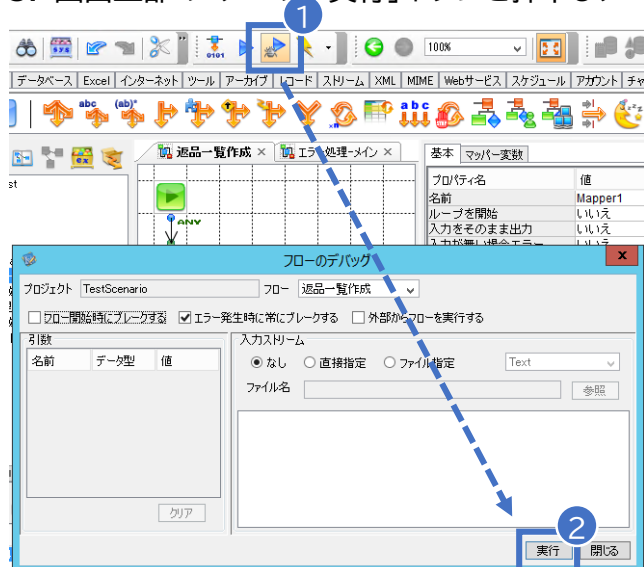
2. Mapperコンポーネントを右クリックし、「ブレークポイントの設定」を選択します。



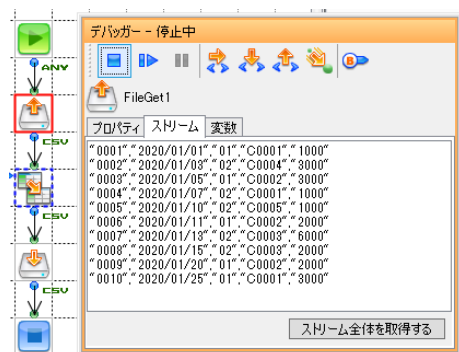


デバッグ実行

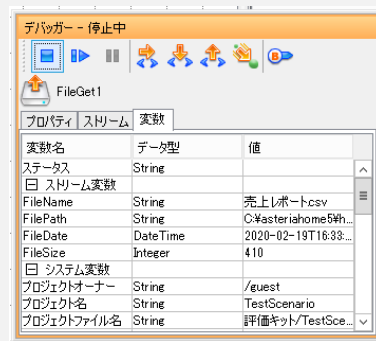
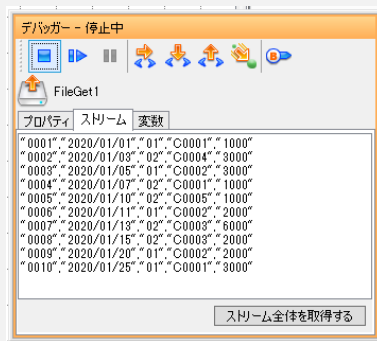
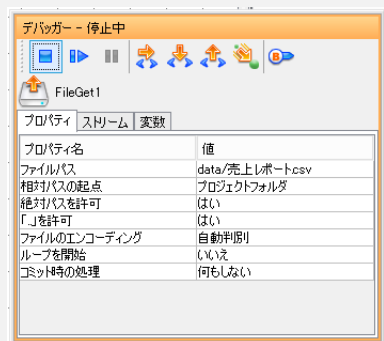
3. 画面上部の「デバッグ実行」ボタンを押下しデバッガーを起動、そのまま実行します。



