

**Asteria<sup>☆</sup> warp**

**環境移行ガイド**

## ご注意

本書は著作権法により保護されています。アステリア株式会社による事前の許可無く、本書のいかなる部分も無断転載、複製、複写を禁じます。本書の内容は、将来予告無しに変更することがあります。

ASTERIA およびアステリアは、アステリア株式会社の登録商標です。

このマニュアルに記載されているその他の会社名および製品名は、あくまでその会社および製品を識別する目的にのみ使用されており、それぞれの所有者の商標または登録商標です。

2025年1月31日

アステリア株式会社

Copyright © 2025 Asteria Corp. All rights reserved.

# 目次

1. 移行する前に【必ずお読みください】.....	4
1.1. ASTERIA Warp 2406 より前のバージョンをご利用の場合 .....	4
1.2. 使用する Java の変更について .....	4
1.3. フォルダー名や Windows サービス名の変更について .....	4
1.4. 初期設定時の注意点について .....	4
1.5. 廃止された機能について .....	5
1.6. チェックポイントについて .....	5
1.7. 3rd パーティ製のアダプターについて .....	5
2. データを移行する .....	6
2.1. 移行ツールについて .....	6
2.2. 移行ツールのダウンロードについて .....	6
2.3. 移行ツールの起動方法について .....	7
2.4. データの移行方法について .....	11
2.5. 移行したフローの動作を確認する .....	13
3. 付録 .....	14
3.1. 移行ツールの起動オプションについて .....	14
■ 改版履歴 .....	15

## 1. 移行する前に【必ずお読みください】

### 1.1. ASTERIA Warp 2406 より前のバージョンをご利用の場合

ASTERIA Warp 2406 より前のバージョンから ASTERIA Warp 2412 以降のバージョンへは移行できません。必ず ASTERIA Warp 2406 へバージョンアップしてから ASTERIA Warp 2412 以降のバージョンへ移行してください。

また、以後この文書では、移行元となる ASTERIA Warp2406 のことを旧バージョン、移行先となる ASTERIA Warp2412 以降のバージョンのことを新バージョンと記します。

### 1.2. 使用する Java の変更について

ASTERIA Warp 2412 以降では、実行環境として JDK Development Kit 21 以降の LTS 版の JDK を使用します。ASTERIA Warp 2412 以降をインストールする前に JDK21 以降の LTS 版(x64)のインストールが必要になります。また、使用する OS が Windows の場合は、MSI インストーラーを使用して JDK をインストールしてください。

### 1.3. フォルダー名や Windows サービス名の変更について

ASTERIA Warp 2412 からインストーラーで設定するインストールフォルダーの初期フォルダーなどが次のように変更されましたのでご注意ください。

	変更前	変更後
フローサービス インストールフォルダー	asteria5	asteria6
フローデザイナー インストールフォルダー	ASTERIA Warp Flow Designer	ASTERIA Warp Flow Designer6
Windows サービス名	ASTERIA5.FlowService	ASTERIA6.FlowService
Windows サービス表示名	ASTERIA Warp	ASTERIA Warp6
フローサービス スタートメニュー	ASTERIA Warp	ASTERIA Warp6
フローデザイナー スタートメニュー	ASTERIA Warp Flow Designer	ASTERIA Warp Flow Designer6

### 1.4. 初期設定時の注意点について

インストール後に行う初期設定で指定するポート番号には、移行元となる旧バージョンと同じポート番号を指定してください。また、初期設定で指定するデータフォルダーには、移行元となる旧バージョンのデータフォルダーと**同じフォルダーを絶対に指定しないようにしてください**。同じフォルダーを指定した場合、旧バージョンのデータフォルダーに新バージョンのデータフォルダーが上書きされて移行できなくなります。

## 1.5. 廃止された機能について

ASTERIA Warp 2412 で次の機能が廃止されました。

- Subversion を使用したバージョン管理機能
- SOAP 実行設定(SOAPトリガー)
- SOAP テンプレートフロー
- Web サービス関連コンポーネント
- Notes 関連コンポーネント
- TableauMakeHyper/TableauMakeTde/TableauPublishTde コンポーネント
- Handbook 関連コンポーネント
- JSONEncode/JSONDecode コンポーネント
- 旧 PDF コンポーネント、旧 PDFMerge コンポーネント  
その代わりに PDF2 が PDF に PDFMerge2 が PDFMerge に名称変更になりました
- ImageConverter コンポーネント
- StrToDate マッパー関数
- FormatDate マッパー関数