4. デバッガーが起動し、ブレークポイントの直前のコンポーネントの状態を確認出来ます。



■一口メモ デバッグの情報



■ プロパティ

実行したコンポーネントに設定されているプロパティの値が表示されます。プロパティ式やマッピングによる設定を行ったプロパティ値を確認することができます。

■ ストリーム

実行後に出力したストリームの一部が表示されます。すべてのデータを表示するには、「ストリーム全体を取得する」をクリックします。

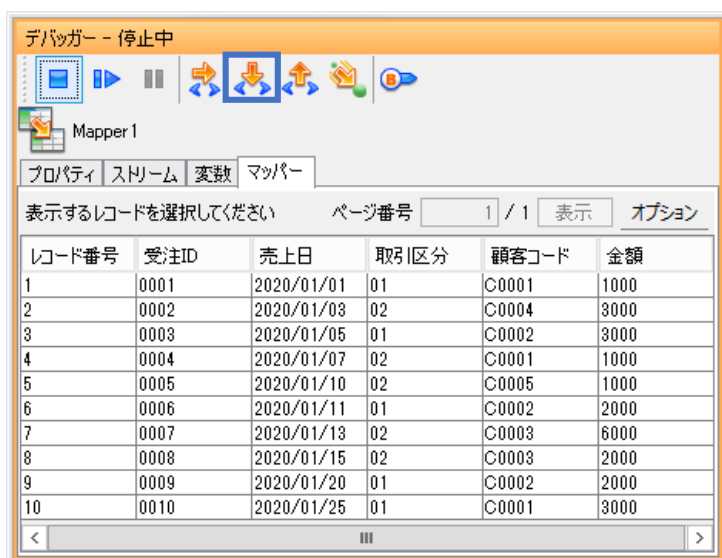
■ 変数

コンポーネント実行後の各種変数の値が表示されます。



デバッグ実行

5. デバッガーの「ステップイン」ボタンを押下すると次のコンポーネントに処理が進みます。



このようにフローデザイナーは「デバッグ実行」を利用することで処理の流れとデータの加工/変換の様子を確認出来るので、便利です。

■一口メモ デバッグの実行モード

■ ステップオーバー

- コンポーネントを1つ分実行する。
- 実行するコンポーネントがサブフローコンポーネントである場合またはエラーが発生してエラー処理フローを呼び出す場合に、フロー内に制御を遷移せずに次のコンポーネントに制御を遷移する。

■ ステップイン

- コンポーネントを1つ分実行する。
- 実行するコンポーネントがサブフローコンポーネントである場合またはエラーが発生してエラー処理フローを呼び出す場合に、フロー内に制御を遷移する。

■ ステップアウト

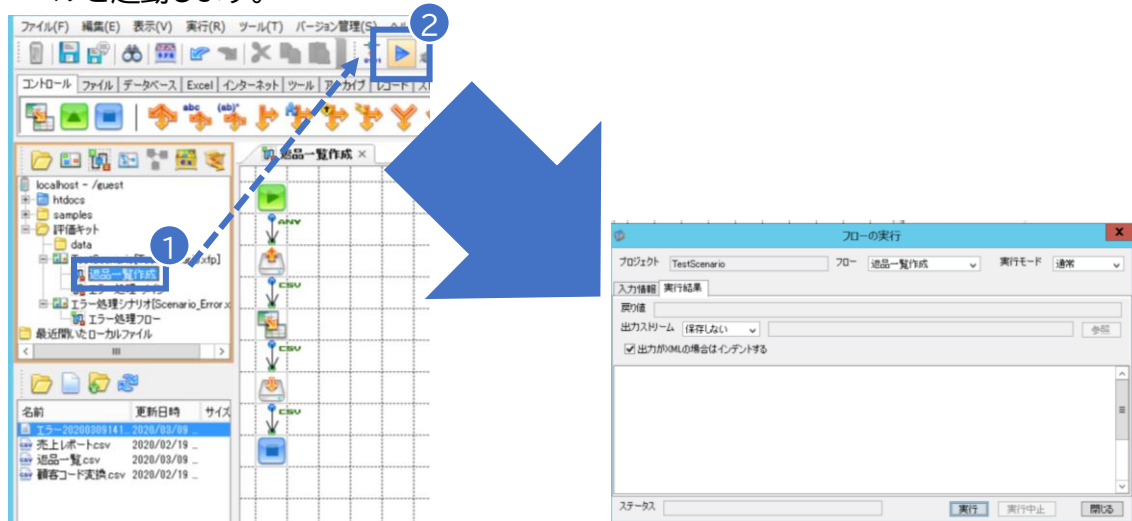
- 実行しているコンポーネントがあるフローがサブフローまたはエラー処理フローの場合、呼出し元フローに戻った段階でデバッガーが停止します。
- フローに呼出し元フローが無い場合、つまりメインフローの場合は、次のブレークポイントまで実行、またはブレークポイントが無いときはフローが最後まで実行されて終了します。