上記リストの赤文字の機能を使用しているフローは移行できませんので、事前に ASTERIA Warp 2406 にてこれらのコンポーネントを使用しない方法でフローを実装しなおしてください。上記リストの黒文字の機能は代替機能がございます。

## 1.6. チェックポイントについて

エンタープライズエディションでチェックポイントを使用している場合には、エラーとなっているチェックポイントのリクエストは移行されません。移行前にエラーとなっているリクエストを再実行してください。

## 1.7. 3rd パーティ製のアダプターについて

3rd パーティ製のアダプターの移行方法につきましてはアダプター製造元にお問い合わせください。

## 2. データを移行する

### 2.1. 移行ツールについて

旧バージョンから新バージョンへの移行には、移行ツールと呼ばれるコマンドラインツールを使用します。移行ツールには次のような機能が含まれます。

- 定義ファイルやプロジェクトファイルなどのコピー機能
- 代替コンポーネントへの置き換え機能
- 移行前の事前確認機能
  - 「1.5 廃止された機能について」で対応が必須の項目についてログ出力を行うなど、どのようなデータ移行が行われるかを事前に確認できます。

### 2.2. 移行ツールのダウンロードについて

移行ツールはユーザーサイト上で公開されています。ユーザーサイトにログイン後に「マイページ>ダウンロード>最新のリリース製品」画面からダウンロードしてください。ユーザーサイトのログイン URL は次の通りです。

【URL】<https://asteria.com/mng/login>

ダウンロードした移行ツールのファイル名は次のようなファイル名となっています。

migrationtool-XXXX.XXXX.zip

ここで、上記ファイル名の「XXXX.XXXX」の部分はバージョン番号となっており、例えば、

migrationtool-2412.2229.zip

のようなファイル名となります。

## 2.3. 移行ツールの起動方法について

移行ツールをダウンロードしたら次の手順で移行ツールを起動します。

### (1) 移行ツールをコピーします

新バージョンのインストールされているサーバーにダウンロードした移行ツールを配置します。新バージョンのインストールされている OS に応じてコピー & ペースト、FTP や SCP などファイルを配置してください。

### (2) 移行ツールを解凍します

#### ◆Windows の場合

エクスプローラーで展開、もしくは、zip 解凍ツールで展開します。

#### ◆Linux の場合

必要に応じて zip コマンドをインストールしてから unzip コマンドで展開します。

### (3) 移行ツールを解凍すると次のようなファイル構成になっています

項番	ファイル/フォルダー名	説明
1	migration-tool.bat	Windows 用の実行ファイル(バッチファイル)
2	migration-tool	Linux 用の実行ファイル(シェルスクリプト)
3	libs フォルダー	実行に必要なライブラリが格納されているフォルダー

新バージョンのインストールされている OS に応じて、項番1のバッチファイルまたは項番2のシェルスクリプトのどちらか一方のファイルを使用します。

### (4) 実行情報を設定します

上記の実行ファイルをテキストエディターで開きます。

#### ◆Windows の場合

実行ファイルの 5 行目と 6 行目の JDK\_DIR と INSTALL\_DIR を、新バージョンをインストールした環境に合わせて設定します。

```
set JDK_DIR=C:¥Program Files¥Java¥jdk-21
set INSTALL_DIR= C:¥Program Files¥asteria6
```

上記の赤字部分を、新バージョンをインストールした環境に合わせて編集してください。

#### ◆Linux の場合

実行ファイルの 3 行目と 4 行目の JDK\_DIR と INSTALL\_DIR を、新バージョンをインストールした環境に合わせて設定します。

```
JDK_DIR=~/jdk-21.0.5  
INSTALL_DIR=~/asteria6
```