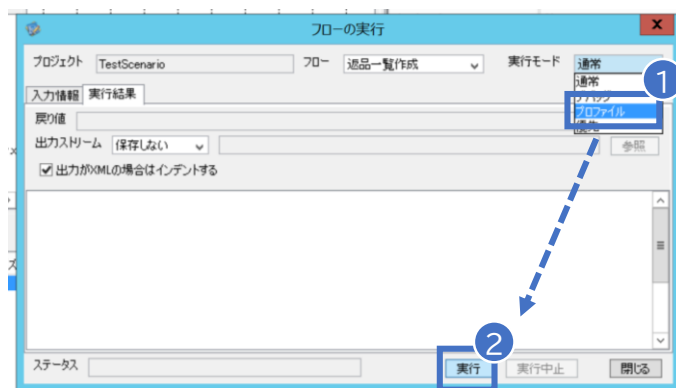
プロファイル実行

開発したフローを「プロファイル実行」で実行し、フロー/各コンポーネントの実行速度を計測しましょう。

1. ツリーペイン上の「返品一覧作成」を選択、画面上部の「実行」ボタンを押下し実行コンソールを起動します。



2. 実行コンソールの「実行モード」を“プロファイル”に変更し、実行します。



■一口メモ 実行モード

フロー実行時にフローサービスが出力するログ情報レベルを切り替えたり、フローの実行を優先実行する場合に指定できます。指定できる実行モードは下記の4モードです。

■ 通常

フロー実行の開始、終了の情報のみを出力します。運用段階でのフローは、通常このモードを使用します。

■ デバッグ

各コンポーネントが出力するデバッグ情報と実行詳細情報を出力します。主に開発時にフローの実行を確認する場合などに使用します。

■ プロファイル

各コンポーネントおよびフロー全体での実行時間を出力します。処理時間の長いフローのボトルネックがどこにあるかを調べる場合などに使用します。

■ 優先

フローを優先実行します。フローサービスに高負荷がかかり、Workerスレッドに空きがない場合でも、このモードを指定したフローは通常とは別のWorkerスレッドを使用して処理されます。



プロフィール実行

3. フローが正常終了し、実行コンソールに「プロフィール」タブが追加されます。

The screenshot shows the 'Flow Execution' window with the 'Profile' tab selected. The window displays the project name 'TestScenario', the flow '返品一覧作成', and the execution mode 'プロフィール'. The '実行結果' (Execution Results) tab is active, showing a table with columns for 'プロジェクト' (Project), 'フロー' (Flow), 'コンポーネント' (Component), '初期化' (Initialization), and '実行' (Execution). The status bar at the bottom indicates '実行開始: 12:04:01... 正常終了: 76ms'.

4. 「プロフィール」タブを選択すると、各コンポーネントの実行時間を確認できます。

The screenshot shows the 'Flow Execution' window with the 'Profile' tab selected. The table displays execution times for various components. The status bar at the bottom indicates '実行開始: 12:04:01... 正常終了: 76ms'.