上記の赤字部分を、新バージョンをインストールした環境に合わせて編集してください。

#### (5) 移行ツールを仮実行する

ここまでの設定が正しく行われているかどうかを確認するため、移行ツールを試しに実行します。

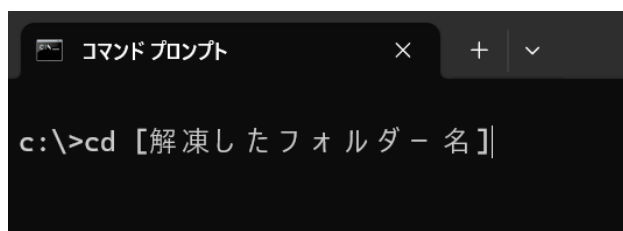
#### ◆Windows の場合

まず、コマンドプロンプトを開きます。コマンドプロンプトは様々な方法で開くことができますが、簡単な方法として次のような方法があります。最初に Windows のスタートメニューを開き、検索ボックスに「cmd」と入力します。



検索結果に表示される「コマンドプロンプト」をクリックすることで、コマンドプロンプトを開くことができます。

次に、コマンドプロンプト上で、移行ツールを解凍したフォルダーに移動します。



例としては、`cd c:¥migrationtool` のようになります。



最後に、移行ツールを実行します。コマンドプロンプト上で **migration-tool** と入力して Enter キーを押します。

```
コマンド プロンプト
c:\migrationtool>migration-tool|
```

しばらくしてから、次のように使用方法が表示されれば設定は正しく行われています。

```
コマンド プロンプト
c:\migrationtool>migration-tool
-oldのパスを指定してください

使い方: warpmaigration.bat [-check|-overwrite] -old DATA_DIR_2406 [-new DATA_DIR_2412 -force -debug]

オプション:
-check
  - コピーや上書きなどを行わず、ファイルのチェックのみを行うモードで実行します。
  - -checkも -overwriteも指定されていない場合は checkモードで実行します。
-overwrite
  - -forceが指定されていない場合は、チェックを行い[要対応]のメッセージがなければ、ファイルのコピーやバックアップを行った後、自動変換可能なファイルを上書きします。
-old <2406のデータディレクトリ>
  - 2406のデータディレクトリを指定します。
  - checkと overwrite両方で指定が必須です。
-new <2412のデータディレクトリ>
  - 2412のデータディレクトリを指定します。
  - overwriteの場合に指定が必須です。
-force
  - overwriteモードでチェックを行わずに強制的にコピーや上書きを行います。
-debug
  - ファイルへの変更内容を表示します。

c:\migrationtool>
```

#### ◆Linux の場合

まず最初に、コンソール画面上で移行ツールを解凍したフォルダーに移動します。

```
$cd [解凍したフォルダー名]
```

例としては、**cd migrationtool** のようになります。

次に、移行ツールのシェルスクリプトに実行権限を付与します。

```
$chmod u+x migration-tool
```

最後に、移行ツールを実行します。コンソール画面上で **migration-tool** と入力して Enter キーを押します。

```
$migration-tool
```

しばらくしてから、次のように使用方法が表示されれば設定は正しく行われています。

### \$migration-tool

-old のパスを指定してください

使い方: warpmaigration.bat [-check|-overwrite] -old DATA\_DIR\_2406 [-new DATA\_DIR\_2412 -force -debug]

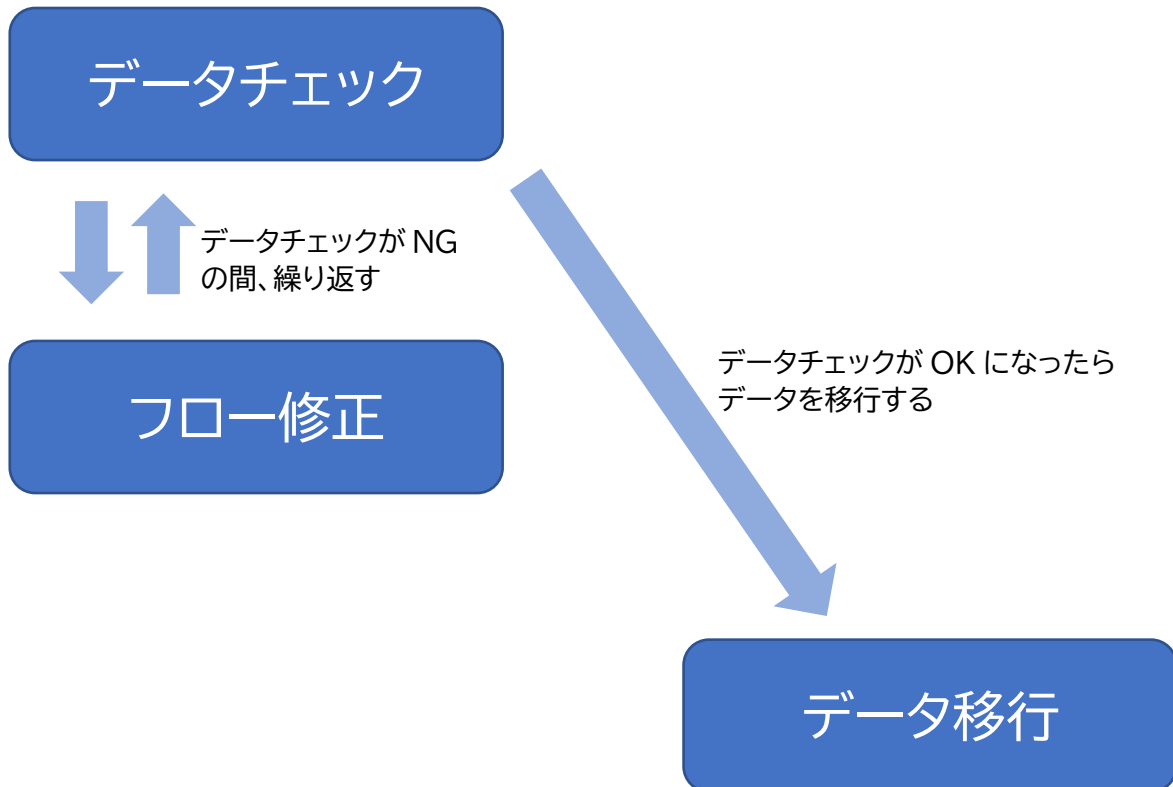
オプション:

- check
  - コピーや上書きなどを行わず、ファイルのチェックのみを行うモードで実行します。
  - -check も-overwrite も指定されていない場合は check モードで実行します。
- overwrite
  - -force が指定されていない場合は、チェックを行い[要対応]のメッセージがなければ、ファイルのコピーやバックアップを行った後、自動変換可能なファイルを上書きします。
- old <2406 のデータディレクトリ>
  - 2406 のデータディレクトリを指定します。
  - check と overwrite 両方で指定が必須です。
- new <2412 のデータディレクトリ>
  - 2412 のデータディレクトリを指定します。
  - overwrite の場合に指定が必須です。
- force
  - overwrite モードでチェックを行わずに強制的にコピーや上書きを行います。
- debug
  - ファイルへの変更内容を表示します。

\$

## 2.4. データの移行方法について

移行ツールを使って旧バージョンから新バージョンへ移行するためには次のステップが必要になります。



まず最初に、移行ツールを使って、旧バージョンから新バージョンへデータ移行が可能かどうかをチェックします。データチェックを行うためには、次のようにして移行ツールを実行します。

```
migration-tool -old [旧バージョンのデータフォルダー]
```

実行例としては、`migration-tool -old c:¥asteriahome5` となります。

チェックを実行するとチェック結果として「**要対応**」、「**警告**」、「**自動変換**」の 3 種類のメッセージが出力されます。チェック結果に、「**要対応**」と表示されたフローが存在していた場合は、そのフローには新バージョンでは使用できない機能が含まれています。次ページの例の場合、「852902.xfp」に SOAPDecoder コンポーネントが2つ存在することが分かります。SOAPDecoder コンポーネントはバージョン 2412 で廃止されたコンポーネントなので、バージョン 2412 以降のバージョンではこのフローは動作しません。

\* 実行状況:

チェック対象 : c:\asteriahome5  
実行日時 : 2025/01/30 16:06

\* ホームフォルダ:

/asu : home\asu  
/guest : home\guest

\* Check:

ファイルをチェックします : c:\asteriahome5\system\conf\flowservice.ifx  
[自動変換]FlowSchedule の設定がありません  
[自動変換]FlowSchedule の設定がありません  
[自動変換]FlowMetric の設定がありません  
[自動変換]FlowMetric の設定がありません  
[自動変換]FlowMetric の設定がありません  
[自動変換]SOAP の設定が見つかりました  
[自動変換]SOAP の設定が見つかりました  
[自動変換]SOAP の設定が見つかりました  
要対応: 0, 警告: 0, 自動変換: 8

ファイルをチェックします : c:\asteriahome5\home\asu\TriggerMap.xconf  
要対応: 0, 警告: 0, 自動変換: 0

...(中略)...