プロジェクト	フロー	コンポーネント	初期化		実行			合計
			日時	時間	回数	最小	最大	
TestScenario	返品一覧作成	Start1	2020-03-12...		0	1	0	0
TestScenario	返品一覧作成	FileGet1	2020-03-12...		0	1	16	16
TestScenario	返品一覧作成	Mapper1	2020-03-12...		0	1	0	0
TestScenario	返品一覧作成	FilePut1	2020-03-12...		0	1	0	0
TestScenario	返品一覧作成	End1	2020-03-12...		0	1	0	0

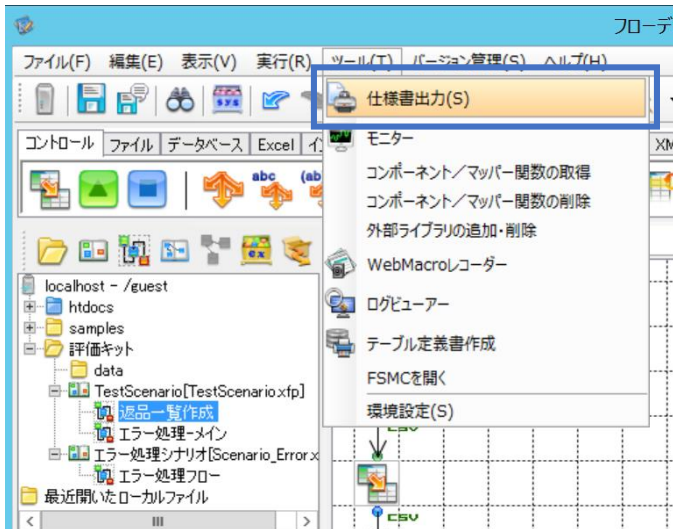
このように「プロフィール実行」で各コンポーネントおよびフロー全体での実行時間が計測され、出力されます。なお、ここで計測される時間はフローの実行にかかった時間のみで、HTTPリクエストを受けてからレスポンスを返すまでの間などのフローの実行前後にかかる時間は含まれません。



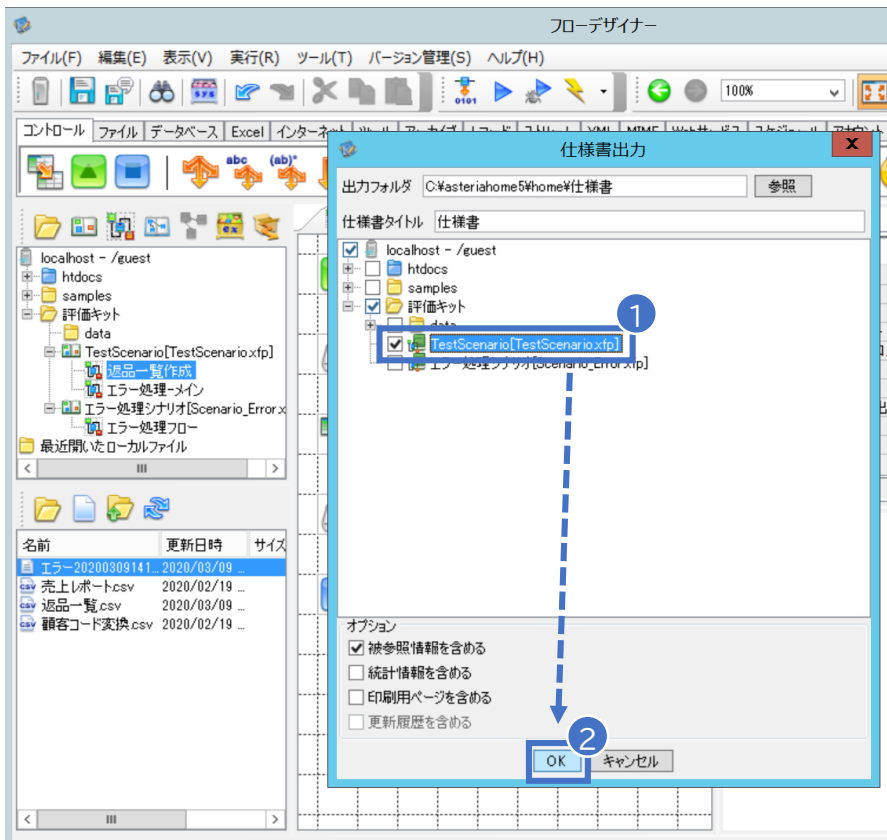
仕様書出力(本機能はStandardエディション以上でご利用いただけます)