プロジェクトファイルをチェックします (5/6) :  
c:\asteriahome5\home\guest\852902\852902.xfp  
[要対応] 廃止されたコンポーネントが見つかりました :  
852902.Flow1.SOAPDecoder1  
[要対応] 廃止されたコンポーネントが見つかりました :  
852902.Flow2.SOAPDecoder1  
要対応: 2, 警告: 0, 自動変換: 0  
プロジェクトファイルをチェックします(6/6) : c:\asteriahome5\home\guest\ftp.xfp  
要対応: 0, 警告: 0, 自動変換: 0

-----  
要対応: 2, 警告: 2, 自動変換: 11  
自動変換した設定やフローが期待通り動くかの確認が必要です  
-----

チェック結果で「要対応」と表示されたプロジェクトはそのままでは移行できません。旧バージョンで修正が必要なプロジェクトなので廃止された機能を使わないフローに修正する必要があります。

旧バージョンでフローを修正したら、再度移行ツールでチェックします。このサイクルを繰り返して「要対応」が0件となるようにします。「要対応」が0件となって初めて新バージョンヘデータを移行することができます。

「要対応」が0件となったらいよいよ旧バージョンから新バージョンへデータを移行します。データを移行するには、次のようにして移行ツールを実行します。

```
migration-tool -overwrite -old [旧バージョンのデータフォルダー] -new [新バージョンのデータフォルダー]
```

実行例としては、

```
migration-tool -overwrite -old c:¥asteriahome5 -new c:¥asteriahome6
```

となります。

コマンドが正常に終了し、チェック時と同じような実行結果が表示されたらデータ変換は完了です。

## 2.5. 移行したフローの動作を確認する

データ移行が完了したので、新バージョンのサーバーを起動することができます。サーバーを起動すると登録済みのスケジュールや実行設定も即座に動作し始めることに注意してください。

移行ツールによるデータ移行は完了しましたが、新バージョンで正しく動作するか検証が必要になります。特に「警告」、「自動変換」と表示されたプロジェクトについてはフローの動作が想定通りであるか必ず検証してください。

「警告」の例として正規表現を使用しているマッパー関数の例が下の例になります。

```
プロジェクトファイルをチェックします(4/6) : c:¥asteriahome6¥home¥guest¥Project1.xfp  
[警告]正規表現を使用したコンポーネント/マッパー関数は挙動が変わる可能性があるため確認  
が必要です : Project1.Flow1.Mapper1.RegexpMatch1  
要対応: 0, 警告: 1, 自動変換: 0
```

Project1 の中の Flow1 の中の Mapper1 に配置されている RegexpMatch1 関数は正規表現を使用しているため想定通りの動作になっているか検証が必要になります。

このようにして、「警告」または「自動変換」と表示されたフローの動作をすべて検証します。すべての検証が終わったらデータ移行は完了です。

### 3. 付録

#### 3.1. 移行ツールの起動オプションについて

移行ツールの起動オプションは次の表の通りです。

オプション	説明	例
-check	旧バージョンのデータのチェックのみを行います。「-overwrite」を指定しない場合はデフォルトの動作となります。	-check
-overwrite	最初に旧バージョンのデータのチェックを行い、「要対応」のメッセージが 0 件の場合のみデータ移行を行います。	-overwrite
-old [移行元データフォルダー]	移行元であるバージョン 2406 のデータフォルダーを指定します。	-old c:¥asteriahome5
-new [移行先データフォルダー]	移行先のバージョンのデータフォルダーを指定します。	-new c:¥asteriahome6
-force	「-overwrite」と同時に指定した場合、「-overwrite」で最初に行われるチェック処理をスキップしてデータ移行を行います。つまり、「要対応」のメッセージが 0 件でなくてもデータ移行されます。	-overwrite -force
-debug	詳細なログを出力します	-debug

## ■ 改版履歴

版数	日付	内容
第 1 版	2024/12/24	新規作成
第 2 版	2024/12/25	「1.2 使用できる Java の変更について」の「LTS 版(x86)のインストール」を「LTS 版(x64)のインストール」に修正
第 3 版	2025/01/30	全面改訂