開発したフローの仕様書を作成しドキュメント管理にもASTERIA Warpを利用しましょう。

1. フローデザイナー画面上部の「ツール」メニューを選択、「仕様書出力」メニューを選択します。



2. オープンされたプロジェクト一覧画面で「評価キット」フォルダ配下の「TestScenario」プロジェクトのみにチェックを入れ、「OK」ボタンを押下します。



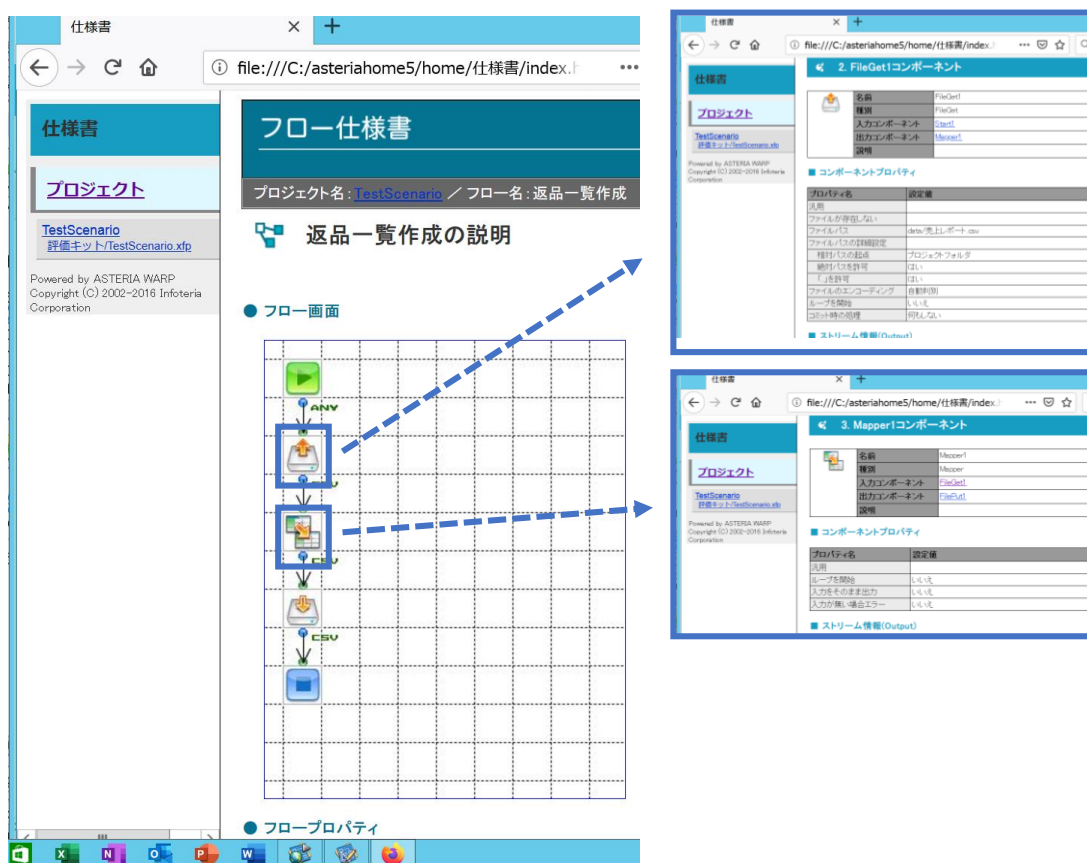


仕様書出力(本機能はStandardエディション以上でご利用いただけます)

3. ブラウザが起動し、仕様書が表示されます。



4. 仕様書はHTML形式となっていて、各リンクを押下することで細かい情報まで確認することが可能です。



仕様書出力機能ではフローデザイナーを通じて設定した内容は全て反映されますので、ドキュメント整備の手間を大幅に減らすことができます。